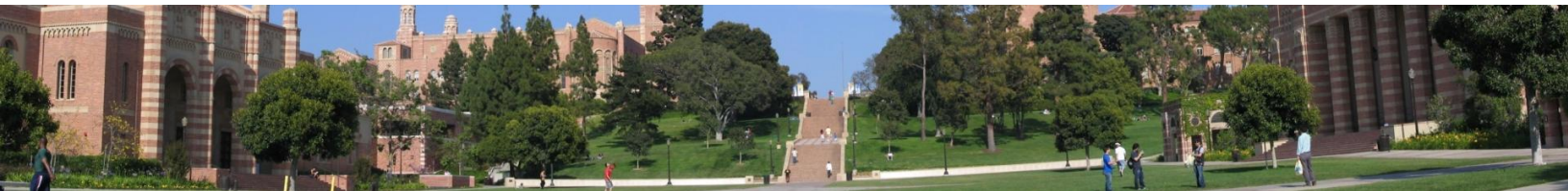
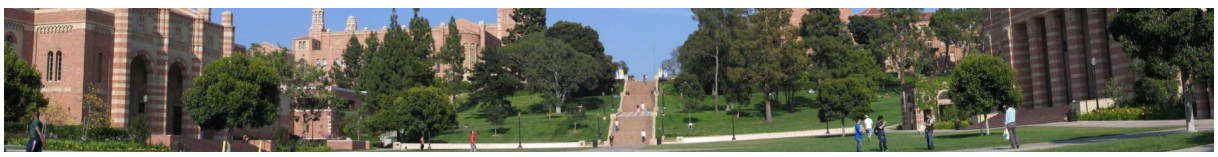
The International R Users Conference,
useR! 2014

Contributed Abstracts

University of California, Los Angeles
June 30 - July 3, 2014



Contributed Talks



A new framework for portfolio management

Ting-Kam Leonard Wong

Department of Mathematics, University of Washington, Seattle
Contact author: wongting@uw.edu

Keywords: Portfolio management, rebalancing, relative arbitrage, diversity, stochastic portfolio theory

Stochastic portfolio theory provides a new mathematical framework for studying the behaviors of portfolios and the structure of equity markets. In contrast to the commonly adopted mean-variance framework which is essentially a dynamic optimization problem with significant estimation difficulty, stochastic portfolio theory focuses on empirically observable characteristics of equity markets such as the stability of capital distribution and the presence of sufficient volatility. Under mild conditions, explicit portfolios can be constructed which outperform a capitalization-weighted benchmark in the long run. Such portfolios are called *relative arbitrages* and we refer the reader to [1] and [2] for precise statements of these results.

In a recent paper [3], the authors develop a simple but novel *energy-entropy framework* which clarifies when and how rebalancing works and provides a consistent framework for attributing the performance of a hierarchical portfolio using information-theoretic concepts such as relative entropy and free energy. In [4] some central results in stochastic portfolio theory are given a geometric and intuitive interpretation.

Although there are several very good *R* packages for portfolio optimization and performance attribution, there is yet no *R* packages which implement these recent ideas in stochastic portfolio theory and related advances.

The author intends to fill the gap with a new *R* package to be called **RelValAnalysis** (relative value analysis). This package is designed to implement the aforementioned tools for analyzing the performance of portfolios relative to a capitalization-weighted benchmark. Among other things, the package will include classes and functions for measures of diversity of capital distribution, construction of functionally generated portfolios and the associated Fernholz decomposition, the energy-entropy decomposition and attribution, and simulation of common models in stochastic portfolio theory. The package should be of interest for students, researchers and practitioners. The talk will introduce the main functions of the package.

References

[1] Fernholz, E. R. (2002). *Stochastic portfolio theory*. Springer.

[2] Karatzas, I. and R. Fernholz (2009). Stochastic portfolio theory: an overview. *Handbook of numerical analysis 15*, 89–167.

[3] Pal, S. and T.-K. L. Wong (2013). Energy, entropy, and arbitrage. *arXiv preprint arXiv:1308.5376*.

[4] Pal, S. and T.-K. L. Wong (2014). The geometry of relative arbitrage. *arXiv preprint arXiv:1402.3720*.

Regression Fit Diagnostics Using freqparcoord

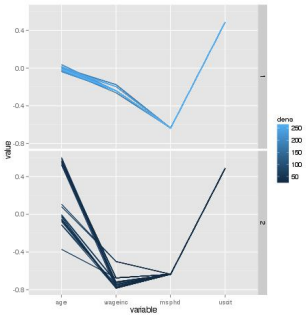
Norm Matloff[1⋆], **Yingkang Xie**[1]

1. University of California, Davis
⋆Contact author: matloff@cs.ucdavis.edu

Keywords: regression diagnostics, freqparcoord, parallel coordinates

The **freqparcoord** package, available on CRAN, takes a new approach to the parallel coordinates visualization method for multivariate data. Parallel coordinates (Unwin, 2006) is an exploratory method aimed at visualizing interrelations among variables, especially within groups. But it becomes difficult or impossible to use when the number of data points becomes even moderately large, which causes the "black screen problem," uninterpretable, dense clutter. This problem is solved in **freqparcoord** by plotting only a few "typical" lines in the graph, meaning the ones with the highest estimated multivariate density.

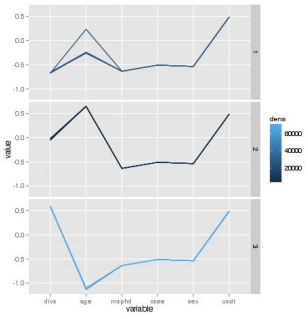
As an example, here is a **freqparcoord** plot of data from the 2000 Census data, for engineers and programmers in Silicon Valley, showing the 25 most typical data points for each gender. Compared to men (upper panel),, we see a much greater range of age among women, with lower wages, but with both genders typically being U.S. citizens with at most a bachelor's degree.



In the present work, we apply **freqparcoord** to assessing the fit of parametric regression models. The first axis is the "divergences," the differences beween the parametric and nonparametric estimates of the population regression function, while the other axes are the predictor variables. Note that the divergences are NOT the parametric model residuals, e.g. differences between fitted model values and response ("Y") values.

The question addressed is, "In what regions is the parametric fit poorer?" To answer that, the divergences are grouped into upper and lower tails; the default finds the data points that have divergences in the lower and upper 40%, then plots both groups, as well as the middle.

As an example, we fit a linear regression model, predicting wages from age, MS/PhD, CSEE, gender and U.S. citizenship in the Census data. There is a definite trend of overpredicting the young. Moreover, the text output (not shown) finds that the nonparametric R^2 is more than 10% higher than the (adjusted) one from `lm` (though both are low). This suggests adding a quadratic term in age to the model, which then indeed raises the R^2 value to a level similar to the nonparametric one. On the other hand, the graph does not suggest adding any interaction terms.



Any parametric regression model may be used. For instance, in data on graduate school admissions, we fit a logistic model predicting admission from grades, GRE scores and rank. The plot suggested a quadratic effect for grades and possible interactions.

References

Unwin, A., Theus, M. and Hofmann, H. (2006). *Graphics of Large Datasets: Visualizing a Million*, Springer, 2006.

R For Improving Consumer Engagement and Health Outcomes

Ken Yale[1 *], Frank Norman[2], Na'im Tyson[2]

1. ActiveHealth Management, Inc.
2. Knowledgent Group, Inc.
*Contact author: kyale@activehealth.net

Oral Talk Abstract for the The R User Conference 2014

Keywords: R, K-Means, CART, Segmentation, Mining

Background

Health reform shifts our focus from wholesale populations to retail individual consumers. Our care management platform and services has peer-reviewed and published documentation of improved patient care quality and affordability over the past decade, but the new ecosystem requires us to go much further in improving patient engagement, health outcomes and lower costs.

Working with Knowledgent, we utilized an advanced analytical approach using *R* combined with A-B testing and segmentation techniques to increase engagement rates and lower the cost of each engagement.

Methods

Combining internal member data and claims history with externally-purchased lifestyle & behavioral data, we built a segmentation model using K-Means Clustering and CART classification trees with a 1.3 million training population. Clustering showed clearly identifiable micro-segments with common behavioral characteristics based on input factors such as age, health status, socioeconomic factors, purchasing behaviors, technical savvy, education level, geography, etc.

We then conducted live A-B tests, varying messaging & channel usage, and measured response rates. Findings showed differing response rates among segments to the use of digital communication channels vs. traditional communications, for example. In addition, communication design alterations affected response rates for segments in varying degrees.

Results

By optimizing message & channel selection for micro-segments, response rates increased 74 percent.

Mining and visualization of operational data, including time & duration & outcome of calls, produced insights which enabled the cost of each individual engagement to be reduced as well.

R packages used for the segmentation modeling included k-means for the cluster analysis and rpart for analyzing the classification and regression trees.

Models were originally built using an underlying MySQL data store, but have been subsequently ported to access Hadoop data via HiveQL as well.

Conclusion

The improvements identified client savings of $6 million in avoidable health costs and labor cost savings producing a 900% internal ROI. We shall also cover future plans for additional work.

Plyrmr: a data manipulation DSL for big data

Antonio Piccolboni[1,*]

1. Revolution Analytics
*Contact author: rhadoop@revolutionanalytics.com

Keywords: data manipulation, hadoop, big data

plyrmr is the latest package to spawn from RHadoop, an open source project aimed at making *R* work seamlessly with the Hadoop system for the storage and processing of big data on commodity clusters. Like another RHadoop package, **rmr2**, it is specifically targeted to work with Mapreduce, Hadoop's batch computing subsystem. The goal for **plyrmr** was to strike a different compromise of power and ease of use biased toward the latter, thus helping to expand the circle of people who can access and process big data directly. The main compromise we accepted is that **plyrmr** is focused on structured data, specifically data organized in columns, like a `data.frame`, as feedback from our users indicated this was the most important use case. With this in mind, we set five design guidelines:

- Reduce the need to define functions even for the simplest tasks: to this end we have adopted a programming jargon popularized by the package **plyr** [2] (to which **plyrmr** also owes half of its name). Simple calculations such as the ratio of two columns or the average of another one can be described with expressions thanks to a non-standard but well understood evaluation method.
- Whenever users need to define functions to access advanced functionality, make them simpler and less specialized to their use in a mapreduce context, thus promoting reuse. In fact, all user defined function in the **plyrmr** API accept a data frame as their first argument and return a data frame, enabling the reuse, for instance, of functions from **plyr**, **dplyr** and **reshape2** for processing big data.
- Replace the potentially unfamiliar concept of a *key* with an SQL-like function `group` and related.
- Whereas **rmr2** was more a foundational package with a *minimalist* API, **plyrmr** includes mapreduce equivalents of many popular and useful functions, to show Hadoop conversion can be accomplished and to provide a useful set of tools even without any programming by the user. According to [1] **plyrmr** is more in the camp of a *humane interface*, whereby common use cases are worth defining and implementing, no matter how trivial their implementation. Converting more functions for Hadoop use can require as little as a call to the function `magic.wand`.
- using a technique known as delayed evaluation, reduce the cost of abstraction eliminating redundant I/O when possible.

A touch of syntactic sugar is a Unix-like `%|%` operator to make nested expressions more readable. The result are programs like `input("path/to/data-set") %|% where(var1/var2 > x) %|% group(id) %|% select(mean(var1))` that can run on the largest commodity clusters in use today, and process the largest data sets.

References

[1] Fowler, M. (2005, December). HumaneInterface. http://martinfowler.com/bliki/HumaneInterface.html. Accessed 2014-3-20.

[2] Wickham, H. (2009). plyr: Tools for splitting, applying and combining data. *R package version 0.1 9*, 651.

Beyond R CMD check: Helping *R* developers to detect CRAN package conflicts

Malick Claes*, Tom Mens, Philippe Grosjean

COMPLEXYS Research Institute, University of Mons, Belgium
*Contact author: maelick.claes@umons.ac.be

Keywords: CRAN, package dependencies, conflicts, maintainability, tool support

CRAN is a large software collection containing thousands of *R* packages maintained by thousands of different maintainers. The number of packages is growing very rapidly (currently there are over 5000 packages), which is considered by some as problematic [2]. Another problem is a lack of coordination between developers of dependent software components. Maintainability problems may arise and packages may cease to function correctly because of unexpected changes made to the packages they depend upon. In addition, problems with the dependency versioning system of *R* have been reported and possible directions for improvement have been proposed, such as staged package distributions (as in Debian) and versioned package management [3].

Currently, *R* package developers can use the `R CMD check` command to detect possible problems in CRAN contributed packages. This tool is also used to ensure conformance of accepted packages to the CRAN quality policy, and to check that packages don't break over time. However, since the number of packages is growing quickly, it becomes harder and harder to solve problems that are due to updates of packages that one directly or indirectly depends upon. We have studied the extent of this problem through an empirical analysis of CRAN's `R CMD check` results [1][1]. We observed that package quality and maintainability varies with the operating system considered. We also observed that a non-negligible amount of errors are caused by dependency updates and need to be fixed by the maintainers. Maintenance effort hence needs to take into account changes made to package dependencies. This may become detrimental to package maintainability in the long run if the number of CRAN packages keeps on growing at the same pace.

Therefore, there is a need for more specific tools dedicated to R package developers, that allow them to gain insight and deal with the implications and problems raised by package updates. Will changes to their packages cause potential problems to other CRAN packages? Do package updates or changes in the dependencies of other packages cause potential problems in one's own package? Being able to address such problems *a priori* during package development and maintenance, i.e., long before submitting it to CRAN, will reduce the effort of maintaining contributed CRAN packages.

We will report on a prototype tool that we have developed for the above.[2] It is more specific and fine-grained than the `R CMD check` and it considers potential conflicts with *all* CRAN packages, not only those currently tested. It aims to help *R* package maintainers to identify and avoid problems that could break their own package or those of others *before* sending it to CRAN. The tool is based on a fine-grained function-level analysis of dependencies, conflicts and clones (copy-paste reuse of code) between packages.

References

[1] Claes, M., T. Mens, and P. Grosjean (2014). On the maintainability of CRAN packages. In *IEEE CSMR-WCRE 2014 International Conference*.

[2] Hornik, K. (2012). Are there too many R packages? *Austrian Journal of Statistics 41*(1), 59–66.

[3] Ooms, J. (2013, June). Possible directions for improving dependency versioning in R. *R Journal 5*(1), 197–206.

[1] R package available at `github.com/maelick/extractoR`. CRAN historical data available at `github.com/maelick/CRANData`.

[2] R package available at `github.com/maelick/maintaineR`.

statsTeachR.org: A New Framework for Collaborative, Open-Access Curriculum Development

Nicholas G Reich[1*], **Jeff Goldsmith**[2], **Andrea S Foulkes**[1]

1. University of Massachusetts, Amherst
2. Columbia University
*Contact author: nick@umass.edu

Keywords: Teaching, Data Visualization, Reproducible Research

statsTeachR.org is a new, open-access, online repository with modular lesson plans for teaching statistics using R at the undergraduate and graduate level. Each curricular "module" focuses on teaching a particular statistical subject or concept. This provides teachers with flexibility to build their own course à la carte, choosing only modules that are relevant for their course. The modules range from introductory lessons in statistics and statistical computing to more advanced topics in statistics and biostatistics. A unifying goal for statsTeachR is to facilitate the use of hands-on exercises in statistical computing, data visualization, and reproducible research with R to teach fundamental concepts in statistics. For example, the **resamp** module teaches resampling inference by having students run their own bootstrapping routines. In our graduate-level biostatistics courses, we have piloted successfully a curriculum where students develop statsTeachR modules as final projects. We also have used statsTeachR as a platform and framework for sharing curriculum and teaching materials for similar courses being taught at different institutions (UMass-Amherst and Columbia University) and for interdisciplinary workshops. In addition to serving as a central location for interactive and modern lesson plans in statistics and statistical computing, statsTeachR has defined a standardized file structure for modules and supplies templates for LaTeX documents, including lab assignments and slides (both with optional **knitr** compatibility). For each module on statsTeachR.org, curricular materials are available either as a direct download from the site or by direct link to a external website such as OpenIntro.org or GitHub.com. Additionally, registered users on statsTeachR.org can curate and share their own statsTeachR course by choosing from the slate of existing modules.

Rapid Prototyping with R/Shiny at McKinsey: A New Way of Delivering Value for Our Clients

Aaron Horowitz[1*] , Frank Kroell[1]

1. McKinsey & Company
*Contact author: aaron_horowitz@mckinsey.com

Keywords: prototype, shiny, rCharts, visualization, analytics consulting, tool development

McKinsey & Company is a global management consulting firm, serving 90% of the world's largest companies. As part of the growing analytics team at McKinsey, we strive to ensure that leaders at these organizations recognize the importance and value advanced analytics can make. Our work frequently entails building analytics teams, piloting new methodologies, and finding new and innovative ways to serve our current and future clients. Our clients and colleagues are unfamiliar with advanced analytics, but increasingly understand its importance. With the help of **Shiny**, other packages and even externally integrated software, we now create rapid analytic prototypes that change the types of end-products we offer, and the ways in which we interact internally and externally.

We plan to discuss the means by which these prototypes solve multiple problems we face in delivering advanced statistical analysis including 1)demystifying the analytics "black box" 2)Productizing rapid tool development so it can be made by statistical professionals and 3)offering end-products we can pass off to enterprise software developers for full-scale applications. We'll then demonstrate products we've created which go well-beyond most toy examples, and are full-fledged applications. Finally, we'll discuss our process for doing so at speed, thanks to a custom-built framework we share and develop with each new product we create.

References

[1] RStudio, Inc. (2014). Shiny home page, http://rstudio.com/shiny/.

[2] McKinsey & Company (2014). McKinsey & Company home page http://www.mckinsey.com/

subsemble: Ensemble learning in *R* with the Subsemble algorithm

Erin LeDell [1,2,*], **Stephanie Sapp**[1,3], **Mark van der Laan**[1,2,3]

1. University of California, Berkeley
2. Division of Biostatistics
3. Department of Statistics
*Contact author: ledell@berkeley.edu

Keywords: machine learning, ensemble methods, cross-validation, prediction, big data

We present the **subsemble** *R* package, which implements the Subsemble ensemble machine learning algorithm (Sapp et al., 2013), a new variant of Super Learning (van der Laan et al., 2008). Ensemble methods that combine models trained on different subsets of observations have recently received increased attention as practical prediction tools for massive datasets. Subsemble is a general subset ensemble prediction method that partitions a full dataset into subsets of observations and trains a user-specified learning algorithm on each subset. Then a unique form of V-fold cross-validation is used to learn a final prediction function which combines the subset-specific fits via a user-specified metalearner algorithm. Instead of simply averaging subset-specific fits, Subsemble differentiates fit quality across the subsets and learns an optimal combination of the subset-specific fits.

This implementation allows the user to ensemble subset-specific fits which are trained using the same or different learning algorithms. The package uses the machine learning algorithm API provided by the **SuperLearner** *R* package. This user-friendly API currently provides a uniform interface to nearly 30 machine learning algorithms (e.g. `randomForest`, `gbm`) and allows the user to define custom algorithm wrappers. Each of the default algorithm wrappers can also be customized by specifying unique model parameters.

The user can either explicitly define which observations belong to each subset or simply specify the desired number of subsets. In the case of the latter, the subsets will be created randomly, with or without stratification. The package provides the ability to compute the V-fold cross-validation step as well as the model fitting across the subsets in parallel using the *R*-core **parallel** package.

The **subsemble** package will be released to CRAN soon and the current version of the package can be found here: `http://www.stat.berkeley.edu/~ledell/R/subsemble.tar.gz`

References

Sapp, S., van der Laan, M. J., and Canny, J. (2013). Subsemble: an ensemble method for combining subset-specific algorithm fits. *Journal of Applied Statistics*.
Article: `http://dx.doi.org/10.1080/02664763.2013.864263`
Tech report: `https://biostats.bepress.com/ucbbiostat/paper313`

van der Laan, M. J., Polley, E. C. and Hubbard, A. E. (2008). Super Learner. *Statistical Applications of Genetics and Molecular Biology*, 6, article 25.
Article: `http://dx.doi.org/10.2202/1544-6115.1309`
Tech report: `http://biostats.bepress.com/ucbbiostat/paper222`

Representing Model Ensembles in PMML

Tridivesh Jena[1,*] , Alex Guazzelli[1,] , Wen Ching Lin[1], Michael Zeller[1]

1. Zementis, Inc.
*Contact author: tridivesh.jena@zementis.com

Keywords: Predictive Analytics, PMML, Random Forest, R, Standards

PMML, the Predictive Model Markup Language, is the de facto standard to represent predictive analytics models [1,2,3]. It is currently supported by many of the leading commercial and open-source data mining systems, including R. With PMML, it is extremely easy to build a predictive model in one system (PMML producer) and exchange with another solution (PMML consumer) avoiding incompatibility problems and custom coding. The R **pmml** package was designed to export many popular predictive algorithms into the PMML standard [4,5]. Given the recent interest in ensemble models and their applicability to large datasets, it was only natural to add functionality to the R **pmml** package to convert these models into PMML.

The R **pmml** package is now able to export PMML for ensemble models via the `ada` and `randomForest` functions. In this presentation, we describe all the steps necessary to export random forest and stochastic boosting models from R into PMML and show how the PMML standard is capable of representing not only model ensembles but also any R specified treatments for missing and invalid values as well as outliers. Additional functions available to the data scientist through the R **pmml** package include the ability to perform data pre- and post-processing. By using the R **pmml** and the **pmmlTransformations** package [6,7], a scientist can read in input data in R, perform transformations on the input data, build the ensemble model and finally output the entire predictive workflow containing model and any pre or post-processing steps in PMML format. Once operationally deployed, the resulting PMML then generates predictions directly from raw input data.

Being able to export the entire model ensemble in PMML format, together with any data validation and transformation steps is remarkable, since it allows for these models to be moved to the operational environment without the need for any recoding. Once in PMML, models can be deployed in minutes and executed in a variety of Big Data platforms, including Hadoop, in-database or cloud computing.

References

[1] The Data Mining Group (DMG) website: www.dmg.org

[2] A. Guazzelli, W. Lin, T. Jena (2010). *PMML in Action: Unleashing the Power of Open Standards for Data Mining and Predictive Analytics (2nd Edition)*. CreateSpace (available on Amazon.com).

[3] A. Guazzelli (2010). What is PMML? Explore the power of predictive analytics and open standards. IBM developerWorks website.

[4] A. Guazzelli, M. Zeller, W. Lin, G. Williams (2009). **PMML: An Open Standard for Sharing Models**. *The R Journal*, Volume 1/1.

[5] The R pmml package: http://cran.r-project.org/web/packages/pmml/index.html

[6] T. Jena, A. Guazzelli, W. Lin, M. Zeller (2013). The R *pmmlTransformations* Package. *In Proceedings of the 19th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.*

[7] The R pmmlTransformations package: http://cran.r-project.org/web/packages/pmmlTransformations/index.html>

Handling conditional correlation GARCH models with the ccgarch2 package

Tomoaki Nakatani[1,⋆]

1. Department of Agricultural Economics, Hokkaido University
⋆Contact author: naktom2@gmail.com

Keywords: Multivariate GARCH, Conditional correlations, Volatility spillovers

The **ccgarch2** package is designed to provide functions for estimation and simulation of conditional correlation (CC-) GARCH models. It can estimate the Constant Conditional Correlation (Bollerslev, 1990), Dynamic Conditional Correlation (Engle, 2002) and corrected Dynamic Conditional Correlation (Aielli, 2013) GARCH models in a relatively large dimension. The package is also capable of simulating multivariate time series from the CC-GARCH models.

A couple of *R* packages are available for handling the major variants of the CC-GARCH models. An advantage of **ccgarch2** over the other existing packages is that it allows for modeling a multivariate counterpart of the univariate GARCH model in the conditional variance part. With this modeling strategy, volatility spillovers in the GARCH part can be incorporated into the model (Nakatani and Teräsvirta, 2009). In particular in the bivariate model, the estimating functions are constructed in such a way that it can capture negative volatility spillovers (Nakatani and Teräsvirta, 2008; Conrad and Karanasos, 2010).

ccgarch2 is a successor of the **ccgarch** package available at CRAN. In addition to inheriting many of the functionalities from its predecessor, **ccgarch2** improves user-interface by defining classes and associated methods. Numerical optimization of the likelihood function is now carried out by the `solnp()` function in the **Rsonlp** package, which makes it possible to impose non-linear restrictions on the parameters. These restrictions are necessary to keep the time-varying conditional covariance matrices positive definite as well as to keep the sequence of conditional variances stationary. Performance issues are improved by the use of *C* code.

In the presentation, basic usage of **ccgarch2** will be illustrated by analyzing real data. Extension to handling negative volatility spillovers in larger dimensional models will also be discussed.

References

Aielli, G. P. (2013). Dynamic conditional correlation: On properties and estimation. *Journal of Business & Economic Statistics 31*(3), 282–299.

Bollerslev, T. (1990). Modeling the coherence in short-run nominal exchange rates: A multivariate generalized ARCH approach. *Review of Economics and Statistics 72*, 498–505.

Conrad, C. and M. Karanasos (2010). Negative volatility spillovers in the unrestricted eccc-garch model. *Econometric Theory 26*, 838–862.

Engle, R. F. (2002). Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroskedasticity models. *Journal of Business and Economic Statistics 20*, 339–350.

Nakatani, T. and T. Teräsvirta (2008). Positivity constraints on the conditional variances in the family of conditional correlation GARCH models. *Finance Research Letters 5*, 88–95.

Nakatani, T. and T. Teräsvirta (2009). Testing for volatility interactions in the constant conditional correlation GARCH model. *Econometrics Journal 12*, 147–163.

Generating R reference classes in rClr with software reflection

Jean-Michel Perraud[1]

1. Commonwealth Scientific and Industrial Research Organisation, Australia
*Contact author: jean-michel.perraud@csiro.au

Keywords: interoperability, R reference classes, .NET, CLR

rClr (http://rclr.codeplex.com) is a package for *R* to access arbitrary *.NET* code executing on a Common Language Runtime (CLR) implementation: Microsoft's implementation or the multi-platform runtime Mono. **rClr** complements and in part re-uses the *R.NET* library (http://rdotnet.codeplex.com) that makes *R* programmatically accessible to *.NET* programmers, mostly but not only from the *C#* and *F#* languages. Work has been done in the growing *F#* community to expose R functionalities with *F#* idioms [3]. **rClr** should similarly leverage the most appropriate R idioms to expose CLR objects. The style of object oriented programming supported by the CLR makes R reference classes (see `help(ReferenceClasses)` in R) a natural candidate to access CLR objects from R. R reference classes have been used by at least two packages for interoperability: **RCpp** (Eddelbuettel [2]) and **rJavax** (Danenberg [1]). **rClr** will have an updated public release with support reference classes by or around the time of this conference. Given the closer similarities of the CLR with the *Java* runtime than *C++*, the design and implementation handling reference classes naturally shares similarities. In particular the capacity to reflect on software types/classes in the CLR and *Java* is useful to generate R reference classes, with a minimum of custom code. There are of course language difference between *R* and the CLR that remain and require choices to find a balance between the accessibility from *R* and the faithful representation of the CLR objects, properties and methods. Using R reference classes with **rClr** is demonstrated in a case study, the calibration of hydrological models. The general programming workflow and techniques that generate R reference classes is presented.

References

[1] Danenberg, P. (2011). rJavax, http://cran.r-project.org/web/packages/rJavax/index.html

[2] Eddelbuetel, D. (2013). Seamless R and C++ Integration with Rcpp. *Springer*, pp 236.

[3] Mansel, H. and contributors, (2014). An F# type provider for interoperating with R, https://github.com/BlueMountainCapital/FSharpRProvider

Predicting insurance industry churn

Pramod Kunju[1]

1. Founder and CEO, Dataversal Inc
*Contact author: pramod.kunju@dataversal.com

Keywords: Predictive analytics, Customer retention, Customer churn, Insurance churn, Big data

Customer retention is a critical success factor for insurance industry, much more so than for other industries. Losing one customer can result in lost revenue for several years. For example, an auto policy might last for 10 years, and a home policy for 30 years. An existing customer canceling a policy after 1 year of inception can result in lost revenues of 9 years and 29 years for auto and home policies, for example.

Customer retention in insurance industry is very complicated. For example, let's assume that a customer is identified as a churn prospect, and a call is made to her to assess how happy she is with her insurance. If she is really a churn prospect, the insurance company can try to address any concern she has, and proactively retain her. However, if she is a perfectly happy customer, that call can potentially trigger a churn as well, since calls from customer service agents are not something everyone looks forward to. So, it is crucial that any customer churn prediction is accurate.

The presentation will cover real-life examples of how *R* is instrumental in accurately modeling customer churn for insurance companies. The typical output of such a model-based analytics solution is a list generation. The list will contain existing customers who are likely to churn, and the reasons why they are churn candidates. The input data to *R* will come from internal transactional systems, and external sources such as social media. In fact, an increasing percentage of the input data is from a big data platform. The presentation will lay out a business approach, and an architectural pattern for solving the very important and real customer churn problem in insurance industry.

popKorn: An *R* package for Inference on Selected Populations

Vik Gopal[1,*], **Claudio Fuentes**[2], **Andrew Womack**[3]

1. National University of Singapore
2. Oregon State University
3. University of Florida
*Contact author: stavg@nus.edu.sg

Keywords: confidence intervals, selected means, selected populations, asymmetric intervals

Consider an experiment in which p independent populations π_i, with corresponding unknown means θ_i are available and suppose that for every $1 \leq i \leq p$, we can obtain a sample $X_{i1}, X_{i2}, \ldots, X_{in}$ from π_i. In this context, researchers are sometimes interested in selecting the populations that give the largest sample means as a result of the experiment, and to estimate the corresponding population means θ_i's. In [1], the authors present an approach to the problem and discuss how to construct confidence intervals for the mean of $k \geq 1$ selected populations, assuming the π_i are independent and normally distributed with a common variance σ^2. The *R* package **popKorn** implements this approach, which is based on minimisation of the coverage probability.

The **popKorn** package is the next generation of **kPop**, which was presented at useR! 2013 [2]. The new version contains a better implementation of functions for estimating the optimal asymmetric intervals to be used. This provides a practical yet formal tool for estimating (simultaneously) the mean of several selected populations.

In this talk, we shall motivate the problem, introduce this package, and demonstrate how its main functions can be used. We shall compare it with traditional methods, some of which do not account for the selection phase.

References

[1] Fuentes, C., G. Casella, and M. Wells (2013). Interval estimation for the mean of the selected populations. *Submitted*.

[2] Gopal, V. and C. Fuentes (2013). **kPop**: An *R* package for the interval estimation of the mean of selected populations. In *useR! 2013, The R User Conference (Albacete, Spain)*.

Approximate text matching with the stringdist package

Mark van der Loo

Statistics Netherlands
mark.vanderloo@gmail.com

Keywords: approximate string matching

Comparing text strings in terms of distance functions is a common and fundamental task in many statistical text-processing applications. Thus far, string distance functionality has been somewhat scattered around *R* and its extension packages, leaving users with inconsistent interfaces and encoding handling. The newly developed **stringdist** package is designed to offer an easy to use interface to several popular string distance algorithms which have been re-implemented in *C* for this purpose. The package offers distances based on counting *q*-grams, edit-based distances, and some lesser known heuristic distance functions [1].

For example, to compute the true Damerau-Levenshtein distance between two strings, one uses the `stringdist` function as follows.

```
> library(stringdist)
> stringdist('leia', 'leela', method='dl')
```

```
[1] 2
```

The distance of two corresponds to two edit operations necessary to turn `'leia'` into `'leela'`. For example: replace `'i'` with `'e'` and insert an `'l'`.

For approximate dictionary lookup one may use the `amatch` function:

```
> companions <- c('adric', 'ace', 'leia')
> amatch('leela', companions, method='dl', maxDist=2)
```

```
[1] 3
```

Here, `'leela'` matches with the third element of `companions` since it is both the closest match and its DL-distance is less than or equal to `maxDist`.

In this presentation I will review the string distance algorithms offered by the package, show how to apply them and point out some particularities related to special values (`NA`) and character encoding.

References

[1] M.P.J. van der Loo (2014). *The stringdist package for approximate string matching.* Accepted for publication in the R Journal.

Permutation Tests in Multidimensional Scaling

Patrick Mair[1*], **Jan De Leeuw**[2], **Ingwer Borg**[3]

1. Harvard University
2. University of California, Los Angeles (UCLA)
3. Leibniz Institute for the Social Sciences (GESIS)
*Contact author: mair@fas.harvard.edu

Keywords: Multidimensional Scaling, MDS, Unfolding, Permutation Tests

The **smacof** package [3] implements various methods of multidimensional scaling (MDS) such as metric and nonmetric MDS, spherical MDS, individual difference scaling, and unfolding for preference data [1]. MDS is a family of methods that optimally map proximity data of objects into distances between points of a multidimensional space with a given dimensionality (usually 2 or 3 dimensions). In **smacof** we use a majorization approach that minimizes the "Stress" target function.

MDS has been used heavily to model and to explore (dis)similarity data in psychology, the social sciences, and in market research. The Stress value of a p-dimensional MDS solution with n points is usually evaluated by going to tables listing the expected Stress of the "nullest of all null models" [2], i.e. the Stress value of random data for the (p, n) case. In this talk we present clearly sharper tests based on the Stress distribution that results from (matrix- or row-wise) randomly permuted dissimilarity data.

References

[1] Borg, I., P. J. F. Groenen, and P. Mair (2012). *Applied Multidimensional Scaling*. New York: Springer.

[2] Cliff, N. (1973). Scaling. *Annual Review of Psychology 25*, 473–506.

[3] De Leeuw, J. and P. Mair (2009). Multidimensional scaling using majorization: **SMACOF** in *R*. *Journal of Statistical Software 31*(3), 1–30.

Sensory discrimination testing with the sensR package

Rune Haubo Bojesen Christensen[1,*] , Per Bruun Brockhoff[1]

1. Technical University of Denmark, Applied Mathematics and Computer Science, Statistics Section.
*Contact author: rhbc@dtu.dk

Keywords: Thurstonian models, Discrimination tests, Signal detection theory, Binomial data analysis, sample size computations

Development of the *R*-package **sensR** [1] was motivated by problems in modeling sensory and signal detection theory (SDT) discrimination tasks. On a basic level **sensR** provides the means for standard discrimination testing, d-prime estimation and sample size estimation in sensory discrimination protocols such as the Triangle, Duo-Trio and Alternative-Forced-Choice tasks. Other commonly used testing protocols like A-not A (Yes-No), Same-Different, Paired preferences optionally with a no-preference option are also supported with estimation, profile likelihood and power estimation functions.

On a more advanced level **sensR** facilitates modeling of psychometric and sensory discrimination experiments with generalized linear models (GLMs) where special purpose link functions derived from the psychometric functions for the discrimination protocols directly relate the probability of success to the underlying Thurstonian d-prime [2]. Family objects to be used with `glm()` unleash the full power of GLMs facilitating in-depth modelling via the linear predictor, profile likelihood intervals of parameters with psychometric interpretations and much more.

The **sensR** package also facilitates analysis of ordinal ratings in the context of the Degree-of-Difference test and A-not A with sureness (yes-no rating) tests. This is partly obtained by interfacing the **ordinal** package [3]. Power estimation functions are also provided in the ordinal setting. In addition, support is provided for replicated tests via Beta-Binomial and chance-corrected Beta-Binomial models, ANOVA-like models for d-prime values with posthoc testing, as well as ROC curves and AUC facilities.

References

[1] Christensen, R.H.B and P.B Brockhoff (2014). **sensR** – An *R*-package for sensory discrimination testing, version 1.3-2, `http://www.cran.r-project.org/package=sensR/`.

[2] Brockhoff, P.B. and Christensen, R.H.B. (2010). Thurstonian models for sensory discrimination tests as generalized linear models. *Food Quality and Preference*, **21**, pp. 330-338.

[3] Christensen, R. H. B. (2014). **ordinal** - Regression Models for Ordinal Data, *R*-package version 2013.10-31, `http://www.cran.r-project.org/package=ordinal/`.

Rcpp11

Romain François*

R Enthusiasts
*Contact author: romain@r-enthusiasts.com

Keywords: *R*, *C++*, *C++11*, high performance

Extending R with compiled code is a great way to achieve good performance. Using C++ to rewrite critical parts has become a popular approach in R package development. For years, **Rcpp** [1] has defined the best bridge between *R* and *C++*.

In 2011, the *C++* standards committee released the *C++11* standard [2]. In 2013, major compilers (gcc, clang) have shipped versions of their tools that are feature complete. In April 2014, *R* version 3.1.0 was released with some special support for *C++11*. It is time to start using it. *C++11* is a major update over the previous version of the *C++* standard. It is a combination of various features that make *C++* a much better, more expressive language [3].

Rcpp11 is a complete rewrite of the `Rcpp` library, it takes advantage of *C++11* to make the user experience of combining *R* and *C++* even better and more future proof. **Rcpp11** was also an opportunity to review the code base of **Rcpp**, identify mistakes and fix them. **Rcpp11** is a smaller, cleaner implementation of the **Rcpp** api, written with *C++11* in mind.

During this talk, I will introduce **Rcpp11** with a few simple examples.

References

[1] Eddelbuettel, D. and R. François (2014). *Rcpp: Seamless R and C++ Integration*. R package version 0.11.1.

[2] ISO/IEC (2011). C++ 2011 standard document 14882:2011. ISO/IEC Standard Group for Information Technology / Programming Languages / C++.

[3] Stroustrup, B. (2013). *The C++ Programming Language* (4th ed.). Addison-Wesley.

Using SPRINT and parallelised functions for analysis of large data on multi-core Mac and HPC platforms

Eilidh Troup[1*], Thorsten Forster[2], Luis Cebamanos[1], Terence Sloan[1], Peter Ghazal[2]

1. Edinburgh Parallel Computing Centre, University of Edinburgh, Edinburgh, UK
2. Division of Pathway Medicine, University of Edinburgh Medical School, Edinburgh, UK
*Contact author: e.troup@epcc.ed.ac.uk

Keywords: HPC, Big Data, Genomics, SPRINT, Parallelisation

We here present computation performance (CPU time, memory requirements) increases we can obtain in the analysis of large biological (or other) data sets through use of the **SPRINT** package (www.r-sprint.org).

With the arrival of "big data" (microarrays, screens, next-generation sequencing) in the life sciences, standard analyses of these data for regular users of *R* now run into severe issues of computation time or computer memory. Many projects (including parallelisation efforts of the *R* core) offer *R* packages and functions that allow programming of solutions for large-scale analysis problems. However, these usually require familiarity with HPC programming as well as sufficient and funded time to employ, which is feasible for one-off analysis problems but impractical for common analysis methods.

To make High Performance Computing (HPC) solutions available to *R* users without HPC experience, we started development on the **SPRINT** package in 2008. It allows these users straightforward use of already implemented parallelised versions of many relevant *R* functions on multi-core Macs as well as large-scale clusters/HPC platforms like the UK's HECToR or ARCHER (we have also tested on Amazon Elastic Compute Cloud). In addition to addressing speed-critical problems, we also address memory-critical problems.

We will here introduce recent upgrades to **SPRINT**, discuss for regular *R* users how to use **SPRINT** and for users with HPC background how our parallelisation strategies are particularly aimed at problems that go beyond 'simple' task farming. We outline case examples for use of **SPRINT** as well as performance and limitations of our approach in context of biological high-throughput data (although most individual functions are generically usable for other larger data sets).

Based on our needs and those we established in *R* user surveys, we currently support parallelised versions [1] of original [2] functions (our function names add prefix 'p', apart from `pmaxt`, which is based on `mt.maxT`) that are essential in clustering, classification and non-parametric statistics when applied to very large data sets: `pstringdistmatrix`, `pboot`, `papply`, `pcor`, `ppam`, `prandomForest`, `pmaxt`, `pRP`, `psvm`.

References

[1] Publications of our function implementations can be found on www.r-sprint.org -> Publications

[2] Source citations for these packages can be found on www.r-sprint.org -> Overview and R functions

TestR: generating unit tests for R internals

Roman Tsegelskyi*, Jan Vitek

Purdue University, West Lafayette, IN, USA
*Contact author: rtsegels@purdue.edu

Keywords: testing, R implementation

When implementation a programming language like *R*, one of the biggest challenges is ensuring correctness of the many runtime functions that are part of Rs environment. For example, most of the extensively used operations like arithmetics are implemented in *C*. Overall current *GNU R* relies on 695 internal functions implemented in *C*. Constructing test cases for every function separately does not seem practical. We attack this problem with an automated method using the existing *GNU R* test suite. This test suite emerged with the evolution of *R*, but two main problems arise when trying to use it for testing a new implementation of *R* such as *FastR*. Firstly, it is not possible to tests functions separately, thus running the whole suite requires full implementation of all internal functions. Another problem is that errors in this test suite are hard to interpret. Where did the error come from exactly? How do we localize it?

This project presents an approach to generating a concise test suite that can be used for unit testing Rs internals. Our goal was to create a test suite for in internal functions that can be used to test each function separately, while maintaining the same code coverage level as regression test suite does. Our approach is based on instrumenting the *R* VM to capture calls to built in/special functions and generates test cases based on captured information. As a lot of those calls are redundant in terms of code coverage, we are also filtering them based, on the impact on code coverage.

We will explain the status of the project, and current results compared to full *R* test suite. Currently, we were able to generate tests that cover more that 80% of what *R* test suite covers while shrinking test suite size to only 3000 function calls. We will provide some thoughts about how this can be used for creating test suites for *R* packages and where to go from there.

Version 2 of the R Commander

John Fox[1*], **Milan Bouchet-Valat**[2]

1. McMaster University, Hamilton, Ontario, Canada
2. OSC-CNRS & Sciences Po, Paris, France
*Contact author: jfox@mcmaster.ca

Keywords: R, graphical user interface (GUI), Tcl/Tk, reproducible research

The R Commander provides a graphical user interface (GUI) for *R* based on standard menus, buttons, and dialog boxes. Implemented in the **Rcmdr** package [4], the R Commander uses *Tcl/Tk* widgets via the **tcltk** package [6], which is included in the standard *R* distribution. As a consequence, the R Commander is simple to install and use on all of the computing platforms on which *R* commonly runs — Windows, Mac OS X, and Linux/Unix. The R Commander also uses the message-translation facilities in *R* [7], and it has been translated into a number of different languages.

Since its introduction about 10 years ago [2], the R Commander has evolved substantially. An early innovation was the capacity to accommodate plug-in packages [3], which extend the capabilities of the R Commander, and more than 30 of these are currently available on CRAN. Recently, we introduced version 2 of the R Commander, which incorporates a variety of interface improvements, including tabbed dialogs; a more consistent appearance employing themed *Tcl/Tk* widgets (thanks to the **tcltk2** package, 5); *Apply* buttons that reopen dialogs in their current state after executing *R* commands; and the automatic generation of reports using editable R Markdown and LaTeX documents, via the **markdown** [1] and **knitr** [8] packages. These documents compile respectively to HTML and PDF files at the press of a button in the R Commander interface.

The overall goal of the R Commander remains largely unchanged, however: To provide a simple, intuitive, extensible GUI to *R*, primarily for basic and occasional use of *R*, such as in introductory and intermediate-level statistics courses, as an open-source alternative to GUI-based commercial software such as *SPSS*.

References

[1] Allaire, J., J. Horner, V. Marti, and N. Porte (2014). *markdown: Markdown rendering for R*. R package version 0.6.4.

[2] Fox, J. (2005). The R Commander: A basic statistics graphical user interface to R. *Journal of Statistical Software 14*(9), 1–42.

[3] Fox, J. (2007). Extending the R Commander by "plug-in" packages. *R News 7*(3), 46–52.

[4] Fox, J. and M. Bouchet-Valat (2013). *Rcmdr: R Commander*. R package version 2.0-3.

[5] Grosjean, P. (2013). *SciViews-R: A GUI API for R*. MONS, Belgium: UMONS.

[6] R Core Team (2013). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

[7] Ripley, B. D. (2005). Internationalization features of R 2.1.0. *R News 5*(1), 2–7.

[8] Xie, Y. (2013). *Dynamic Documents with R and knitr*. New York: Chapman and Hall/CRC.

Rapid Prototyping with R/Shiny at McKinsey: A New Way of Delivering Value for Our Clients

Aaron Horowitz[1*] , Frank Kroell[1]

1. McKinsey & Company
*Contact author: aaron_horowitz@mckinsey.com

Keywords: prototype, shiny, rCharts, visualization, analytics consulting, tool development

McKinsey & Company is a global management consulting firm, serving 90% of the world's largest companies. As part of the growing analytics team at McKinsey, we strive to ensure that leaders at these organizations recognize the importance and value advanced analytics can make. Our work frequently entails building analytics teams, piloting new methodologies, and finding new and innovative ways to serve our current and future clients. Our clients and colleagues are unfamiliar with advanced analytics, but increasingly understand its importance. With the help of **Shiny**, other packages and even externally integrated software, we now create rapid analytic prototypes that change the types of end-products we offer, and the ways in which we interact internally and externally.

We plan to discuss the means by which these prototypes solve multiple problems we face in delivering advanced statistical analysis including 1)demystifying the analytics "black box" 2)Productizing rapid tool development so it can be made by statistical professionals and 3)offering end-products we can pass off to enterprise software developers for full-scale applications. We'll then demonstrate products we've created which go well-beyond most toy examples, and are full-fledged applications. Finally, we'll discuss our process for doing so at speed, thanks to a custom-built framework we share and develop with each new product we create.

References

[1] RStudio, Inc. (2014). Shiny home page, http://rstudio.com/shiny/.

[2] McKinsey & Company (2014). McKinsey & Company home page http://www.mckinsey.com/

Hierarchical Bayesian Estimation - Consumers' Change in Recognition and Behavior toward Advertisements by Elaboration Likelihood Model

Fumiyo N. Kondo[1,*] , Satoshi Nakano[1]

1. University of Tsukuba
*Contact author: kondo@sk.tsukuba.ac.jp

Keywords: Elaboration likelihood model, Identification of information processing routes, Identification of initial states, Single source data, Hierarchical Bayesian model

This paper proposes a hierarchical Bayesian Binomial logit model to verify the effectiveness of identification of information processing route via the elaboration likelihood model (ELM). The proposed model evaluates the influence of changes in recognition and behavior of consumers by cue information and the number of advertisement contacts. The concept of ELM leads to the identification of the two information processing routes, but the proposed model identifies also three initial states of advertisement targets. We examined the effectiveness of segment classification by estimating different parameters of 6 segments through a unified model. Analyses were conducted by using two sets of single-source data of two new products in different categories and two data sets of different elapsed time for one of the new products. Four models of different combinations of information processing route/initial states were compared to examine the validity of our proposed model. Our proposed model with six segments were selected as the best model by the information criterion, DIC. This shows that there are statistically significant differences on recognition and behavior changes among different groups in terms of the information processing routes and the initial states.

References

[1] Batra, R., and Ray, M. L. (1985), "How advertising works at contact", Psychological processes and advertising effects, pp.13-43.
[2] Bitner, M. J., and Obermiller, C. (1985), "The elaboration likelihood model Limitation and extensions inmarketing",Advances in Consumer Research, 12, pp.420-425.
[3] Braverman, J. (2008), "Testimonials Versus Informational Persuasive Messages The Moderating Effect of Delivery Mode and Personal Involvement", Communication Research, 35 (5), pp.666-694.
[4] Cacioppo, J. T. and Petty, R. E. (1979), "The Effects of Message Repetition and Position on Cognitive Responses, Recall, and Persuasion," Journal of Personality and Social Psychology, 37 (9), pp.7–109.
[5] Kim, J. U., Kim, W. J., and Park, S. C. (2010), "Consumer perceptions on web advertisements and motivation factors to purchase in the online shopping", Computers in human behavior, 26(5), pp.1208-1222.
[6] Te'eni-Harari, T., Lampert, S. I., and Lehman-Wilzig, S. (2007), "Information processing of advertising among young people: the elaboration likelihood model as applied to youth", Journal of Advertising Research, 47(3), pp.326-340.

rapport: a report templating system in *R*

Gergely Daroczi[1,2,⋆] **and Aleksandar Blagotic**

1. Founder of Easystats Ltd, United Kingdom
2. PhD candidate at Corvinus University of Budapest, Hungary

Contact authors: ⋆ daroczig@rapporter.net

Keywords: literate programming, reproducible research, markdown, reports

rapport is an *R* package aimed at creating reproducible and reusable statistical report templates. The goal of this talk is to discuss **rapport**'s unique approach to report reproducibility through a blend of literate programming and template-based reporting, that allows the user to replicate his analysis against any suitable dataset, by means of a simple *R* command.

We will discuss the usage of *YAML*-flavoured inputs of the template's header that one can match against the dataset variables or custom *R* objects in order to produce a report, then we will focus on how **rapport** uses **brew**-style tags in order to evaluate *R* chunks and how is the output of the evaluated expressions automatically converted to *markdown* with the **pander** package – to be transformed to various document formats with *pandoc*.

The **pander** backend will also be highlighted, which provides a robust cache engine, applies a uniform look to all the **graphics**, **lattice** or **ggplot2** plots and permits the manipulation of template parts via *R* control structures.

The talk will end with an overview on the (dis)similarities with packages like **brew**, **Sweave** and **knitr**.

References

Aleksandar Blagotic and Gergely Daroczi (2013). **rapport**: a report templating system. http://cran.r-project.org/package=rapport

Friedrich Leisch (2002). Sweave: Dynamic generation of statistical reports using literate data analysis. In Wolfgang Hrdle and Bernd Rnz (editors). *Compstat 2002 - Proceedings in Computational Statistics*.

Gergely Daroczi (2013). **pander**: An R Pandoc Writer. http://cran.r-project.org/package=pander

Gergely Daroczi (2014). **rapportools**: Miscellaneous (stats) helper functions with sane defaults for reporting. http://cran.r-project.org/package=rapportools

Jeffrey Horner (2011). brew: Templating Framework for Report Generation. http://CRAN.R-project.org/package=brew

Yihui Xie (2013). knitr: A general-purpose package for dynamic report generation in R. http://CRAN.R-project.org/package=knitr

RForcecom: an R package which provides a connection to Force.com and Salesforce.com

Takekatsu Hiramura*

*Contact author: thira@plavox.info

Keywords: Cloud computing, Software-as-a-Service, CRM, Web API, Text mining

Nowadays, cloud computing systems are becoming more common, especially the SaaS (Software-as-a-Service) enterprise system. One of the key features of an enterprise system is the CRM (Customer Relationship Management) area. Salesforce.com [1] is a well-known SaaS-based CRM service [2]. Companies using these kinds of SaaS services need to retrieve datasets from them, in order to run their statistical analyses, since the data is stored inside cloud servers.

I have developed an R-Package called "**RForcecom"** [3], which provides a connection to Salesforce.com and Force.com via REST API. **RForcecom** has various features to which enable datasets to exchange information with Salesforce.com and retrieve a dataset, delete, create, update and upsert records in Salesforce.com. In addition, it has an SOQL (Salesforce Object Query Language) and an SOSL (Salesforce Object Search Language) interface to query and search records.

In this presentation, I'm going to illustrate with the aid of an example relating to the analysis of CRM data using **RForcecom**. My example contains some procedures: first, retrieving a dataset from Salesforce.com using **RForcecom**; second, extracting high-frequency keywords and buzz-words using a Natural Language Processing (NLP) algorithm; and finally, visualizing the words as a word cloud, which will assist corporate staff, managers and executives to grasp their consumers' voice.

References

[1] Salesforce.com, Inc. Salesforce 1 Service Cloud, http://www.salesforce.com/.

[2] Louis Columbus (2013), CRM Market Share Update: 40% Of CRM Systems Sold Are SaaS-Based. *Forbes*, http://www.forbes.com/sites/louiscolumbus/2013/04/26/2013-crm-market-share-update-40-of-crm-systems-sold-are-saas-based/.

[3] Takekatsu Hiramura (2012). RForcecom, http://rforcecom.plavox.info/.

DataCamp: online interactive learning platform for R

Andreas Alfons[1,2], Jonathan Cornelissen[2,3,*], Dieter De Mesmaeker[2], Bram Jans[2], Albert Jorissen[2], Martijn Theuwissen[2]

1. Erasmus Universiteit Rotterdam
2. DataCamp
3. Vrije Universiteit Brussel
*Contact author: Jonathan@datacamp.com

Keywords: Learning, interactivity, web-based R apps.

DataCamp [1] is an online interactive learning platform for *R*. It offers learners the ability to work with *R* in their browser and receive instant feedback on their statistical analysis (through automated correction). Furthermore, it offers teachers and trainers the possibility to create interactive online courses themselves using *R Markdown [2]* and using the same syntax as the **slidify** package [3].

To enable the use of R on the web, DataCamp leverages the functionality of the **Rserve** package with enhanced security provided by the **RAppArmor** package. In terms of web technologies, DataCamp uses open-source frameworks like *AngularJS*, *NodeJS* and *Ruby on Rails*. We briefly discuss the advantages and disadvantages of these technologies to bring R to the web and how they have been used in both DataCamp and our side-projects: Rdocumentation.org and R-Fiddle.org.

The creation of interactive R exercises is easy and transparent through the use of *R Markdown* and the structure imposed by the **slidify** package. We discuss step-by-step how to create these interactive exercises with a focus on how the interactivity is achieved through Submission Correctness Tests, aided by the **datacamp** R package [4]. After the talk, you should be able to create your own interactive course on DataCamp.

Students learning through a web-based interface has many advantages. One key advantage is that large amount of data can be collected on how students are learning. It's our goal to collect useful information that allows teachers to improve their courses and grade their students in a data-driven way. By partnering with the *Coursera* courses of Dr. Mine Çetinkaya-Rundel [5] and Prof. Dr. Eric Zivot [6], in the last 2 months DataCamp has taught over 30.000 students who submitted over 1 million exercises about basic R, basic statistics and computational finance. We discuss the key insights on online learning for R based on this data.

References

[1] http://www.DataCamp.com/.

[2] http://rmarkdown.rstudio.com/.

[3] http://slidify.github.io/.

[4] https://github.com/Data-Camp/datacamp and https://github.com/Data-Camp/datacampSCT.

[5] https://www.coursera.org/course/statistics and https://www.datacamp.com/courses/data-analysis-and-statistical-inference_mine-cetinkaya-rundel-by-datacamp

[6] https://www.coursera.org/course/compfinance and https://www.datacamp.com/courses/introduction-to-computational-finance-and-financial-econometrics

Visualizing Lack of Fit in Complex Regression Models: Adding Partial Residuals to Effect Displays

John Fox[1,*], **Sanford Weisberg**[2]

1. McMaster University, Hamilton, Ontario, Canada
2. University of Minnesota, Minneapolis, Minnesota, U.S.A.
*Contact author: jfox@mcmaster.ca

Keywords: Regression graphics, Statistical modeling, Nonlinearity

Effect displays, introduced by Fox [2] for generalized linear models, visualize the response surface of complex regression models by conditioning and slicing the surface, producing a sequence of 2D line graphs representing the response surface. These displays are implemented in the **effects** package for *R* [3, 4].

Partial-residual plots, also called *component-plus-residual plots*, visualize lack of fit, traditionally in relatively simple regression models. The properties of these graphs were systematically explored by Cook [1].

We combine partial residuals with effect displays to visualize lack of fit in complex regression models, plotting residuals from a model around 2D slices of the fitted response surface. Employing Cook's fundamental results, we discuss and illustrate both the strengths and limitations of the resulting graphs.

This extension to effect displays is implemented for generalized linear models of arbitrary complexity in the development version of the **effects** package.

References

[1] Cook, R. D. (1993). Exploring partial residual plots. *Technometrics 35*, 351–362.

[2] Fox, J. (1987). Effect displays for generalized linear models. In C. C. Clogg (Ed.), *Sociological Methodology 1987 (Volume 17)*, pp. 347–361. Washington, D. C.: American Sociological Association.

[3] Fox, J. (2003). Effect displays in R for generalised linear models. *Journal of Statistical Software 8*(15), 1–27.

[4] Fox, J., S. Weisberg, M. Friendly, and J. Hong (2014). *effects: Effect Displays for Linear, Generalized Linear, Multinomial-Logit, Proportional-Odds Logit Models and Mixed-Effects Models*. R package version 3.0-0.

Automated Business Reporting with R

Zhengying (Doro) Lou[1*], Van Morgan

1. ZestFinance

*Contact author: dzl@zestfinance.com

Keywords: automated reporting, RMySQL, xlsx, Markdown, knitr, ggplot2

In ZestFinance, we have developed a fully automated process with *R* to improve the quality and efficiency of business reporting. This process uses **RMySQL** to retrieve raw data from our database, summarizes and structures the data into report ready format in *R*, and then uses **xlsx** to write data into Excel template to create the final report. The *R* script is run automatically by a cron job and the final report is sent to users by email. Additionally, **Markdown** and **knitr** are used to publish **ggplot2** objects on an internal website that is updated on a daily basis. The automated process improved report quality and reduced analyst staffing needs by 0.5 FTE (full-time equivalent).

The Arborist: a Scalable Decision Tree Implementation

Mark Seligman[1,⋆]

1. Rapidics LLC
⋆Contact author: mseligman@rapidics.com

Keywords: machine learning, high performance, big data

The Arborist is an implementation of the Random Forest algorithm (Breiman 2001) invocable through an *R* package interface. The software is tailored for high performance, with execution time scaling linearly in both predictor and row count. Both regression and categorical cases are supported, with no limit on the number of factor levels in either the response or the predictors. In addition to standard features, **The Arborist** offers quantile regression, missing-value handling, and automatic resampling.

The Arborist's interface with *R* employs the `Rcpp` template extension, allowing the software to be invoked by package. Specialized GPU versions are under development, but a general-purpose, multicore version is being made available under MPL-2 license.

We describe the organization of the software and compare performance with other implementations of the Random Forest algorithm.

References

[1] Breiman, L. (2001). Title of an article. *Machine Learning 45*, 5–32.

[2] Breiman, L. and A. Cutler (2006). Random forests. `http://www.stat.berkeley.edu/~breiman/RandomForests/`.

[3] Liaw, A. and M. Wiener (2002). Classification and regression by randomforest. *R News 2*(3), 18–22.

Categorical Data Visualization Reordered

Alexander Pilhöfer[1,*], **Antony Unwin**[1]

1. Department of Computer Oriented Statistics and Data Analysis, Institute of Mathematics, University of Augsburg, Germany
*Contact author: alexander.pilhoefer@math.uni-augsburg.de

Keywords: Categorical Data, Reordering, Visualization

The analysis of categorical data can often be challenging, especially when multiple nominal variables are involved. The major problem is the lack of naturally defined distances and orders. Visualisations reveal information in datasets, especially when they are backed up with flexible ordering options.

This talk presents *joint reordering*, an approach for emphasising associations between categorical variables: Each variable is reordered in a way that is optimal given the orderings of the other variables. The optimality criterion used is the *Bertin Classification Criterion* (BCC) [1]. Two algorithms are presented, the general BCC reordering algorithm and a stepwise procedure tailored for the optimization of CPCP plots [2], an extension of Parallel Coordinates for categorical data.

The main example discussed makes use of the big US airport dataset that was the subject of the 2009 JSM Data Expo. Results are visualised using both fluctuation diagrams and CPCP plots.

The reordering and visualisation techniques described are available in the *R* package **extracat** [2].

References

[1] Pilhöfer, A., A. Gribov, and A. Unwin (2012). Comparing clusterings using Bertin's idea. *Visualization and Computer Graphics, IEEE Transactions on 18*(12), 2506–2515.

[2] Pilhöfer, A. and A. Unwin (2013, 5). New approaches in visualization of categorical data: R package extracat. *Journal of Statistical Software 53*(7), 1–25.

Adaptive Resampling in a Parallel World

Max Kuhn[1⋆]

1. Pfizer Global R&D, Groton CT
⋆Contact author: max.kuhn@pfizer.com

Keywords: Machine Learning, Classification, Regression, Parameter Tuning

Many predictive models require parameter tuning. For example, a classification tree requires the user to specify the depth of the tree. This type of "meta parameter" or "tuning parameter" cannot be estimated directly from the training data. Resampling (e.g. cross-validation or the bootstrap) is a common method for finding reasonable values of these parameters (Kuhn and Johnson, 2013). Suppose B resamples are used with M candidate values of the tuning parameters. This can quickly increase the computational complexity of the task.

Some of the M models could be disregarded early in the resampling process due to poor performance. Maron and Moore (1997) and Shen *el at* (2011) describe methods to adaptively filter which models are evaluated during resampling and reducing the total number of model fits. However, model parameter tuning is an "embarrassingly parallel" task; model fits can be calculated across multiple cores or machines to reduce the total training time. With the availability of parallel processing is it still advantageous to adaptively resample?

This talk will briefly describe adaptive resampling methods and characterize their effectiveness using parallel processing via simulations.

References

Kuhn, M. and Johnson, K. (2013). *Applied Predictive Modeling*. Springer Verlag.

Maron, O. and Moore, A. (1997). The Racing Algorithm: Model selection for lazy learners. In D. Aha (Ed.), *Lazy Learning*, 193225. Springer.

Shen, H., Welch, W. J. and Hughes–Oliver, J. M. (2011). Efficient, adaptive cross–validation for tuning and comparing models, with application to drug discovery. *The Annals of Applied Statistics*, 5(4), 26682687.

Knitr Ninja

Yihui Xie[1,*]

1. RStudio, Inc

*Contact author: xie@yihui.name

Keywords: knitr, Dynamic document, Quick reporting, Package vignette, Tricks

The **knitr** package (Xie 2013a) is a general-purpose tool for dynamic report generation in R. In this talk, we first introduce the basic idea of dynamic documents (Xie 2013b) using simple LaTeX and Markdown examples. Then we show some less well-known applications of **knitr**, including

- `spin()` for the lazy and impatient: generate a report from an R script
- some RPubs gems (`http://rpubs.com`)
- web applications such as the Rcpp gallery and Vistat
- R package vignettes using **knitr**, and the Gangam style (Docco style)
- how the knitr book was written using LyX + **knitr**
- boosting Markdown using Pandoc (yes, you can have Word as desired)
- AsciiDoc for O'Reilly books
- language engines: executing shell scripts, Python, and Julia code via **knitr**

We hope these demos can uncover some potential of **knitr** and R, especially when they are integrated with third-party software packages.

References

Xie, Yihui. 2013a. *knitr: A General-Purpose Package for Dynamic Report Generation in R*. `http://CRAN.R-project.org/package=knitr`.

———. 2013b. *Dynamic Documents with R and Knitr*. Chapman; Hall/CRC. `http://yihui.name/knitr/`.

ctsmr package - Continuous Time Stochastic Modelling in R

Rune Juhl*, Peder Bacher, Henrik Madsen

Department of Applied Mathematics and Computer Science, Technical University of Denmark
*Contact author: ruju@dtu.dk

Keywords: Stochastic Differential Equations (SDE), grey-box modelling, continuous time modelling, nonlinear and nonstationary modelling, maximum likelihood estimation

ctsmr (`http://ctsm.info`) is a new *R* package providing a framework for identifying and estimating stochastic grey-box models. **ctsmr** is the continuation of CTSM[4]. A grey-box model consists of a set of stochastic differential equations coupled with a set of discrete time observation equations, which describe the dynamics of a physical system and how it is observed. The grey-box models can include both system and measurement noise, and both nonlinear and nonstationary systems can be modelled using **ctsmr**.

The estimation is based on one or more independent datasets using maximum likelihood (or maximum a posteriori estimation) and Kalman filtering. **ctsmr** automatically distinguishes between linear or nonlinear stochastic state space formulations and applies either the regular or extended Kalman filter.

A model in **ctsmr** is built around `ReferenceClasses`. State and measurement equations are added sequentially forming a complete model. The model is subsequently translated into Fortran and compiled for speed. The likelihood function is usually optimized using a quasi Newton method using finite difference approximations of the gradient. The gradient is computed in parallel using *OpenMP*.

CTSM and **ctsmr** has been successfully applied to a range of applications. A few examples are: heat dynamics of thermal systems (walls and buildings[1], BIPV[5]), solar and wind power forecasting[3], solar-activity[7], pharmacokinetic/pharmacodynamic[2] and rainfall-runoff forecasting[6].

The upcoming version of **ctsmr** includes sensitivity analysis for computing the gradients and a mixed-effects extension.

References

[1] Bacher, P. and H. Madsen (2011). Identifying suitable models for the heat dynamics of buildings. *Energy & Buildings 43*(7), 1511–1522.

[2] Hansen, A. H., A. K. Duun-Henriksen, R. Juhl, S. Schmidt, K. Nørgaard, J. B. Jørgensen, and H. Madsen (2014, March). Predicting plasma glucose from interstitial glucose observations using bayesian methods. *Journal of Diabetes Science and Technology 8*(2), 321–330.

[3] Iversen, E. B., J. M. Morales, J. K. Møller, and H. Madsen (2013, October). Probabilistic forecasts of solar irradiance by stochastic differential equations. *arXiv:1310.6904 [stat]*.

[4] Kristensen, N. R., H. Madsen, and S. B. Jørgensen (2004, February). Parameter estimation in stochastic grey-box models. *Automatica 40*(2), 225–237.

[5] Lodi, C., P. Bacher, J. Cipriano, and H. Madsen (2012, July). Modelling the heat dynamics of a monitored test reference environment for building integrated photovoltaic systems using stochastic differential equations. *Energy and Buildings 50*, 273–281.

[6] Löwe, R., P. S. Mikkelsen, and H. Madsen (2014, March). Stochastic rainfall-runoff forecasting: parameter estimation, multi-step prediction, and evaluation of overflow risk. *Stochastic Environmental Research and Risk Assessment 28*(3), 505–516.

[7] Vio, R., P. Rebusco, P. Andreani, H. Madsen, and R. V. Overgaard (2006, June). Stochastic modeling of kHz quasi-periodic oscillation light curves. *Astronomy and Astrophysics 452*(2), 383–386.

translateR – A cloud based translator for SPSS and SAS Code

Oliver Bracht

*Contact author: oliver.bracht@eoda.de

Keywords: Code Translation, Code Migration, Transcompiler, SPSS, SAS

One of the major hurdles for companies and research institutions that are already using statistical software and are willing to move to R is the migration of the existing scripts. In many cases, these scripts contain thousands and thousands lines of code and have grown dynamical over years and decades. Given this, it is time consuming and expensive to prove that R is capable to solve the specific analytic processes with a similar outcome.

translateR is an approach to ease the migration of code and scripts to R. It consists of a cloud based translation engine and an R-package. The translation engine takes an SPSS or SAS script and returns a script that does the same in R. The translation is not “literally”, but in functions that are provided by the translateR R-package. These functions resemble their SPSS/SAS counterparts in naming and input parameters. Additionally, the package provides a dataframe-like class which can keep attributes like user defined missing values or value labels with negative or non-consecutive values.

translateR can be used for migration and proof-of-concepts projects, as a starting point to learn R or as a convenience layer. In addition, its data-class can be used standalone by R users who want to benefit from enhanced data attributes.

R as a PaaS cloud computing service for Computational Intelligence tasks

Jos M. Bentez⋆, Lala S. Riza, C. Bergmeir, D. Peralta, F. Herrera

Dept. Computer Science and Artificial Intelligence, CITIC-UGR, iMUDS, Universidad de Granada, Granada, Spain
⋆Contact author: J.M.Benitez@decsai.ugr.es

Keywords: Cloud Computing, Big Data, Computational Intelligence

Computational Intelligence (CI) is a field within Artificial Intelligence that has drawn the attention of a numerous community of researchers and practitioners. This field is concerned with computational methods inspired on nature and language and targeted for complex real-world problems for which traditional approaches are innefective or infeasible. While a number of different techniques are included within CI, a special effort is made towards their fusion and hybridization looking for systems that gather the stong points of the original components. In particular, CI hosts artificial neural networks, evolutionary algorithms, fuzzy systems and rough sets. Our group is actively involved in developing *R* packages for different heavily used CI techniques: e.g. **RSNNS**, **Rmalschains**, **frbs** and **RoughSets**.

On the other hand, Cloud Computing has emerged along last years as a new computing paradigm and is steadily gaining traction. It represents an attractive alternative for short usages of supercomputing facilities, particularly boosted by the Big Data push.

We present the advances in a new project whose objective is to develop a Platform as a Service (PaaS) in a cloud computing platform —OpenNebula is used as IaaS— which offers the computing processing capabilities of R for Big Data. This software allows the development and easy scaling of data analysis and modeling tasks based on CI techniques.

References

[1] Christoph Bergmeir and Jos M. Bentez. Neural networks in r using the stuttgart neural network simulator: Snns. *Journal of Statistical Software*, 46:7:1–26, 2012.

[2] Christoph Bergmeir, Daniel Molina, and Jos M. Bentez. *Rmalschains: Continuous Optimization using Memetic Algorithms with Local Search Chains in R*, 2012.

[3] Rajkumar Buyya, James Broberg, and Andrzej Goscinski. *Cloud Computing. Principles and Paradigms*. John Wiley & Sons, 2011.

[4] R. Moreno-Vozmediano, R.S. Montero, and I.M. Llorente. Iaas cloud architecture: From virtualized datacenters to federated cloud infrastructures. *IEEE Computer*, 45:65–72, 2012.

[5] Lala S. Riza, Christoph Bergmeir, Francisco Herrera, and Jos M. Bentez. *frbs: Fuzzy Rule-based Systems for Classification and Regression Tasks*, 2012.

[6] Lala S. Riza, Andrzej Janusz, Chirs Cornelis, Francisco Herrera, Dominik Slezak, and Jos M. Bentez. *RoughSets: Data Analysis using Rough Set and Fuzzy Rough Set Theories*, 2014.

waveCUDA: an R package for performing CUDA-accelerated wavelet analysis

Julian Waton[1,⋆], **Dr Emma McCoy**[1]

1. Imperial College London, Department of Mathematics
⋆Contact author: julian.waton08@imperial.ac.uk

Keywords: Wavelet, Lifting, CUDA, GPU, Parallelisation

We introduce a new *R* package **waveCUDA** that performs *CUDA*-accelerated Discrete Wavelet Transforms (DWTs). Whilst there are existing libraries available in *R* for wavelet analysis, notably **waveslim**, **wavethresh** and **wmtsa**, these do not use the GPU (Graphics Processing Unit) for accelerated computation. GPUs are highly parallel by construction, and *CUDA* allows programmers to take advantage of this using explicit parallel programming.[1] There are already some *R* packages that do parallel computing using GPUs such as **gputools** and **HiPLARM**.

The DWT, with the exception of the Haar transform, is not parallelisable using the traditional pyramid algorithm. However, Wim Sweldens developed the Wavelet Lifting Scheme which is both parallelisable and allows for calculations to be made in-place in memory.[2] Some authors have successfully written *CUDA*-accelerated DWTs (e.g. [3]) - but **waveCUDA** will provide a suite of functions for *CUDA*-accelerated wavelet analysis in R. This package is in development and will gain features over the course of time. At the time of writing, we have implemented transforms with filter lengths up to 4, with significant speed-ups over serial *C* code.

References

[1] Nvidia (2011). NVIDIA CUDA programming guide.

[2] Sweldens, W. (1996, April). The Lifting Scheme: A Custom-Design Construction of Biorthogonal Wavelets. *Applied and Computational Harmonic Analysis 3*(2), 186–200.

[3] van der Laan, W. (2011). Accelerating wavelet lifting on graphics hardware using CUDA. *IEEE transactions on parallel and distributed systems 22*(1), 132–146.

Opportunities through the use of Open-Street-Map data in social sciences

Jan-Philipp Kolb[1]

1. GESIS - Leibniz Institute for the Social Sciences
Survey Design and Methodology
P.O. Box 122155
68072 Mannheim

Keywords: Open-Street-Map, Georeferencing, spatial information

A high portion of information can be related to place. But the processment of this relating information has long been difficult due to lacking data sources and processing power. Numerous *R*-packages have been developed recently, which provide tools to read, visualize, and analyze spatial data.

In social sciences it is increasingly common to deal with observational data in their spatial context. The analysis of geographic locations and their attributes is of growing importance for research in social sciences and humanities.

In the presentation these two areas are combined. The *R*-packages **sp**, **ggmap** and **rgeos** are employed to process spatial informations. Examples are applied to highlight the additional benefit of geographic associations. The *R*-package **osmar** enables the Use*R* to benefit from the high level informations of the Open-Street-Map project. The derived data is for example used to describe geographic disparities in the selection of pre-school establishments. The underlying hypothesis is that the neighborhood social context plays a big role in this coherence.

The aim of this work is to highlight possibilities of using relevant information on the surroundings to provide a more accurate picture of social phenomena.

References

[1] Bivand, R. S., E. J. Pebesma, and V. Gómez-Rubio (2008). *Applied Spatial Data Analysis with R*. New York: Springer.

[2] Croner, C. M., J. Sperling, and F. R. Broome (1996). Geographic information systems (gis): new perspectives in understanding human health and environmental relationships. *Statistics in Medicine 15*(18), 1961–1977.

[3] Dearwent, S. M., R. R. Jacobs, and J. B. Halbert (2001). Locational uncertainty in georeferencing public health datasets. *Journal of Exposure Analysis & Environmental Epidemiology 11*(4).

[4] Eugster, M. J. and T. Schlesinger (2012). osmar: Openstreetmap and r.

[5] Hill, L. L. (2009). *Georeferencing: The geographic associations of information*. MIT Press.

Monitoring Patients with Ongoing Reduced Kidney Function

A. Jonathan R. Godfrey[1,*], **Greig K. G. Russell**[2], **Brigid D. Betz-Stablein**[1]

1. Institute of Fundamental Sciences, Massey University, Palmerston North, New Zealand
2. MidCentral District Health Board, Palmerston North, New Zealand
*Contact author: a.j.godfrey@massey.ac.nz

Keywords: automated process, health care, filtering, quality control, Shewhart methodology

Chronic Kidney Disease (CKD) is an increasingly concerning worldwide health problem [1]. Our client is a local independently owned pathology laboratory which processes approximately 1200 renal function tests per day from a catchment population of approximately 750,000 people. When our client needed to filter test results for healthy patients out from those that were showing decreased kidney function we decided to create an automated filtering system. With respect to those patients showing less than fully healthy kidney function, we needed to identify which patients needed more active human monitoring due to changes in their circumstances, versus those that were currently stable and could have less active human monitoring.

We have processed each patient's history of test results using *R* to create a single-page pdf document that can then be supplied to the relevant medical staff. The automated system is running in near-real time, as batches of test results are being processed every fifteen minutes.

The document created shows the medical practitioners the patient's history in graphical form, along with some additional information inspired by Shewhart methodology. The standard control chart is presented in conjunction with exponentially weighted moving average and CUSUM charts. A recommendation for the timing of the patient's next renal function test is also given. While all patients who are tested have the graphic-based pdf file created, this is not sent to the patient's general practitioner but is stored in a repository that is available to them and the specialist clinician. We do generate an e-mail message that informs the general practitioner when the patient should next be tested and indicates how the patient's condition is going to be monitored in future. Meanwhile, the specialist clinician is sent a prioritised list of patients whose cases do need more active attention.

The system is implemented in such a way that allows the client's IT team (not versed in *R* or statistics) to make changes to the messages (in textual form) sent to the patient's general practitioner and/or specialist. Any cosmetic changes requested by our client can be implemented in the source code and tested offline, before being transferred over to the implementation server.

Our findings thus far are that our automated filtering has not made a single false negative call, although we do make a higher number of false positive calls as a consequence. Time savings for the clinicians that had to monitor the test results one by one prior to implementation, look extremely promising.

References

[1] Levey, A. S., K. Eckardt, Y. Tsukamoto, A. Levin, J. Coresh, J. Rossert, D. de Zeeuw, T. H. Hostetter, N. Lameire, and G. Eknoyan (2005). Definition and classification of chronic kidney disease: a position statement from Kidney Disease: Improving Global Outcomes (KDIGO). *Kidney international 67*(6), 2089–2100.

The R package FANet: sparse Factor Analysis model for high dimensional gene co-expression Networks

Yuna Blum[1*], **Magalie Houée-Bigot**[2] **Sandrine Lagarrigue**[3] **David Causeur**[2]

1. Department of Human Genetics, University of California, Los Angeles CA. USA
2. Agrocampus Ouest- Applied Mathematics Department, Rennes, FRANCE
3. Agrcoampus Ouest - UMR PEGASE INRA, Agrocampus Ouest, Rennes, FRANCE.
*Contact author: yuna.blum@gmail.com

Keywords: co-expression networks, factor analysis, high dimension, sparsity, R

Inference on gene regulatory networks from high-throughput expression data turns out to be one of the main current challenges in systems biology. Such interaction networks are very insightful for the deep understanding of biological relationships between genes. In particular, a functional characterization of gene modules of highly interacting genes enables the identification of biological processes underlying complex traits as diseases. Inference on this dependence structure shall account for both the high dimension of the data and the sparsity of the interaction network.

The R package **FANet** provides a powerful method for estimating high dimensional co-expression networks. Extending the idea introduced for differential analysis by Blum et al. [1] and Friguet et al. [2] we suggest to take advantage of a low-dimensional latent linear structure of dependence to improve the stability of correlation estimations. We propose an EM algorithm to fit a sparse factor model for correlations and demonstrate how it helps extracting modules of genes and more generally improves the gene clustering performance. Two functions are available in **FANet** package in order to introduce sparsity in the network estimation. One function is based on a LASSO estimation using a cyclic coordinate descent algorithm. As an alternative, the second function is based on biological knowledge integration as Gene Ontology annotation. Finally, **FANet** results can serve as an input for WGCNA (Langfelder and Horvath [3]) procedure for gene modules detection.

References

[1] Blum, Y., G. Le Mignon, S. Lagarrigue, and D. Causeur (2010). A factor model to analyze heterogeneity in gene expression. *BMC bioinformatics 11*(1), 368.

[2] Friguet, C., M. Kloareg, and D. Causeur (2009). A factor model approach to multiple testing under dependence. *Journal of the American Statistical Association 104*(488), 1406–1415.

[3] Langfelder, P. and S. Horvath (2008). Wgcna: an r package for weighted correlation network analysis. *BMC bioinformatics 9*(1), 559.

Fostering the next generation of open science with R

Karthik Ram [1,2,*] , Scott Chamberlain [1]

1. The rOpenSci project, University of California, Berkeley. Berkeley CA 94720
2. Berkeley Initiative in Global Change Biology, University of California Berkeley, 94720
*Contact author: foo@bar.com

Keywords: open science, reproducible research, data sharing, open data

Research is becoming increasingly data intensive and computation driven across various scientific domains from the social and life sciences all the way to particle physics. Many new scientific insights will likely emerge from vast stores of existing data, rather than from new data collection efforts. In addition, funder and journal mandates now require that researchers share at least the final datasets at the time of publication.

rOpenSci is an effort to foster such data driven science among researchers that use R. Our suite of tools (http://ropensci.org/packages/) allow access to these data repositories through a statistical programming environment that is already a familiar part of the workflow of many scientists. Our tools not only facilitate drawing data into an environment where it can readily be manipulated, but also one in which those analyses and methods can be easily shared, replicated, and extended by other researchers. In this talk we highlight some our recent efforts in advancing open and transparent practices in the sciences.

Simulating Influenza Transmission with Real Network Data

Henry Bongiovi

BS Statistics, California Polytechnic State University, San Luis Obispo
bongiovihenry@gmail.com

Keywords: Network Data, Simulation, Education, Influenza, Epidemic

Disease has been humanities arch rival since the dawn of our existence. As such, we have been trying our best to understand its spread and proliferation. One of the most common diseases, Influenza, is also one of the most complex. To understand the complexities of its spread would greatly improve our ability to combat it and other diseases like it. Using *R* in conjunction with the package **statnet**, I have created a simulation of influenza transmission in an American high school based on real data collected from a study using RFID chips to collect information about the duration of close contacts (within 3 meter) between students and faculty (Salathé[1]). Combing this network data with simplified research done on influenza transmission (Potter[2]), I have crated baseline predictions for final size, duration and probability among other summary statistics per theoretical probability of transmission for a particular strain of the virus. After these baseline prediction have been simulated, I then used data on a known intervention strategy (Potter[3]) to determine the effectiveness of it in terms of a side-by-side comparison.

After modeling the natural course of a disease alongside potential intervention strategies, the next natural step was to make a function using easily changeable attributes so that the simulation can encompass up to date information about transmission probabilities, contact duration length, or other variables used to simulate the epidemic or intervention. From these changeable attributes, one could easily specify other diseases so long as its transmission is known to be similar to influenza.

Perhaps the most important function of this project is to create an interactive simulation to educate people on the effectiveness of intervention strategies as well as risks of epidemic given a certain social structure based on a given network of contact durations. These simulations are simple to understand and could be used and experimented on by anyone from middle-schoolers to policy makers.

References

[1] Salathé (2007). "A High-Resolution Human Contact Network for Infectious Disease Transmission", http://www.pnas.org/content/107/51/22020.full.pdf%20html

[2] Potter (2011a). Estimating Within-School Contact Networks to Understand Influenza Transmission. *The Annals of Applied Statistics 2012, Vol 6*. 10-11

[3] Potter (2011b). Estimating Within-School Contact Networks to Understand Influenza Transmission. *The Annals of Applied Statistics 2012, Vol 6*. 14-15

Package mlr: Machine Learning in R

Michel Lang[1,*], **Jakob Richter**[1], **Bernd Bischl**[1,*]

1. TU Dortmund University
*Contact author: {lang,bischl}@statistik.tu-dortmund.de

Keywords: machine learning, model selection, parameter tuning, variable selection, parallelization, survival analysis

Constructing a suitable machine learning model for a given data set or conducting large-scale comparison experiments in this domain can be a tedious task in *R* for various reasons: First, there is no standardized interface for learning algorithms in *R*. The technical ins and outs of each model and operation must be understood and unified for comparison. Many implementations additionally require some sort of special treatment, e.g. transformations of the input data or output results. For benchmark experiments, all this must be embedded in a resampling strategy, where every learner works on the same training sets and is evaluated on the same test sets. More complex statistical learners offer a large set of arguments to allow fine-grained control of the algorithm. For a fair comparison, the arguments must be systematically varied, i.e. using a grid, or, as more efficient approach, tuned by a modern algorithm configurator. Variable reduction and selection is another routine matter. Combining all of these operations with a larger number of machine learning models is not only time-consuming and error-prone to program, but can also easily lead to runtime issues for more comprehensive benchmark studies.

The package **mlr** [3] tries to solve this problem by providing abstractions for learning task, learning machines, resampling strategies, performance measures, tuning algorithms, variable selection methods and other common operations. Its intent is to offer a clean, easy-to-use and flexible domain specific language for machine learning experiments in *R*. It currently offers around 30 classifiers, 20 regression models, most well-known resampling strategies and all popular performance measures. There is also a novel support for survival models and cost-sensitive learning. The package provides several hyperparameter tuning algorithms. These range from very simple random searches to modern approaches based on evolutionary strategies or iterated f-racing. Variable selection is possible through various filter and wrapper approaches.

The clean, abstract interface enables learning algorithms to be enhanced or extended, by chaining the basic method with other useful operations. Examples are generic bagging, adding feature imputation (or other preprocessing) or adding self-tuning.

Parallelization can be triggered on many different levels of a typical **mlr** workflow, e.g. for outer or inner resampling, tuning or variable selection operations. Many popular parallelization back-ends, e.g. **parallel** [4] or **BatchJobs** [2], can easily be selected by the user, as **mlr** internally uses the **parallelMap** [1] package for this.

References

[1] Bischl, B. and M. Lang (2014). *Unified interface to some popular parallelization back-ends for interactive usage and package development*.

[2] Bischl, B., M. Lang, and O. Mersmann (2014). *BatchJobs: Batch computing with R*.

[3] Bischl, B., M. Lang, and J. Richter (2014). *mlr: Machine Learning in R*.

[4] R Core Team (2014). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

data.table : fast and flexible data manipulation

Matt Dowle

Keywords: Large data, ordered joins, update by reference, aggregation

The `data.table` package inherits from and extends `data.frame` aiming to reduce two types of time :

1. programming time (fewer function calls, less variable name repetition)
2. compute time on large data (e.g. 64bit with 8GB+ RAM)

The package offers fast aggregation of large datasets, fast ordered joins, fast add/modify/delete of columns by group using no copies at all, list columns where each cell can itself be a vector/object and a fast file reader: `fread()`. Although the speed benefits are greatest on large datasets (1GB – 100GB), many also use it on small datasets for its brief and flexible syntax.

The general form, including chaining is :

```
DT[where, select|update, group by][order by][ ... ]...[ ... ]
```

Currently there are 5 active contributors to the project, mainly from Genomics and Finance.

The presentation covers the essential syntax illustrated with examples.

Creating a `data.table`
Fast and friendly file reading with `fread`
Basic query syntax
Keys (`setkey`)
Update by reference (`:= and set*`)
Ordered joins forwards, backwards, limited and nearest
List columns (each cell can itself be a vector)
Why R?
Recent new features
Future directions
Quality assurance (1,000 tests, release procedures)
A review of online help (1,200 Q&A on Stack Overflow's `data.table` tag)

References

M Dowle, T Short, S Lianoglou, A Srinivasan, R Saporta, E Antonyan (2008-2014). data.table: Extension of data.frame. http://datatable.r-forge.r-project.org/.

Approximate Bayesian Inference for Spatial Econometrics with R-INLA

Roger S. Bivand[1], **Virgilio Gómez-Rubio**[2,*], **Håvard Rue**[3]

1. Norwegian School of Economics
2. University of Castilla-La Mancha
3. Norwegian University for Science and Technology
*Contact author: virgilio.gomez@uclm.es

Keywords: Bayesian Inference, INLA, Spatial Econometrics

LeSage and Pace [2] describe several models for Spatial Econometrics and provide some software to fit them in the Spatial Econometrics Toolbox for Matlab (`http://www.spatial-econometrics.com/`). Many of these models rely on spatially autoregressive effects, such as

$$y = \rho W y + X\beta + \varepsilon \tag{1}$$

where y is a vector of observed data, ρ a spatial autocorrelation parameter, X a matrix of covariates with associated coefficients β and ε is a Gaussian random error. Equation (1) can be rewritten as

$$y = (I_n - \rho W)^{-1}(X\beta + \varepsilon) \tag{2}$$

which shows how the response y depends on some latent spatial effects.

In this work we will introduce the use of the Integrated Nested Laplace Approximation [3, INLA], as implemented in the **R-INLA** package, to fit a wider range of spatial econometrics models. INLA provides a suitable methodology to estimating the posterior marginal of the model parameters when the latent effects are Gaussian Markov Random Fields. Some latent effects are implemented in the **R-INLA** package so that models can be defined and fitted similarly with the `inla()` function, similarly as with `glm()` or `gam()`.

We will consider two different approaches. The first one is described in Bivand et al. [1] and it is very helpful when the latent model that we need is not implemented in **R-INLA**. Instead of fitting the required model, this method is based on fitting that model after conditioning on different values some of the parameters in the model and then combining them using Bayesian model averaging (with package **INLABMA**) to obtain the desired model. These conditioned models are often simpler than the original model and can be easily fitted with **R-INLA**.

The second approach is based on a newly implemented latent model that provides random effects required for several spatial econometrics models as in equation (2). This approach to model fitting is preferable because it is completely implemented in **R-INLA** and does not require any other external code.

Finally, we will describe how to use these models on two real datasets on the housing value in Boston and the probability of re-opening a business in New Orleans in the aftermath of hurricane Katrina.

References

[1] Bivand, R. S., V. Gómez-Rubio, and H. Rue (2014). Approximate bayesian inference for spatial econometrics models. *Spatial Statistics*, To appear.

[2] LeSage, J. and R. K. Pace (2009). *Introduction to Spatial Econometrics*. Chapman and Hall/CRC.

[3] Rue, H., S. Martino, and N. Chopin (2009). Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society, Series B 71*(Part 2), 319–392.

Spatial Tweetstistics with R: Geographical Distribution of English Loan Words in Spanish Tweets

Irene Checa-Garcia [1*], Virgilio Gómez-Rubio [2]

1. University of Wyoming 2. Universidad de Castilla-La Mancha
*Contact author: irene.checa@uwyo.edu

Keywords: Loan Words, Dialectology, Anglicisms, Spatial Statistics, Twitter

In this study we look at the geographical and frequency distribution of different types of Anglicisms in tweets in Spanish. In the first part of the study we distinguish different types of Anglicisms according to several criteria: their degree of official acceptance into the language, their structural characteristics, their extension, and the availability of alternative expressions in the Spanish language for the same concept. One recurrent question in the literature about loan words, particularly in language contact situations, such as Spanish in the USA, is how frequent loan words really are. Does a language contact situation heavily influence –and if so, how heavily- the frequency of use of loan words from one language to another? A related question is which types of loan words are more frequent [1]. Previous works have contrasted loan word presence in bilingual and monolingual speakers [2] or offered a percentage of their frequency in different corpora [3], but no further statistical analysis was presented. In this work, we won't look at who is producing loan words, but rather where they are being produced.

Using a spatial data analysis [4] we will study the geographical frequency of a set of over 250 Anglicisms. We have used *R* packages **streamR** for the data collection and different packages from the *Spatial Task View* for the visualization and spatial data analysis. We have produced maps to summarize the main results and to show the spatial distribution of Anglicisms types that we have found in tweets in different Spanish speaking countries. In this way, we will show the "hottest spots" for the use of the different types of Anglicisms. And we hope to answer geographically the question of how language contact might influence loan word frequency.

References

[1] Medieta, Eva (1999) *El préstamo en el español de los Estados Unidos*. New York: Peter Lang.

[2] Otheguy, Ricardo & García, Ofelia (1988) Diffusion of lexical innovations in the Spanish of Cuban Americans. In Jacob Ornstein-Galicia & George K. Green (eds.) *Research Issues and Problems in United States Spanish: Latin American and Southwestern Varieties.*

[3] Silva-Corvalán, Carmen (1994) *Language contact and change. Spanish in Los Angeles.* Oxford: Oxford University Press.

[4] Bivand, R., Pebesma, E. and Gómez-Rubio, V. (2013). *Applied Spatial Data Analysis with R*. Springer, New York.

Why I heart (not) parentheses, a journeyman's toolkit path from S to R

Colin Goodall[1,*]

1. AT&T Labs - Research
*Contact author: cgoodall@att.com

Keywords: "data model" "standalone tools" jt Perl Python

The data model introduced in S [1], which includes vectors (no scalars), multi-dimensional arrays, lists, data frames, functions as objects, and metadata in object attributes, is clearly one of the most successful and long-lived in data analysis. The introduction of open source R, r-project.org [2], and CRAN, set in motion the pervasive use of the R software environment for data analysis and statistical computation we participate in today. Indeed, the early limitations of S and R, notably in the analysis of very large data sets, in highly interactive graphics, and in providing an integrated collaborative development environment, are even now being effectively addressed and overcome.

A different approach to computational data analysis is to shed R (for now) as a software environment. Instead, computation is by discrete lightweight standalone tools in a shell environment. Data objects sit in distinct files instead of as R objects. UNIX tools for system administration, such as sort, cat, paste, and join, are a starting point in assembling a large set of standalone executables for data analysis, the journeyman's toolkit. The functions are programmed in Perl, Python, Korn shell, and Cymbal [3]. The specific language used is not important, except that each language supports some constructs more easily than does R, which in turn influences how a data analysis task is designed. For example, in Perl, each array may contain data of different types (as in, rows of a data frame in R), text manipulation is primary, numerical analysis is secondary, and hierarchical hashes and arrays are very general.

The data model is at core that of S and R, and the journeyman's toolkit has analogs of `apply`, `aggregate`, `reshape`, `duplicated`, `print`, `subset`, `seq`, etc., as one would expect. However, compared to their R counterparts these standalone functions typically do more operations (with one or more versions possibly of the base function), accept more complex inputs, and offer more output options. For example, duplicated can return duplicated values in one or more variables conditional on a set of conditioning variables (`tapply` anyone?). Or print will return formatted output with multi-row column headers, empty lines for spacing, and exception formatting for values with unusual width.

To take this a step further, the toolkit contains novel handling of metadata (for example multi-valued names attributes), list-valued fields (support for multiple field separators in the same record), analysis of arbitrarily large data files (including parallelization opportunities), and mini-languages (used for example in building Excel spreadsheets from flat files).

I describe an R package **jt** which brings these modes of standalone analysis back into R.

References

[1] Becker, R.A., Chambers, J.M., and Wilks, A.R. (1988). The New S Language. Wadsworth and Brooks/Cole, Pacific Grove CA.

[2] R Project (…2014). The R Project for Statistical Computing, http://www.r-project.org

[3] Greer, R (1999). Daytona and the Fourth-Generation Language Cymbal. In *Proceedings ACM SIGMOD Intl Conf on Management of Data*, **28**(2): 525-526.

Practical use of *R* by blind People

A. Jonathan R. Godfrey

Institute of Fundamental Sciences, Massey University, Palmerston North, New Zealand
a.j.godfrey@massey.ac.nz

Keywords: accessibility, screen reader, synthesized speech, low vision, blindness

This presentation will highlight the successes and failures blind people have had when using *R* for their university studies and in their working lives. Primary focus is on the Windows operating system but consideration of the use of *R* under Linux will also be briefly discussed. A practical demonstration from the author (himself blind) completing some basic tasks in *R* using a free open source screen reader known as NVDA will be given.

The benefits of *R* to the blind user are numerous [2]. Of particular note are: *R* is extensible; *R* is not reliant on a graphical user interface (GUI); *R*'s help functionality is available in plain *HTML*; *R* has strong links to LaTeX; and, *R* can be used within minutes of installation as there are no additional setup tasks to complete.

Major issues with *R* are therefore relatively few when compared to other statistical software options [3]. In the majority of situations, these difficulties are easily avoided — some might say that the shortcomings are limited to those bells and whistles that are optional for any other *R* user. perhaps the most important request would be to use scalar vector graphics (*SVG*) as a standard file type for graphs, including being an option for the `savePlot` command and the `save as` item in the pull down menu of a graphics device window.

Further to this is a secondary list of "nice to have's" including: Creation of all package vignettes into *HTML* instead of pdf files, perhaps using the **knitr** package [4] which is currently only an option for package developers; An accessible integrated development environment (IDE) because RStudio is not yet an accessible option; and, more accessible options for a GUI that aids the novice user that is blind.

Developments in the **BrailleR** package [1] have started to meet the needs of blind users but there are some stumbling blocks. Critical among these are the lack of assigned classes to graphical objects, and finding substitutes for the not fully accessible *R* console under Windows, such as being able to open a script window when working with *R* in terminal mode.

References

[1] Godfrey, A. J. R. (2013a). *BrailleR: Improved Access for Blind UseRs*. R package version 0.9.

[2] Godfrey, A. J. R. (2013b). Statistical software from a blind person's perspective: R is the best, but we can make it better. *The R Journal 5*(1), 73–80.

[3] Godfrey, A. J. R. and M. T. Loots (2014). Statistical software (*R*, *SAS*, *SPSS*, and *Minitab*) for blind students and practitioners. *Journal of Statistical Software*, under review.

[4] Xie, Y. (2013). *knitr: A general-purpose package for dynamic report generation in R*. R package version 1.5.

Integrating R-INLA with *R* spatial packages and ggplot2

Thomas Jagger and James Elsner

Florida State University
*Contact author: Thomas Jagger

Keywords: Bayesian, Spatial, Graphics

We develop a model for the space time distribution of tornado count data aggregated yearly over counties in Kansas in the USA. We use a Bayesian model to analyze local counts. A hierarchical model is chosen, in which the local counts are assumed to have a negative binomial distribution when conditioned on the distribution's mean and count parameters. Next in the hierarchy, the mean and count parameters each have a distribution, where the mean is assumed to be a spatial Gaussian process. The spatial Guassian process is composed of a mean that is linearly regressed onto local and global covariates. Both the intercept and the coefficients are allowed to vary spatially using intrinsic conditional autoregressive (ICAR) priors [1]. Finally, the model is fit by an INLA (Integrated Nested Laplace Approximation) [4] using the **R-INLA** [3].

We demonstrate the use of the *R* package **rgdal** to read in the tornado shape files, with *R* packages **maps** and **maptools** to generate a counties shape file. We use functions from the *R* **sp** and **spdep** packages to overlay tornado tracks onto the counties and generate a spatial neighborhood list [2], followed by `ddply()` from the **plyr** *R* package to generate yearly county tornado counts. We perform our analysis using **R-INLA** package to estimate posterior densities and means. We generate publication quality plots of these results using the **ggplot2** *R* package [5]. The example code and presentation will be created using *R-Studio* and published on Rpubs.

References

[1] Besag, J., J. York, and A. Mollie (1991). Bayesian image restoration, with two applications in spatial statistics. *Annals of the Institute of Statistical Mathematics 43*(1), 1–59.

[2] Bivand, R., E. Pebesma, and V. Gomez-Rubio (2008). *Applied Spatial Data Analysis with R*. New York: Springer.

[3] Rue, H. (2012). *The R-INLA project*. R-INLA `http://www.r-inla.org`.

[4] Rue, H., S. Martino, and N. Chopin (2009). Approximate Bayesian inference for latent Gaussian models using integrated nested Laplace approximations (with discussion).

[5] Wickham, H. (2009). *ggplot2: elegant graphics for data analysis*. Springer New York. GGPLOT2 `http://had.co.nz/ggplot2/book`.

Software Testing and the R Language

Stephen Kaluzny[1*]

1. TIBCO Software Inc.
*Contact author: skaluzny@tibco.com

Keywords: R, TERR, testing

We have been developing TIBCO Enterprise Runtime for R (TERR), an Open Source R compatible engine for several years. I will describe the testing processes we have created for this engine. We are testing for both numerical correctness as well as compatibility with Open Source R. We have developed a framework that uses a large portion of the tests from our extensive collection of S-PLUS test suites. Several packages that we have developed for this testing will be presented.

Testing of packages from CRAN is another challenge. We have created a system for testing packages with both TERR and R. We will describe how the system automatically creates tests from a package's source files. Issues with stochastic algorithms and testing on multiple platforms will be discussed. Suggestions for improving packages with tests will also be presented.

Interactive Visualizations from R

Ramnath Vaidyanathan[1,*]

1. McGill University

*Contact author: ramnath.vaidyanathan@mcgill.ca

Keywords: rcharts, d3js, interactive, visualization

In this talk, I will discuss rCharts [1], an R package to create, customize and share interactive visualizations straight from R, using a consistent plotting interface, leveraging several existing javascript visualization libraries.

The advent of javascript visualization frameworks like d3.js [1] and raphaeljs have made it easier to create sophisticated interactive visualizations. However, it requires deep knowledge of web development tools, making it harder to use for data scientists, who often spend a lot of their time analyzing data using languages like R/Python/Julia. Moreover, data scientists are often used to creating plots with a few lines of code, as opposed to building a plot up from scratch, as is often the case with interactive plotting libraries.

The main motivation behind this work is to provide data scientists a seamless workflow that allows them to execute all steps of the data visualization process, from acquiring data to exploring it, visualizing it, and sharing the results as an interactive presentation, without having to leave the comfort of their primary language for data analysis.

References

[1] Vaidyanathan, R. (2014) rCharts: Interactive Visualizations from R. http://rcharts.github.io

[2] Bostock, M., Ogievetsky, V., & Heer, J. (2011). D3: data-driven documents. Visualization and Computer Graphics, IEEE Transactions on, 17(12), 2301-2309.

Spyre: Exploratory Data Analysis in the Browser

Erik Iverson[1,*]

1. MEI Research
*Contact author: erik@sigmafield.org

Keywords: EDA, browser, JSON, websockets, data analysis

Many exploratory data analysis (EDA) tools involve the use of a GUI that exposes the analyst to a very limited set of options for viewing and summarizing data. Alternatively, interactive languages such as *R* offer an incredibly flexible environment to facilitate data exploration, at the expense of having to repeat basic operations such as bivariate plots and tabular summaries repeatedly to gain deeper insight into the data.

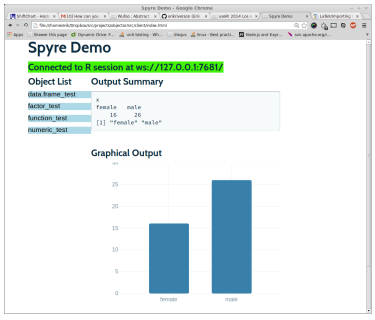
Two popular *R* IDEs, Emacs ESS and RStudio, have incorporated useful shortcuts to help analysts with the mechanics of EDA. For example, the user can view which *R* objects have been created in the current workspace, create plots of those objects where appropriate, and inspect the structure and contents of objects. While useful, these features are IDE-specific, and neither flexible nor extensible.

Increasingly, the web browser is a platform not just for information retrieval and display, but for interactive applications. External processes such as *R* can communicate with modern web browsers through standards like websockets. The **spyre** package is a hybrid of the two EDA approaches above. It allows an *R* user to inspect data through the familiar interactive command-line environment, while *simultaneously* offering browser-based GUI features to facilitate common tasks when exploring data. Spyre brings R-based EDA to the browser.

Spyre communicates with a running *R* process through the use of the **websockets** package, and does *not* block the *R* process while running. In other words, the user is still able to use *R* interactively through the command line while Spyre is running. As new objects are created in the *R* process, Spyre becomes aware of them and allows the user to view basic information about the objects, plot them, and view customized summaries. Because the underlying communications are done through websockets, Spyre can be used with any *R* IDE.

In the figure below, a basic Spyre session is shown. A list of currently available objects is displayed. When clicked, a summary function based on that object's class, in this case a factor, is called. Summary data is sent to the browser via JSON. Spyre's modular design allows an analyst to easily override or augment any of the information sent between the *R* process and Spyre, using the **jsonlite** package. This allows the summaries to be displayed in simple text form, or more advanced actions to be taken, such as using the d3.js library to draw graphical summaries of the data. The wide availability of JavaScript libraries opens up many possibilities for interesting applications using Spyre.

The next extensions planned for **spyre** include a data.frame explorer and an interactive regression tool.



Data Warehousing for Interactive Visualization of Student Data

Kim Speerschneider[1,2,*]

1. University at Albany, SUNY
2. Excelsior College
*Contact author: kimkspeer@gmail.com

Keywords: Data Warehousing, Reproducible Research, Visualization

This talk will explore the intersection of data warehousing and data visualization. Data warehousing allows us to collect a depth of information about longitudinal trends while also retaining student-level details. In this way, the interactive graphics we share with our colleagues can be manipulated to serve specific needs, but the process remains transparent and reproducible. I will describe how, at our institution, we warehouse student data and in turn share this data with our colleagues. Specifically, I will describe some of the warehousing functions in our privately used package, **ecir**, and why we created a package specific to our own institution and data. Additionally, I will share some of the visualizations that we share throughout our institution that make use of **ggplot2** (Wickham, 2009), **shiny** (RStudio, Inc., 2014), and **googleVis** (Gesmann et al., 2013) and explain how data warehousing allows us to tell a more complete story of our students' experiences. The figure below shows one such visualization. Although applications here are in the field of Education, this talk is relevant to anyone interested in how data warehousing can expand how data is shared, particularly with highly interactive graphics.

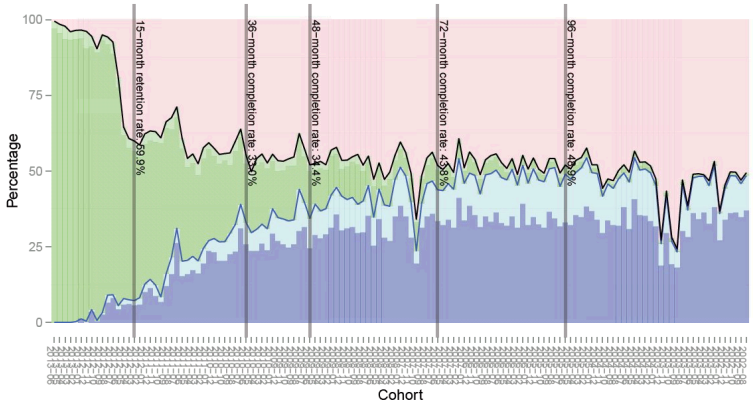


Figure 3.2: Institutional Beginning Retention by Cohort

References

Gesmann, M, de Castillo, D., & Cheng, J. (2013). *googleVis: Interface between R and the Google Chart Tools*. R package version 0.4.7

RStudio, Inc. (2014). *shiny: Web Application Framework for R*. R package version 0.9.1

Wickham, H. (2009). *ggplot2: elegant graphics for data analysis.* Springer New York.

ETD: A Design Pattern for Building Web-Based Analytics Dashboards in *R*

Bhaskar Rao[1*], Eric Colson[1], Alan Eng[1],

1. Stitch Fix Inc.
*Contact author: bhaskar@stitchfix.com

Keywords: reports, shiny, analytics dashboards, design pattern, ETD.

ETD, an abbreviation for extract-transform-display, is a design pattern that the Stitch Fix data team observed while building reporting and analytics dashboards in *R*, using the **Shiny** package. Formalizing this pattern reduces the complexity involved in creating web-based dashboards. It also provides a templatized approach for creating dashboards and promotes re-use and encapsulation of *R* and data extraction code.

Developing interactive web-based dashboards typically involves three distinct stages. First, the 'Extract' stage pulls data from a data source - typically a relational database using *SQL*. This extracted data is pulled into an *R* data structure where complex calculations can be applied (e.g. cross-tabulation, cleansing routines, conditional probabilities, complex metric definitions, …etc.). This is the 'Transform' stage. Finally, the transformed information is displayed using standard *R* visualization packages like **ggplot** or **googleVis.** This is the 'Display' stage. This 3-staged workflow is analogous to the extract-transform-load[2] (ETL) pattern prevalent in data warehousing. The important distinction is the final stage where, rather than loading the data for system consumption, we are rendering the information for end-user consumption.

Many data scientists lack the requisite skills to build web-based analytics dashboards. However, packages like **Shiny** provide a layer of abstraction that enables them to build web-based application in *R* without having to learn *HTML, Javascript* and *CSS*. Our ETD design pattern takes it one step further by taming the complexities of **Shiny**'s reactive programming framework and making it possible to templatize the creation of typical analytics dashboards. Using the ETD pattern in the development of Shiny dashboards helps our data scientists build complex web-based dashboards quickly while keeping our *R* code-base modular, clean, and extensible.

References

[1] RStudio (2013). Shiny by RStudio, http://shiny.rstudio.com/articles/
[2] Kimball, Ralph, and Joe Caserta. The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data. Indianapolis, IN: Wiley, 2004. Print.

Creating a network of women *R*-users

Gabriela de Queiroz[1,2,*]

1. R-ladies
2. Alpine Data Labs
*Contact author: gabkroz@gmail.com

Keywords: Teaching, Women, Statistics, Gender, Networking

Even though there are many women using *R*, they are underrepresented at conferences and meetups. Thinking about that I created, in October of 2012, the *R*-ladies meetup group. It provides a friendly environment for women to learn and practice *R* and for networking. Based in San Francisco, the group has over 300 members with an average of 25 ladies at each event. Twenty meetups have happened so far, ranging from workshops for *R* beginners to multi-week meetings accompanying online courses such as Johns Hopkins' Computing for Data Analysis [1] and Stanford's Statistical Learning [2]. The group also provides a mailing list where members can post job opportunities and give and receive feedback on resume development and interview preparation. The goal of this work is to share my experience running the *R*-ladies meetup, and ideas on how to get more women into the *R* community.

References

[1] Coursera (2014). Computing for Data Analysis, `http://www.coursera.org/course/compdata`.

[2] Stanford (2014). Statistical Learning, `http://class.stanford.edu/courses/HumanitiesScience/StatLearning/Winter2014/info`.

A Comparison of Rc^2, RStudio, and RCloud

E. James Harner[1*] **and Mark Lilback**[1]

1. Dept. of Statistics, West Virginia University
*Contact author: jharner@stat.wvu.edu

Keywords: Cloud computing, R frontend, Rc2, RStudio, RCloud

Various cloud-based frontends have been or are being developed for the *R* statistical computing environment. This talk compares and contrasts three: Rc^2, RStudio, and RCloud.

Rc^2 is a cloud-based, collaborative interface to *R* that currently works with client iPads and computers running OS X. Rc^2 is highly scalable and allows real-time research collaborations and high-performance, big-data computing. The clients provide a native look and feel, but use HTML5 for generated output, thus allowing development for other platforms. With Rc^2, *R* sessions are no longer tied to a specific device/computer or user.

Rc^2 is an Integrated Development Environment (IDE) designed for simplicity and ease-of-use, allowing students or researchers to learn *R* without solely imposing a command-line interface. At the same time, power users will find the system flexible enough to meet most of their needs, including the development of *R* packages. Rc^2 is organized by projects and projects can have multiple sharable workspaces. This allows researchers to collaborate over the Internet without concern for code or data becoming out of sync.

The Java-based server side of Rc^2 spawns *R* sessions (using Rserve). Rc^2 is driven by and interfaces with both SQL and NoSQL databases. Full support for Sweave allows users to easily include, update, and format *R* output within LaTeX documents for publishable papers. R markdown and other input types, e.g., SAS, are also supported.

RStudio is a powerful, open-source IDE for *R*. It is provides a productive user interface to *R* that works on all major platforms. A server version is also available for *R* code development over the web.

As an IDE, RStudio supports syntax highlighting, code completion, and smart indentation. *R* code can be directly executed from the source editor. It supports integrated *R* help, the use of projects, and has a workspace browser. An interactive debugger allows the developer to find and fix errors quickly. It has extensive support for developing packages.

RStudio supports both Sweave and R Markdown. It also supports interactive web application development using Shiny and Shiny Server.

RCloud is an HTML5 frontend to *R* for data analysis, which allows users to collaboratively create and share R scripts. Since it is HTML5 based, it is platform independent. It provides a notebook interface that lets you easily record a session and annotate it with text, equations, and supporting images.

RCloud allows you to easily browse other users's notebooks, comment on notebooks, fork notebooks, and use them as function calls in your own notebooks. It provides an environment in which *R* packages can create rich HTML content, e.g., using D3. It also provides a transparent, integrated version control system. RCloud notebooks are Github gists.

Rc^2, RStudio, and RCloud target different audiences. Rc^2 is an accessible IDE for students and researchers who have limited technical skills. Rc^2 sessions allow real-time collaboration which is ideal for students taking distance-based courses and researchers in different locations. On the other hand, Rc^2 is not yet platform independent. RStudio is a powerful IDE, but its completeness necessarily involves complexity. It does not support collaboration although users could share information using group permissions on the Linux server version. RCloud is HTML5 based and thus platform independent. Its most powerful feature is its implementation of notebooks. This allow users to flexibly share and extend notebooks.

PivotalR: A Package for Machine Learning on Big Data

Hai Qian[1,*]

1. Pivotal Inc.
*Contact author: hqian@gopivotal.com

Keywords: big data, machine learning, database, usability

PivotalR [1] is an R package that provides a front-end to PostgreSQL [2] and all PostgreSQL-like databases such as Pivotal Inc.'s Greenplum Database (GPDB) [3], HAWQ [4] on Hadoop. PivotalR also provides the R wrapper for MADlib [5]. MADlib is an open-source library for scalable in-database analytics. It provides data-parallel implementations of mathematical, statistical and machine-learning algorithms for structured and unstructured data. Thus PivotalR also enables the user to apply machine learning algorithms onto big data.

In recent years, Big Data has become an important research topic and a very realistic problem in industry. The amount of data that we need to process is exploding, and the ability of analyzing big data has become the key factor in competition. Big data sets do not fit into computer's memory and it would be really slow if the big data sets were processed sequentially. On the other hand, most contributed packages of R are still strictly sequential, single machine, and they are restricted to small data sets that can be loaded into memory. As computing shifts irreversibly to parallel architectures and big data, there is a risk for the R community to become irrelevant.

PivotalR, which provides an R front-end with data.frame oriented API for R users to access big data stored in distributive databases or Hadoop distributive file system (HDFS). PivotalR puts more emphasis on machine learning by providing a wrapper for MADlib, which is an open-source library of scalable in-database machine learning algorithms. Actually PivotalR offers more than what MADlib has. It adds functionalities that do not exist in MADlib, for example, the support for categorical variables.

PivotalR makes it easier to work on big data sets in databases or HDFS. Many queries that are difficult to construct in SQL client can be easily constructed using PivotalR. This make sit suitable for data preprocessing. PivotalR also makes it easy to create many algorithm prototypes that can directly run in database using familiar R syntax. Besides, PivotalR is portable onto many different platforms, and the prototype code is the same on all supported platforms.

Although PivotalR is targeted at big data, database, and Hadoop, no prior knowledge is needed. The objective of PivotalR is to give the normal R users an easy access to all of these without learning extra knowledge.

References

[1] PivotalR, http://cran.r-project.org/web/packages/PivotalR/index.html. version 0.1.15.1.
[2] PostgreSQL, http://www.postgresql.org/
[3] Greenplum Database, 2013a. http://www.gopivotal.com/products/pivotal-greenplum-database. version 4.2.4.
[4] HAWQ, http://www.gopivotal.com/pivotal-products/pivotal-data-fabric/pivotal-hd. version 1.2 (to be released).
[5] MADlib, http://madlib.net. version 1.5 (to be released).

Swimming in clear lakes: How model coupling with R helps to improve water quality

Thomas Petzoldt[1*], **René Sachse**[1,2,3]

1. Technische Universität Dresden, Institute of Hydrobiology, Dresden, Germany
2. Leibniz Institute for Freshwater Ecology and Inland Fisheries, Berlin, Germany
3. Potsdam University, Institute of Earth and Environmental Science, Potsdam, Germany.
*Contact author: thomas.petzoldt@tu-dresden.de

Keywords: aquatic ecology, lakes, differential equations, model coupling

Scientific understanding of complex ecological systems is inherently difficult, and numerous theoretical and management models have emerged. Each model has its own purpose, philosophy, and implementation. Often, tunnel-vision and wheel re-invention are hindering scientific exchange [3] and in response, ecological modelers have started to develop platforms and interfaces to support scientific and technical exchange.

We will present an approach to implement the core models platform independent in *C/C++* or *Fortran*, while using *R* for model coupling, data management, numerical treatment and visualization. The partial differential equations of the models were solved with package **deSolve** [7] and matter transport with **ReacTran** [6]. This enables the separation of process equations from numerical techniques.

The feasibility of the approach is shown by coupling an ecological lake model (**SALMO**) describing nutrient turnover and growth of planktonic algae [1, 4] with a model for water plants, based on a lake model (**PC Lake**) of another group, [2]. The coupled model (package **rSALMO**) was used in a case study for a stratified German lake [5], where the presence of submerged macrophytes resulted in significant improvement of water quality.

The case study shows, that the *R* language with its pool of application-specific packages is a powerful resource for scientific computing even beyond statistics. It can be used as an efficient general-purpose development platform for coupling of complex models and for sharing code and ideas.

References

[1] Benndorf, J. and F. Recknagel (1982). Problems of application of the ecological model SALMO to lakes and reservoirs having various trophic states. *Ecological Modelling 17*, 129–145.

[2] Janse, J. H., E. van Donk, and T. Aldenberg (1998). A model study on the stability of the macrophyte-dominated state as affected by biological factors. *Water Research 32*(9), 2696–2706.

[3] Mooij, W. *et al*. (2010). Challenges and opportunities for integrating lake ecosystem modelling approaches. *Aquatic Ecology 44*(3), 633–667.

[4] Rolinski, S., T. Petzoldt, H. Z. Baumert, K. Bigalke, H. Horn, and J. Benndorf (2005). Das physikalisch-ökologisch gekoppelte Talsperrenmodell. *Wasserwirtschaft 95*, 34–38.

[5] Sachse, R., T. Petzoldt, M. Blumstock, S. Moreira, M. Pätzig, J. Rücker, J. H. Janse, W. M. Mooij, and S. Hilt (2014). Extending one-dimensional models for deep lakes to simulate the impact of submerged macrophytes on water quality. *Environmental Modelling and Software*, accepted.

[6] Soetaert, K. and F. Meysman (2012). Reactive transport in aquatic ecosystems: Rapid model prototyping in the open source software *R*. *Environmental Modelling and Software 32*(0), 49–60.

[7] Soetaert, K., T. Petzoldt, and *R*. Woodrow Setzer (2010). Solving differential equations in *R*: Package **deSolve**. *Journal of Statistical Software 33*(9), 1–25.

An R tools platform in Cosmetic Industry*

Jean-François COLLIN

Contact author: jfcollin@live.fr

Keywords: GUI, platform

In our company there are hundreds of in vitro, vivo and ex vivo studies per year. They range from efficacy to safety results on raw materials and formulae. All these studies must be analyzed, the results must be stored and we must keep track of the programs used to have a reproducible research. In this context we choose to create a statistical platform with specific tools for researchers so they can make their analysis by themselves and more general tools for statisticians so they can quickly produce accurate statistical reports.

This platform contains mainly tools that are built entirely within R (GUI's and statistical programs). In this presentation we would like to show the platform and its functionalities with some technical aspects concerning the technology we choose. We also like to show the tools we build to make exploratory analysis, mixed model analysis, an example of a 'non statistician' tool and a presentation of some packages that we build exclusively for our purpose.

The purpose of this conference is to show how we've been able to grasp the incredible potential of R to analyze our data, and to show that today R has concrete and very effective applications in the Industry.

*The name of our company will be told once we got the approval of our own scientific committee.

An R tools platform in Cosmetic Industry

Jean-François COLLIN[1]

1. L'Oréal Research and Innovation, Aulnay-sous-bois, France
*Contact author: jfcollin@rd.loreal.com

Keywords: Graphical User Interface**,** platform

In vitro, in vivo or ex vivo studies are continually performed on ingredients and formulae in our laboratories for efficacy and safety purposes. Results must be statistically analyzed, and stored together with the programs used affording a sustainable research policy. In such a context, a statistical platform was created including R (a language and environment for statistical computing) tools that integrate automatic reporting system connected to a centralized data basis.

Some applications aim at helping researchers to carry on analysis of their data by their own whereas more general tools are dedicated to statisticians for quickly producing accurate statistical reports with a wide variety of statistical methods: mixed models, multidimensional analysis, exploratory analysis etc…

This platform comprises tools that are entirely built within R (Graphical User Interface and statistical programs). This presentation will focus upon the platform, its functionalities and technical aspects. The tools built to perform exploratory analysis and mixed model analysis will be presented. Examples of a 'non statistician' tool and some packages specifically conceived for answering to our laboratories needs will be given.

BCP Stability Analytics and Markov Chain Monte Carlo

Tobias Setz[1,*], Jan Hendrik Witte[2], Diethelm Würtz[1]

1. Swiss Federal Institute of Technology, Zurich, Switzerland
2. Record Currency Management, Windsor, UK
*Contact author: tobias.setz@rmetrics.org

Keywords: Portfolio, Bayesian Change Points, MCMC, Shiny, Forex Markets

Modern Portfolio Theory goes back to Harry Markowitz [1]. When he published his article more than half a century ago, our knowledge of mathematical finance, econometrics, and statistics, as well as computer science, was much less developed than the options and the tools we have available today. We would like to share new ideas based on modern concepts of stability analytics, which allow for an alternative view on performance and risk in funds and portfolios and their impact on indexation techniques and tactical asset management.

The main topics we would like to address are based on statistical methods for the identification of instabilities and vulnerabilities in the dynamics behind financial markets. Our approach (Bayesian Change Point (BCP) Stability Analytics [4]) is based on the work of Barry and Hartigan [2] about Bayesian change point detection and parameter estimation. The method makes use of Bayesian Statistics and a Markov Chain Monte Carlo approach (implemented by Erdman and Emerson [3]).

The analytics can be used to explore financial markets and financial investments before, during and after critical financial and economic periods (e.g. the recent sub-prime or European debt crises). We will demonstrate how such vulnerabilities to external forces can be detected, analyzed and quantified. Thus, we can define figures to measure the structure and strength of instabilities appearing over time.

As a practical example, we assess the fragility of different currencies in spot and forward FX markets. We show ideas on how to construct wealth protected FX indices and how they can be combined in an FX portfolio. As a valuable tool to visualize the results we will additionally demonstrate an *R* **shiny** web application.

References

[1] Markowitz Harry (1952). Portfolio Selection. Journal of Finance Vol. 7 No. 1, 77-91.

[2] Barry Daniel, and Hartigan John A. (1993). A Bayesian Analysis for Change Point Problems. Journal of the American Statistical Association 35, 309-319.

[3] Erdman Chandra, and Emerson John W. (2008). A fast Bayesian change point analysis for the segmentation of microarray data. Bioinformatics 24, 2143-2148.

[4] Würtz Diethelm, Chalabi Yohan, Ellis Andrew, and Theussl Stefan (2010). Proceedings of the Singapore Conference on “Computational Finance and Financial Engineering”, pp. 205 – 213.

Don't Optimize! - Portfolios with Bayesian Change Point Analytics

Diethelm Würtz[1,*], Tobias Setz[1] Venetia Christodoulopoulou[1]

1. Swiss Federal Institute of Technology, Zurich, Switzerland
*Contact author: diethelm.wuertz@rmetrics.org

Keywords: Portfolio, Bayesian Change Points, MCMC, Shiny

In this talk we present a new unconventional method based on a predictive Bayesian Change Point (BCP) Stability Analytics [4] and Markov Chain Monte Carlo method ([2] and [3]) to design portfolios. Our approach makes optimization obsolete and comes with many additional advantages compared to standard investment strategies [1]. The portfolios are characterized by a high degree of stability of the underlying price process resulting in a steady increase of returns, low drawdowns, short recovery times, and low volatilities.

Two examples are presented: (i) an Euro based ETF portfolio build from Large and Small Cap Equities, REITS, and Government Bonds, and (ii) an USD based sectored portfolio of MSCI Emerging Market Equities. All calculations were done in *R* using the **Rmetrics** package family. Furthermore, a real time *R* **shiny** web application which visualizes stability forecasts and portfolio rebalancing will be demonstrated.

References

[1] Markowitz Harry (1952). Portfolio Selection. Journal of Finance Vol. 7 No. 1, 77-91.

[2] Barry Daniel, and Hartigan John A. (1993). A Bayesian Analysis for Change Point Problems. Journal of the American Statistical Association 35, 309-319.

[3] Erdman Chandra, and Emerson John W. (2008). A fast Bayesian change point analysis for the segmentation of microarray data. Bioinformatics 24, 2143-2148.

[4] Würtz Diethelm, Chalabi Yohan, Ellis Andrew, and Theussl Stefan (2010). Proceedings of the Singapore Conference on “Computational Finance and Financial Engineering”, pp. 205 – 213.

Plotly: Online Plotting and Collaboration with R

Chris Parmer [1], Matt Sundquist[1,*]

1. Plotly
*Contact author: "mailto:Matt@plot.ly", Matt@plot.ly

Keywords: Plotting, GUI, Visualization, Open Science, Open Data.

Plotly is an online plotting platform. Think of it like GitHub, but for sharing data, graphs, and scripts for plotting. Plotly has a GUI and APIs for making graphs with *R*, *Python*, *MATLAB*, *Perl*, *Julia*, *Arduino*, *Ruby*, *Raspberry Pi*, and *REST*. The APIs let users make and share web-based graphs and interface a desktop environment with Plotly. Public sharing is free, users own their data, and users control whether data and graphs are public or private.

Using Plotly and *R* to Make and Share Graphs

For our talk, we would like to demonstrate our *R* API. We will show how to import and graph data and use *R* and the Plotly GUI for graphing, sharing, and embeding graphs. We will also use Plotly with **knitr** to make an **RPub** [1], use the **maps** package to create interactive Plotly graphs [2] , and use **ggplot2** to make Plotly graphs with Plotly's own **ggplotly** package. We will demonstrate how to use Plotly in conjunction with **rOpenSci**, a package that Plotly is now a part of. We will also demonstrate the use of Plotly as an interactive graphing element when used with **Rmagic** and **IPython**.

These are currently available features. By the time of the conference, we anticipate having a more robust **ggplotly** experience available. The RStudio team informed us that they plan to sandbox their viewer. Currently you cannot serve web content into the viewer, but with a sandbox, we could serve Plotly graphs in an iframe into the viewer, pairing graphs, data, and scripts together online. Authors, and journalists from the Washington Post [5] and Wired Science [6] use these Plotly features and the capacity to embed graphs in an iframe.

References

[1] Dylan Matthews (2013). HYPERLINK "washingtonpost.com/blogs/wonkblog/wp/2013/06/14/do-low-taxes-on-the-rich-leave-the-middle-class-with-lower-wages/" washingtonpost.com/blogs/wonkblog/wp/2013/06/14/do-low-taxes-on-the-rich-leave-the-middle-class-with-lower-wages/.

[2] Rhett Allain (2014). HYPERLINK "www.wired.com/wiredscience/2014/02/much-real-olympic-gold-medal-cost/" www.wired.com/wiredscience/2014/02/much-real-olympic-gold-medal-cost/.

Imputation of Missing Values with the R Package VIM

Matthias Templ[1,2,3,⋆], **Alexander Kowarik**[1,3]

1. Statistics Austria & Vienna University of Technology
2. Vienna University of Technology
3. data-analysis OG
⋆Contact author: matthias.templ@gmail.com

Keywords: Missing Values, Imputation, Survey, R

The package **VIM** [4, 3] is developed to explore and analyze the structure of missing values in data using visualization methods, to impute these missing values with the built-in imputation methods and to verify the imputation process using visualization tools, as well as to produce high-quality graphics for publications.

The most common imputation methods such as hotdeck, kNN and regression imputation are implemented, but also more advanced methods like iterative robust regression estimation are available.

A point- and click graphical user interface has been developed to give access to these methods and tools to users with limited R skills. It is available in the package **VIMGUI** [2]. All important methods are supported by the flexible point- and click-interface.

The presentation describes the application of the methods available in the package **VIM** and demonstrates the usage of the graphical user interface of the **VIMGUI** package. Special attention is also given to the VIM integration of survey [1] objects.

References

[1] Lumley, T. (2012). *survey: analysis of complex survey samples*. R package version 3.28-2.

[2] Schopfhauser, S., M. Templ, A. Alfons, A. Kowarik, and B. Prantner (2013). *VIMGUI: Visualization and Imputation of Missing Values*. R package version 0.9.0.

[3] Templ, M., A. Alfons, and P. Filzmoser (2012). Exploring incomplete data using visualization techniques. *Advances in Data Analysis and Classification 6*, 29–47.

[4] Templ, M., A. Alfons, A. Kowarik, and B. Prantner (2013). *VIM: Visualization and Imputation of Missing Values*. R package version 4.0.0.

BBRecapture for capture-recapture data modelling with behavioural effects

Danilo Alunni Fegatelli[1], **Luca Tardella**[2*]

1. Department of Public Health and Infectious Diseases, Sapienza Universit di Roma (Italy)
2. Department of Statistics, Sapienza Universit di Roma (Italy)
*Contact author: luca.tardella@uniroma1.it

Keywords: Mark-recapture; Behavioral response; Ecological model; Memory effect; Bayesian inference

This **BBRecapture** package has been built up to help researchers to fit some relevant classes of capture-recapture models within the framework of Bayesian inference. Special emphasis is given on recently developed tools to take into account flexible behavioral response to capture. The main function developed in the package relies on the generalized linear model framework in the spirit of Huggins [6] and Alho [1] for regressing the capture occurrence on previous partial capture histories although shortcuts have been embedded to reduce computational complexity whenever possible. There are also some functions which fit the same class of models maximizing the unconditional likelihood as opposed to the most frequently used approach based on the conditional likelihood [5]. There are theoretical arguments related to the so-called *likelihood failure* [3, 4] which support the use of a Bayesian approach for the estimation of the unknown population size in the presence of behavioral response to capture. Some simulation studies have been also carried out in [2] to highlight the occurrence of the likelihood failure pathology and the loss of inferential performance of the conditional likelihood approach even in the absence of failure. In the same circumstances the unconditional likelihood approach should be preferred to the conditional likelihood but it is in any case outperformed by the Bayesian approach. Functions in the package are designed to allow minimal efforts by the researcher although optional arguments often allow for a more customized and refined model building.

References

[1] Alho, J. M. (1990). Logistic regression in capture-recapture models. *Biometrics 46*, 623–635.

[2] Alunni Fegatelli, D. (2013). *New methods for capture-recapture modelling with behavioural response and individual heterogeneity*.

[3] Alunni Fegatelli, D. and L. Tardella (2013). Improved inference on capture recapture models with behavioural effects. *Statistical Methods & Applications 22*(1), 45–66.

[4] Carle, F. L. and M. R. Strub (1978). A new method for estimating population size from removal data. *Biometrics 34*, 621–630.

[5] Huggins, R. and W. Hwang (2011). A review of the use of conditional likelihood in capturerecapture experiments. *International Statistical Review 79*(3), 385–400.

[6] Huggins, R. M. (1989). On the statistical analysis of capture experiments. *Biometrika 76*, 133–140.

Image analysis and statistics: an introduction using *R* and RIPA

Talita Perciano[1,*]

1. Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, United States
*Contact author: tperciano@lbl.gov

Keywords: *R*, statistics, image analysis, **RIPA**, high performance

R is a public domain, open source, language and environment for statistical computing and graphics [4]. As a tool, *R* is very popular among scientists because it is free; it includes a diverse set of algorithms, and entails a large and broad community that participates in augmenting the tool. *R* provides a large variety of statistical techniques such as linear and non-linear models, classical statistical tests, time series models, classification and clustering. Moreover, it offers various graphical techniques and it is highly extensible. *R* is an integrated collection of software packages for data manipulation, performing calculations and graphics production.

The field of digital image processing refers to any process applied to a digital image performed by a computer [2]. A 2D image can be defined as a bidimensional function $f(x, y)$, where x and y are the spatial coordinates, and the amplitude of f at a coordinate (x, y) is called the intensity or the gray level of the image at that point. An image has a finite number of elements, and each one of them has a particular position and a scalar or vectorial value. For instance, a gray level image has a unique value for each position. This definition can be easily extended to n-dimensional images. This field of study is very broad, comprising numerous operations, from common ones such as contrast enhancement and noise reduction, to more complex ones such as image segmentation/classification and pattern recognition. The range of applications which can take advantage of image processing techniques is immense and its importance to scientific research increases drastically. Moreover, the advent of Big Data and Data Science [3] demands the use and development of such techniques in order to discover knowledge from datasets in several research fields: Geoscience and Remote Sensing, Bioinformatics, Biology, Medicine, Physics, Astronomy, Geology, to name a few.

RIPA [5] is a user friendly package that provides several image processing tools, which can be applied to different types of images such as binary, gray level, color and multispectral images. Analysis and exploration tools, such as those provided by **RIPA** can be applied to diverse domains and can be very useful and important in many challenging imaging problems. Besides a general tool that can be used in such manner, **RIPA** is presented here also with the purpose to provide an auxiliary tool for learning statistics and image analysis. This package, along with the book entitled "Introduction to Image Processing Using *R*: learning by examples" [1], has been used as a text book around the world for disciplines such as "introduction to image processing" and "introduction to statistics". As its new version comes up with new image analysis tools and with high performance additions, it becomes a valuable tool both for teaching and scientific research purposes involving Big Data.

References

[1] Frery, A. C. and T. Perciano (2013). *Introduction to Image Processing Using R: learning by examples (SpringerBriefs)* (1 ed.). Springer. ISBN: 978-1447149491.

[2] Gonzalez, R. C. and R. E. Woods (2008). *Digital Image Processing* (3 ed.). Prentice Hall.

[3] Hey, T., S. Tansley, and K. Tolle (Eds.) (2009). *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Redmond, Washington: Microsoft Research.

[4] Hornik, K. (2002). The R Project for Statistical Computing. http://www.r-project.org. Accessed March 25, 2014.

[5] Perciano, T. and A. C. Frery (2009). R Image Processing and Analysis. http://cran.r-project.org/src/contrib/Archive/ripa/. Accessed March 25, 2014.

The OpenCPU system: towards a universal interface for scientific computing

Jeroen Ooms[1,*]

1. UCLA Statistics
*Contact author: jeroen.ooms@stat.ucla.edu

Keywords: Systems, Embedded Scientific Computing, Reproducible Research, Web Applications

Even though bridges to embed `R` in general purpose software have been available for several years, they have not been able to facilitate the big break through of `R` as a ubiquitous statistical engine. In my experience, the primary cause for the limited success is that low-level tools are difficult to implement, do not scale very well, and leave the most challenging problems unsolved. Substantial plumbing and expertise of `R` internals is required for building actual applications on these tools. What is needed to scale up embedded scientific computing is a system that seperates the application layer from the computational back-end, similar to how e.g. `SQL` separates application from database. The `OpenCPU API` defines an interface that captures the domain logic of scientific computing and abstracts implementation details, in a way that allows for independent development of client and server components, by different people that do not speak each others language.

References

[1] Dirk Eddelbuettel, Murray Stokely, and Jeroen Ooms. `RProtoBuf`: Efficient Cross-Language Data Serialization in R. *arXiv:1401.7372*. URL http://arxiv.org/abs/1401.7372.

[2] Jeroen Ooms. The `jsonlite` Package: A Practical and Consistent Mapping Between `JSON` Data and `R` Objects. *arXiv:1403.2805*. URL http://arxiv.org/abs/1403.2805.

[3] Jeroen Ooms. The `RAppArmor` Package: Enforcing Security Policies in R Using Dynamic Sandboxing on Linux. *Journal of Statistical Software*, 55(7):1–34, 2013. URL http://www.jstatsoft.org/v55/i07/.

[4] Jeroen Ooms. Possible Directions for Improving Dependency Versioning in `R`. *The R Journal*, 5/1, June 2013. URL http://journal.r-project.org/archive/2013-1/ooms.pdf.

Enhancing Medical Reporting by Combining Electronic Health Records with REDCap: Applications of the REDCap API

Benjamin Nutter[1*]

1. Cleveland Clinic Foundation, Quantitative Health Sciences
*Contact author: nutterb@ccf.org

Keywords: REDCap, API, Electronic Health Records

Electronic health records (EHR) can be a rich source of complex data. Unfortunately, much of the most interesting data is buried in text notes that are not easily parsed into discrete data. Combining the discrete data elements with the non-discrete usually requires a clinician to perform a chart review. Although chart reviews can be costly in terms of time and money, these costs can be reduced by providing clinical staff with narrowly focused patient lists that are prepopulated with the available discrete data from the EHR. This allows the clinical staff to spend more time collecting data that cannot be easily obtained electronically. Providing these lists for clinicians can be done dynamically with ongoing registries to monitor quality of patient care and medical outcomes. The *R*esearch *E*lectronic *D*ata *Cap*ture system (REDCap) and its API provide a mechanism by which select data may be uploaded from the health records to a REDCap database to populate as many informative fields as possible. Clinicians may then focus on the narrowly selected patient population and collect only the data that cannot be queried from the health records. This presentation highlights the process used by the Cleveland Clinic Ob/Gyn and Women's Health Institute to streamline data collection and improve reports on patient management and quality of care using the **redcap** and **RODBC** packages.

References

[1] Harris, P. A., R. Taylor, R. thielke, J. Payne, N. Gonzalez, and J. G. Conde (2009). Research electronic data capture (redcap) - a metadata-driven methodology and workflow process for providing translational research informatics support. *Journal of Biomedical Informatics 42(2)*, 377–81.

[2] Horner, J. (2012). *redcap: R interface to REDCap*. R package version 0.1. (available on GitHub).

[3] Pickett, M., A. Milinovich, and S. John (2013). Creating and managing research registries and datasets from a clinical data repository. *AMIA Clinical Research Informatics Summit*.

[4] Ripley, B. and from 1999 to Oct 2002 Michael Lapsley (2012). *RODBC: ODBC Database Access*. R package version 1.3-6.

A real time, responsive Quantitative trading analysis Mobile App using R

Nilesh N. Shah[1]

1. Investygator Inc.
*Contact author: nilsha@yahoo.com

Keywords: JQuery Mobile App,AWS, Quantitative Strategy, Web Services, Sentiment Analysis.

Quantitative Trading is typically associated with Hedge Funds, high frequency trading firms/ "Quant shops", and knowledgeable investment professionals. The sophisticated individual investor now has access to several specialized software tools, assuming they know what they're looking for and how to interpret the results. That still leaves a gap for the common every day person who, while not armed with sophisticated knowledge, would still like to have sophisticated information available to help them make investment decisions.

We present an easily extensible/scalable Cross platform, real time, responsive, Quantitative trading analysis Mobile App using R as the primary backend compute engine. This App allows the user to ask investment questions and receive responses in simple English. The backend R engine runs quantitative algorithms to answer the questions. The Mobile App consists of a "Front End" User interface, implemented using the popular JQuery Mobile framework, and a backend server hosted on Amazon AWS, running R on a headless Linux server. The app utilizes several R modules (quantstrat, quantmod, sentiment, RMySQL) for data storage, retrieval and analysis, as well as custom algorithms. We also use FastRWeb, a Fast Interactive Web Framework in R. For real time ticker lookup, we use PHP/ Ajax. Data is delivered from the server to the App Front End user interface using JSONP, a popular Web Service data exchange format. This enables remote procedure calls to be made from the Web App to the server running R, and data returned back to the app using a JSONP callback function.

References

[1] Jay Emerson (2011). *Setting up FastRWeb/Rserve on Ubuntu* , http://jayemerson.blogspot.com/2011/10/setting-up-fastrwebrserve-on-ubuntu.html.

[2] Timothy P. Jurka **(2012)**. **Sentiment**: **sentiment** analysis including bayesian classifiers for positivity/negativity and emotion classification, http://www.icesi.edu.co/CRAN/web/packages/sentiment/sentiment.pdf**.**

[3] Simon Urbanek, Jeffrey Horne (2012). *FastRWeb*: *Fast Interactive Framework for Web Scripting Using R*, http://cran.r-project.org/web/packages/FastRWeb/index.html.

[4] Humme, Peterson (2013). R in Finance Conference presentation "*Using quantstrat*" http://www.rinfinance.com/agenda/2013/workshop/Humme+Peterson.pdf.

[5] The Jquery Foundation *JQuery Mobile, A Touch Optimized Web framework*. http://jquerymobile.com/.

An R package for creating Microsoft Word, Power Point and HTML documents

David Gohel[1*]

1. Lysis-consultants
*Contact author: david.gohel@lysis-consultants.fr

Keywords: tables, graphics, reporting, reproducible research

The ReporteRs package provides a framework that allows users to create Microsoft Word (>=2007), Power Point (>=2007) and HTML documents. It makes easy to add pretty output from *R* in a Microsoft Word or Power Point document with a particular corporate template. ReporteRs can be used as a tool for fast reporting within *R* and can be used as a reporting automation tool.

A rich API is available for producing documents with simple or sophisticated tables, formatted texts, raster or editable vector graphics, those functions are implemented for the three document types available. There are also functions dedicated to specific contents (e.g. a table of content in a Word document, a slide in a Power Point presentation). By default, content is added at the end of a copy of the template; with a Word document or a Power Point presentation; content can be inserted at a specific location (a bookmark or a slide number).

The only requirement is to have a Java Runtime on the machine, it makes it easy to deploy in business environments.

This talk will include an introduction to ReporteRs, simple and complex reporting demonstrations and future plans.

iwplot: An R Package for Creating web Based Interactive Graphics for Big Data

Ganesh Subramaniam, Todd Larchuk, Simon Urbanek and Robert Archibald

AT&T Labs - NJ, USA
*Contact author: mkg@research.att.com

Keywords: iPlots, Interactive R Graphics, RCloud, Big Data

We discuss **iwPlot**, an *R* package that provides a graphical environment on the browser for exploring big data. This was motivated by **iPlots** (Urbanek and Theus 2003), a package that provides advanced interactive features. **iPlots** provides several key interactive features like selection of individual points or subsets of data, as well as zooming, brushing and linking between different plots. The existing **iPlots** package is limited to R running in a standard desktop environment. With the growing popularity of RCloud and other *R* applications that are deployed on the browser for the consumption of end users, there is a need for developing interactive plots in the browser environment. The **iwPlot** package brings some of these advanced interactive features like selections, zooming, brushing to a web browser. The other consideration is **iwPlot**'s ability to handle big data. The advanced interactive features that are currently available in the desktop environment in **iPlots** package are extended to the browser environment. The graphics in **iwPlots** package is rendered using the HTML5 Canvas API. This API is used by writing JavaScript that can access the canvas area through a full set of drawing functions, thus allowing for dynamically generated graphics. The current architecture can accommodate large amounts of data. The other benefit of this interactive environment is the ability to deploy *R* applications that runs higher order analytics to end users who are not *R* users. We will demonstrate this by showing two web applications, one that uses functional data analysis for informal classification for a large collection of time series and other application that involves spatial temporal data.

References

Urbanek. S, Theus. M (2003). *iPlots: High Interaction Graphics for R*. In *Proceedings of the 3rd International Workshop on Distributed Statistical Computing*.

Subramaniam. G and Varadhan. R (2007). *Feature Extraction Using FDA*. *JSM 2007, (Salt Lake City, UT,USA)*, Aug. 2007.

Varadhan. R and Subramaniam. G (2009). *Automatic Numerical Differentiation of Noisy, Time-Ordered Data in R*. In *useR! 2009, The R User Conference, (Renne, France)*, Jul. 2009.

Subramaniam. G, Varadhan. R, Urbanek. S and Epstein. S (2010). *tsX: An R package for the exploratory analysis of a large collection of time-series*. In *useR! 2010, The R User Conference, (Gaithersburg, USA)*, Jul. 2010.

rctrack: An R that Package Automatically Collects and Archives Details for Reproducible Computing

Stan Pounds* **and Zhifa Liu**

Department of Biostatistics, St. Jude Children's Research Hospital, Memphis, TN
*Contact author: stanley.pounds@stjude.org

Keywords: Reproducible Computing, Automatic Archiving

It is scientifically and ethically imperative that the results of statistical analysis of biomedical research data be computationally reproducible in the sense that the reported results can be easily recapitulated from the study data. Literate programming tools such as **Sweave** [1] and **knitR** [2] are very useful tools for reproducible computing that internally document how analysis results were generated and transferred into a report file. However, literate programming tools do not archive the supporting data files, program files, code library files, and other details that are subject to updates, corrections, or other modifications over time. These details must be archived in order to ensure that a particular statistical analysis is computationally reproducible at a later time.

Therefore, we developed the **rctrack** package [3] that automatically collects and archives read only copies of program files, data files, and other details needed to computationally reproduce an analysis. The **rctrack** package uses the `trace` function to temporarily embed detail collection procedures into functions that read files, write files, or generate random numbers so that there is no need to modify the R program that performs the statistical analysis. At the conclusion of the analysis, **rctrack** uses these details to automatically generate a read only archive of data files, program files, result files, and other details needed to recapitulate the analysis results. Information about this archive may be included as an appendix of a report generated by **Sweave** or **knitR**. Here, we describe the usage, implementation, and other features of the **rctrack** package.

References

[1] Leish F (2002). Sweave: Dynamic generation of statistical reports using literate data analysis. In *Compstat 2002 – Proceedings in Computational Statistics* (Heidelberg, Germany), pp. 575-580.

[2] Xie Y (2013) knitR, http://yihui.name/knitr/.

[3] Liu Z and Pounds S (2014) rctrack: An R package to Track Details for Reproducible Computing, http://www.stjuderesearch.org/site/depts/biostats/rctrack.

How Popular is R?

Robert Muenchen[1,*]

1. The University of Tennessee
*Contact author: muenchen.bob@gmail.com

Keywords: r-project, popularity, market share

R is popular software, but how popular? Compared to what? Without metrics like *sales* at our disposal, measuring open source software popularity is an imperfect science. This talk will cover several ways of measuring R's popularity[1], including: counting prevalence of relevant job advertisements [2], scholarly articles, books, blogs, examining software usage surveys, discussion forum activity, and more. Each of these methods has inherent advantages and disadvantages, and each shows *R* in a slightly different competitive position. We will examine each of these approaches and consider the latest forecasts regarding when *R* will surpass the current analytics market leaders. Finally, a new *R* package for collecting popularity measures for any software will be presented.

[1] Muenchen, R.A. (2014). The Popularity of Data Analysis Software, http://r4stats.com/articles/popularity/.

[2] Muenchen, R.A. (2014). How to Search for Analytics Jobs, http://r4stats.com/articles/how-to-search-for-analytics-jobs/.

Zillow's Big Data and Real-time Services in R

Zillow is the leader in providing data and analysis for consumers and professionals in the real estate market place. The flagship product is the Zestimate™, an estimate of the value for over 90 million homes in the US. Millions of models are behind the Zestimate and the models are constantly being retrained. The model fitting and scoring infrastructure rests on top of R, allowing rapid prototyping and deployment to production servers. In order to more fully integrate the Zestimate infrastructure with the Zillow production environment, we have developed ZillowRServe, a client/server interface based on the RServe package. ZillowRServe allows the Zestimate to operate as a service as well as providing basic monitoring and controlling of the Zestimate application. ZillowRServe provides a simple generic application that can automatically spawn and manage a number of parallel R processes. It also provides utilities specific to Zillow for dealing with real-estate data. For false tolerance and load balancing, we deploy multiple redundant ZillowRServe on multiple machines. We also embedded a ZillowRServe client in stored procedures for use within SQL server to handle real-time data import.

R and Reproducibility: a Proposal

David Smith[1,*], Joseph Rickert[1]

1. Revolution Analytics
*Contact author: david@revolutionanalytics.com

Keywords: R, reproducibility, packages, CRAN

The R ecosystem is in a state of near constant change. While a new version of the R engine is now released just once a year, 2-3 patches are usually released in the interim. On top of that, new versions of R packages on CRAN are released at rate of several per day (and that's not counting packages that are part of the BioConductor project or hosted elsewhere on the Web).

While this rapid change is a boon for the advancement of R, it can cause problems for package authors[1] and also for scientists and their peers who may need to reliably reproduce the results of an R script (possibly dependent on a number of packages) months or even years down the line. In this talk we propose a downstream distribution of R and CRAN packages that provides for the reproducibility of R scripts and reduces the impact of dependencies for packages authors.

References

[1] Ooms, Jeroen (2013) "Possible Directions for Improving Dependency Versioning in R", *The R Journal* Vol. 5/1, June 2013

Abstract for UseR! 2014

RLint: Reformatting R Code to Follow the Google Style Guide

Alex Blocker[1], Andy Chen[1]*, Andy Chu[1], Tim Hesterberg[1], Jeffrey D. Oldham[1], Caitlin Sadowski[1], Tom Zhang[1]

[1] Google Inc.
*Contact author: andych@google.com

Keywords: lint, rlint, R style, R format, R

RLint (proposed external location:https://code.google.com/p/google-rlint/) both checks and reformats R code to the Google R Style Guide. It warns of violations and optionally produces compliant code. It considers proper spacing, line alignment inside brackets, and other style violations, but like all lint programs does not try to handle all syntax issues.

Code that follows a uniform style eases maintenance, modification, and ensuring correctness, especially when multiple programmers are involved. Thus, RLint is automatically used within Google as part of the peer review process for R code. We encourage CRAN package authors and other R programmers to use this tool. A user can run the open-source Python-based program in a Linux, Unix, Mac or Windows machine via a command line.

LivelyR: Making R charts livelier

Aran Lunzer[1], **Amelia McNamara**[2*], **Robert Krahn**[3]

1. Viewpoints Research Institute
2. University of California-Los Angeles
3. SAP
*Contact author: amelia.mcnamara@stat.ucla.edu

Keywords: Visualization, statistics education, subjunctive interfaces

Traditional *R* graphics are static; if you want to modify an aspect of a graphic or the data preparation leading to it, you must edit and re-run your code. Even working in an interactive `R` console, this process is laborious, and the resulting sequence of static outcomes is unhelpful for understanding what has changed from one to the next. Thus users are discouraged from parameter tinkering. For example, when you draw a histogram in *R* but don't specify a bin width, a default will be chosen for you. However, for a given dataset the choice of the bin width and the origin of binning can lead to dramatically different-looking histograms. If you don't tinker, you may not realise that your histogram is failing to reveal an important story in your data, or that the story it appears to tell is no more than an accident of the chosen bin divisions.

`LivelyR` builds on and extends **shiny** [3] and **ggvis** [4], which work together to bridge from *R* to the interaction possibilities of *Javascript* code running in a web browser. In our browsers we run `Lively` (in full `Lively Web` [1]), a full-fledged *Javascript* programming environment with rich facilities for building interfaces and for hooking into external applications (such as `R`) and web-page display abstractions (such as `SVG` and `d3`). `LivelyR` is, at this point, a research exploration into how a powerful combination such as this can be used to further enliven `R` charts by (a) turning rendered chart elements into interactive controls over the chart; and (b) providing built-in support for side-by-side comparison of alternative chart settings, based on subjunctive interface [2] techniques and a manipulable history of the user's interactions.

Unlike **ggvis**, all `LivelyR` execution is initiated and controlled from the *Javascript* side. From `Lively` we start an `R` process to act as a server that accepts fragments of *R* code for processing, and returns to `Lively` any textual results. A `LivelyR` web page uses this channel to start a **shiny** session and specify the dataset and desired charts; this session thereafter communicates with `Lively` through a web socket, using an extended form of the **ggvis** protocols.

In this talk we will demonstrate the latest version of `LivelyR`, showing various lively chart examples—including our approach to helping a user choose the parameters for a histogram—and will talk about where we imagine this type of system being useful. For example, we envisage using it to assist students in introductory statistics classes to understand the purpose of *R* functions' parameters, by making it easy to manipulate parameter values and see how the results are affected. Similar features could also be integrated into the charts in a newspaper article or academic paper, allowing a reader to adjust parameter values in the code and see how the authors' chosen values affect the analysis outcome. It would allow for a sort of code audit.

Acknowledgements: We thank Bret Victor for ideas and inspiration, and Dan Ingalls and Alan Kay for the freedom to explore.

References

[1] Ingalls, D. and R. Krahn, et al. (2013). The Lively Web. `http://lively-web.org/`.

[2] Lunzer, A. and K. Hornbæk (2010). Subjunctive interfaces for the Web. In A. Cypher, M. Dontcheva, T. Lau, and J. Nichols (Eds.), *No Code Required: Giving Users Tools to Transform the Web*, pp. 267–285. Morgan Kaufmann Publishers.

[3] RStudio, Inc. (2013). *shiny: Web Application Framework for R*. R package version 0.8.0.

[4] RStudio, Inc. (2014). *ggvis: Interactive grammar of graphics for R*. R package version 0.1.0.99.

A team's story in the IMPROVER Species Translation Challenge

Adi L. Tarca[1,2] and Roberto Romero[2]

1. Wayne State University, Detroit, MI, USA;
2. Perinatology Research Branch, Perinatology Research Branch, *Eunice Kennedy Shriver* National Institute of Child Health and Human Development, NICHD/NIH, USA;

Keywords: gene expression, protein phosphorylation, pathway analysis, crowdsourcing, machine learning

The IMPROVER [1] Species Translation Challenge (STC) [2], organized by IBM Research and Philip Morris International, tested the limits of the fundamental assumption that underpins the use of animal models for gaining insight into human biology, i.e., that there is conservation in the nature of the responses to injury and therapy. Gene expression and protein phosphorylation data measured in rat and human cells after treatment with 26 stimuli were made available at https://www.sbvimprover.com. International teams were challenged to 1) use rat gene expression to predict rat protein phosphorylation, 2) use rat gene expression and protein phosphorylation to predict human protein phosphorylation, and 3) use rat gene expression to predict pathway activity in human. Teams were ranked based on performance on a test dataset generated from other 26 stimuli.

We participated in these challenges relying on good old friend *R* and *Bioconductor* packages to process data and build prediction models to address challenges 1 and 2, and derive a data driven homology between rat and human genes to tackle challenge 3. Our team's rank in these challenges was first, second, and third for sub-challenges 1, 3, and 2 respectively.

In this talk we will present the methods we have used in this crowdsourcing effort and the R code that we have developed to implement these methods. Lessons learned about the translatability of findings from rat to human and about best strategies to modeling sparse outcomes in STC will also be discussed.

1. Meyer P et al. (2011) Verification of systems biology research in the age of collaborative competition. *Nat.Biotechnol.* 29(9):811-815.

2. Rhrissorrakrai,K. et al. (2014) Understanding the limits of animal models as predictors of human biology: lessons learned from the sbv improver species translation challenge. *Bioinformatics*, in press.

Teaching R to high school students (and teachers)

Amelia McNamara[1*], **James Molyneux**[1]

1. University of California-Los Angeles
*Contact author: amelia.mcnamara@stat.ucla.edu

Keywords: Statistics education

As graduate students on the Mobilize grant [1], we have had the opportunity to interact with Los Angeles high school teachers and students who are trying to learn *R*. Over the years of the grant, we have learned some factors that can help or hinder learning of *R* at the high school level.

In this talk, we will reflect on lessons we have learned about teaching *R* to this particular audience. We will discuss some of the pedagogical decisions surrounding *R* that have been made in the various pieces of Mobilize curriculum, including the use of the **mosaic** package [3], as well as our own package of wrapper functions, **MobilizeSimple** [2].

References

[1] Mobilize. http://www.mobilizingcs.org/.

[2] McNamara, A. and J. Molyneux (2013). MobilizeSimple: Simple functions for Mobilize. github.com/mobilizingcs/MobilizeSimple. R package version 1.2.8.

[3] Pruim, R., D. Kaplan, and N. Horton (2013). *mosaic: Project MOSAIC (mosaic-web.org) statistics and mathematics teaching utilities*. R package version 0.7-30.

Computer Vision in R: Enabling Flyby Science at Comets and Asteroids

Thomas J. Fuchs*, David R. Thompson, Brian D. Bue, Julie Castillo-Rogez, Kiri L. Wagstaff

Jet Propulsion Laboratory, California Institute of Technology
*Contact author: thomas.fuchs@jpl.nasa.gov

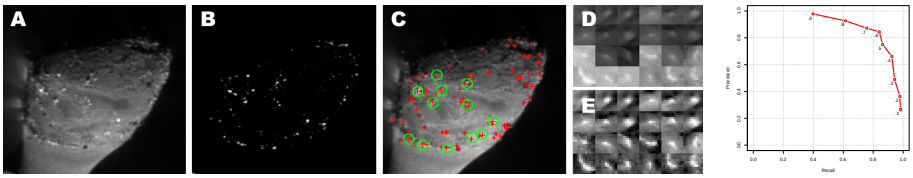
Keywords: Computer Vision, Space Exploration, Classification, Asteroids, Comets

Computer Vision is a vast field of research and integral part of scientific inquiry from biology and medicine to earth science and space exploration. While *R* provides an exceptional portfolio of statistical packages it has historically been very weak in computer vision. Some packages provide wrappers to external software like GDAL, ImageJ or ImageMagick but an *R* -specific computer vision framework is still missing to date. To close this gap we are developing **visionaRy** . The purpose is to provide an *R* package which allows for fast prototyping of image intensive algorithms in *R* with standard *R* data structures and standard *R* syntax. We are convinced that the benefit of directly manipulating images and easily using the complete statistical machinery of *R* outweighs the loss of processing speed for a wide range of scientific scenarios. Furthermore, since this implementation is based on standard *R* structures replacing specific methods in **Rcpp** is straightforward.

While the package is intended for a general computer vision and image processing audience some methods are tailored towards applications in space exploration. To this end **visionaRy** contains methods for retrieving imagery from NASA's Planetary Data System (PDS: pds.nasa.gov) and for decoding raw data from space missions stored in JPL `.IMG` format.

Specifically, we represent images as arrays within a `Reference Class` which allows for image manipulation in place without unnecessary copying of data. Based on that we can provide an extensive set of *drawing* methods for geometric primitives like lines, circles, or rectangles which enables image annotation by modifying pixel intensities in the array itself. Compared to classic *R* plotting operation on some output device, *drawing* yields precise annotations in the same coordinate frame in which all algorithms operate.

visionaRy provides a unified interface for reading and writing a broad range of file formats, image processing functionality such as color conversion, image overlays, and scaling as well as display functions on single and multiple panels, tiling and patch extraction. Furthermore **visionaRy** implements several computer vision algorithms ranging from integral images, 3D color histograms, and edge preserving median filters to binary and intensity weighted mean shift clustering in image space. Since the underlying data structure is a standard *R* array we can employ the full breath of *R* 's statistical tool chest for feature extraction and classification.



We demonstrate the capabilities of **visionaRy** with an application for flyby science at small bodies from JPL (cf. Fig.): **A:** First we retrieve images of comet Hartley 2 taken by the framing camera of the Deep Impact probe during the EPOXI mission from PDS. **B:** The difference of the grayscale and median filtered image is renormalized. **C:** Set of possible surface features as a result of weighted mean shift clustering (red crosses) and ground truth labels from a domain expert (green circles). **D:** Patches of labeled surface features from 9P/Tempel. **E:** Locally normalized patches, constituting the positive class for training. **F:** Finally we train a random forest classifier to differentiate surface features from background and report precission and recall on the test set.

R in the Midst of Exploding Stars: Distributed, Time-Domain Transient Classification

Thomas J. Fuchs[1,2,*], **Michael Turmon**[1], **Matthew J. Graham**[2], **Ciro Donalek**[2], **Ashish Mahabal**[2]
Andrew J. Drake[2], **S. George Djorgovski**[2]

1. Jet Propulsion Laboratory, California Institute of Technology
2. California Institute of Technology
*Contact author: thomas.fuchs@jpl.nasa.gov

Keywords: Astronomy, Machine Learning, Transients, Time Series, Feature Extraction

Modern synoptic sky surveys are enabling novel research frontiers in time domain astronomy and posing new machine learning challenges for early detection and classification Donalek et al. [2]. We present a novel framework for time domain astronomy using *R*. Our system incorporates machine learning algorithms for an iterative, dynamical classification of astronomical transient events, based on the initial detection measurements, archival information, and newly obtained follow-up measurements from robotic telescopes. Specifically we use *R*for (i) preprocessing, (ii) to post observed light curves to the Caltech Time Series Characterization Service for feature extraction and (iii) for classification.

Astronomical time series tend to be heterogeneous. They can vary widely in their temporal coverage, sampling rates and regularity, number of points and error bars, even from the same survey, and hence need to be homogenized prior to analysis. For a large number of learning algorithms each time series needs to be recast in terms of a set of features – individual measurable heuristic properties of an object that can be used to characterize it. The Caltech Time Series Characterization Service (CTSCS: nirgun.caltech.edu:8000) aims to provide easy access to different time series characterization statistics used in the literature. These descriptors are then used to train and test a hierarchical model based on random forest classifiers. We compare various modeling choices and analyze the contribution of CTSCS features to the classification accuracy.

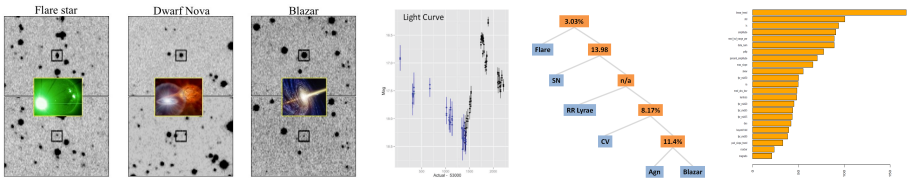


Figure 1: **Left:** Examples of transient events from the Catalina Real-time Transient Survey (CRTS: crts.caltech.edu) Djorgovski et al. [1]. Images in the top row show objects which appear much brighter that night, relative to the baseline images obtained earlier (bottom row). On this basis alone, the three transients are physically indistinguishable, yet the subsequent follow-up shows them to be three vastly different types of phenomena: a flare star (left), a cataclysmic variable powered by an accretion to a compact stellar remnant (middle), and a blazar, flaring due to instabilities in a relativistic jet (right). Accurate transient event classification is the key to their follow-up and physical understanding. **Center left:** A single light curve plotted with **ggplot**. **Center right:** Classification error of a hierarchy of random forest models. **Right:** Random forest based importance plot of CTSCS features.

References

[1] Djorgovski, S. G., A. Mahabal, C. Donalek, M. J. Graham, A. J. Drake, B. Moghaddam, and M. Turmon (2012). Flashes in a star stream: Automated classification of astronomical transient events. In *E-Science (e-Science), 2012 IEEE 8th International Conference on*, pp. 1–8. IEEE.

[2] Donalek, C., A. Kumar, S. G. Djorgovski, A. Mahabal, M. J. Graham, T. J. Fuchs, M. J. Turmon, N. S. Philip, M. T.-C. Yang, and G. Longo (2013). Feature selection strategies for classifying high dimensional astronomical data sets. In *BigData Conference*, pp. 35–41.

Recovering Risk Neutral Density from Options Using RND Package

Kam Hamidieh
USC Marshall Business School
& Zoolytics LLC
hamidieh@marshall.usc.edu

Keywords: Options, Finance, Risk Neutral Density, S&P 500

Practitioners and researchers in finance are often interested in recovering the risk neutral density implied by market options. The risk neutral density can be used to obtain information about the underlying asset. The RND package is the only R package that brings together various methods for recovering the risk neutral density. Numerous examples illustrate how different methods can be used to recover the risk neutral density. With the practitioner in mind, a few simple functions can be used to create numerical and graphical summaries which then can be imported into various software such as Excel.

The peer performance of hedge funds

David Ardia[a,b,*], Kris Boudt[c,d]

[a]*Département de finance, assurance et immobilier, Université Laval, Québec, Canada*
[b]*Centre interuniversitaire sur le risque, les politiques économiques et l'emploi, Québec, Canada*
[c]*Solvay Business School, Vrije Universiteit Brussel, Belgium*
[d]*Faculty of Economics and Business, VU University Amsterdam, The Netherlands*

Abstract

An important component in the analysis of a (hedge) fund returns is to measure the fund's performance with respect to the group of peer funds. The industry standard is to rank funds based on their risk-adjusted return and conclude that the fund outperforms the peers with a lower percent rank. When all funds perform equally well, this rate of outperformance is a random number between zero and one, depending on how lucky the fund is. We use the false discovery rate approach to construct relative performance ratios that account for the uncertainty in estimating the performance differential of two funds. Our application is on hedge funds, which leads us to develop a test for equality of the modified Sharpe ratio of two funds. The effectiveness of the method is illustrated with a Monte Carlo study and an empirical study is performed on the Hedge Fund Research database. Our regression analysis shows that the larger the fund is, the more similar the fund returns are to its peers, and that small funds have a higher tendency to underperform.

Keywords: Equal–performance ratio, false discovery rate, hedge fund, modified Sharpe ratio, out–performance ratio, peer group, performance measurement

*Corresponding author.
Email addresses: david.ardia@fsa.ulaval.ca (David Ardia), kris.boudt@vub.ac.be (Kris Boudt)

Working with *R* and *SAS*: Some initial experiences from Statistics Norway

Susie Jentoft[1,*]

1. Statistics Norway
*Contact author: Susie.Jentoft@ssb.no

Keywords: Official Statistics, *SAS* software, Interactive Matrix Language, IML

Statistics Norway relies heavily on the statistical software *SAS* in its statistical production. While this is an excellent piece of software, *R* provides additional functionally and flexibility which is not always available in *SAS*. Recently, we have been exploring ways of incorporating and expanding the use of *R* within Statistics Norway. It is not currently viable to transfer all our statistical production over to *R*, however, *R* may be used in some steps of the process. One option is to utilize both of the software through calling *R* functions or scripts from within *SAS*. This way, we can add some of the best functionality of *R*, without the cost of re-writing all existing *SAS* programs.

In particular, I show examples of using *R* within our unified sampling system and creating *R* figures within data editing procedures. We explore a starting point of using `R CMD BATCH` as a step in a process that also includes *SAS* programs. We look at the *SAS* Interactive Matrix Language (*IML*) using `Proc IML` to call *R* functions from within the *SAS* environment. Some alternative *SAS* macros are also investigated. *SAS Enterprise Guide* is a relatively new tool within Statistics Norway, but provides a visual approach to statistical production with a lower programming threshold. It also provides an opportunity to incorporate discrete steps which call on *R* programs. Not only does this provide opportunities to utilize some of *R*'s best functions but is a way of softly introducing the program to new users.

Embedding Shiny Apps in R Markdown documents

Garrett Grolemund[1,*]

1. RStudio, Inc.
*Contact author: garrett@rstudio.com

Keywords: **rmarkdown**, **shiny**, **knitr**, RStudio, reports

Shiny and R Markdown provide two versatile platforms for generating reports from *R*. Shiny creates reactive web apps that allow users to explore a data set and initiate analysis in an interactive fashion. R markdown creates attractive static reports that are fully reproducible; they can be automatically regenerated whenever *R* code or data changes. R markdown achieves this by combining *markdown* (an easy-to-write plain text format) with embedded *R* code chunks that are run so their output can be included in the final document.

But can R markdown be extended further — can a user use it to embed live Shiny apps into an *R* report? If so, the results would be remarkable. Not only could *R* users augment their reports with live content in the form of Shiny Apps, but they could also include things that are less obviously Shiny apps, such as interactive **ggvis** plots.

However, this approach is fraught with challenges. These range from the simple to the subtle — How can multiple Shiny apps be included in a single web document? How should the resulting documents be deployed? How can we let users know they have a dynamic document? How can **knitr** convert an R Markdown file that contains a Shiny app into an *HTML* file? How should Shiny apps respond to the *CSS* files of a R Markdown document? And, how can multiple Shiny apps be managed in the same *R* session without unintended side effects?

This talk will describe RStudio's progress at solving these issues and explain how *R* users can embed Shiny apps in their own R Markdown reports.

Pricing credit derivatives with *R*

Giuseppe Bruno[1⋆]

1. Bank of Italy, Economics and statistics department
⋆ giuseppe.bruno@bancaditalia.it

Keywords: Credit risk, Collateralized Debt Obligations, Monte Carlo methods, Variance reduction.

Since the end of the eighties, in order to allow the exchange of credit risk, new financial instruments have arisen. These instruments are generically dubbed credit derivatives. Among the most important instruments traded in this market there are: Credit Default Swaps (CDS), Basket Default Swaps (BDS) and Collateralised Debt Obligations (CDO). The relevance of these instruments in the market for securitisation springs from their features that allow to trade credit risk in the same way as it is possible to trade market risk (for example see Gibson [1]). Securitization is grounded on the principles of pooling and successive tranching of a portfolio of assets. This process is schematically shown in the following figure. (1)

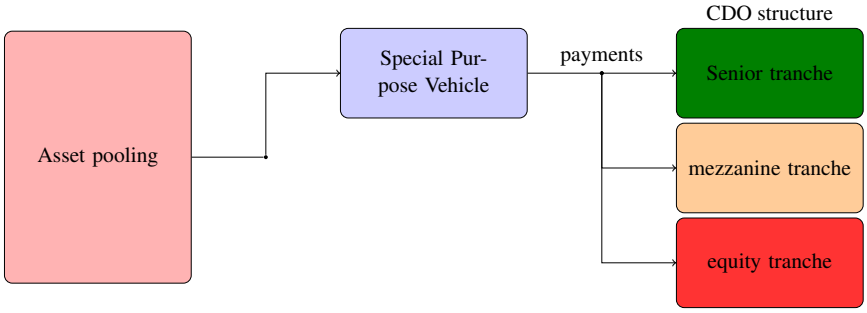


Figure 1: CDO scheme

Pricing of BDS and CDO often lacks analytical solution. For this reason it is required to resort to some kind of numerical simulation.

The most important quantity in pricing multi-name credit derivatives is the portfolio loss process. In the case of CDOs this stochastic process can be derived from individual times to default of underlying reference obligors (see Mounfield [2]).

In this paper we address the issues of implementing in the *R* software environment some of the most widespread algorithm for pricing credit risk derivatives such as Basket Default Swaps and synthetic CDO instruments. We first present the models allowing a closed form solution and then we tackle the issue of improving the Monte Carlo methods by comparing the relative performances with the employment of some variance reduction techniques such as antithetic variates and two different Quasi Random numbers procedures. On theoretical grounds, all these techniques should improve upon the traditional $(\sqrt{N})$ speed of convergence of the classical Monte Carlo methods. The use of variance reduction techniques along with the frameworks for parallelization available in the *R* environment allow the deployment of reasonably fast Monte Carlo algorithms.

References

[1] Gibson, M. S. (2004). Understanding the Risk of Synthetic CDOs. *FRB Working Paper*.

[2] Mounfield, C. C. (2009). *Synthetic CDOs, Modelling, Valuation and Risk Management*. Cambridge: Cambridge University Press.

Bayesian First Aid: A Package that Implements Bayesian Alternatives to the Classical `*.test` Functions in *R*

Rasmus Bååth[1,⋆]

1. Lund University Cognitive Science, Sweden
⋆Contact author: rasmus.baath@gmail.com

Keywords: Bayesian estimation, Bayesian statistics, Classical statistics, Teaching aid

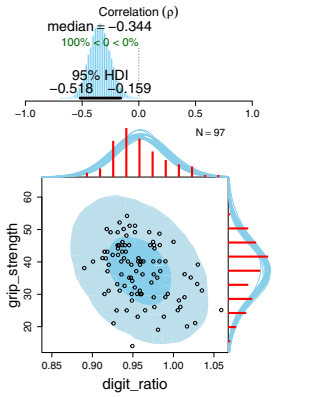
This talk will introduce **BayesianFirstAid**[1], an R package that implements Bayesian alternatives to the most commonly used statistical tests. It is inspired by the **BEST** package [2] and is similarly intended both as a practical tool and as a teaching aid. A main feature of the package is that the Bayesian alternatives are called in the same way as the corresponding classical test functions, save for the addition of `bayes.` to the beginning of the function name. For example, if `binom.test(x=7, n=10)` runs a classical binomial test then `bayes.binom.test(x=7, n=10)` runs the Bayesian alternative. This makes the package easy to pick up and use, especially if you are already used to the classical `*.test` functions, and it also facilitates comparing the output of the different approaches. All models are implemented using the JAGS modeling language, called from R using the **rjags** package. The generic function `model.code` makes it straightforward to start modifying the models underlying the package. It takes a **BayesianFirstAid** object and prints out the underlying model code which is ready to be copy-n-pasted into an R script and tinkered with from there. All **BayesianFirstAid** objects have default `plots` that show the posteriors of the parameters of interest together with a display that enables a quick posterior predictive check.

Below is an example of the output from the Bayesian First Aid alternative to `cor.test(...)`. The data is the hand grip strength (in kg) and index / ring finger ratio for the male group in [1].

```
> bayes.cor.test(digit_ratio, grip_strength)

  Bayesian First Aid Pearson Correlation Coefficient Test

data: digit_ratio and grip_strength (n = 97)
Estimated correlation:
  -0.34
95% credible interval:
  -0.52 -0.16
The correlation is more than 0 by a probability of 0.001
and less than 0 by a probability of 0.999

> model.code(bayes.cor.test(digit_ratio, grip_strength))

## Bayesian First Aid model code ##
require(rjags)

# Setting up the data
x <- digit_ratio
y <- grip_strength
xy <- cbind(x, y)

# The model string written in the JAGS language
model_string <- "model {
  for(i in 1:n) {
    xy[i,1:2] ~ dmt(mu[], prec[ , ], nu)
  }
```

[*output is truncated*]

```
> plot(bayes.cor.test(digit_ratio, grip_strength))
```



References

[1] Hone, L. S. and M. E. McCullough (2012). 2d: 4d ratios predict hand grip strength (but not hand grip endurance) in men (but not in women). *Evolution and Human Behavior 33*(6), 780–789.

[2] Kruschke, J. K. (2013). Bayesian estimation supersedes the t test. *Journal of Experimental Psychology: General 142*(2), 573.

[1]The **BayesianFirstAid** development can be followed at https://github.com/rasmusab/bayesian_first_aid

Simulations for regulatory decision making: How many simulations do we need to run?

Paul Schuette[1*]

1. FDA, Center for Drug Evaluation and Research, Office of Translation Sciences, Office of Biostatistics
*Contact author: Paul.Schuette@fda.hhs.gov

Keywords: Simulation, Modeling, Regulatory Science, sample size

Simulations are being used more frequently in statistics and in other sciences, and are an important component of strategic plans of regulators such as the US FDA. Simulations offer scientists the ability to employ complex models which are not analytically tractable. As a statistical programming language *R* is particularly suited for use with simulations. However, an implicit assumption for many simulations is that a fixed number of repetitions of the simulation such as $n = 1{,}000$ or $n = 10{,}000$ is more than sufficient to establish accurate results. Focusing on binomial proportion estimation, we use *R* to establish that simulation of size $n = 1{,}000$ or $n = 10{,}000$ are generally inadequate for commonly used levels of precision in a regulatory context. Using both standard normal approximations and exact methods, we establish the required number of replications can approach $4{,}000{,}000$ for some scenarios of interest. Thus, the number of simulations should be determined by the context of use. Additionally, we show that simple quantile estimation using simulation, a method used with Bayesian estimation as well as confidence interval estimation in conjunction with naive resampling and bootstrap methods, is fraught with potential problems. Finally, we suggest possible methods to enable large scale simulation efforts, with an emphasis on parallel computing methods.

Massive Predictive Modeling

Mark F. Hornick

Oracle
mark.hornick@oracle.com

Keywords: Advanced Analytics, Predictive Modeling, Oracle, Big Data

As enterprises continue to amass data at ever increasing rates and with greater variety – what is being called *big data* – the ability to extract value from that data demands high performance and scalable tools – both in hardware and software. In various industries, enterprise take on *massive predictive modeling* projects, where the goal is to build models, one per customer, to understand behavior and tailor predictions at the customer level. These predictions can then be aggregated to assess future demand. When there are millions of customers, each with their own accumulated data, such as frequent utility meter readings or retail sales, the scale of such projects takes on a new dimension. Massive predictive modeling comes with challenges: effectively partitioning data, storing and managing resulting models, associating models with customers during prediction, as well as backup, recovery, and security.

While *R* has parallel capabilities to facilitate taking advantage of clusters of computers, significant coding is usually required to meet the challenges noted above. In this talk, we present the business problem and illustrate how *Oracle R Enterprise*, one of Oracle's *R* technologies [1], facilitates massive predictive modeling in a pair of succinct *R* scripts. With Oracle R Enterprise, the data, *R* scripts, and models all reside in Oracle Database, which simplifies and speeds production deployment.

References

[1] Oracle R Technologies, http://www.oracle.com/technetwork/database/database-technologies/r/r-technologies/overview/index.html.

eegR: an R package to analyze electrophysiological (EEG) signals

Dénes Tóth[1]

1. Research Centre for Natural Sciences, Hungarian Academy of Sciences
*Contact author: toth.denes@ttk.mta.hu

Keywords: electrophysiology, electroencephalography (EEG), event-related potentials (ERP), biosignal processing

eegR is the first and only *R* package which provides a comprehensive tool to analyze electrophysiological signals. In typical cognitive electrophysiological studies, scalp-recorded voltage fluctuations (electroencephalography, EEG) are measured with a sampling rate of 250-1000 Hz from 32-128 channels (electrodes) while participants are exposed so systematic sensory stimulation (events). The number of trials is generally at least 40 up to several hundred per experimental condition. The main focus is on the event-related changes of the ongoing EEG signals (event-related potentials, ERPs).

The most popular open source software applications like EEGLAB (Delorme & Makeig, 2004) and FieldTrip (Oostenveld et al., 2011) have been written in the *MATLAB* programming language (www.mathworks.com). Numerous other *MATLAB*-toolboxes have been developed for specific EEG and ERP analysis methods; several of them can be used as EEGLAB plug-ins. However, since *MATLAB* is a commercial software, those toolboxes are either not free or if used as free stand-alone applications (created via *MATLAB* Compiler), they lack most of the scripting possibilities provided by the *MATLAB* environment. Other disadvantages of these toolboxes are their suboptimal performance regarding CPU-time and memory management, and the relatively low user-friendliness compared to commercial EEG softwares.

eegR is an experimental attempt to develop a package in the *R* programming language which 1) is freely available with the same functionality for *Windows*, *Linux* and *OS X* operating systems, 2) covers all basic steps in the processing of EEG/ERPs (raw data import, filtering, artifact rejection, segmentation, frequency decomposition, statistical analyses of single-trial and averaged data, etc.) but its main strength lies in the extensively documented general methods, classes, and utility functions which additional packages can rely on, 3) can handle out-of-memory data and allows easy parallelization, 4) provides user-friendly workflows and GUIs for users with limited *R* knowledge but does not restrict power-users in developing custom scripts, 5) provides functions for interactive and/or animated plotting, 6) builds on existing *R* packages if possible. The basic structure and functionality of the package will be presented, along with illustrative examples of add-on analysis possibilities like threshold-free cluster enhancement and permutation statistics.

References

Delorme, A. & Makeig, S. (2004) EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics. *Journal of Neuroscience Methods*, *134*, 9-21.

Oostenveld, R., Fries, P., Maris, E., Schoffelen, J-M. (2011) FieldTrip: Open source software for advanced analysis of MEG, EEG, and invasive electrophysiological data. *Computational Intelligence and Neuroscience*, Article ID 156869, 9 pages, doi:10.1155/2011/156869

Probabilistic Programming in R with Bruno

Robert Zinkov[1,*], **Chung-Chieh Shan**[1]

1. Department of Computer Science, Indiana University
*Contact author: zinkov@indiana.edu

Keywords: Machine learning, Graphical models, Bayesian modeling, MCMC

Markov Chain Monte Carlo (MCMC) methods have proven to be very effective for estimating posterior distributions. Their utility has led to the creation of general-purpose libraries for automatically performing this sampling. Unfortunately, many existing libraries have difficulty modeling distributions which vary in dimension. In addition, they have problems representing nonparametric Bayesian models as their dimensionality is defined to grow with the amount of data available [2].

We present *Bruno*. A universal probabilistic programming language for computing probability distributions over programs. By providing a programming language where every term can denote a probability distribution, we are able to model domains with changing dimensionality and have nonparametric components. We also provide a *R* interface called **rbruno** in the same spirit as **rjags** [1] to access the samples produced by these probabilistic programs. Lastly, our software allows users to specify how they wish to perform inference on their model. This flexibility makes it possible to finely tune the MCMC sampler if the problem domain demands it.

References

[1] Plummer, M. (2012). rjags: Bayesian graphical models using mcmc. http://CRAN.R-project.org/package=rjags.

[2] Teh, Y. W., M. I. Jordan, M. J. Beal, and D. M. Blei (2006). Hierarchical Dirichlet processes. *Journal of the American Statistical Association 101*(476), 1566–1581.

dplyr: a grammar of data manipulation

Hadley Wickham[1,*]

1. RStudio

*Contact author: hadley@rstudio.com

Keywords: data manipulation, databases

`dplyr` is a new package which provides a set of tools for efficiently manipulating datasets in R. `dplyr` is the next iteration of `plyr`, focussing on only data frames. `dplyr` is faster, has a more consistent API and should be easier to use. There are three key ideas that underlie `dplyr`:

1. Your time is important, so Romain Francois has written the key pieces in Rcpp to provide blazing fast performance. Performance will only get better over time, especially once we figure out the best way to make the most of multiple processors. For some cases, dplyr is 10,000x faster than dplyr.
2. Tabular data is tabular data regardless of where it lives, so you should use the same functions to work with it. With dplyr, anything you can do to a local data frame you can also do to a remote database table. PostgreSQL, MySQL, SQLite and Google bigquery support is built-in; adding a new backend is a matter of implementing a handful of S3 methods.
3. The bottleneck in most data analyses is the time it takes for you to figure out what to do with your data, and dplyr makes this easier by having individual functions that correspond to the most common operations (`group_by()`, `summarise()`, `mutate()`, `filter()`, `select()` and `arrange()`). Each function does one only thing, but does it well.

The "CUtil" Package for GPU-Accelerated Computing

Kazutaka Doi[1,*†], Kei Sakabe[†]

1. Fukushima Project Headquarters, National Institute of Radiological Sciences, Japan
† Both authors contributed equally to this work
*Contact author: kztkdi@gmail.com

Keywords: GPU computing, CUDA, Windows, parallel computing

Since we made presentation about our package **CUtil** (**CU**DA™ U**til**ity package) in useR! 2011[1], we were reconstructing the library for further improvements. Though **CUtil** package has been developed for providing computing power of graphical processing units (GPUs) for *R* users (especially for Windows® users) easily, our package did not accelerate calculations much compared with standard *R*. Therefore, we have improved our package drastically as follows.

The major time consuming procedure in the previous version was video memory allocation in GPUs, and this kind of operations is suppressed as possible. Because *R* codes appeared frequently in the previous version, which caused performance loss in some part, exposures of *R* codes are minimized. Furthermore, the required time for CPU tasks conducted before and after GPU tasks was not negligible, so our package is now multi-threaded by using Boost C++ library. With other minor improvements not mentioned above, the new version of our package will be ready at the time of the useR! 2014 with following features.

There are three main features in this package. The first feature is the *R*-native GPU function calls. Users can call GPU functions from *R* like other default functions in *R*. The second feature is the functionality to keep data on video memory even after GPU computing is over and the control is returned to *R*. When users call the functions of our package, the given data is automatically transferred to video memory and the pointer to the memory is stored into the *R* object as an external pointer. As the data is kept in the video memory, no time is needed for transfer in further function calls. As it is known that the proportion of time needed for the data transfer is relatively large, this advantage will be beneficial especially for a long series of computations, such as Markov chain Monte Carlo methods in Bayesian statistics. The third feature is the override of default functions. Users can override the default *R* functions by our package's functions. By using this functionality, users can accelerate their own codes by our package with minimum modifications. Other features are that double precision floating-point calculations are fully supported, and complex number computations are also partly supported. Garbage-collection, which enables us to use a small amount of video memory effectively, will be implemented, and the execution will be multi-threaded. This package will require computers with NVIDIA®'s GPU (compute capability is equal to or greater than 2.0). We are now preparing the Windows binary package. Linux® binary package will be available soon.

References

[1] Doi K, Sakabe K (2011). The "CUtil" package which enables GPU computation in R. In *useR! 2011, The R User Conference (Coventry, UK)*, pp. 24.

muHVT: Computational Geometry for Visual Analytics

Pravin Venugopal[¶,⋆], **Subir Mansukhani**[¶], **Zubin Dowlaty**[¶]

[¶]Innovation and Development, Mu Sigma Business Solutions Pvt Ltd
[⋆]Contact author: pravin.v@mu-sigma.com

Keywords: Voronoi Tessellations, Vector Quantization, Hierarchical K-Means, Non linear Dimensionality Reduction, Visual Analytics

Given our current capability to store and process vast amounts of data, there is a need to translate data into a visual form. This makes it easy to highlight important features including groups that share common features and outliers along different dimensions of the data.Visual representation of high dimensional data enables users to perceive features in their data quickly thereby augmenting the cognitive reasoning process with perceptual reasoning and enabling the process of generating insight from data to become faster and more directed.To this end, we use techniques and algorithms from *computational geometry* and *nonlinear dimensional reduction* to build a hierarchical "map" of the data and add the ability to overlay features on top of this map in order to visually discover patterns and also validate hypotheses that the user might have about the data at hand.

The muHVT package is a collection of *R* functions for clustering and construction of **Hierarchical Voronoi Tessellations** as a visualization tool to visualize clusters and generate insights from them. The data is compressed in a hierarchical manner to form clusters at various levels using either the *Hierarchical K-means*[1] algorithm where a quantization error governs the number of levels in the hierarchy for a set *k* parameter (the maximum number of clusters at each level) or the *LBG Vector Quantization* (LBG VQ)[3] algorithm which detects the number of clusters for the first level in the hierarchy, based on a specified quantization threshold. The LBG VQ algorithm is useful if the number of clusters in the dataset is not known a priori. The benefit of using hierarchical tessellations lies in the fact that we can analyze data at different levels of granularity. We can think of this as the way digital maps are used, where the user zooms in until the desired information is available. Also, plotting heat maps of the variables in the dataset on the tessellations at various levels of the hierarchy helps in deriving further insight from the data. This next generation segmentation technique gives an edge over the traditional segmentation techniques. For instance, in the example provided in the package we apply this technique to find hierarchical customer segments and overlay the distribution of features across the "map" of the entire dataset in order to understand the characteristics of the different regions of the map. Such datasets are typically found in the CRM systems at Fortune 500 companies. In this package, we use a non linear dimensionality reduction technique called *Sammon's projection*[4] from the **MASS** package. While various distance metrics like Euclidean, Manhattan, Minkowski, etc can be used for computing the distance between the data points, this package also provides a function for the *Jensen-Shannon-Bregman Divergence*[2] distance metric.

References

[1] A Bocker, S Derksen, E. S. G. S. (2004). Hierarchical k-means clustering.

[2] Arindam Banerjee, Srujana Merugu, I. S. D. and J. Ghosh (2005, October). Clustering with bregman divergences. *Journal of Machine Learning Research 6*, 1705–1749.

[3] Linde Y, Buzo A, G. R. (1980, January). An algorithm for vector quantizer design. *IEEE Transactions and Communications 28*, 84–95.

[4] Sammon, J. (1969, May). A nonlinear mapping for data structure analysis. *IEEE Transactions and Communications C-18*, 401–409.

Scagnostics

Katrin Grimm[1,*], **Antony Unwin**[1]

1. University of Augsburg
*Contact author: katrin-grimm@web.de

Keywords: Visualization, Scagnostics, Scatterplots

The neologism scagnostics is derived from the words scatterplots and diagnostics. The term was first mentioned by John and Paul Tukey in the middle of the nineteen-eighties. Scagnostics are measures for characterising scatterplots and identifying different features. The aim is to give an initial overview of unknown datasets with a large number of variables. Concrete measures were proposed by Wilkinson et al. and implemented in the *R* package **scagnostics**. The package assumes that simplifications are necessary in order to make the measures applicable for large datasets and it uses hexagonal binning to speed up calculations. For eight of the nine measures from Wilkinson et al. the graphs' minimum spanning trees, complex hulls and alpha hulls are used as basis for the calculations. This can lead to some imprecision, as the measures are limited by the utilized graphs. Some alternative concepts and measures will be presented with the focus on finding interesting scatterplot structures quickly. Although computers continually become faster, the calculation of two-dimensional measures is still computationally intensive and methods for reducing calculation time are important.

An obvious idea to reduce the computing time is to exclude discrete variables. Additionally, variables with significant anomalies in 1-D — for example multimodal or skew variables — can also be left out of the calculation of two-dimensional measures, as a scatterplot in two dimensions will almost always be dominated by an anomaly in one dimension.

Another important question in the context of computer analyzed graphics is, how a good selection of graphics can be found. In this case Wilkinsons idea was to present scatterplots which are significant different from the others (*Outliers*) on the one hand and to do a clustering of the measures on the other hand. Each cluster stands for a group of similar scatterplots and it is sufficient to look at one representative from each cluster (*Exemplars*). Although this approach is optimal in theory, there are difficulties in practice which can be explained by the measures themselves. We propose a different approach for automatically selecting scatterplots using the measures.

The talk includes alternative criteria for functional dependencies — based on splines and distance correlation — and a presentation of an implementation of some of these ideas in *R*, using a German election and demographic dataset with 70 different variables.

References

[1] Szkély, G. J., M. L. Rizzo, and N. K. Bakirov (2007). Measuring and testing dependence by correlation of distances. *The Annals of Statistics 35*(6), 2769–2794.

[2] Wilkinson, L., A. Anand, and R. Grossman (2005). Graph-theoretic scagnostics. *Proceedings of the 2005 IEEE Symposium on Information Visualization*, 157–164.

Deploying R into Business Intelligence and Real-time Applications

Louis Bajuk-Yorgan*

Tibco Software Inc.
*Contact author: lbajuk@tibco.com

Keywords: Commercial, Real time, Streaming Data, TERR, TIBCO

Open source *R* provides a powerful environment for statisticians to analyze data and develop new analytic approaches. However, their organizations often struggle with the best way to reliably and repeatably integrate their statisticians' work into other parts of the business. This presentation will demonstrate how **TIBCO Enterprise Runtime for R (TERR)**[1] can be used to deploy *R* language analyses into Business Intelligence applications and into real-time, event-driven applications, so that the organization can more easily benefit from their statisticians' work. Examples will include fraud detection & marketing upsell.

References

[1] Louis Bajuk-Yorgan (2013). TIBCO Enterprise Runtime for R, http://spotfire.tibco.com/terr .

Text processing with R: `exact.matches` and other functions

Stefan Th. Gries[1,*]

1. University of California, Santa Barbara
*Contact author: stgries@linguistics.ucsb.edu

Keywords: text processing, pattern matching, regular expression, corpus linguistics

While *R* has been extremely widely and successfully adopted as a programming language for statistical data processing and analysis, its use for text processing and analysis is much less widespread. For instance, in many digital humanities or linguistics contexts, *Perl* or *Python* are still more common even though *R* offers very much the same functionality for text processing plus the advanced statistical and graphical tools that usually follow textual analyses in these fields. Over the last few years, however, R has become more popular in these fields, too, in part because of (i) the availability of a first textbook on text processing with *R* (Gries, 2009) as well as workshops and bootcamps and (ii) because of a variety of functions that provide convenient text processing abilities that are more challenging to implement with the regular base *R* functions. In this talk, I will showcase several functions that are now enjoying wider use in linguistics; specifically, I will discuss

– `exact.matches`: a function that allows to retrieve the exact matches of a search expression in a character vector with many output options: just the matches, matches with the tab-delimited rest of the elements of the character vector with matches, as shown below, …

```
> cat(exact.matches("qwe", "1 1 1 qwe 2 2 2 qwe 3 3 3")[[4]], sep="\n")
[1] 1 1 1               qwe         2 2 2 qwe 3 3 3
[2] 1 1 1 qwe 2 2 2     qwe         3 3 3
```

… matches with user-defined numbers of characters or vector elements as contexts; in addition, contrary to competing functions, the function allows to find multiple overlapping matches:

```
> exact.matches.new(c("s", "s"), "this is a second sentence")[[1]]
[1] "s is"           "s is a s"        "s is a second s" "s a s"
[5] "s a second s"   "second s"
```

– `word.grammy`: a function that generates *n*-grams of text vectors:

```
> word.grammy(c("this","is","a","brand","news","example"), 3, " ")[[1]]
[1] "this is a"          "is a brand"         "a brand news"
[4] "brand news example"
```

– `char.grammy`: a function that generates *n*-grams of characters of text vectors:

```
> char.grammy(c("expressions"), 2)[[1]]
[[1]]
[1] "ex" "xp" "pr" "re" "es" "ss" "si" "io" "on" "ns"

[[2]]
[1] "te" "ex" "xt" "ts"
```

References

Gries, Stefan Th. (2009). *Quantitative Corpus Linguistics with R*, London & New York: Taylor & Francis. URL <http://tinyurl.com/QuantCorpLingWithR>.

An R Package for Parallel Matrix Powers

Norm Matloff[1*], **Jack Norman**[1]

1. University of California, Davis
*Contact author: matloff@cs.ucdavis.edu

Keywords: matrix powers, parallel computation, Markov chains, graph connectedness, GPUs

Powers of square matrices are useful in a variety of contexts. Two examples in the statistics/data science field are calculating the stationary distribution of a discrete-time Markov chain and determining whether a graph is connected. Also, the exponential of a matrix M, defined as $e^M = \sum_{r=0}^{\infty} M^r/r!$, can be used to calculate the finite-time distribution of a continuous-time Markov chain (Stewart, 2009).

The CRAN package **expm** calculates matrix exponentials and powers. However, in many applications, the matrices involved are quite large, so parallel computation is needed. Our package **parmatpows** (`http://heather.cs.ucdavis.edu/parmatpows`) fills this need. Two parallel platforms are supported, multicore and GPU, based on the CRAN packages **Rdsm** and **gmatrix**, affording excellent speedups.

The package includes special functions for the applications mentioned above. The Markov chain example exploits the fact that for an irreducible, aperiodic chain with transition matrix P and stationary distribution π, $\lim_{n\to\infty} P(X_n = i) = \pi_i$, a fact that implies that $\lim_{n\to\infty} P^n$ is a matrix having each of its rows equal to π; the column means of P^k will then provide a reasonable approximation to π for a suitably large k. In other words, finding the stationary distribution reduces to a matrix-powers problem. As with the matrix power function in **expm**, repeated squaring is used to compute a large enough power—P is squared to yield P^2, which itself is then squared to yield P^4 and so on—thus reducing $O(k)$ time to $O(\log_2 k)$.

It can be shown that an $n \times n$ graph with adjacency matrix A is connected if and only if $(A+I)^k$ consists of all positive values for all sufficiently large k (one need check only values of k through n-1). Thus connectedness can be determined again by repeated squaring.

Many mathematical operations used often in statistics, such as matrix inversion and QR factorization, are difficult to parallelize effectively on GPUs (Buckner, 2009). Thus it is not surprising that we found that in the Markov chain stationary distribution problem, use of matrix inverse on the GPU (`gpusolve()` in **gputools**) was not very effective.

But matrix multiplication is an exemplar data science application for GPU platforms, due to its regular pattern of data access and "embarrassingly parallel" nature, suggesting a matrix-powers approach to the stationary distribution problem. However, one needs to minimize data transfer back and forth between the CPU and GPU, which in turn means intermediate results must be saved between GPU kernel launches. In the context here, that means avoiding copying intermediate matrix powers if possible. This is not possible in **gputools**, but the **gmatrix** package, used here does allow saving on the GPU the result of a GPU matrix operation. (See also **RCUDA**, currently under development by D. Temple Lang and P. Baines.)

Similarly, on multicore platforms, our use of **Rdsm** is aimed at minimizing data copying, as it uses **bigmemory** to directly access shared memory (though there may be some copying behind the scenes due to cache coherency actions).

References

Buckner, J., Justin Wilson, M. Seligman, B. Athey, S. Watson and F. Meng (2009). The gputools package enables GPU computing in R. *Bioinformatics*, 26, 1, 134-135.

Stewart, W. (2009). *Probability, Markov chains, queues and simulation*, Princeton University Press.

PSAboot: An R Package for Bootstrapping Propensity Score Analysis

Jason M. Bryer[1,⋆]

1. Excelsior College
⋆Contact author: jason@bryer.org

Keywords: propensity score analysis, matching, bootstrapping

Propensity score analysis [5] is an approach to estimate causal effects in observational studies where randomization is not used. However, even when balance has been achieved between treatment and control units on observed covariates, there is a risk of bias due to unobserved covariates. Rosenbaum[4] has suggested testing one hypothesis multiple times using multiple matching and/or stratification methods to avoid exaggerated bias due to method selection. The **PSAboot** extends Rosenbaum's suggestion by implementing bootstrapping [1] for PSA. Like typical bootstrapping methods, the **PSAboot** package will draw *n* random samples and provide a pooled effect size estimate and confidence intervals. However, for each bootstrap sample multiple PSA methods will be used. The package includes implementations for matching (using the **Matching**[6] and **MatchIt**[2] packages), stratification with logistic regression, and stratification using classification trees (using the **rpart**[7] and **party**[3] packages). The package emphasizes the use of visualizations for evaluating balance as well as summarizing results. Additionally, examples on extending the package for other PSA methods will be discussed.

References

[1] Efron, B. and R. Tibshirani (1993). *An Introduction to the Bootstrap*. Chapman & Hall/CRC.

[2] Ho, D. E., K. Imai, G. King, and E. A. Stuart (2011). MatchIt: Nonparametric preprocessing for parametric causal inference. *Journal of Statistical Software 42*(8), 1–28.

[3] Hothorn, T., K. Hornik, and A. Zeileis (2006). Unbiased recursive partitioning: A conditional inference framework. *Journal of Computational and Graphical Statistics 15*(3), 651–674.

[4] Rosenbaum, P. R. (2012). Testing one hypothesis twice in observational studies. *Biometrika 99*, 763–774.

[5] Rosenbaum, P. R. and D. B. Rubin (1983). The central role of the propensity score in observational studies for causal effects. *Biometrika 70*, 41–55.

[6] Sekhon, J. S. (2011). Multivariate and propensity score matching software with automated balance optimization: The Matching package for R. *Journal of Statistical Software 42*(7), 1–52.

[7] Therneau, T., B. Atkinson, and B. Ripley (2013). *rpart: Recursive Partitioning*. R package version 4.1-3.

Shiny Demos of Statistical Modelling

Heather Turner[1], **Paul Hewson**[2]

1. Independent statistical/R consultant
2. University of Plymouth, UK
*Contact author: ht@heatherturner.net

Keywords: Shiny apps, teaching, statistical modelling

In teaching statistical modelling, particularly to students with little mathematical background, it can be helpful to visualise properties of the model and associated statistics. Rather than producing several static visualisations, interactive or dynamic visualisations enable a number of concepts to be illustrated on the same data set. Furthermore, dynamic visualisations can be used to explore differences between data sets or alternative parameter settings.

Some interactive demos to illustrate ordinary least squares regression are available in the CRAN packages **rpanel** and **TeachingDemos**, implemented via Tcl/Tk and base graphics respectively. Both require a small amount of R code to set up the demo for a particular data set, which is unlikely to be a problem for the instructor but makes the demos less accessible to students without knowledge of R. In this talk we present our work on the development of statistical modelling demos using Shiny. Shiny provides a platform to implement R-based web apps, that may be accessed by students on their mobile computing devices via a browser.

Our initial focus has been on simple linear regression, where many fundamental concepts can be illustrated. The `linreg` app explores the optimisation process using absolute or quadratic loss functions and illustrates the fitted line, (squared) residuals, and fitted density in two or three dimensions. Several well known data sets that are often used in introductory courses are made available in the app, such as the `anscombe` data sets. The `linreg` app is hosted on a Shiny Server at the University of Plymouth and is publicly accessible at `http://141.163.66.244:3838/linreg/` (a domain name is forthcoming). Further demos for more complex models, such as Poisson regression models, are in the pipeline and developments can be tracked in the GitHub repository: `https://github.com/hturner/shiny-demos`.

Domino: A Platform-as-a-Service for Industrialized Data Analysis

Nick Elprin[1*]

1. Domino Data Lab, Inc.
*Contact author: nick@dominodatalab.com

Keywords: Cloud computing, reproducibility, version control

As data volumes have increased and analytical techniques have become more sophisticated, the tools necessary to do industrialized analysis — analysis that is scalable, reproducible, and collaborative — have lagged in their ease of use. We have built Domino, a Platform-as-a-Service for data analysis, to equip a larger group of users with functionality that has typically been inaccessible to people without engineering abilities and/or a massive amount of time to set up infrastructure and plumbing.

Although Domino is language agnostic, it has particularly deep integration with *R*, including integration with RStudio and first-class support for packages from CRAN and other repositories.

Domino address three core areas of functionality:

(1) It lets you run your *R* code (or *Python*, *Julia*, *Matlab*, and more) in the cloud without any setup or configuration. Domino handles AMI and package management, job distribution and secure data transfer. It allows you to change your hardware with one-click, or to distribute your analysis across multiple machines.
(2) It automatically keeps a revisioned history of your project — code, data, and results — so you can browse and reproduce past work. Unlike traditional source control, Domino tracks large data files, and creates a first-class association between your results artifacts (e.g., charts) and the code/data that produced them.
(3) It facilitates collaboration so you can easily share results and co-author analyses.

In this talk we will demonstrate Domino's core functionality and describe its architecture, with an emphasis on the technical challenges involved in enabling reproducible work (e.g., an immutable, revisioned data store for large files). We will also describe some case studies highlighting how Domino is being used in the real world.

Selection Effects of Common Variables on Statistical Matching

Yukiko Kurihara[1*]

1. Department of Humanities, Hirosaki University, Japan
*Contact: yukuri@cc.hirosaki-u.ac.jp

Keywords: Bayesian regression imputation, Mahalanobis method, Multiple imputation, Sampling experiment

This study verifies the precision of correlation coefficients based on statistical matching and multiple imputation under different matching methods and combinations of common variables. The matching methods for verification are a non-parametric approach based on Mahalanobis distance (package **StatMatch**) and the Bayesian regression imputation method (NIBAS)—a parametric method [5]. Questionnaire data from the Financial Statements Statistics of Corporations by Industry (Ministry of Finance) were used to clarify the effectiveness of matching data created from different sample datasets.

The three main findings are as follows: First, NIBAS enables the estimation of correlation coefficients with lesser bias than those of the Mahalanobis matching method when one aims to obtain only one set of statistics. Second, the primary condition for high-precision estimation is a combination of common variables with both low conditional dependence and strong correlation with target variables. Finally, the confidence interval computed by multiple imputation with NIBAS suitably covers the true value and measures the uncertainty inherent in statistical matching, except in the case of point estimates with extremely large bias.

References

[1] D'Orazio, M., M. Di Zio, & M.Scanu (2006), *Statistical Matching: Theory and Practice*, Wiley, West Sussex.

[2] Goel, P.K. & T. Ramalingam (1980), *The Matching Methodology: Some Statistical Properties*, Springer, Berlin.

[3] Little, R.J.A. & D.B. Rubin (2002), *Statistical Analysis with Missing Data*, Wiley, New York.

[4] Mase, S (2010). *R Programming Manual* (in Japanese), Surikougaku-sha, Tokyo.

[5] Rässler, S. (2002), *Statistical Matching*, Springer, New York.

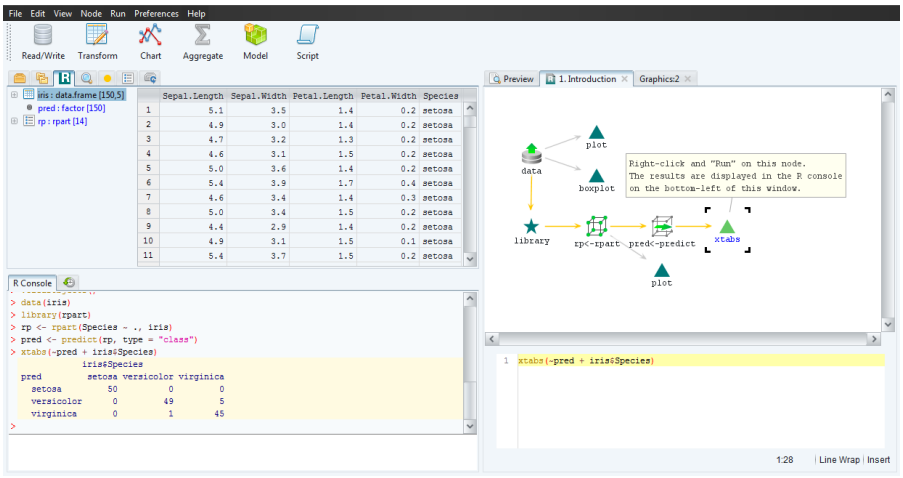
R AnalyticFlow 3: An Environment for Data Analysis with *R*

Ryota Suzuki*, Tatsuhiro Nagai

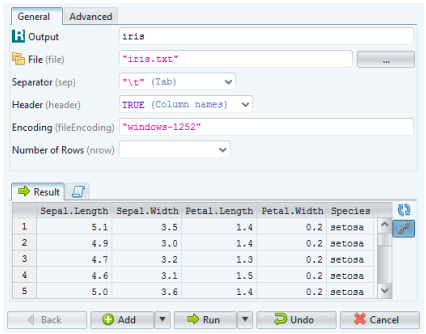
Ef-prime, Inc.
*Contact author: suzuki@ef-prime.com

Keywords: R AnalyticFlow, GUI, *Java*, **JRI**

R AnalyticFlow is software which provides a flowchart-style GUI for *R*. A user draws an *analysis-flow*, a graphical representation of analysis process in *R*. Once a flow is written, anyone can easily execute it by "right-click and run" mouse operation. It is written in *Java* and communicates with *R* via **JRI**. The software is freely available on our website for Windows, Linux and Mac OS X.



We introduce the newest version of the software: **R AnalyticFlow 3**. It has a tab-based interface that you can see everything – analysis-flows, *R* scripts, console, *R* objects, data files, etc. – in one window.



We have redesigned the software so that many common tasks of data analysis can be done without writing code. The figure on the left shows the UI for reading a data file. Once you select options the corresponding *R* script is generated, and a preview of the result is shown with a small sample of the original data.

This software will reduce the burden of data analysis, help beginners to use *R*, and facilitate collaborations between people with varying skills.

References

Suzuki (2008). R AnalyticFlow: A flowchart-style GUI for R. In *useR! 2008, The R User Conference, (Dortmund, Germany)*, pp. 176.

FADA: an *R* package for variable selection in supervised classification of strongly dependent data

Emeline Perthame [1,*], **Chloé Friguet** [2], **David Causeur** [1]

1. Agrocampus Ouest - Applied Mathematics Department, Rennes, France
2. Laboratoire de Mathématiques de Bretagne-Atlantique (LMBA), Université de Bretagne-Sud, Vannes, France
*Contact author: perthame@agrocampus-ouest.fr

Keywords: High dimension, variable selection, dependent data, supervised classification, factor analysis

Handling dependence or not in feature selection is still an open question in supervised classification issues where the number of covariates exceeds the number of observations. Some recent papers surprisingly show the superiority of naive Bayes approaches based on an obviously erroneous assumption of independence (see [2]), whereas others recommend to infer on the dependence structure in order to decorrelate the selection statistics (see [6, 1]). In the classical Linear Discriminant Analysis (LDA) framework, the present talk first highlights the impact of dependence in terms of instability of feature selection. A second objective is to revisit the above issue using a flexible factor modeling for the covariance.

Latent components of dependence are introduced in the LDA model, conditionally on which a new Bayes consistency is defined. The linear Bayes classifier derived on factor-adjusted data, namely decorrelated data obtained by subtracting the effects of latent factors, is shown to be conditionally consistent. A procedure is then proposed for the joint estimation of the expectation and variance parameters of the model, based on an iterative algorithm which alternates the estimation of the fixed parameters of the supervised factor model and the latent factors. As in [5], the estimation method adapts an EM algorithm for factor models to the present supervised classification situation. The Factor-Adjusted Discriminant Analysis (FADA) method is compared to recent regularized Diagonal Discriminant Analysis approaches (DDA), assuming independence among features, and regularized LDA procedures, both in terms of classification performance and stability of feature selection.

The talk will also focus on a demonstration of an *R* package which implements FADA. The main function of the package provides an efficient way to decorrelate data before applying a feature selection procedure. Several methods of classification are available: DDA, penalized logistic regression [4], shrinkage discriminant analysis [1] which proposes a shrunken estimation of covariance matrix through James-Stein estimator and variable selection by controlling false non-discovery rate and LDA penalized by Lasso [3].

References

[1] Ahdesmäki, M. and K. Strimmer (2010). Feature selection in omics prediction problems using cat scores and false non-discovery rate control. *Annals of Applied Statistics 4*, 503–519.

[2] Bickel, P. and E. Levina (2004). Some theory for fisher's linear discriminant function, naive bayes, and some alternatives when there are many more variables than observations. *Bernoulli 10*(6), 989–1010.

[3] Clemmensen, L., T. Hastie, D. Witten, and B. Ersbøll (2011). Sparse discriminant analysis. *Technometrics 53(4)*, 406–413.

[4] Friedman, J., T. Hastie, and R. Tibshirani (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software 33*, 1–22.

[5] Friguet, C., M. Kloareg, and D. Causeur (2009). A factor model approach to multiple testing under dependence. *Journal of the American Statistical Association 104:488*, 1406–1415.

[6] Xu, P., G. Brock, and R. S. Parrish (2009). Modified linear discriminant analysis approaches for classification of high-dimensional microarray data. *Computational Statistics and Data Analysis 53*, 16741687.

Performance Analysis for R: Towards a Faster R Interpreter

Helena Kotthaus*, Ingo Korb, Markus Künne, Peter Marwedel

Department of Computer Science 12, TU Dortmund University
*Contact author:helena.kotthaus@tu-dortmund.de

Keywords: Profiling, Performance Analyses, Machine Learning, Tools

The *R* language has a large set of dynamic features which allow for rapid development of new algorithms for statistical applications. However, this flexibility comes at a price: *R* is considered to be a rather slow language that needs a large amount of memory during runtime. Our goal is to resolve these performance problems. An indispensable prerequisite for this is having a more detailed view into the *R* interpreter's internals. For this reason we have developed a profiling tool [3] which enables a detailed analysis of runtime behavior and memory consumption of *R* programs.

As a basis for our profiling tool we have chosen the TraceR framework. TraceR was originally developed at Purdue University for *R* 2.12 [2]. Besides porting it to the current version of *R*, we also improved its usability and analysis capabilities. Since TraceR is directly integrated with the *R* interpreter, we can generate more detailed data compared to other existing profiling tools, such as Rprof. Rprof operates only at the *R* function level and as such does not provide information about the internals of the *R* interpreter or native code. Our new profiling tool, in contrast, can profile the execution time spent in C/Fortran code supplied by *R* packages, or timing characteristics of memory management tasks like garbage collection. Additionally, our tool is able to record the function call hierarchy. This is similar to the context stack feature of Rprof, but independent of the sampling rate. Moreover, it is capable of also recording internal operations of the *R* interpreter. The level of detail of this control flow information is highly configurable and can be changed interactively. The collected data is used to create a call graph which is annotated with other measurement results from our profiling tool like the invocation count.

Even though we originally developed our profiling tool for analyzing the bottlenecks of machine learning *R* programs [1], we believe that the feedback which can be generated with our tool is valuable to guide changes in the original *R* interpreter, as well as to support the development of alternative *R* interpreters. In this talk we will present our profiling tool and how to apply it to analyze the runtime behavior and the memory consumption of *R* programs.

References

[1] Helena Kotthaus, Michel Lang, Jörg Rahnenführer and Peter Marwedel: Runtime and memory consumption analyses for machine learning R programs. Statistical Computing, Schloss Reisensburg, Germany, 2013

[2] The Reactor Project: http://r.cs.purdue.edu, Purdue University, 2012

[3] Ingo Korb, Helena Kotthaus, Markus Künne: TraceR, https://github.com/allr/tracer, TU Dortmund University, 2014

ERP: an *R* package for Event-Related Potentials data analysis

David Causeur[1,2⋆], **Emeline Perthame**[1,2], **Ching-Fan Sheu**[3]

1. Agrocampus, Rennes, France
2. IRMAR, UMR 6625 CNRS, Rennes, France
3. National Cheng-Kung University, Tainan, Taiwan
⋆Contact author: david.causeur@agrocampus-ouest.fr

Keywords: ERP; High-dimensional data; Multiple testing; Time dependence

Experiments involving recording event-related (brain) potentials (ERP) are now widely performed in psychological research to study the time courses of mental events. With the routine collection of massive amount of data from ERP studies, researchers must face the challenge of multiple comparison corrections: in shifting, simultaneously, through thousands or tens of thousands of tests, a balance must be struck between keeping a low false positive error rate while maintaining sufficient power for correct detection.

A number of False Discovery Rate (FDR) controlling procedures ([1]) have become standard tools to achieve the above objective in high-dimensional situations. It is, however, well known ([3]) that highly correlated data, such as ERPs with their strong temporal dependence, can severely affect the stability of simultaneous testing.

In ERP data analysis, before the widespread success of FDR controlling procedures, [5] had already proposed a procedure for identifying significant intervals assuming an auto-regressive dependence structure of order 1 among test statistics. This method exhibits desirable properties but suffers both from an inadequate time-dependence modelling and an uncontrolled proportions of false positives. An alternative approach to dealing with correlation in multiple testing is to account for the multivariate dependence by some flexible data reduction techniques involving latent variables (see [6, 4] or more recently [7]).

We propose a joint modeling of the signal and time-dependence in ERP data (see [2]) to improve the properties of multiple testing procedures, such as the Benjamini-Hochberg ([1]) procedure, the Guthrie-Buchwald method and the decorrelation approaches by [6] (SVA) and [7] (LEAPP). The talk will also introduce the *R* package **ERP**, which implements a variety of multiple testing functions including the aforementioned procedures.

References

[1] Benjamini, Y. and Y. Hochberg (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society, Ser. B 57*, 289–300.

[2] Causeur, D., M. Chu, S. Hsieh, and C. Sheu (2012). A factor-adjusted multiple testing procedure for erp data analysis. *Behavior Research Methods 44*, 635–643.

[3] Efron, B. (2007). Correlation and large-scale simultaneous significance testing. *Journal of the American Statistical Association 102(477)*, 93–103.

[4] Friguet, C., M. Kloareg, and D. Causeur (2009). A factor model approach to multiple testing under dependence. *Journal of the American Statistical Association 104*, 1406–1415.

[5] Guthrie, D. and J. Buchwald (1991). Significance testing of difference potentials. *Psychophysiology 28*, 240–244.

[6] Leek, J. and J. Storey (2008). A general framework for multiple testing dependence. *Proceedings of the National Academy of Sciences of the United States of America 105*, 18718–18723.

[7] Sun, Y., N. Zhang, and A. Owen (2012). Multiple hypothesis testing adjusted for latent variables, with an application to the agemap gene expression data. *The Annals of Applied Statistics 6(4)*, 1664–1688.

"This code is a complete hack, may or may not work, etc.." The Challenges of Validating R

Aimee Gott[1*], Andy Nicholls[1]

1. Mango Solutions
*Contact author: agott@mango-solutions.com

Keywords: validation, knitr, testCoverage, functionMap

Whilst *R* usage has grown hugely in recent years the use of *R* in regulated industries, such as the pharmaceutical industry, is still limited. The *R* core team has provided documentation as guidance for the use of *R* in such industries [1], though *R* still comes with "absolutely no warranty" and there is no formal documentation related to the many additional packages available on CRAN. To comply with FDA guidelines [2] these add on packages must be validated along with core and recommended packages. Mango was first asked to validate a version of *R* in 2009. The growth of *R* and the number of companies wishing to validate *R* has led to a steady stream of *R* validations at Mango in recent months.

In this talk we will consider some of the challenges that we have faced in validating *R* packages and discuss some of the tools that we have developed to aid the process. We will discuss the challenge of creating large amounts of documentation for *R* packages and how **knitr** can be incorporated into an automated process to do this.

We will also talk about two new packages developed for code analysis, **testCoverage** and **functionMap**. The **testCoverage** package has been developed to determine how much of the code in a given R package is covered by associated unit tests, while the **functionMap** package has been developed in order to explore the functional relationship within a package and its dependencies.

References

[1] The R Foundation (2013). R: Regulatory Compliance and Validation Issues A Guidance Document for the Use of R in Regulated Clinical Trial Environments, http://www.r-project.org/doc/R-FDA.pdf.

[2] U.S. Food and Drug Administration (2013). 21 CFR Part 11: Electronic Records, Electronic Signatures, http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfcfr/CFRSearch.cfm?CFRPart=11&showFR=1

ggvis: Interactive graphics in R

Winston Chang[1,*]

1. RStudio, Inc.

*Contact author: winston@rstudio.com

Keywords: visualization, ggvis, ggplot2

The **ggvis** package makes it easy to create interactive data graphics with R, with a declarative syntax similar to that of **ggplot2**. Like **ggplot2**, **ggvis** uses concepts from the grammar of graphics, but it also adds the ability to create interactive graphics and deliver them over the web.

In this talk I will provide an overview of what **ggvis** can do, and how to use it for exploring data. One of the central goals of **ggvis** is to not only make it possible to create interactive graphics, but to make it simple for *R* users who are not experts in programming or data visualization.

With **ggvis**, data manipulation and transformation is performed in R, while the presentation and interaction occur in a web browser. The communication between the two sides is handled by Shiny, which also provides the basis for the reactive programming model of interaction in **ggvis**.

References

RStudio, Inc. (2014). ggvis web page, http://ggvis.rstudio.com/

Visually Exploring Random Forests with ggRandomForests

John Ehrlinger

Department of Quantitative Health Sciences
Lerner Research Institute
Cleveland Clinic
john.ehrlinger@gmail.com

Keywords: Machine Learning, Random Forests, survival analysis, **ggplot2, randomForestSRC**

Random Forests [1] (RF) are a fully non-parametric statistical method requiring no distributional assumptions on covariate relation to the response. RF are robust, optimizing predictive accuracy by fitting an ensemble of trees to stabilize model estimates. RF utilizes all variables in predicting the specified outcome, effectively weighting the most important covariates by assessing their impact on separating dissimilar groups of observations. Random Forests for survival [3, 5] (RF-S) are an extension of RF techniques to survival settings, allowing efficient non-parametric analysis of time to event data. The **randomForestSRC** [4] package is a unified treatment of Breiman's random forests for survival, regression and classification problems.

Predictive accuracy make RF an attractive alternative to parametric models, though complexity and interpretability of the forest hinder wider application of the method. We introduce the **ggRandomForests** package, an implementation of **ggplot2** [7] graphics for exploring **randomForestSRC** objects. Using both classification (RF-C) and survival (RF-S) examples from our research at the Cleveland Clinic, we will demonstrate the **randomForestSRC** package. We use Variable Importance measure (VIMP) [1] as well as Minimal Depth [6], a property derived from the construction of each tree within the forest, to assess the impact of variables on forest prediction. We will also demonstrate the use of variable dependence plots [2] to aid interpretation RF results in different response settings.

References

[1] Breiman, L. (2001). Random Forests. *Machine Learning 45*(1), 5–32.

[2] Friedman, J. H. (2000). Greedy function approximation: A gradient boosting machine. *Annals of Statistics 29*, 1189–1232.

[3] Ishwaran, H. and U. B. Kogalur (2007). Random survival forests for R. *R News 7*, 25–31.

[4] Ishwaran, H. and U. B. Kogalur (2013). Random Forests for Survival, Regression and Classification (RF-SRC), R package version 1.4.

[5] Ishwaran, H., U. B. Kogalur, E. H. Blackstone, and M. S. Lauer (2008). Random survival forests. *The Annals of Applied Statistics 2*(3), 841–860.

[6] Ishwaran, H., U. B. Kogalur, E. Z. Gorodeski, A. J. Minn, and M. S. Lauer (2010). High-dimensional variable selection for survival data. *J. Amer. Statist. Assoc. 105*, 205–217.

[7] Wickham, H. (2009). *ggplot2: elegant graphics for data analysis*. Springer New York.

Beyond the black box: Flexible programming of hierarchical modeling algorithms for *BUGS*-compatible models using *NIMBLE*

Christopher Paciorek[1,⋆], Perry de Valpine[2], Daniel Turek[1,2], Cliff Anderson-Bergman[1,2], Ras Bodík[3], Duncan Temple Lang[4]

1. Department of Statistics, University of California, Berkeley
2. Department of Environmental Science, Policy, and Management, University of California, Berkeley
3. Department of Electrical Engineering and Computer Science, University of California, Berkeley
4. Department of Statistics, University of California, Davis
⋆Contact author: paciorek@stat.berkeley.edu

We introduce a domain-specific language embedded in *R* for programming hierarchical model algorithms for models declared in the *BUGS* language. Various software packages allow flexible specification of hierarchical models and provide algorithms such as specific types of MCMC, particle filter (PF), Laplace approximation, or others. However, many new and old algorithms remain inaccessible for practical use without re-writing them for each model. Moreover, having different packages for different algorithms makes it difficult to combine methods or try different methods on the same problem. The new *R*-based *NIMBLE* language allows flexible programming of algorithms with access to the model structure declared by *BUGS* code. *BUGS* code is processed into model-specific *C++* code, compiled, and interfaced with *R*. Functions in *NIMBLE* can use the model structure to allow automatic specialization to the details of any model. Once specialized to a model, functions can be processed into *C++*, compiled, and interfaced with *R*. With *R*'s CRAN package system, developers will be able to distribute new algorithms written in *NIMBLE*. We will present examples with MCMC, PF and more.

Creating R-Based Web Browser Applications Using Alteryx

Dan Putler[1]

1. Alteryx, Inc.
*Contact author: dputler@alteryx.com

Keywords: web browser, *R* deployment, applications

Alteryx Analytics is a software platform for data blending, advanced analytics, and application deployment. All three of these capabilities come together in creating web browser applications that allow non-technical users to parameterize and consume the results of a specialized *R* process. Examples of this type of process include applications to generate time series sales forecasts for different items at different locations in a retail store chain, the probability that a patient has a particular disease condition given a set of diagnostic inputs, and the expected number of claims a new customer is likely to have on an automobile insurance policy given their personal characteristics and their residence location in order to determine that customer's policy price. In this presentation, we will illustrate how these types of specific *R*-based use case applications can be created in *Alteryx* and then easily deployed within an organization (or more broadly) for use within a web browser environment via *Alteryx's* Analytics Gallery technology.

Talk: Debugging in R

Jonathan McPherson[1,*]

1. RStudio, Inc.

*Contact author: jonathan@rstudio.com

Keywords: debugging, techniques, tools, R 3.1.0

This talk will cover a selection of both beginning and advanced topics in *R* debugging, with the goal of being helpful both to those not yet familiar with *R*'s debugging facilities and to those who have mastered the tools and want to learn about the capabilities offered by the next version of R and RStudio.

The talk's format will be example-based: we will begin with some *R* code that has several problems, and then walk through the diagnosis and resolution of those problems using *R* debugging tools. We will focus more on bug discovery than on techniques for avoiding bugs, as the latter topic is larger, and excellent resources exist (Wickham 2014).

First, **code instrumentation** for debugging will be discussed, including content-based instrumentation such as `browser()` and runtime-based instrumentation such as `debug()` and editor breakpoints.

Next, **runtime inspection and bug discovery** will be covered, including inspecting local environments and stepping through code using the interactive browser. Here we'll discover and correct the first set of bugs in the code.

We'll then discuss **handling and inspecting errors**, with a focus on finding error sources using `traceback()` and `recover()`. We'll analyze errors both after the code has finished running (post-mortem error handling) and at the moment they occur (just-in-time error handling), then use these tools to discover and correct a second set of bugs.

Finally, for attendees already familiar with debugging *R* code, we'll cover **new R debugging tools**, including stepping commands introduced in *R* 3.1.0 (R Core 2014), and new debug tools in RStudio, such the integration of *R* 3.1.0 commands and support for stepping through deparsed code.

References

R Core. 2014. "Changes in R 3.1.0." http://stat.ethz.ch/R-manual/R-devel/doc/html/NEWS.html.

Wickham, Hadley. 2014. "Debugging, Condition Handling, and Defensive Programming." http://adv-r.had.co.nz/Exceptions-Debugging.html.

Shiny: R made interactive

Joe Cheng[1,*]

1. RStudio, Inc.

*Contact author: joe@rstudio.com

Keywords: web applications, interactive, reactive programming, shiny

R has long been an excellent platform for writing reports, thanks to tools like `Sweave` (and more recently, **knitr** and **rmarkdown**). But these tools have focused primarily on generating static artifacts, like PDF and HTML documents. As the reader of a report, it's impossible to tweak any of the parameters used, or provide your own data to be subjected to the same analysis, without going back to the report's author and asking them to modify and recompile the report.

In contrast, the **shiny** package makes it easy for *R* users to create interactive artifacts, in the form of web applications. Shiny has built-in functions that:

- Create input widgets like sliders, numeric inputs, and dropdowns
- Include output widgets for graphical, textual, and tabular data
- Lay out these widgets and any other content using grids, tabs, navigation lists

No knowledge of web technologies is necessary, but Shiny users who do know HTML and JavaScript can extend the framework with new types of input/output widgets and visual themes. These Shiny extensions can then be bundled into *R* packages for easy reuse by other Shiny users.

This talk will illustrate just how easy it is to write Shiny applications, then show some of the interesting ways we have been improving and extending Shiny.

References

RStudio, Inc. 2014. "Shiny." `http://shiny.rstudio.com`

10 R packages to win Kaggle competitions

Xavier Conort, DataRobot

*Contact author: xavier@datarobot.com

Keywords: Kaggle, Machine Learning

R is rapidly growing in popularity among statisticians, data scientists and actuaries. An actuary by training, I became an *R* enthusiast myself 3 years ago when I discovered that *R* offered me a powerful platform for statistical and actuarial analysis. The main draw for me was the large palette of Machine Learning algorithms to tackle predictive modeling problems outside my comfort zone -- ranging from churn prediction to essay scoring, sales forecasts, flight arrival prediction, recommendation engine and credit scoring. Machine Learning helped me significantly reduce the modeling effort compared to traditional statistical parametric techniques, work on dirty data with limited domain insight and extract value from unstructured or large dimensionality datasets. Thanks to *R*, I placed in the top 10 of more than 15 Kaggle competitions and won several of them. 10 *R* packages were the key ingredients of my Kaggle solutions. In this talk, I will cover why I found those 10 packages particularly powerful and how I used them to build winning solutions.

Packrat - A Dependency Management System for R

J.J. Allaire[1,*], **Kevin Ushey**[1]

1. RStudio, Inc

*Contact author: jj@rstudio.com

Keywords: reproducible research, dependency management, project management

Dependency management in *R* is difficult. Different *R* projects can have different dependencies, and can often depend on different versions of the same *R* packages. The suite of *R* packages served by CRAN and BioConductor is constantly evolving and growing, and while *R core* and package authors make large efforts to maintain backwards compatibility, it is not guaranteed as *R* and its packages evolve.

There has been a lot of discussion as to how the *R* project, alongside the CRAN repository, could be augmented to support better versioning in projects (Ooms 2013). **packrat** uses a form of **local versioned package management**, to ensure a project and its versioned dependencies are coupled together – similar to JavaScript's *node.js* and the packages on its associated repository *NPM*.

As a result, **packrat** helps the user by isolating dependencies within a project, ensuring that they do not conflict with package requirements in other projects. In addition, package sources are recorded, so that packages can be easily upgraded and rolled back, using the archives available on CRAN and BioConductor, or to local package sources packaged alongside the project. Furthermore, users collaborating on a project can use **packrat** to ensure that their *R* environments are compatible, hence avoiding compatibility problems in collaborative projects.

packrat helps solve the following problems:

- **Local Dependency Management:** Because each **packrat** project uses its own private library, dependencies are effectively isolated from other projects, and so versioning conflicts can be controlled and avoided. A user can *bootstrap* a project to infer and set up the local library the project requires, and with later modifications to this local library, the user can *snapshot* and save the current library state, or *restore* and roll back to the last *snapshot*ted state.
- **Portability: packrat** makes it easy to *bundle* a project for sharing. A *bundle*d project can easily be *unbundle*d to restore the same *R* environment that was originally used in the project, even across different platforms.
- **Reproducibility: packrat** records the exact package versions a project depends on, and ensures those exact versions are the ones that get installed wherever the **packrat** project is used.

  Replication is the ultimate standard by which scientific claims are judged. (Peng 2011)

In this talk, we will outline a number of common usage scenarios with **packrat**, and demonstrate how it can be used to control and manage dependencies within your project.

References

Ooms, Jeroen. 2013. "Possible Directions for Improving Dependency Versioning in R." *CoRR* abs/1303.2140.

Peng, Roger D. 2011. "Reproducible Research in Computational Science." *Science* 334 (6060): 1226–27. doi:10.1126/science.1213847. http://www.sciencemag.org/content/334/6060/1226.abstract.

The Next Generation of R Markdown

Jeff Allen[1,*]

1. RStudio

*Contact author: jeff@rstudio.com

Keywords: reproducible, knitr, markdown, reports

R Markdown is an authoring format that enables easy creation of dynamic documents, presentations, and reports from *R*. It combines the core syntax of markdown (Gruber 2004), an easy-to-write plain text format, with embedded *R* code chunks whose output is included in the final document. R Markdown documents are fully reproducible (they can be automatically regenerated whenever underlying *R* code or data changes).

This talk will describe R Markdown v2 (Allaire 2014), a next generation implementation of R Markdown based on **knitr** (Xie 2012) and pandoc (MacFarlane 2006). This implementation brings many enhancements to R Markdown, including:

- Create HTML, PDF, and MS Word documents as well as Beamer, ioslides and reveal.js presentations.
- New markdown syntax including expanded support for tables and bibliographies.
- Hooks for customizing HTML and PDF output (include CSS, headers, and footers).
- Embedding of Shiny applications and reactive expressions within R Markdown documents.
- Compilation of HTML, PDF, or MS Word notebooks from R scripts.
- Extensibility: easily define new formats for custom publishing requirements.

This talk will cover the new features of R Markdown in depth and provide many examples of their use. We'll also discuss creating custom templates and output formats as well as extending R Markdown to include new types of rich web content.

References

Allaire, J.J. 2014. "R Markdown: Dynamic Documents for R." http://rmarkdown.rstudio.com.

Gruber, John. 2004. "Markdown." https://daringfireball.net/projects/markdown/.

MacFarlane, John. 2006. "Pandoc: A Universal Document Converter." http://johnmacfarlane.net/pandoc/.

Xie, Yihui. 2012. "knitr: A General-Purpose Package for Dynamic Report Generation in R." http://yihui.github.com/knitr/.

RcppZiggurat: Faster Normal Random Draws

Dirk Eddelbuettel

Debian and R Projects
edd@debian.org

Keywords: Simulation, Random Number Generation, Rcpp

Random numbers following a Standard Normal distribution are of great importance when using simulations as a means for investigation. The Ziggurat method [3, 4] is one of the fastest methods to generate normally distributed random numbers while also providing excellent statistical properties. However, the original papers only introduced 32-bit versions.

This talk introduces the **RcppZiggurat** package [1]. It provides updated implementations of the Ziggurat generator suitable for 32- and 64-bit operating system. It compares the original implementations to several popular Open Source implementations of the Ziggurat generator.

The package provides a new implementation which embeds the generator into an appropriate *C++* class structure [2]. The performance of the different generator is investigated both via extended timings and through a series of statistical tests, including a suggested new test for testing Normal deviates directly.

The new generator can be called via the package; further integration into *R* is discussed briefly as well.

References

[1] Eddelbuettel, D. (2013a). *RcppZiggurat: Rcpp integration of different Ziggurat Normal RNG implementations*. R package version 0.1.1.

[2] Eddelbuettel, D. (2013b). *Seamless R and C++ Integration with Rcpp*. Use R! New York: Springer.

[3] Leong, P. H. W., G. Zhang, D.-U. Lee, W. Luk, and J. Villasenor (2005, 2). A Comment on the Implementation of the Ziggurat Method. *Journal of Statistical Software 12*(7), 1–4.

[4] Marsaglia, G. and W. W. Tsang (2000, 10). The Ziggurat Method for Generating Random Variables. *Journal of Statistical Software 5*(8), 1–7.

Generalized Linear Models on Large Data Sets

Joseph Rickert[1,*], Susan Ranney[1]

1. Revolution Analytics
*Contact author: joseph.rickert@revolutionanalytics.com

Keywords: GLM, Tweedie, R, Big Data

Since their introduction by Nelder and Wedderburn over forty years ago, Generalized Linear Models (GLMs) have been a mainstay of statistical inference. Moreover, while they were originally employed with samples of relatively modest size, GLMs are now being employed with very large data sets in machine learning and data science applications. In this talk, we will briefly review the history of using GLMs in *R*, discuss the issues involved in using GLMs on large data sets, and show examples of various models including logistic regression, Poisson and Tweedie models running on large data sets using the Parallel External Memory algorithms implemented in **`RevoScaleR`** package of Revolution R Enterprise.

References

[1] Nelder, J.A. and R.W.M. Wedderburn. 1972 "Generalized linear models." *Journal of the Royal Statistical Society*, Series A 135:370--84

Visualizing Diseased Transcriptomes with *R*

Tracy Nance[1*], **Stephen B. Montgomery**[1]

1. Department of Pathology, Stanford University, USA
*Contact author: tracy.nance@gmail.com

Keywords: rna-seq, IPF, genes, visualization, transcriptome

Recent advances and decreasing costs in high-throughput sequencing have made possible the generation of vast quantities of biological data. Hundreds of experiments analyzing gene expression in diseases have been performed using RNA sequencing or older microarray technology, and a correspondingly large amount of statistical machinery has been built within R to analyze this data. However the results of these studies are usually presented to the community as lists of genes with p-values, and are difficult for biological researchers to reproduce, query or visualize.

We performed RNA-Seq analysis on lung tissue from 8 individuals with idiopathic pulmonary fibrosis (IPF) and 7 healthy controls, and found evidence for substantial differential gene expression and differential splicing in IPF as compared to healthy lungs [5]. Using a combination of R packages like **shiny** [6], **ggplot2** [8], **DESeq** [1], **DEXSeq** [2], **limma** [7], and the *JavaScript* library **D3** [3], we created a web application [4] which allows researchers to explore our results interactively, and to compare them with previously published microarray studies of the same disease. This application shows the power and ease of combining multiple R packages for visualization and making them available to the public, and we hope that it will serve as a toy model for making results accessible in large-scale genomic and transcriptomic studies.

References

[1] Anders, S. and W. Huber (2010). Differential expression analysis for sequence count data. *Genome Biology 11*, R106.

[2] Anders, S., A. Reyes, and W. Huber (2012, Oct). Detecting differential usage of exons from RNA-seq data. *Genome Res. 22*(10), 2008–2017.

[3] Bostock, M., V. Ogievetsky, and J. Heer (2011). D3: Data-driven documents. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)*.

[4] Nance, T. (2014). IPF Explorer. `http://stanford.edu/~tnance/ipf.html`.

[5] Nance, T., K. S. Smith, V. Anaya, R. Richardson, L. Ho, M. Pala, S. Mostafavi, A. Battle, C. Feghali-Bostwick, G. Rosen, and S. B. Montgomery (2014). Transcriptome Analysis Reveals Differential Splicing Events in IPF Lung Tissue. *PLoS ONE 9*(3), e92111.

[6] RStudio and Inc. (2013). *shiny: Web Application Framework for R*. R package version 0.7.0.

[7] Smyth, G. K. (2005). Limma: linear models for microarray data. In R. Gentleman, V. Carey, S. Dudoit, R. Irizarry, and W. Huber (Eds.), *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*, pp. 397–420. New York: Springer.

[8] Wickham, H. (2009). *ggplot2: elegant graphics for data analysis*. Springer New York.

dendextend: an R package for easier manipulation and visualization of dendrograms

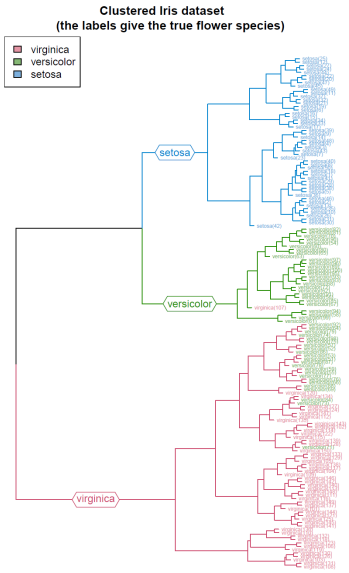
Tal Galili [1,*]

1. Tel Aviv University
*Contact author:Tal.Galili@gmail.com

Keywords: dendrogram, hierarchical clustering, hclust, visualization, tanglegram

In this talk I will introduce the **dendextend** package [1] which extends the palette of functions and methods for the `dendrogram` object.

A dendrogram is a tree diagram which is often used to visualize a hierarchical clustering of items. Dendrograms are used in many disciplines, ranging from Phylogenetic Trees in computational biology to Lexomic Trees in text analysis. Hierarchical clustering in *R* is commonly performed using the `hclust` function. When a more sophisticated visualization is desired, the `hclust` object is often coerced into a `dendrogram` object, which in turn is modified and plotted. While **base** *R* comes with several very useful methods for manipulating the `dendrogram` object (namely: `plot`, `print`, `[[`, `labels`, `as.hclust`, `cophenetic`, `reorder`, `cut`, `merge`, `rev`, and `str`), still - the current palette of functions leaves a lot to be desired.



The novel **dendextend** package offers functions and methods for the `dendrogram` object, allowing for easier manipulation of a dendrogram's shape, color and content through functions such as `rotate`, `prune`, `labels<-`, `labels_colors`, `cutree`, `color_branches`, and more. **dendextend** also provides the tools for comparing the similarity of two dendrograms to one another either graphically using a `tanglegram` plot, or statistically with association measures ranging from `cor_cophenetic` to `Bk_plot`, while enabling bootstrap and permutation tests for comparing the trees.

Since tree structure often requires the use of recursion, which can be slow in *R*, some of the more computationally intensive aspects of the **dendextend** package can be handled with its sister package, **dendextendRcpp** [2], which overrides several basic functions (namely: `cut_lower_fun`, `heights_per_k.dendrogram`, `labels.dendrogram`), with their *C++* implementation.

References

[1] Tal Galili (2014). dendextend: Extending R's dendrogram functionality, http://cran.r-project.org/web/packages/dendextend

[2] Tal Galili (2014). dendextendRcpp: Faster dendrogram manipulation using Rcpp, http://cran.r-project.org/web/packages/dendextendRcpp

Robust model selection: New developments in the *R* package robustHD

Andreas Alfons[*]

Erasmus School of Economics, Erasmus Universiteit Rotterdam

[*]Contact author: alfons@ese.eur.nl

Keywords: Outliers, Robust groupwise LARS, Sparse S-regression, Sparse MM-regression, *C++*

Variable selection is a common task in regression analysis to improve prediction performance by variance reduction, and to increase interpretability of the resulting models due to the smaller number of variables. In the presence of outliers, robust methods are necessary to prevent unreliable results. The *R* package **robustHD** [2] provides functionality for robust linear model selection with a focus on methods for high-dimensional data. New developments include robust groupwise least angle regression, sparse S-regression and sparse MM-regression. The package implements an object-oriented design, while large parts of the code are written in *C++* to reduce computing time. Cross-validation functionality to select the final model is implemented via package **perry** [1] such that taking advantage of parallel computing is easy. In addition, diagnostic plots to evaluate the model selection procedures are available in **robustHD**.

References

[1] Alfons, A. (2013). **perry***: Resampling-based prediction error estimation for regression models*. *R* package.

[2] Alfons, A. (2014). **robustHD***: Robust methods for high-dimensional data*. *R* package.

GLARMA Models and the glarma Package

William Dunsmuir[1], **David Scott**[2,⋆],

1. University of New South Wales
2. University of Auckland
⋆Contact author: d.scott@auckland.ac.nz

Keywords: Time series, count data, generalized linear models

In the past 15 years there has been substantial progress made in developing regression models with serial dependence for discrete valued response time series such as arise for modelling Bernoulli, binomial, Poisson or negative binomial counts. In this paper we consider the GLARMA (generalized linear autoregressive moving average) class of models which are a subclass of generalized state space models for non-Gaussian time series described in [2], [1] and [3] for example.

We review the theory and application of GLARMA models for time series of counts with explanatory variables and describe the estimation of these models using the **glarma** *R* package. Diagnostic and graphical methods are illustrated by several examples.

References

[1] Brockwell, P. J. and R. A. Davis (2010). *Introduction to Time Series and Forecasting* (2nd ed.). New York, NY: Springer-Verlag.

[2] Davis, R. A., W. T. Dunsmuir, and Y. Wang (1999). Modeling time series of count data. In S. Ghosh (Ed.), *Asymptotics, Nonparametrics, and Time Series*, Volume 158 of *Statistics Textbooks and Monographs*, pp. 63–114. New York, NY: Marcel Dekker.

[3] Durbin, J. and S. J. Koopman (2012). *Time Series Analysis by State Space Methods*. Oxford: Oxford University Press.

docopt, add beautiful command line options to *R* scripts

Edwin de Jonge[1,⋆]

1. Statistics Netherlands (CBS)
⋆Contact author: e.dejonge@cbs.nl

Keywords: Options, Rscript, docopt, programming

With its increasing popularity *R* scripts are more and more executed in batch mode from the command line. When a script matures and becomes more generic it often is desirable to add command-line options to it. Starting simple you may use `cmdArgs` to parse the extra options given to the script, but it quickly becomes complicated: parsing multiple options, long and short names for options, default values for options, generating sensible information when incorrect input is given and documenting all options for usage can take quite some code and time. The packages **getopt** [1] and **optparse** [4] can be of great help for parsing command-line arguments, but **docopt** makes it super easy.

The R-package **docopt** [2] is a port of the *Python* package `docopt`. Docopt is a command-line interface description language, and helps you to formulate a command-line interface and automatically generate a parser for it, that will parse the command-line arguments automatically for you. The nice part is that the documentation given to the user is equal to its definition.

As an appetizer: `docopt("Usage: prog [-a -b=value]")` will generate a parser and test if `a` and `b` were set in the command line. It returns a list in which the values for `a` and `b` are included.

The presentation will introduce **docopt**, describe its usage, specification format and some of its implementation details.

References

[1] Allen Day, T. L. D. (2013). *getopt: C-like getopt behavior.* R package version 1.20.0.

[2] de Jonge, E. (2013). (docopt), command-line interface description language for r. `http://github.com/edwindj/docopt.R`.

[3] Keleshev, V. (2012). (docopt), command-line interface description language. `http://docopt.org`.

[4] Trevor L Davis, S. L. and J. Nikelski. (2013). *optparse: Command line option parser.* R package version 1.0.2.

[5] Van Rossum, G. et al. (2007). Python programming language. In *USENIX Annual Technical Conference*.

An R tools platform in Cosmetic Industry

Jean-François COLLIN[1]

1. L'Oréal Research and Innovation, Aulnay-sous-bois, France
*Contact author: jfcollin@rd.loreal.com

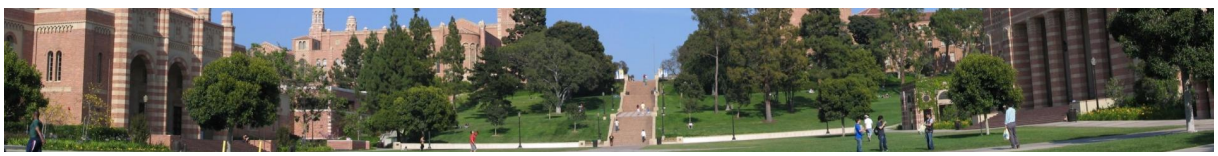
Keywords: Graphical User Interface**,** platform

In vitro, in vivo or ex vivo studies are continually performed on ingredients and formulae in our laboratories for efficacy and safety purposes. Results must be statistically analyzed, and stored together with the programs used affording a sustainable research policy. In such a context, a statistical platform was created including R (a language and environment for statistical computing) tools that integrate automatic reporting system connected to a centralized data basis.

Some applications aim at helping researchers to carry on analysis of their data by their own whereas more general tools are dedicated to statisticians for quickly producing accurate statistical reports with a wide variety of statistical methods: mixed models, multidimensional analysis, exploratory analysis etc…

This platform comprises tools that are entirely built within R (Graphical User Interface and statistical programs). This presentation will focus upon the platform, its functionalities and technical aspects. The tools built to perform exploratory analysis and mixed model analysis will be presented. Examples of a 'non statistician' tool and some packages specifically conceived for answering to our laboratories needs will be given.

Posters



unmixR: Hyperspectral Unmixing in *R*

Conor McManus[1], **Simon Fuller**[2], **Claudia Beleites**[3,4⋆], **Bryan A. Hanson**[5⋆]

1. National University of Ireland Maynooth, Maynooth, Ireland
2. Computer Science Dept., National University of Ireland Maynooth, Maynooth, Ireland
3. Leibniz Institute of Photonic Technolology, Jena, Germany
4. Chemometrische Beratung, Bad Nauheim, Germany
5. Dept. of Chemistry & Biochemistry, DePauw University, Greencastle IN USA
⋆Contact authors: chemometrie@beleites.de & hanson@depauw.edu

Keywords: hyperspectral unmixing, imaging spectroscopy

Hyperspectral images are 3D data sets collected over an x, y grid, where the pixel at each x, y is composed of a spectrum. In hyperspectral unmixing, such a data set $\mathbf{X}$ composed of n observed spectra with p wavelengths or spectral bands is decomposed to identify the pure component spectra. Such data sets are found in airborne land imaging studies, biomedical and art history investigations as well as time series (kinetics) of chemical reactions. The spectra are typically visible, infrared, near-infrared, Raman spectra or mass spectrometric data sets.

Each spectrum is assumed to be a linear mixture of a limited number m of pure component spectra, the so-called endmembers. m is also referred to as chemical rank of the spectra matrix $\mathbf{X}$. Endmembers are suitable for direct interpretation in the domain of the study (e.g. Raman spectra of cancerous tissue, reflectance spectra of minerals) and the goal is to identify them. The spectra matrix $\mathbf{X}^{(n\times p)}$ can be thought of as a sum:

$$\mathbf{X}_{np} = \sum_m \mathbf{A}_{nm}\mathbf{E}_{mp} + \varepsilon \qquad \text{or in matrix notation:} \qquad \mathbf{X}^{(n\times p)} = \mathbf{A}^{(n\times m)}\mathbf{E}^{(m\times p)} + \varepsilon \tag{1}$$

Where n is the number of spectra and m is the number of endmembers. Thus, $\mathbf{E}_{mp}$ is the m^{th} endmember spectrum composed of p bands, and the abundances A_{nm} give the contribution of the m^{th} endmember to the n^{th} sample spectrum. ε is (Gaussian) noise. The abundances $\mathbf{A}$ can be depicted in a mixture diagram of the m components which forms a $(m-1)$-simplex. Thus the spectra $\mathbf{X}$ lie in a $(m-1)$-simplex. If pure component spectra of all m components are available in the data and the noise level is low, the decomposition given above can be obtained by finding the corners of the $(m-1)$-simplex in $\mathbf{X}$.

N-FINDR achieves this by first projecting the data into $(m-1)$-dimensional space, usually by PCA. Starting from a set of m (randomly chosen) potential endmembers, an iterative procedure to maximize the volume of the simplex is used: $m-1$ endmembers are kept fixed and the simplex volume is maximized by varying the remaining point. This is in turn done with each corner of the simplex until a stable solution is reached.

Vertex component analysis (VCA), in contrast, projects the spectra onto an orthogonal set of m axes and chooses the extreme points as endmembers.

Physically, both $\mathbf{E}$ and $\mathbf{A}$ are subject to non-negativity constraints as $\mathbf{E}$ takes the role of pure component spectra and $\mathbf{A}$ correspond to concentrations or molar fractions. If $\mathbf{A}$ is formulated as molar fraction, the rows of $\mathbf{A}$ must sum to 1. Thus, once the endmember spectra are found, $\mathbf{A}$ is obtained by a non-negative least squares fit of the remaining spectra.

unmixR (`http://github.com/Chathurga/unmixR`) provides different N-FINDR and VCA algorithms as an *R* package. The Google Summer of Code 2013 supported Conor McManus to implement the algorithms, supervised by Claudia Beleites, Simon Fuller and Bryan Hanson. Claudia Beleites now maintains the package. Claudia Beleites thanks the BMBF for funding via the project "RamanCTC" (13N12685).

Learning *R*: Needs Analysis, Learning Taxonomies, Methodology, and Visualization

Alon, Friedman PhD[1], Edd Schneider, PhD[2]
1 & 2 School of Information, University of South Florida
*Contact: alonfriedman@usf.edu

Keywords: New user, College students, curriculum, statistics, taxonomies, visualization.

The growing popularity of *R* has presented new challenges for educators as well as for new users. This paper addresses *R* from an educational standpoint, providing a methodological approach for teaching statistics and programming in any arts and science college curriculum. For many years, the main feature of the development of *R* focused on the growth of its popularity and the widening accessibility of the technology across platforms. However, with this growth more attention is needed to address specific educational concerns of people wanting to teach *R*. The open source nature of the application presents unique problems for educators, since the mechanics and functionality of *R* are constantly evolving. This in turn makes *R* a challenge in terms of instructional design.

In order to address this challenge, this presentation will share the results of the first year of an experiment teaching the *R* to undergraduate students in the University of South Florida's School of Information. Faculty taught the subject in both face-to-face and online class environments. The educational context of the lessons utilizing *R* focused on the power of the application to produce visualization; specifically visualization of large data sets. While the results of measuring student feedback from the class are still being measured and analyzed, early results indicate that using visualization can be guiding a factor for new users to understand why they are learning about *R*, while simultaneously improving their understanding of statistics by using *R*. Essentially, the classroom model discussed here uses the visualization aspects of the application to motivate students to further their own understanding of statistics through a better command of *R*'s data analysis functionality.

During the spring of 2013, we taught two different courses in *R* in our undergraduate program. A pedagogical framework was developed which encouraged students to become more personally involved in statistical production and control by creating visualization. Learning in an academic course on data analysis is more complex than merely getting students to learn a language for its own sake, and faculty have found that students do not necessarily learn through having an example explained step-by-step through the basic procedure of manipulating data. Many authors discussed the subject of "delivery" of statistics in the classroom and many point out the lack of standardized "visualization delivery". Further more, there are numerous educational taxonomies that could potentially be applied in the context of improving *R* education. However, a survey of materials found little in the way of attempts to explore a theoretical framework to teach statistics, programming and visualization using an open source platform.

We conducted a methodology study in the context of the specific statistics and programming languages courses taken prior to the students' course in *R*. Our framework blends Constructivist and Cognitivist approaches to instruction. Details of the results of this methodology will be presented, along with an analysis of how these results will influence teaching in the future.

Investigating cold light: The *R* package Luminescence - signal, statistics and dating of environmental dynamics -

Sebastian Kreutzer[1,2,*], Steve Grehl[3], Michael Dietze[4], Christoph Burow[5], Margret C. Fuchs[6], Manfred Fischer[7], Christoph Schmidt[7]

1. IRAMAT-CRP2A, Université Bordeaux-Montaigne, Maison de l'Archéologie, Esplanade des Antilles, 33607 Pessac Cedex, France
2. Department of Geography, Justus-Liebig-University Giessen, Senckenbergstrasse 1, 35390 Giessen, Germany
3. Freiberg Instruments GmbH, Delfter Str. 6, 09599 Freiberg, Germany
4. Section 5.1 Geomorphology, GFZ German Research Centre for Geosciences, 14473 Potsdam, Germany
5. Institute for Geography, University of Cologne, 50923 Cologne, Germany
6. Alfred Wegener Insitute for Polar and Marine Research, Department of Periglacial Research, Telegraphenberg A43, 14473 Potsdam, Germany
7. Geographical Institute, Geomorphology, University of Bayreuth, 95440 Bayreuth, Germany

[*]Contact author: sebastian.kreutzer@geogr.uni-giessen.de

Keywords: Luminescence Dating, Geosciences, Dosimetry, Signal Analysis, Dynamic GUI

Earth surface processes decisively shape our planet. They are the primary mediators of environmental change and directly affect human societies. To decipher the timing and rates of Earth surface processes, throughout the last 250,000 years one numerical dating method has reached paramount importance: Luminescence dating. This method provides robust numerical data on environmental changes (a) due to the fact that the luminescence signal is reset by daylight exposure or heating and (b) the advantage of using nearly ubiquitously available mineral grains of quartz or feldspar.

During the last decades more and more ages based on luminescence dating have been requested and the method has been considerably enhanced. However, an increasing data complexity demands for a flexible and scalable software solution for data analysis. For more innovative measurements, existing software solutions (e.g. 'Analyst', Duller, 2007) are limited, especially regarding new experimental measurements.

Therefore, in 2012 the *R* package **Luminescence** has been introduced (Kreutzer et al., 2012, Dietze et al., 2013, Fuchs et al., subm.). The package is a toolbox intended to provide customised solutions for a variety of requirements (e.g. data import, statistical analysis, graphical output). The used algorithms and statistical treatments are always transparent and the user remains in control, of combining and adjusting algorithms by taking advantage of the wide range of functions available in *R*.

Our contribution (1) summarizes the concept of the *R* package **Luminescence** and focusses on some conceptional aspects and selected practical examples. (2) We present a sneak preview on a dynamic graphical user interface written in *Java* to make the functions of the package available to users who are not familiar with *R* but wants to access the package functionality.

References

Dietze, M., Kreutzer, S., Fuchs, M.C., Burow, C., Fischer, M., Schmidt, C. (2013). A practical guide to the R package Luminescence, Ancient TL 31, 11–18.

Duller, G.A.T. (2007). Analyst. Unpublished manual.

Fuchs, M.C, Kreutzer, S., Burow, C., Dietze, M., Fischer, M., Schmidt, C., Fuchs, M. (subm.). A practical workflow for data analyses in luminescence dating using the R package 'Luminescence': A case study from the Panj River, Pamir, submitted to Quaternary International

Kreutzer, S., Schmidt, C., Fuchs, M.C., Dietze, M., Fischer, M., Fuchs, M. (2012). Introducing an R package for luminescence dating analysis. Ancient TL 30, 1–8.

EpiDynamics 0.1: Dynamic Models in Epidemiology

Fernando S Marques[1,*], **Oswaldo Santos**[1,**], **Fernando Ferreira**[1], **Marcos Amaku**[1]

1. School of Veterinary Medicine and Animal Science, University of São Paulo
Contact authors: *fernandosix@gmail.com, **oswaldosant@gmail.com

Keywords: Epidemiology, Models, Disease Spread

Models in epidemiology are important tools that help understand the complexity and evolution of disease spread. However, teaching these models might be troublesome since many students are not experienced in writing code and simulation techniques. The **EpiDynamics** package provides pre-built epidemiological models to help students understand the evolution of disease spread through *R* language. We collected and implemented models from well-known book written by Matt J. Keeling and Pejman Rohani [1], from basic to more complex models, e.g.:

- SIR model with births and deaths
- SIR model with disease induced mortality and density dependent transmission
- SIS model with n risk groups
- SEIR model with n age groups and yearly aging
- SIR model with partial immunity
- SIR model for mosquito vectors
- SIR model with sinusoidal forcing

References

[1] Keeling, M. J. and P. Rohani (2008). *Modeling infectious diseases in humans and animals*.

The PMMLTransformations package

Tridivesh Jena[1,*] , Alex Guazzelli[1,] , Wen Ching Lin[1] , Michael Zeller[1]

1. Zementis, Inc.
*Contact author: tridivesh.jena@zementis.com

Keywords: R, Predictive Analytics, PMML, Data Transformations, Standards

PMML, the Predictive Model Markup Language, is the de facto standard to represent predictive analytic models [1,2,3]. With PMML, is is extremely easy to move the model from the scientist's desktop to the operational IT environment for real-time execution or batch scoring, since there is no recoding necessary.

Typically, the analytic process involves quite a bit of data pre-processing before a predictive model is build; the R **pmmlTransformations** package was designed to cover just such a need [4,5]. This package not only enables a data scientist to transform input data but also, when used along with the R **pmml** package [6,7], makes it possible to represent the transformation steps in PMML. The produced PMML file will then contain the combination of the predictive model itself together with all the steps necessary to pre-process incoming data. This is remarkable, since it allows for systems and applications to connect directly to the raw input data and leave it up to the PMML consumer to deliver the expected predictions.

The **pmmlTransformations** package implements many of the commonly used transformation operators used by data scientists, among them the Z-transform, linear transformation, data discretization, data normalization and value mapping. The result is not only the transformed data itself but also information to represent the transformation operators in PMML format. In this work, we illustrate the steps necessary to: 1) acquire raw data; 2) pre-process the raw data through available operators; 3) gain access to the manipulated data; and 4) output all the performed operations in PMML format. We also describe the various settings and options associated to each transformation operator and available to the data scientist. After the entire predictive workflow is exported into PMML, the predictive model or solution can easily be operationally deployed and executed in a variety of platforms including Hadoop, in-database or cloud computing.

References

[1] The Data Mining Group (DMG) website: www.dmg.org

[2] A. Guazzelli, W. Lin, T. Jena (2010). *PMML in Action: Unleashing the Power of Open Standards for Data Mining and Predictive Analytics (2nd Edition)*. CreateSpace (available on Amazon.com).

[3] A. Guazzelli (2010). What is PMML? Explore the power of predictive analytics and open standards. IBM developerWorks website.

[4] T. Jena, A. Guazzelli, W. Lin, M. Zeller (2013). The R *pmmlTransformations* Package. *In Proceedings of the 19th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.*

[5] The R pmmlTransformations package: http://cran.r-project.org/web/packages/pmmlTransformations/index.html>

[6] A. Guazzelli, M. Zeller, W. Lin, G. Williams (2009). **PMML: An Open Standard for Sharing Models**. *The R Journal*, Volume 1/1.

[7] The R pmml package: http://cran.r-project.org/web/packages/pmml/index.html

And you want to interact with it using a spreadsheet? Simple connections between R and Microsoft Excel

Scott Porter[1,*]

1. Added Value
*Contact author: scott.porter@added-value.com

Keywords: Interoperability, GUI, Excel, prototype interface

Although we are moving to more web applications as deliverables for clients, we still have occasion where the desired method of delivery is a Microsoft Excel spreadsheet. Our clients are used to simulators and scenario tools built in Excel, and they generally take simulation results and further format or analyze them in Excel.

There are methods of deeply integrating *R* and Excel, such as the RExcel [1] add in for Excel. However, when building an application for a client that will need to be installed on multiple computers in a computing and network environment that you do not control, this type of deep integration is not necessarily desired.

The criteria that have emerged as the most important in our past engagements have been reduced likelihood of failure (and doing so visibly when it occurs), ease of adjusting for differences that may occur in different computing environments, requiring the least amount of software to be installed, and causing minimum interference with existing software setups.

This paper describes simple ways to make clean handoffs between Excel and *R* where Excel will provide most of the user interface and *R* will provide the heavy lifting in terms of simulation or other calculations. By keeping the handoff between the two pieces of software limited, it is also easy to replace the Excel component in the future, so this approach can also be a way to easily prototype analysis applications whose interface will eventually be replaced (for example, with a web-based interface).

References

[1] Thomas Baier and Erich Neuwirth (2007). Excel :: Com :: *R. Computational Statistics*, 22, 91–108.

Faster FastR through Partial Evaluation and Compilation

Michael Haupt[1,*], **Christian Humer**[2], **Mick Jordan**[1], **Prahlad Joshi**[3], **Jan Vitek**[3], **Adam Welc**[1], **Christian Wirth**[1], **Andreas Wöß**[2], **Mario Wolczko**[1], **Thomas Würthinger**[1]

1. Oracle Labs
2. Johannes Kepler University, Linz, Austria
3. Purdue University, West Lafayette, IN, USA
*Contact author: michael.haupt@oracle.com

Keywords: R Language Runtime, Java, R Performance, AST Interpretation, Partial Evaluation

FastR, first introduced at *useR! 2013* [3], is an implementation of the *R* programming language in *Java* [2]. It uses the concept of self-specialising abstract syntax tree (AST) interpretation [5]. In such interpreters, AST nodes replace themselves with nodes that are specialised for handling the types and data actually occurring during execution. This saves considerable time in the implementation of dynamically typed programming languages.

The implementation introduced in 2013 was a pure interpreter. We introduce the next version of *FastR*. The current implementation is based on Truffle [4]. Truffle is a framework for the implementation of specialising AST interpreters. Truffle-based language implementations transparently employ partial evaluation of specialised ASTs, and dynamic compilation, to obtain performance competitive with that of dedicated dynamic compilers.

The performance of *FastR* running the b25 benchmarks and an *R* version of a subset of the Computer Language Benchmarks Game ("shootout") is, on average, more than an order of magnitude faster than the GNU R byte code interpreter, and significantly faster than the purely interpreted version of FastR. *FastR* is available as an open source project [1] under the terms and conditions of the GNU General Public License 2.

We will describe the status of the implementation and outline our plans for the future. An important long-term goal of the *FastR* project is to dispense with the need for implementing performance-critical parts of *R* applications in lower-level languages.

References

[1] BitBucket (2014). FastR project. http://bitbucket.org/allr/fastr.

[2] Kalibera, T., P. Maj, F. Morandat, and J. Vitek (2014). A Fast Abstract Syntax Tree Interpreter for R. In *Proceedings of the 10th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments*, VEE '14, New York, NY, USA, pp. 89–102. ACM.

[3] Kalibera, T., P. Maj, and J. Vitek (2013). R in Java: Why and How? In *The R User Conference, useR! 2013, Book of Contributed Abstracts*, pp. 111. http://www.edii.uclm.es/~useR-2013/docs/useR2013_abstract_booklet.pdf.

[4] Würthinger, T., C. Wimmer, A. Wöß, L. Stadler, G. Duboscq, C. Humer, G. Richards, D. Simon, and M. Wolczko (2013). One VM to rule them all. In *Proceedings of the 2013 ACM international symposium on New ideas, new paradigms, and reflections on programming & software*, pp. 187–204. ACM.

[5] Würthinger, T., A. Wöß, L. Stadler, G. Duboscq, D. Simon, and C. Wimmer (2012). Self-optimizing AST interpreters. In *Proceedings of the 8th Symposium on Dynamic Languages*, DLS '12, New York, NY, USA, pp. 73–82. ACM.

Using RGraphviz as a first pass for layout of small structural model graphs

Scott Porter[1,*]

1. Added Value
*Contact author: scott.porter@added-value.com

Keywords: Structural models, Graphs, Visualization, Readability, Layout

We use automated search algorithms to find plausible structural models (such as Structural Equation Models or Bayes Networks) in the context of marketing and consumer perceptions. We are often trying to lay out some subset of these models with a limited set of variables for presentation to stakeholders to discuss reasonability of the discovered models.

The **RGraphviz** [1] package provides an interface to the AT&T Labs Research graphviz library [2] for plotting graphs. The graphviz library includes several layout engines. We have found it to be useful to use **RGraphviz** to do an initial layout of the graphs.

This initial layout produced by graphviz is a useful starting point for various next steps we may take with the graphs. Interactive graph display methods (either within *R*, or within a web display framework) can use the layout to set starting positions for each of the variables. The initial layout can also be used as a first pass for creating printable versions of the graph, such as for posters or slides, although we generally make additional manual adjustments to the layout.

References

[1] Jeff Gentry, Li Long, Robert Gentleman, Seth Falcon, Florian Hahne, Deepayan Sarkar and Kasper Daniel Hansen (). Rgraphviz: Provides plotting capabilities for R graph objects. R package version 2.2.1.

[2] Graphviz.org, AT&T Labs Research

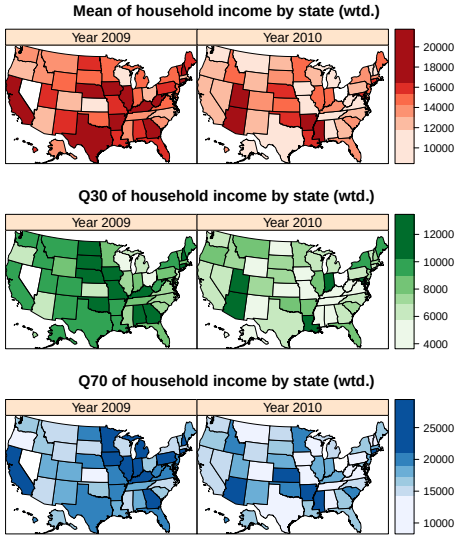
mapStats: An R Package for Geographic Visualization of Survey Statistics

Samuel S. Ackerman*

Department of Statistics, Temple University
*Contact author: ackerman@temple.edu

Keywords: maps, lattice, shapefile

mapStats is an *R* package that produces color-coded maps of survey data using **lattice** graphics. It calculates variable statistics (mean, quantiles, total), either weighted or unweighted, from survey data where observations have a geographic element, and produces a map with colors representing the levels of the statistic. For example, one may want to visualize how the median and mean income of respondents by state in a survey changes over several years or differs by education level. Visualizing survey statistics geographically, which this package greatly simplifies, is an important way to understand characteristics of the survey data quickly. **mapStats** has flexible user control over appearance, including options such as overplotting maps with another shapefile or graphical element. Generating quality graphics without the package would require familiarity with **lattice** utilities and knowledge of manipulating shapefiles to merge them with data for plotting, which may be difficult for a casual user. Methods in *R* for calculation of summary statistics, such as quantile regression, require manual case-by-case manipulation of output, and do not necessarily perform well in cases where levels of class combinations are missing. The package includes functions for calculating variable statistics which deal well with missing data or class combinations, and that can be used for class variables in general (not necessarily geographies) or without plotting. An example of output on synthetic data is shown below:



References

Ackerman, Samuel (2013). "mapStats: An R package for geographic display of survey statistics."

An Integrated Environment for Social Research Analysis

Yasuto NAKANO[1,*]

1. School of Sociology, Kwansei Gakuin University, JAPAN
*yasuto@kwansei.ac.jp

Keywords: Social Research, DDI

The purpose of this presentation is to propose an environment for social research and its analysis.

If there were clear research questions, prosess of social research remains as follows; (1)editing questions, (2)editing questionaire, (3)conducting a survey, (4)inputing data, (5)cleaning data, (6)analysing data, (7)publishing report and article. In each step, same informations are used repeatedly but independently. For example, there is a quetion about gender. Items of answer are [1. Male 2. Female]. They might be typed in a qustionaire using some wordprosessing software (e.g. ms-word). In the step of inputing data, they could be treated as a code (1 or 2) using some spreadsheet software (e.g. ms-excel). In the step of analysis, data could be read from excel file, and labeled as 'male' or 'female' for each code using R functions(e.g. factor(), levels()). In the step of publishing a report, result of analysis could be pasted to a file of wordprocessing software. What a waste of time. How convenient it would be, if we could use informations of step (1) throughout all process of social research. There is a standard documentation and data format for social research, that is DDI (Data Documentation Initiative). DDI is a XML protocol to describe informations related to social research including questionnaire, data and report. We propose a package **DDIR** which utilize informations in DDI format. Once a questionnare is described in DDI format, we could utilize that informations as variable labels and value labels in analysis process by **DDIR**. These informations could be also automatically divert to publishing process. **DDIR** proposes an efficient integrated environment for social research analysis.

lsh, Nearest neighbor search in high dimensions

Edwin de Jonge[1,⋆]

1. Statistics Netherlands (CBS)
⋆Contact author: e.dejonge@cbs.nl

Keywords: Machine learning, Locality Sensitive Hashing, high dimensional nearest neighbor

Data sets with many variables and rows are very common nowadays. The number of dimensions p can run from ten to thousands of variables. The number of observations n typically runs from thousands to millions. Both a large n and p are challenging for traditional nearest neighbor techniques.

Calculating distance pairs is $O(n^2)$ in memory and time and finding the nearest neighbor is $O(n)$ in time. Tree indexing techniques like kd-tree [2] were developed to cope with large n, however their performance quickly breaks down for $p > 3$ [3]. Locality sensitive hashing (LSH) [3] is a technique for generating hash numbers from high dimensional data, such that nearby points have identical hashes. This enables efficient nearest neighbor search for (very) high dimensional data sets. It has been successfully applied to several problems including text similarity search [5].

R package **lsh** [4] (in development) is an implementation of locality sensitive hashing in *R*. We will describe the implemented locality sensitive hashing technique, the several distance functions and the functionality of **lsh**. Suggestions for further tuning performance will be provided.

References

[1] Andoni, A. and P. Indyk (2006). Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In *Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on*, pp. 459–468. IEEE.

[2] Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM 18*(9), 509–517.

[3] Datar, M., N. Immorlica, P. Indyk, and V. S. Mirrokni (2004). Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the twentieth annual symposium on Computational geometry*, pp. 253–262. ACM.

[4] de Jonge, E. (2014). lsh, locality sensitive hashing in r. `http://github.com/edwindj/lsh`.

[5] Slaney, M. and M. Casey (2008). Locality-sensitive hashing for finding nearest neighbors [lecture notes]. *Signal Processing Magazine, IEEE 25*(2), 128–131.

Hansel: An Econometrics Plug-In for Deducer

R. Scott Hacker[1,*]

1. Jönköping International Business School, Jönköping, Sweden
*Contact author: Scott.Hacker@jibs.hj.se

Keywords: GUI, **Deducer**, econometrics, time series, panel data

This presentation discusses the development of a **Deducer** plug-in, referred here to as **Hansel**, that can deal with techniques typically found in undergraduate courses in econometrics, along with some more advanced econometric techniques (the final package name should be something like **DeducerHansel**). Currently the **Deducer** package (FellowsI [1]) provides an exceptional interface that deals with a number of areas including generalized linear models. Thus it can already deal with ordinary least squares, weighted least squares, probit models and logit models. However it is not currently well-suited for dealing with time-series data, panel data, or censored data, or for dealing with instrumental variables. That is where **Hansel** helps. The following areas are among those covered by **Hansel**: two-stage least squares; tobit models; smoothing, filtering, and forecasting; unit root testing; vector autoregressive models; cointegration testing; and various panel data and spatial data techniques. **Hansel** can deal with the time series classes ts, zoo, and xts in addition to data frames. **Hansel** is similar in ease to the commercial software *EViews* and another open-source econometric software package called *gretl*, which is written in C. **Hansel** is not only useful for students in econometrics courses, but also provides an opportunity for those unacquainted with *R* to quickly get down to the business of using it for estimation. This can provide a gateway for deeper use of *R*.

References

[1] FellowsI (2012). Deducer: A Data Analysis GUI for R. *Journal of Statistical Software 49*(*8*), 1-15.

R users all around the world

Gergely Daroczi[1,2,*]

1. Founder of Easystats Ltd, United Kingdom
2. PhD candidate at Corvinus University of Budapest, Hungary

Contact authors: * daroczig@rapporter.net

Keywords: R activity, users, conferences, user groups, CRAN statistics

The poster shows an annotated but mainly visual map of the world, which highlights the activity of *R* users from various points of view. The plots and the infographics were created in *R*, inspired by some recent blogposts of cartograms (see the references below), but now we build on a lot wider variety of date sources already collected, cleaned, merged and aggregated by the author.

The data sources include the number of visitors of `R-bloggers.com`, the attendees of all previous useR! conferences, the members and supporters of the R Foundation, the number of users on GitHub with *R* repositories and package download statistics from CRAN mirrors.

Besides these raw data, the poster will also present a population-weighted scale of *R* activity for all countries of the world – for the last 10 years since the first useR! conference.

References

James Cheshire (2013). Where is the R Activity? `http://spatial.ly/2013/06/r_activity/`

Gergely Daroczi (2013). The attendants of useR! 2013 around the world. `http://blog.rapporter.net/2013/11/the-attendants-of-user-2013-around-world.html`

SeekR: A Search Engine for R users

Takekatsu Hiramura*

*Contact author: thira@plavox.info

Keywords: Search engine, The R Ecosystem, Learning R

For most people using the statistical computing environment R, it is difficult to search for information about the search engine R, since R is a single letter. The website **Rseek** [1] provides a solution to this problem. **Rseek** is a search engine that only provides results whose content is related to R, and it does so by using Google Custom Search [2]. In order to configure Google Custom Search, a search engine administrator registers the URLs of website to be searched. **Rseek** brings up R-related websites, which are principally written in English.

In July 2009 I launched a search engine named "**SeekR**" [3] which brings up Japanese R-related websites, for Japanese users of R. Basically, SeekR is a Japanese version of **Rseek**, but it has some additional user-friendly features. One of these features is a Web browser add-on, which allows users to add an option in right-click menu for redirecting the **SeekR** search results page. This Add-on is available for Mozilla Firefox and Google Chrome. In addition, **SeekR** is compatible with OpenSearch, a standard format for search engines. OpenSearch allows users to add a standard search engine on many Web browsers. Another feature of **SeekR** is speech input. **SeekR** has speech input feature: with HTML5 speech input technology, users can input a search query simply by speaking aloud. It is available for Web browsers support speech input feature, such as the latest version of Google Chrome.

In March 2014, I launched a Chinese version of **SeekR** [4] which allow users to switch between simplified Chinese and traditional Chinese. This version brings up search results for Websites whose R-related content is written in Chinese. This version of **SeekR** is still being improved, in order to make the R Ecosystem [5] more productive and user-friendly.

References

[1] Sasha Goodman. Rseek, http://www.rseek.org/.

[2] Google Inc. Google Custom Search, https://www.google.com/cse/.

[3] Takekatsu Hiramura (2009). SeekR (Japanese version), http://seekr.jp/.

[4] Takekatsu Hiramura (2014). SeekR (Chinese version), http://seekr.asia/.

[5] David Smith (2011). The R Ecosystem. *The R User Conference, useR! 2011 Book of Contributed Abstracts*, 12.

Shiny-*ing* compareGroups

Joan Vila[1,2,*], **Isaac Subirana**[2,1,3], **Héctor Sanz**[4], **Judith Peñafiel**[1,2], **David Gimenez**[1]

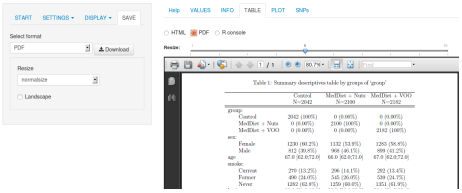
1. Cardiovascular Epidemiology & Genetics group, Inflammatory and Cardiovascular Disease Programme, IMIM, Barcelona
2. CIBER Epidemiology and Public Health (CIBERESP), Spain
3. Statistics Department, University of Barcelona, Spain
4. Barcelona Centre for International Health Research (CRESIB, Hospital Clínic-Universitat de Barcelona), Barcelona, Spain
*Contact author: jvila@imim.es

Keywords: Shiny, Software Design, Bivariate Table, LaTeX, Descriptive Analysis

The **compareGroups** package is an utility tool available on CRAN designed to build tables containing descriptions of several variables stratified by groups that can be displayed in a clear, easy to read format on the *R* console, or included in a LaTeX report, or exported to CSV or HTML file. Since 2010 when **compareGroups** package was first presented at useR! 2010 conference [1], a lot of work has been done to debug and improve it till the 2.1 current version, incorporating several important changes and innovations to improve its functionality and its capability, improving customisation of output tables (including number of decimals, categorisation, character formatting etc.), reading and coding raw SNP data using the functionality of the **SNPassoc** package[2], with quality control processes and summary outputs. The vignette has been extended with several examples, in line with new functionality, using data from a longitudinal RCT with >7,000 individuals: the PREDIMED study [3]. A paper about this package has been reviewed and accepted at the Journal of Statistical Software and it's only pending to be shortly published .

Shiny[4] is an elegant and powerful web framework for building interactive reports and visualizations using R - with or without web development skills. By Shiny analyses can be turn into interactive web applications that anyone can use. The users choose input parameters using easy and fiendly controls. The output incorporates any number of plots, tables, and summaries. We use the Shiny Server software to put our **compareGroups** package in an open free-access web. Users need only a web browser to access at our applications URL.

It opens a world of possibilities for analysts and students. Users only need prepare a data base in *SPSS*, *TEXT*, *EXCEL* or *R*. The menu (see figure) helps the users to produce the tables and figures they want. The output can be seen in the screen or be saved in different formats: the tables in *PDF*, *HTML* or Text file and the figures in *PDF*, *BMP*, *JPG*, *RPNG* or *TIF*.



References

[1] Sanz H, Subirana I, Vila J. Bivariate analyses. In *useR! 2010, The R User Conference (National Institute of Standards and Technology, Gaithersburg, Maryland, US)*, July 2010.

[2] González JR, Armengol L, Guinó E, Solé X, Moreno V.*SNPassoc: SNPs-based whole genome association studies*, 2012.URL http://CRAN.R-project.org/package=SNPassoc. R package version 1.8-5.

[3] Estruch R, Ros E, Salas-Salvadó J, Covas MI, Corella D,Arós F, et al. *Primary Prevention of Cardiovascular Disease with a Mediterranean Diet*. New England Journal of Medicine. vol 368-14, 1279-1290. 2013.

[4] URL http://www.rstudio.com/shiny/.

capm 0.4: an *R* package for Companion Animal Population Management

Oswaldo Santos[1,2,*], **Marcos Amaku**[1] **and Fernando Ferreira**[1]

1. Department of Veterinary Preventive Medicine and Animal Health, University of S ao Paulo, Brazil.
2. Education and Animal Control Technical Institute - ITEC, São Paulo, Brazil.
*Contact author: oswaldosant@gmail.com

Keywords: dog, cat, population management, population dynamics, survey

Thousands of people die from rabies and get diseased of visceral leishmaniasis each year. Also, millions of abandoned dogs and cats die each year. Because dogs and cats are involved in the transmission dynamics of those diseases (in the case of leishmaniasis mainly dogs), companion animal population management is a requirement not only to improve animal welfare but to prevent and to control zoonoses such as the mentioned above. Companion animal population management can be regarded as a set of interventions to modify demographic characteristics (e.g., population size, proportion of fertile and abandoned animals). The **capm** package facilitates to users the implementation of a workflow to collect and analyze data typically needed in companion animal population management. Users can design complex surveys and map selected sampling units; estimate population parameters; simulate population dynamics; simulate the effect of interventions; and prioritize the interventions according with their effect, using sensitivity analysis. The current stable version can be installed from CRAN (`install.packages("capm")`) and unstable version can be installed from a Github repository after loading the **devtools** package (`install_github("capm", "oswaldosantos")`). Additional information and documentation can be found in the web page for the package.

Rdocumentation.org: online documentation for all R packages

Jonathan Cornelissen[1,2], Dieter De Mesmaeker[1,*], Bram Jans[1], Albert Jorissen[1], Martijn Theuwissen[1]

1. DataCamp
2. Vrije Universiteit Brussel
*Contact author: Dieter@datacamp.com

Keywords: documentation.

Rdocumentation [1] is a tool that helps you to easily find and browse the documentation of all current and some past *R* packages on CRAN. It enables you to instantly search for functions and use advanced search on the documentation of all R packages. The fast increase in the number of R packages, makes it sometimes hard for (new) R users to find the right tools for their job. By showing the number of downloads per R package and the evolution over time, Rdocumentation.org increases transparency and improves the discovery process of R users. Most interestingly, we provide an API that allows anyone to query our database and integrate R documentation in their own project. We'd like to present a poster on the API of Rdocumentation.org and how you can contribute.

References

[1] http://www.rdocumentation.org

A land-use regression-based confidence predictor for modeling of Munich air pollution data

Olga Ivina[1*]

1. Postdoctoral researcher, Institute of Epidemiology I, Helmholtz Zentrum München
*Contact author: lyolya@gmail.com

Keywords: Air pollution, Land-use regression, Conformal predictors, Spatial modeling

This work provides valid predictions with confidence estimation for air pollution levels in Munich region, Germany. Data from the ESCAPE study have been used, and they include annual mean concentrations of traffic-related air pollutants observed at the monitoring sites together with their geographical positioning, as well as land-use covariates obtained from the geographic information system (GIS) .

This research takes as a basis a well-established statistical method for air pollution modeling, land-use regression (LUR), and transforms it into a machine learning method, creating a conformal predictor around it. This helps provide confidence to the prediction, guaranteed by the definition of a conformal predictor. Also, any conformal predictor always yields valid predictions, and very high levels of confidence can be guaranteed starting from very small-sized datasets. Different spatial covariance functions for the data can be considered in a LUR-based conformal predictor employing the “kernel trick”.

R programming language has been used to perform the task. LUR models have been fitted with the use of the **SpatioTemporal** package. *R* functions allowing to create conformal predictors that have LUR as the underlying method have been written, and they provide the possibility to implement several spatial covariance models. These functions use the framework set by the function `iidpred` from the **PredictiveRegression** package.

References

[1] Kees de Hoogh et al. (2013). Development of Land Use Regression Models for Particle Composition in Twenty Study Areas in Europe. *Environmental Science and Technology 47 (11)*, 5778–5786.

[2] Bergen S., Lindström J (2013), SpatioTemporal: An R Package for Spatio-Temporal Modelling of Air-Pollution. Comprehensive Tutorial for the Spatio-Temporal R-package. http://cran.r-project.org/web/packages/SpatioTemporal/index.html

[3] Vovk V., Nouretdinov I., Gammerman A. (2009), On-line predictive linear regression, *Annals of Statistics 37 (3)*, 1566–1590.

[4] Ivina, O., Nouretdinov, I., Gammerman, A. (2012) Valid predictions with confidence estimation in air pollution problem. *Progress in Artificial Intelligence 1 (3)*, 235-243.

Running R with 120 threads on the Intel® Xeon® E7-4870 v2

Eric Kramer[1*], William Shipman[1], Ali Torkamani[1]

1. Scripps Translation Science Institute, 3344 North Torrey Pines Court, Suite 300, La Jolla, CA 92037
*Contact author: ekramer@scripps.edu

Keywords: Parallel Processing, Machine Learning, Big Data, High Performance Computing

The increasing size of datasets and the proliferation of multicore CPUs has increased demand for parallel processing in *R*. Existing packages, such as the **parallel**, **snow** [6]**, foreach** [4] and **multicore** [8] packages, allow users to parallelize loops in *R*. Similarly, several groups have demonstrated impressive performance gains by compiling *R* with multithreaded BLAS libraries, such as Intel's Math Kernel Library [5]. We use these strategies to deploy *R* on a 60-core server, which is capable of operating 120 concurrent threads on four Intel Xeon E7-4870 v2 CPUs. Using Urbanek's benchmarking script [7], we see a 230-fold increase in performance for calculating cross products as compared to the base *R* installation, but only a 1.7-fold performance increase for sorting random numbers. We also trained tumor classifiers using methods from the **nnet** [9], **kernlab** [2], and **caret** [3] packages with data from The Cancer Genome Atlas [1]. We see a 20-fold performance gain for training support vector machines as compared to the base R installation, and a 21-fold performance gain for training neural networks with three layers. Overall, these findings suggest that using dozens of threads can result in large performance gains for matrix operations, support vector machines and neural networks in *R*.

References

[1] The Cancer Genome Atlas Network (2012). Comprehensive molecular protraits of human breast tumors. Nature 490(7418), 61-70.

[2] Alexandros Karatzoglou, Alex Smola, Kurt Hornik, Achim Zeileis (2004). kernlab - An S4 Package for Kernel Methods in R. Journal of Statistical Software 11(9), 1-20. http://www.jstatsoft.org/v11/i09/

[3] Max Kuhn (2014). Caret: Classification and Regression training. http://cran.r-project.org/web/packages/caret/index.html

[4] Revolution Analytics (2013). Foreach: foreach looping construct for R. http://cran.r-project.org/web/packages/foreach/index.html

[5] Revolution Analytics (2013). High performance R. http://www.revolutionanalytics.com/high-performance-r

[6] Luke Tierney, A. J. Rossini, Na Li, H. Sevcikova (2013). Snow: Simple Network of Workstations. http://cran.r-project.org/web/packages/snow/index.html

[7] Simon Urbanek (2008). R Benchmarking Script. http://r.research.att.com/benchmarks/R-benchmark-25.R

[8] Simon Urbanek (2011). Multicore: Parallel processing of R code on machines with multiple cores or CPUs. http://cran.r-project.org/web/packages/multicore/index.html

[9] W. N. Venables, B. D. Ripley (2002). Modern Applied Statistics with S. Fourth Edition. Springer, New York. ISBN 0-387-95457-0

Data Works: An Interactive Data Visualization Application Built with Shiny

Christian A. Gonzalez[1*], **Robert J. Youmans**[1]

1. George Mason University
*Contact author: cgonza12@gmu.edu

Keywords: Data Visualization, Web Applications, Interface Design, Big Data, Usability

The use of data and interactive data applications to inform decision making is becoming ubiquitous in many domains. However, there are few examples in the literature of how to ensure that these complex and cognitively demanding applications are usable and effective[1][2]. **Shiny**, a recently released package for the proglangR programming language, provides a flexible, easy-to-use open-source tool for developers and analysts wishing to create interactive web-based data applications.

Data Works presents a case study of the development and iterative user testing of a data visualization application built with **Shiny** for a Challenge.gov competition sponsored National Endowment for the Arts (NEA). Data Works provides the general public as well as domain-experts access to a large multi-year data set from a nationally representative survey of public arts participation, the Survey of Public Participation in the Arts. The application was recently selected by NEA as the challenge winner and will be displayed on the NEA's main website.

This work explores the relevance of usability principles to data applications and potential pitfalls developers and analysts may face. In addition, we provide novel use cases for integrating predictive modelling and geospatial information via the **googleVis** package. Furthermore, we provide recommendations for interface design aesthetics and mobile application experiences. Finally, Results from user testing and eye tracking suggest that interpretation context and responsive feedback may be critical features to ensure effective use.

References

[1] Carr, D. B. & Pickle, L. W. (2010). *Visualizing data patterns with micromaps*. CRC Press.

[2] Ward, M., Grinstein, G., & Keim, D. (2010). *Interactive data visualization: foundations, techniques, and applications*. AK Peters, Ltd.

Logic Programming in *R* using FactsRules

Neal Fultz [1,2,⋆]

1. UCLA Statistics
2. NJNM Consulting
⋆Contact author: nfultz@stat.ucla.edu

Keywords: Logic Programming, Tidy Data

Logic programming is a programming paradigm based on first order logic that is especially well suited for applications in natural language processing, pattern matching, combinatorics, artificial intelligence and social network analysis. First order logic is closely linked to relational algebra and therefore also generally useful for querying and managing tidy data.

The **FactsRules** package provides three components for logic programming in *R*: a pure-*R* implementation of unification; a formula interface for declaring rules; and wrapper functions for `data.frame` and `data.table`.

This talk will review logic programming, relational algebra and unification and explain how they are implemented in **FactsRules** with a focus on pragmatic examples.

Multi-center Clinical trials reporting with R

Scott Gillespie[1], Courtney McCracken[1], Dane Van Domelen[2], Traci Leong[2*]

1. Emory University School of Medicine
2. Emory University School of Public Health
*Contact author: tleong@emory.edu

Keywords: Sweave, Automation, Reporting

Reporting in clinical trials is a requirement from both local and funding source agencies. Regular reporting also aids the study team in evaluating enrollment rates and patient eligibility characteristics. For multi-center clinical trials, there is often a need for reports to be sent to the study team and enrolling sites to monitor each institution's contribution to enrollment. Regular reporting can become a time burden without an automated system. We will evaluate several methods of clinical trials reporting and will demonstrate the computing speed differences as well as the output for each method. We will also consider the amount of manual time required to implement each method.

Results from a critical care clinical trial [1] will be shown using methods incorporating the packages **tab** [2] and **xtable** [3]. In conjunction with the xtable package, we will use **Sweave** [4] to produce high quality summary tables in Latex. The method requiring xtable allows for the greatest flexibility, but requires that the user first summarize the data, then manually fit the output to a matrix prior to obtaining the Latex output from the **xtable** function. An alternative to this approach utilizes the **tab** package in RStudio. Here, the user simply specifies the data as either continuous or discrete, indicates any specific levels to summarize by, and runs the appropriate command. These results can be easily copied and pasted into Microsoft Word for general and quick reporting.

Both **tab** and **xtable** methods are reasonable approaches to displaying data summaries and results. The **xtable** method would be best utilized for presenting publication quality tables and output; whereas, the **tab** approach could be best used for monthly summary reports to be presented to study teams and enrolling sites in clinical trials.

References

[1] Rigby MR, Leong, T, Preissig C, et al. (2014). Introducing a protocol to control hyperglycemia in pediatric critical illness: Associations of hyperglycemia, glycemic control and hypoglycemia. In SCCM 2014, Society of Critical Care Medicine (San Francisco, CA).

[2] Van Domelen DR. (2014). tab: Functions for creating summary tables for statistical reports.. R package version 1.0. http://CRAN.R-project.org/package=tab

[3] Dahl DB. (2013). xtable: Export tables to LaTeX or HTML. R package version 1.7-1. http://CRAN.R-project.org/package=xtable

[4] Leisch F. (2002).Sweave: Dynamic Generation of Statistical Reports Using Literate Data Analysis Physica Verlag, Heidelberg. ISBN 3-7908-1517-9.

[5] R Core Team (2013). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

R graphics and Tidal Wetland Restoration

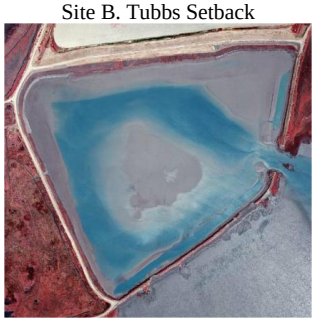
Jeanny Wang[1,2]
1. UC Berkeley - Environmental Science, Policy and Management
2. R-Ladies
Contact: jeannywm@gmail.com

Guadalcanal and Tubbs Setback are two 20-30 ha tidal wetland restorations projects on the San Pablo Bay (north San Francisco Bay) that have been monitored by the US Geological Survey (USGS) for both mitigation and habitat restoration objectives. Both wetlands were restored in part through breaching of a levee to allow tidal inundation, sedimentation and vegetation colonization to occur. *Guadalcanal* [Site A, 1] is considered to have been "engineered" to a greater extent than *Tubbs Setback* [Site B, 2], with considerable design input and channel excavation efforts undertaken before the breach. The Tubbs Setback restoration is considered to be "self-design" with the primary project intervention the single breach to a levee on San Pablo Bay (SPB). Between 2002 and 2008, these two sites were monitored for various biophysical parameters including hydrology, sedimentation rates, vegetation, bird and mammal populations.

R packages including **ggplot2** and **plot.ly** were used to visualize restoration processes in tidal wetland systems, through datasets on hydrology /tidal inundation, sedimentation, vegetation, and bird populations at two wetland restoration sites. The levees were breeched at both sites in order to restore tidal inundation and sediment transport processes. At Site A, the site was designed to specifications, excavated to pre-specified grades, and into a complex engineered channel system. At Site B, the levee was simply breeched and with no additional design or construction of bank elevations or channels. Here we compare the two sites, using defined indicators of restoration efficacy over time. Using remotely sensed imagery and relationships of monitored data, we seek to determine whether there is significant different in restoration outcomes of these representative wetland sites, and predict temporal features of other tidal wetland restoration or mitigation efforts.

Site A. Guadalcanal Village

Site B. Tubbs Setback



References

[1] Woo, I., J. Y. Takekawa and R. Gardiner (2007). Guadalcanal Tidal Marsh Restoration: 2007 Annual Report. Data Summary Report, U. S. Geological Survey, Western Ecological Research Center, San Francisco Bay Estuary Field Station, Vallejo, CA. 57 pp.

[2] Woo, Isa, Takekawa, John Y., Rowan, Aariel, Gardiner, Rachel J. And Giselle T. Block (2006). The Tubbs setback Restoration Project: 2006 Final Report. Administrative Report, US Geological Survey, Western Ecological Research Center, San Francisco bay Estuary Field Station, Vallejo, CA. 70 pp.

~ Monitoring data was provided by the U.S. Geological Survey, Western Ecological Research Center, San Francisco Bay Estuary Field Station, Vallejo, CA, 2013.

EMMAgeo – end-member modelling analysis of grain-size data

Michael Dietze[1*] , Elisabeth Dietze[2]

1. GFZ German Research Centre for Geosciences
Section 5.1 Geomorphology
Telegrafenberg, F427
14473 Potsdam, Germany

2. GFZ German Research Centre for Geosciences
Section 5.2 Climate Dynamics and Landscape Evolution
Telegrafenberg, F451
14473 Potsdam, Germany

*Contact author: mdietze@gfz-potsdam.de

Keywords: EMMA, end-member modelling, grain-size data, Earth surface process, eigenspace

Earth surface processes decisively shape our planet. Identifying and quantifying their rates and impact is one of the major challenges of current geoscientific research. The changing grain-size distributions of deposits (e.g. lake sediments, ocean floor deposits, landforms accumulated by wind-blown sediment) reflect their formation by distinct sediment transport processes and provide a fruitful possibility to unravel the contribution of such processes. However, interpreting sediment transport processes from grain-size data in terrestrial archives runs into problems when source- and process-related grain-size distributions become mixed during deposition. A powerful approach to overcome this ambiguity is to statistically "unmix" the samples. Typical algorithms use eigenspace decomposition and techniques of dimension reduction.

This contribution presents the package **EMMAgeo** (Dietze & Dietze, 2013) for the free statistical software *R*. It bases on an end-member modelling algorithm originally presented as *Matlab*-script (Dietze et al., 2012) and contains several extensions and added functionality. The package comprises 14 functions. It supports simple modelling of grain-size end-member loadings and scores (eigenspace extraction, factor rotation, data scaling, non-negative least squares solving) along with several measures of model quality. It also provides pre-processing tools (grain-size scale conversions, weight factor limit inference, determination of minimum, optimum and maximum number of meaningful end-members) and allows to model data sets with user-defined end-member loadings. **EMMAgeo** also supports uncertainty estimation from a series of plausible model runs and determination of robust end-members.

The contribution depicts important package functions, thereby illustrating how large data sets of artificial and natural grain-size samples from different depositional environments can be analysed to infer and quantify process-related proxies (Dietze et al., 2014) that can be used to better reconstruct environmental conditions in the past (and to learn for future environmental change).

References

Dietze, M., E. Dietze (2013). EMMAgeo: End-member modelling algorithm and supporting functions for grain-size analysis. R package version 0.9.1. http://CRAN.R-project.org/package=EMMAgeo.

Dietze, E., K. Hartmann, B. Diekmann, J. IJmker, F. Lehmkuhl, S. Opitz, G. Stauch, B. Wünnemann, A. Borchers (2012). An end-member algorithm for deciphering modern detrital processes from lake sediments of Lake Donggi Cona, NE Tibetan Plateau, China. *Sedimentary Geology*, 243-244, 169-180.

Dietze, E., F. Maussion, M. Ahlborn, B. Diekmann, K. Hartmann, K. Henkel, T. Kasper, G. Lockot, S. Opitz, T. Haberzettl (2014). Sediment transport processes across the Tibetan Plateau inferred from robust grain-size end members in lake sediments. *Climate of the Past*, 10, 1, 91-106.

Statistics without Numbers: Using Data Visualization to Quantify Trends for Cycling Safety

Jacob Quartuccio[1*] , Simone Erchov[1], Christian Gonzalez[1], David Cades[2]

1. George Mason University
2. Exponent Failure Analysis Associates
*Contact author: jquartuc@gmu.edu

Keywords: visualization, surface transportation, geospatial data

Big data sets can be cumbersome and difficult to understand. User-centered and interactive graphical displays help communicate messages from large and complex data as well as provide a new method to identify data trends outside of tabular or statistical analysis. Not all statistics need to be summarized numerically, sometimes a visualization can convey uncertainty with parameters. Researchers can use data visuals to not only develop questions but also answer them through visual exploration that previously proved difficult. This approach can be especially relevant to the field of surface transportation research where complex plots can incorporate both temporal and geospatial data in an easy-to-digest format. These plots may ease communication of potential issues and solutions between various stakeholders such as policy makers, liability companies, and people using transplantation systems. As a proof of concept, the visualization presented here demonstrates how data showing bike-sharing in 2013 in Chicago and past bicycle collision incidents can meaningfully merge to produce graphical displays that readily identify and communicate potential infrastructure problems for safety. The city of Chicago seemed especially ripe for a safety analysis since the city has actively targeted bicycle safety challenges as the number of cyclists on the road rises [1]. Bicycling routes in the figure were generated via a Bayesian model by estimating the posterior probability that a cyclist traveled between two stations. The model began with an uninformative prior, assuming that every end station had an equal probability of arrival from any other station, and updated with bikeshare trip data via a Bernoulli function in the **LearnBayes** package 2.12. A map of Chicago was acquired from Google to provide a base map for the plots with the **ggmap** v2.3 package. Accident data was overlaid via a heatmap to show areas that have been traditionally prone to cycling accidents; areas in red represent the highest density of collisions, yellow represents a reduction, and cream portions of the image represent lowest levels of collisions. The visualization not only conveys current Bikeshare traffic patterns, but also encourages stakeholders' further exploration of areas that may need further infrastructure development.

References

[1] Chicago Department of Transportation (2013). Chicago Streets for Cycling Plan 2020. Policy Report. Retrieved from http://www.cityofchicago.org/content/dam/city/depts/cdot/bike/general/ChicagoStreetsforCycling2020.pdf.

`plsRglm`, PLS generalized linear models for R

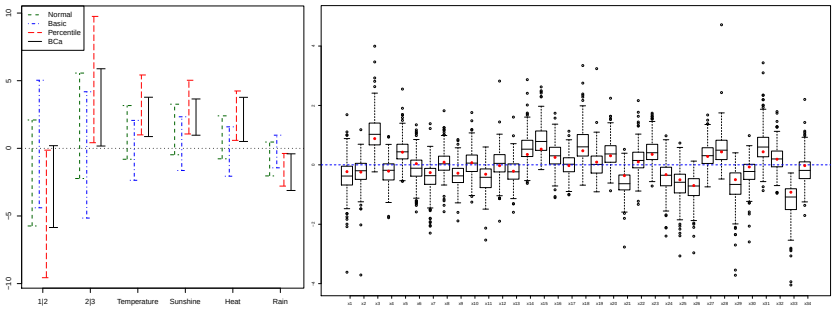
Frédéric Bertrand[1,2,⋆], **Jérémy Magnanensi**[1,2,3], **Nicolas Meyer**[1,3], **Myriam Bertrand**[1,2]

1. Université of Strasbourg
2. Centre national de la recherche scientifique
3. Institut national de la santé et de la recherche médicale
⋆Contact author: fbertran@math.unistra.fr

Keywords: Partial least squares regression, generalized linear models, bootstrap, high dimensional data, R software package

There are mainly two aims for the **plsRglm** library written by the authors for the *R* language. The extension of PLS regression to generalized linear models, and for instance to logistic regression models [2], and the need to provide tools to PLS users to deal with incomplete datasets using cross-validation.

These models were successfully applied to datasets of various kindWe will provide six examples of use: PLSR model fitted to a Mixture design dataset (`Cornell`,[3]), PLSR-multinomial logistic model fitted to a Bordeaux wine quality dataset (`bordeaux`, [2]) and to a Chemotaxonomy dataset (`hyptis` from library **chemometrics**, [5]), PLSR-binary logistic model fitted to a Microarray Colon Cancer dataset (`ColonCA`, [1]), to a Chemometrics dataset (`phenyl` dataset from library **chemometrics**,[5]) and to allelotyping data (`aze_compl`, [4] ; Figure 1 features bootstrap distribution of the coefficients of the predictors and Figure 2 balanced bootstrap confidence intervals).



References

[1] Alon, U., N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and L. A. J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissue probed by oligonucleotide arrays. *Proc. Natl. Acad. Sci. USA 96*, 6745–6750.

[2] Bastien, P., V. Esposito Vinzi, and M. Tenenhaus (2005). Pls generalised linear regression. *Computational Statistics & Data Analysis 48*, 17–46.

[3] Kettaneh-Wold, N. (1992). Analysis of mixture data with partial least squares. *Chemometrics & Intelligent Laboratory Systems 14*, 57–69.

[4] Meyer, N., M. Maumy-Bertrand, and F. Bertrand (2010). Comparaison de variantes de régressions logistiques pls et de régression pls sur variables qualitatives : application aux données d'allélotypage. *Chemometrics & Intelligent Laboratory Systems 151*, 1–18.

[5] Varmuza, K. and P. Filzmoser (2009). *Introduction to Multivariate Statistical Analysis in Chemometrics*. Boca Raton: CRC Press.

Visually Analyzing and Running Multilevel Data in R and BUGS

Jimmy Wong[1,*], Dr. Beth Chance[1]

1. California Polytechnic State University, San Luis Obispo
*Contact author: jwong100@calpoly.edu

Keywords: multilevel models, bugs, lme4, ggplot2, animation

Education data collected for research on students' performance and attitudes in a randomization-based introductory curriculum were analyzed using a multilevel modelling approach. Our data consisted of two sets of observational units that were students and instructors, where we had multiple students nested in each instructor. The intention of applying a multilevel modelling approach originated from the nature of the data, our desire to use variables collected on students and instructors in the same regression model, and our intention to account for any instructor-to-instructor variability in the response variable while running a regression model.

R was the primary statistical software used in running our multilevel models, producing graphical displays, and generating animations/gifs. During the first stages of our analysis, the main packages that we used in the order of importance to our study were **lme4**, **ggplot2**, **animation,** and **gridExtra**. Prior to running the actual multilevel models, we conducted exploratory analysis on the data by using functions from **ggplot2, gridExtra,** and **animation** packages**.** We primarily created multi-dimensional graphs to observe how variables associated and interacted with each other. Examples of such graphs were facetted boxplots conditional on two variables, scatterplots of students and instructors' variables weighted by class size with regression lines imposed, spaghetti plots color coded by an instructor-level variable, mosaic plots, and overlaid marginal and conditional histograms. The codes required to create these graphs extended to more than just using `ggplot` and `geom_point`, for example. There were many tweaks made to our graphs to have them appear more intriguing and original. After generating our graphs, **gridExtra** assisted us in combining multiple graphs to a single cohesive graph when we wanted to compare the idea of marginal versus conditional. These combined graphs were then saved to jpeg files using `ggsave` so we can access them directly in the future without having to run any code in *R*. Shifting our focus from graphs to animations, the **animation** package assisted us in further presenting some of our ideas such as showing the process of creating spaghetti plots and the considerable variability in the many variables among instructors in a more interesting, but yet, educational way by compiling individual graphs to unified gif files with the functions of `saveGIF` and `saveHTML`.

After our exploratory analysis was done, **lme4** assisted us in running multilevel linear and logistic models. The functions of `lmer` and `glmer` in **lme4** were used for quantitative and categorical response variables, respectively. The regression outputs provided us estimates of the coefficients and the variances components as well as giving us information on the significance of the predictor variables. However, we decided to obtain another set of estimates with a Bayesian approach. This idea lead us to the two *R* packages of **arm** and **R2WinBUGS.** We ran our multilevel models in *BUGS* by communicating from *R* to *BUGS* mainly with the function `bugs`. Therefore, our final results were a combination of frequentist and Bayesian methods and comparisons were made between the estimates from these two methods.

In summary, we used a multilevel modelling approach to analyze education data that had variables measured on both students and instructors. Our exploratory analysis consisted of generating graphs and animations to visually analyze the associations of different variables from our data set by using functions from the *R* packages of **ggplot2, gridExtra,** and **animation**. The assistance of **lme4, arm,** and **R2WinBUGS** in running our multilevel models allowed us to combine results from a frequentist and Bayesian method. As future work, we may consider compiling an R package that generalizes our code for graphs to allow users to input their variables of interest and also creating a Shiny application that encompasses as much stages of our analysis as possible to allow users to modify input parameters and to educate multilevel modelling to students or anyone who may be interested in multilevel models.

`plsRcox`, Cox-Models in a high dimensional setting in R

Frédéric Bertrand[1,2,*], **Philippe Bastien**[3], **Nicolas Meyer**[1,4], **Myriam Bertrand**[1,2]

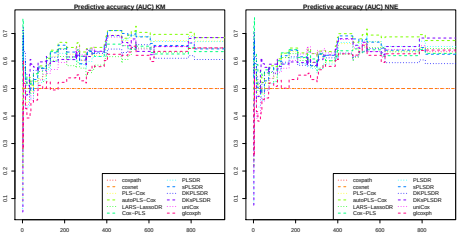
1. Université of Strasbourg
2. Centre national de la recherche scientifique
3. L'Oréal Recherche et Développement
4. Institut national de la santé et de la recherche médicale

*Contact author: fbertran@math.unistra.fr

Keywords: Partial least squares regression, Cox models, survival analysis, high dimensional data, R

A vast literature from the last decade is devoted to relating gene profiles and subject survival or time to cancer recurrence. Biomarker discovery from high-dimensional data, such as transcriptomic or SNP profiles, is a major challenge in the search for more precise diagnoses. The proportional hazard regression model suggested by Cox, [1], to study the relationship between the time to event and a set of covariates in the presence of censoring is the most commonly used model for the analysis of survival data. However, like multivariate regression, it supposes that more observations than variables, complete data, and not strongly correlated variables are available. In practice when dealing with high-dimensional data, these constraints are crippling. Collinearity gives rise to issues of overfitting and model mis-identification. Variable selection can improve the estimation accuracy by effectively identifying the subset of relevant predictors and enhance the model interpretability with parsimonious representation. In order to deal with both collinearity and variable selection issues, many methods based on Lasso penalized Cox proportional hazards have been proposed since the reference paper of Tibshirani, [3]. Regularization could also be performed using dimension reduction as is the case with PLS regression. We propose two original algorithms named sPLSDR and its non linear kernel counterpart DKsPLSDR, by using sparse PLS regression (sPLS) based on deviance residuals. We compared their predicting performance with state of the art algorithms based on reference benchmark datasets.

As sPLSDR and DKsPLSDR compare favorably with other methods in their computational time, prediction and selectivity, as indicated by results based on benchmark datasets, see Figure below, we view them as a useful addition to the toolbox of estimation and prediction methods for the widely used Cox's model in the high-dimensional and low sample size settings.



Model prediction accuracy comparision using iAUC, [2].

References

[1] Cox, D.R. (1972) Regression models and life tables. *Journal of the Royal Statistical Society B*, **74**, 187–220.

[2] Heagerty, P.J., and Zheng, Y. (2005) Survival Model Predictive Accuracy and ROC Curves. *Biometrics*, **61**(1), 92-=105.

[3] Tibshirani, R. (1997). The lasso method for variable selection in the Cox model. *Statistics in Medicine*, **16**, 385–395.

Cascade: a *R*-package to study, predict and simulate the diffusion of a signal through a temporal gene network.

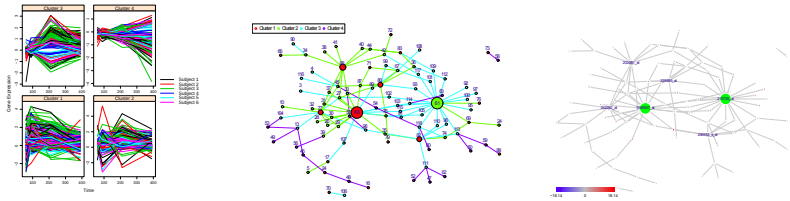
Nicolas Jung[1,2,3], **Frédéric Bertrand**[1,2,*], **Seiamak Bahram**[1,3], **Laurent Valla**[1,3], **Myriam Bertrand**[1,2]

1. Université of Strasbourg
2. Centre national de la recherche scientifique
3. Institut national de la santé et de la recherche médicale
*Contact author: fbertran@math.unistra.fr

Keywords: Partial least squares regression, generalized linear models, bootstrap, high dimensional data, R software package

Temporal gene interactions, in response to environmental stress, is a complex system that can be efficiently described using gene regulatory networks (GRN): a GRN allows to highlight the more influential genes and to spot some targets for biological intervention experiments. Despite that many reverse-engineering tools have been designed, the **Cascade** package is an integrated solution adding several new and original key features such as the ability to predict changes in gene expressions after a biological perturbation in the network and graphical outputs that allow monitoring the spread of a signal through the network.

Since the emergence of high-throughput technologies, many tools have been developed to learn gene expression profiles and reverse-engineer their underlying GRN [1,2]. These tools are either based on static co-expression methods or, if the biological phenomenon shows any temporality, time dependent methods. While the former relies on the assumption that co-expressed genes share some biological characteristics, the latter infers a directed network with temporal dependencies. In this last case, another important distinction should be made between exogenous stress (e.g., growth response) and endogenous phenomenon (e.g., cell cycle) [3,4]. This leads to different network topologies: in exogenous stress, networks' topologies seem to have larger hubs and shorter paths through temporal dependent transcriptional waves [3]. This results in a quick response to environmental modifications [3]. The **Cascade** package is designed to model such "cascade networks" taking advantage of the assignment of genes to temporal clusters which are then used to enforce temporal causality in the network.



The three steps : gene selection with assignment to a time cluster (left), reverse-engineering of the network (center), predicted perturbations in the network after gene expression modulation at a early time (right).

[1] Bar-Joseph, Z. *et al.* (2012). Studying and modelling dynamic biological processes using time-series gene expression data. Nature Reviews Genetics, **13**(8), 552–564.

[2] Bansal, M., *et al.* (2007). Hecker, M. *et al.* (2009). Gene regulatory network inference: data integration in dynamic models-a review. *Biosystems*, **96**, 86–103.

[3] Luscombe, N. M., *et al.* (2004). Genomic analysis of regulatory network dynamics reveals large topological changes. *Nature*, **431**(7006), 308–312.

[4] Zhu, X., Gerstein, M., and Snyder, M. (2007). Getting connected: analysis and principles of biological networks. *Genes and development*, **21**(9), 1010–1024.

Teaching data analysis in R through the lens of reproducibility

Mine Çetinkaya-Rundel[1*]

1. Duke University, Department of Statistical Science *Contact author: mine@stat.duke.edu

Keywords: teaching, reproducibility, RMarkdown, knitr

The issue of reproducibility often comes up in the context of published research and the need to accompany such research with the complete data and analyses, including software/code. As statistics educators who teach data analysis, we should be instilling best practices in students before they set out to do research. We advocate for teaching data analysis and programming in *R* using **R Markdown,** even to students who have no previous programming experience. In this talk we will discuss benefits of this approach, not only with respect to creating opportunities for discussing the importance of reproducible research, but also for learning syntax, avoiding common novice pitfalls, and organizing and unifying output and write-ups. We will present examples from data analysis labs using this approach as well as experiential and statistical evidence that R Markdown can be used effectively in introductory statistics courses.

Distributed Matrix Exponentiation in R

Drew Schmidt[1,*]

1. National Institute for Computational Sciences, University of Tennessee
*Contact author: schmidt@math.utk.edu

Keywords: HPC, linear algebra, ScaLAPACK, MPI, parallel programming

Matrix exponentiation is an important matrix function which is useful in a wide variety of domains and applications. Formally, matrix exponentiation is an easily understood power series; but efficient, numerically stable algorithms for computing this function have been debated for over 30 years. There are several serial implementations of the matrix exponential available to *R*, including those found in the **Matrix** and **rexpokit** packages. We introduce a relatively new algorithm for computing the matrix exponential due to Al-Mohy and Higham, implemented in the **pbdDMAT** package. Our implementation includes both serial and distributed versions of this algorithm, the latter of which fully integrates with the pbdR framework for high performance computing with *R*. Finally, we will conclude by demonstrating the scalability of the implementation with benchmarks on University of Tennessee supercomputing resources.

References

[1] Al-Mohy, A. H. and N. J. Higham (2009). A New Scaling and Squaring Algorithm for the Matrix Exponential.

[2] Blackford, L. S., J. Choi, A. Cleary, E. D'Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker, and R. C. Whaley (1997). *ScaLAPACK Users' Guide*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

[3] Ostrouchov, G., W.-C. Chen, D. Schmidt, and P. Patel. Programming with Big Data in R,.

[4] Schmidt, D., W.-C. Chen, G. Ostrouchov, and P. Patel (2012). pbdDMAT: Distributed Matrix Algebra Computation. R Package, URL http://cran.r-project.org/package=pbdDMAT.

[5] Sidje, R. B. (1998, March). Expokit: A Software Package for Computing Matrix Exponentials. *ACM Trans. Math. Softw. 24*(1), 130–156.

QUANTITATIVE TOOLS FOR MODELING COARSE WOODY DEBRIS DYNAMICS

Md. Abdul Halim and Sean. C. Thomas

Faculty of Forestry, University of Toronto, 33 Willcocks St., Toronto, M5S3B3 Ontario, Canada

Coarse woody debris (CWD) is the standing or fallen dead trees and the remains of large or small branches on the forest floor usually larger than 10cm in diameter. CWD forms major structural features within a forested ecosystem with many vital ecological functions such as habitat for organisms (including endangered and threatened), in energy flow, nutrient cycling and hydrological processes. It is also a good source of soil nitrogen and caps C in the soil. To get these benefits it is very important to understand the ecology CWD. Additionally, due to the rising concern of its steep decline associated with intensive forestry, the need for a suitable forest management approach has become urgent. Modeling the transition of CWD in a forest ecosystem is very important for taking rational decision from biodiversity and economic perspectives. Considering this urgency, we are developing an R package (proposed name 'CWD') which contains five major functions those are useful for modeling CWD transition dynamics. For e.g. the function *vol.cul* can estimate stand volume/ha and cull/ha (volume of trees/ha that have no current or potential commercial value), *vol.cwd.inp* can control the flow of cull/ha used as input/ha for CWD in five years interval up to the rotation age in a stand, *mod.trans.mat* can model the transition rates of CWD among different decay classes (usually five) at five years interval up to the rotation age, *age.vol.asymp* determines at which age the total amount of CWD in a stand reaches asymptote, and *ini.vol.year* can calculate required initial CWD volume/ha to achieve a target CWD (volume/ha) in a given rotation length. With a reliable transition matrix as an input, these functions can be used in finding cost-efficient options for increasing CWD volume in managed forest ecosystems with lower disturbances. By using snapshot-sampling methods together with these functions, it is quite possible to avoid time-consuming long-term studies in different climatic conditions, forest types, and species (plant) groups.

Developing shiny applications for the classroom

Sandra D Griffith[1*], **Michael E Lerner**[2]

1. Department of Quantitative Health Sciences, Cleveland Clinic
2. Beachwood High School
*Contact author: griffis5@ccf.org

Keywords: Teaching, Shiny, Interactive, Visualization, Education

The **shiny** package (RStudio, 2013) provides a flexible framework for developing web applications using *R* and holds potential for use in teaching in a variety of settings. **shiny** allows *R* users to develop interactive applications customized to the specific teaching application and data source without specialized knowledge of web development. Its web-based deployment allows these programs to be accessible on any devices with web access, not requiring software installation. By minimizing the gap between teacher and software developer, **shiny** allows *R* users to directly bring statistical concepts to students in cases where a deeper understanding of programming or statistical software is not feasible or desired.

We illustrate its use in teaching with case studies, including an application for data linearization developed for the physics classroom (Griffith & Lerner, 2014) using the **shiny**, **shinyIncubator**, and **ggplot2** packages. This was prompted by hands-on lab assignments requiring students to collect and enter experimental data, graphically display both raw and model-based visualizations of the data in conjunction with a variety of possible transformations, and grasp concepts of uncertainty. Although many general mathematics and statistical tools exist with similar functionality, none were tailored to the specific audience and available on a web-based platform. We quickly developed a **shiny** application to achieve these needs and piloted it in the classroom. Students found the application intuitive and easy to use. Based on observation and feedback, we were able to rapidly make changes to the application for use in subsequent lab sessions.

References

RStudio and Inc. (2013). shiny: Web Application Framework for R. R package version 0.8.0. http://CRAN.R-project.org/package=shiny

Griffith SD and Lerner M (2014). Physics linearization web app (available at http://spark.rstudio.com/sgriffith/lerner/).

seq2R: Analyzing compositional asymmetries in DNA

Nora M. Villanueva[1⋆], **Marta Sestelo**[2], **Javier Roca-Pardiñas**[1]

1. Department of Statistics and Operation Research, University of Vigo, Spain
2. Department of Mathematics, University Autonomous of Barcelona, Spain
⋆Contact author: nmvillanueva@uvigo.es

Keywords: DNA sequence, change points, nonparametric models, testing procedure, first derivatives.

Understanding the mutational processes that shape DNA sequences is fundamental to better comprehend how genomes evolve. These mutations do not equally affect both complementary strands of DNA when these mutations are associated with molecular processes that are also asymmetric affecting differently both strands (e.g. transcription, DNA repair or replication; Touchon et al., 2005). Over the years, different compositional analyzes were carried out to detect the location of compositional changes points in mitochondrial genomes (Grigoriev, 1998; Reyes et al., 1998, Faith and Pollock, 2003). Identifying these change points in a statistical framework can be a challenging task. Numerous methodological approaches have been developed to analyze change points models, i.e. Bayesian estimation, maximum-likelihood estimation, least squares regression or nonparametric regression. We implement in a user-friendly and simply *R* package, **seq2R**, a methodology that identifies and locates compositional change points in DNA sequences by fitting nonparametric regression models. Our procedure is based on two steps. Firstly, we propose an initial approach of the regions with possible change points in which the first derivative is different to zero. The regression curve and its first derivative are estimated by local linear kernel smoothers (Fan and Gijbels, 1996; Wand and Jones, 1995) and the bandwidths are automatically selected using cross-validation techniques (Golub et al., 1979). Secondly, we asses if there are true change points in those regions, specifically, where and how many they are, with a testing procedure.

References

Faith, J. J., Pollock, D. D., (2003). Likelihood analysis of asymmetrical mutation bias gradients in vertebrate mitochondrial genomes. *Genetics* **165**, pp.735–745.

Fan, J and Gijbels, I (1996). *Local polynomial modelling and its applications. Monographs on statistics and applied probability series 66*. Chapman & Hall.

Golub, G. H., Heath, M. and Wahba, G. (1979). Generealized cross-validation as a method for choosing a good ridge parameter. *Technometrics* **21**, pp. 215–56.

Grigoriev, A., (1998). Analyzing genomes with cumulative skew diagrams. *Nucleic Acids Research* **26** (10), pp. 2286–2290.

Reyes, A., Gissi, C., Pesole, G., Saccone, C., (1998). Asymmetrical directional mutation pressure in the mitochondrial genome of mammals. *Molecular Biology and Evolution* **15** (8), pp. 957–66.

Touchon, M., Rocha, E. P., (2008). From GC skews to wavelets: A gentle guide to the analysis of compositional asymmetries in genomic data. *Biochimie* **90** (4), pp. 648–659.

Wand, M. P. and Jones, M. C. (1995). *Kernel smoothing.*. Chapman & Hall.

Detecting critical points of regression curves. An application to the management of aquatic living resources

Marta Sestelo[1*], **Nora M. Villanueva**[2], **Javier Roca-Pardiñas**[2]

1. Department of Mathematics, Autonomous University of Barcelona, Spain.
2. Department of Statistics and Operation Research, University of Vigo, Spain.

[*]Contact author: sestelo@mat.uab.cat

Keywords: factor-by-curve interactions, kernel, bootstrap, pollicipes, size of capture

In many biological studies, it is necessary to estimate the relationship between two specific variables and, in most cases, to determine how this relationship is influenced by one factor it becomes the main problem analysis. Here, we study the length-weight relationship of the barnacle *Pollicipes pollicipes* on the Atlantic coast of Galicia (NW Spain) taking into account the factor year. Growth curves and their derivatives were estimated using local linear kernel smoothers. Confidence intervals were used to draw inference from the derivates curves and testing procedures were applied to asses the true effect of the factor. These inference methods are based on the use of bootstrap techniques. Additionally, a method for the establishment of an ideal minimum size of capture of this species that would ensure a high commercial yield was developed. All computations were performed in *R* using the graphical and inferential tools from the package **NPRegfast**.

The agridat package is growing

Kevin Wright[1]

1. DuPont Pioneer
*Contact author: kw.stat@gmail.com

Keywords: Data, Graphics, Mixed models

The agridat package is an extensive collection of data sets that have been previously published in books and journals, primarily from agricultural experiments. A sample of datasets in the package are presented graphically with interpretive comments.

Better Data Quality In Clinical Trials

Daniel Dekic[1,2*]

1. Clinical Trials Managment GmbH
2. FH Campus Wien
*Contact author: d.dekic@clinicaltrials.at

Keywords: clinical trial, allergy, reporting, data quality

The VCC Database contains data of over 70 studies from the last 14 years. The VCC Studycenter specializes in testing different allergens. The Provocation Chamber enables monitoring allergic reactions and potential treatments under realistic and reproducible conditions. The data is divided into subjective data collected by the test participants themselves and machine data from rhinomanometric as well as spirometric equipment.

This poster introduces a small reporting tool utilizing reporting results with the **knitr** [5] package and visualizing results using the **ggplot2** [4] package. The reporting tool provides information about the completness and consistency of the study data along side range checks and a preliminary test on carryover effects. These help to improve the quality of the conducted studies.

In a second step the data is used to evaluate the currently used *total nasal symptom score (TNSS)* defined as the sum of scores for sneezing, congestion nasal itching and rhinorhea as an effective screening method.

References

[1] Michael Benninger, Judith R.Farrar, M. B. B. C. B. F. J. K. B. M. W. S. and M. Kaliner (2010). Evaluating approved medications to treat allergic rhinitis in the united states: an evidence-based review of efficacy for nasal symptoms by class. *Ann Allergy Asthma Immunol. 104*, 13–29.

[2] P. Stübner, R. Z. and F. Horak (2004). A direct comparison of the efficacy of antihistamines in sar and par: randomised, placebo-controlled studies with levocetirizine and loratadine using an environmental exposure unit - the vienna challenge chamber (vcc). *Curr Med Res Opin 20(6)*, 891–902.

[3] R Core Team (2013). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

[4] Wickham, H. (2009). *ggplot2: elegant graphics for data analysis*. Springer New York.

[5] Xie, Y. (2013). *Dynamic Documents with R and knitr*. Chapman and Hall CRC.

R Work Journal

Alex Zolotovitski

Medio Inc.
alex@zolot.us

Keywords: R projects, literate programming

R code for a large project often contains many tasks amd may have 500-3000 lines of code that make difficult navigation through the code. Some tasks in a project could have long execution time that makes difficult also usual literate programming methods (as `sweave, knitr`).

We wrote function `Code2RWorkJournal()` that

1. Transforms .R file into self-documented .html file, containing all R code with output pics, headers, table of contents and gallery.

2. The titles in body and contents are clickable to navigate from contents to body and back.

3. The pics are clickable to resize.

4. The html file has partly R syntax highlighted. It is possible to do the full R syntax highlighting in resulting html, but the result file becomes almost twice heavier.

5. Parts of the result html file could be folded.

6. If in a browser you "select all", copy and paste from browser to a text editor, you get the pure original R file.

7. If we modify .R code, recreate .html is fast.

The function is available at [1] .

References

[1] Alex Zolotovitski (2014). R Work Journal, http://github.com/alexzolot/RWorkJournal

Resample package

Tim Hesterberg[1,*]

1. Google
*Contact author: timhesterberg@gmail.com

Keywords: Bootstrap, Permutation Test, Confidence Interval

The **resample** packages makes the most common resampling applications easy: one and two sample bootstrap and permutation tests. For example,

- `bootstrap(x, mean) # a vector`
- `bootstrap(data, mean(x, trim = .25)) # variable in a data frame`
- `permutationTest2(data1, mean(x), data2 = data2)`
- `permutationTest2(data, mean(x), treatment = arm)`

There are methods for plotting and confidence intervals.

I'll also talk about what is wrong with common bootstrap confidence intervals, remedies, and why you should use the package instead of programming yourself from scratch.

Exploring Different Options for Interactive Spatial Data Visualization in *R*: Case Studies based on Crime Data in UK

Jo-fai Chow[1,2,3,*]

1. University of Exeter, College of Engineering, Mathematics and Physical Sciences, Exeter, United Kingdom
2. XP Solutions, Newbury, United Kingdom
3. STREAM Industrial Doctorate Centre for the Water Sector, Cranfield, United Kingdom
*Contact author: jofai.chow@gmail.com

Keywords: Spatial visualization, case studies, open crime data, web application, interactive maps

The maturality and extensive graphical abilities of *R* and its packages make *R* an excellent choice for professional data visualisation. This talk focuses on interactive spatial visualization and illustrates two different approaches with case studies based on open crime data in UK (Home Office, 2014).

Previous work has shown that it is possible to combine the functionality in packages **ggmap**, **ggplot2**, **shiny** and **shinyapps** for crime data visualization in the form of a web application named 'CrimeMap' (Chow, 2013). The web application is user-friendly and highly customizable. It allows users to create and customize spatial visualization in a few clicks without prior knowledge in *R* (figure 1). Moreover, **shiny** automatcially adjusts the best application layout for desktop computers, tablets and smartphones.

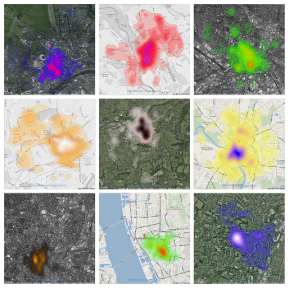


Figure 1 : 'CrimeMap' example.

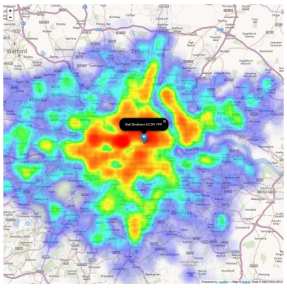


Figure 2 : **rCrimemap** example.

Following the release of **rMaps** (Vaidyanathan, 2014), Chow built upon the original 'CrimeMap' and created a new package **rCrimemap** (Chow, 2014). Leveraging the power of *JavaScript* mapping libraries such as 'leaflet' via **rMaps**, **rCrimemap** allows users to create an interactive crime map in *R* with intuitive map controls using only one line of code. Both zooming and navigation are similar to what ones would expect from using a typical digital map (see figure 2 above).

The availability of these packages means *R* developers can now easily overlay both graphcial and numerical results from complex statistical analysis with maps to create professional and insightful spatial visualization. This is particularly useful for effective communication and decision making.

References

Chow, J. (2013). Blend it like a Bayesian: Introducing CrimeMap - A Web App Powered by ShinyApps, http://bit.ly/bib_crimemap_blog2.

Chow, J. (2014). Package **rCrimemap** on GitHub, http://bit.ly/rCrimemap.

Home Office (2014). Open Data about Crime and Policing in England, Wales and Northern Ireland, http://data.police.uk.

Vaidyanathan, R. (2014). Package **rMaps** on GitHub, https://github.com/ramnathv/rMaps.

Package ATPR for Statistical Analyses of Men's Professional Tennis

Stephanie Kovalchik[1,*]

1. RAND Corporation
*Contact author: skovalch@rand.org

Keywords: XML parsing, Web scraping, Sports, Tennis,

Since the publication of *Moneyball*, the use of statistical analysis to address challenging problems in sports has been more popular than ever. At the same time, professional sports organizations are collecting and publicizing more data about athletic events than ever. Although these data should ostensibly support quantitative analysis of sports, their extraction is frequently a barrier because the data needed for a complete analysis are often embedded within a web markup language and spread across multiple web pages. In this talk, I present a package ATPR that makes numerous data about the Association of Tennis Professionals (ATP) World Tour readily accessible in the R environment. The package consists of a collection of specialized `HTML`/`XML` parsing tools that enable users to extract data about tournament results, match statistics, and player rankings for more than 20 years of singles play on Tour without the need of manual web scraping. In this presentation, I will provide an overview of these tools and the specific tennis data they can be used to gather. It is hoped that the availability of ATPR will encourage more statistical research on tennis and also serve as a useful resource for teachers of statistics.

The choroplethr package

Ari Lamstein[1*]

1. Senior Software Engineer, Trulia Inc.
*Contact author: arilamstein@gmail.com

Keywords: choropleth, map, data visualization, census data

Choropleth maps are maps which a) show boundaries and b) color each region according to a certain metric. The most common choropleth in the US is the presidential election map, which colors states according to which presidential candidate they voted for. In general, choropleth maps are useful ways to understand regional patterns in spatial data.

Despite the utility of these maps, *R* has lacked a consistent interface for creating choropleth maps. In the *R Graphics Cookbook* [1], Winston Chang explains how to create a state choropleth map in the popular **ggplot2** graphing library. His method requires several lines of code and is a different technique than that required for creating a choropleth of US Counties. And that, in turn, is a different technique than that required for creating a map of US ZIP codes. **choroplethr** provides a consistent, one-line, interface to create maps at these three different levels of detail. Currently **choroplethr** renders ZIP level maps as scatterplots; technically they are no longer choropleths because they do not show geographic boundaries. A discussion of this design decision is provided.

choroplethr also provides native support for accessing and viewing data from the US Census Bureau via the **acs** package. This allows **choroplethr** to create hundreds of thousands of choropleths of modern demographic data for the US, at multiple levels of detail (state, county and ZIP), with minimal effort on the part of the user.

Once choropleths become easy to create new discussions arise. A thorough discussion of the impact of level of detail and scale type is provided.

References

[1] Chang, Winston. R Graphics Cookbook. Sebastopol: O'Reilly, 2012. Print.

Rcircle: an R package for Integrating and Visualizing multiple "–omics" data for Knowledge Discovery

Xing Li [1*], Almudena Martinez-Fernandez[2], Terry Therneau[1], Jean-Pierre Kocher[1], Timothy J. Nelson[2]

1. Division of Biomedical Statistics and Informatics, Department of Health Sciences Research, 200 First Street, S.W., Rochester, MN 55906, USA.
2. Department of General Medicine, Center of Regenerative Medicine, Transplant Center, 200 First Street, S.W., Rochester, MN 55906, USA.

*Contact author: Li.Xing@mayo.edu

Keywords: R package, Rcircle, visualization, genome, transcriptome

Biomedical science has entered the big data era and biologists have access to an overwhelming abundance of data due to the rapid advance of high-throughput technology in sequencing and microarray[1]. The tremendous volume and high dimensions pose an unprecedented challenge on data visualization and integration for efficient data exploration and effective scientific communication. Herein, we developed an R package, **Rcircle**, based on **grid** package to integrate and visualize interactome, time-course transcriptome, disease information, disease-affected pathways or networks to facilitate knowledge discovery[2]. Starting with a curated list of congenital heart disease (CHD) genes, we identified their top 10 partners for each CHD gene and built a network for both disease genes and their partners. Pathway analysis is performed on the entire gene list. The R package visualized the gene network in the inner circle with line width representing the interaction confidence. Hub genes in the network were represented by the size of bubbles circling the genes. Transcription profile, disease information, and pathways are shown in the outer layers linked directly to the genes in the network. By integrating different types of information together, we discovered disease hub genes and established cirtical pathways turned on at different stages of cardiogenesis. Furthermore, the **Rcicle** package is able to reveal vital genes in charge of the crosstalk among those disease associated pathways. The case application of **Rcircle** package in CHD analysis indicates that our strategy goes beyond visualization of information to highlight the pattern, prioritize vital candidates, and facilitate scientific discovery.

References

[1]. Wong B. Points of Views: Visualizing Biological Data. Nat Methods 9, 1131 (2102)

[2]. R Core Team (2013). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

The *R* Package ThreeWay For Three-Way Component Analysis

Paolo Giordani[1,*], **Henk A.L. Kiers**[2], **Maria Antonietta Del Ferraro**[1]

1. Sapienza University of Rome
2. University of Groningen
*Contact author: paolo.giordani@uniroma1.it

Keywords: Multi-way data, Principal Component Analysis, Tucker3, Candecomp/Parafac

Data generally refer to the observations of some variables on a set of units and are stored in a (two-way) matrix, say $\mathbf{X}$ of order $(I \times J)$, where I and J denote the numbers of units and variables, respectively. However, in several situations, the available data consist of some variables collected on a set of units on different occasions and are usually stored in a three-way array, say $\underline{\mathbf{X}}$ of order $(I \times J \times K)$, where K denotes the number of occasions. The array can then be seen as a box in which the ways (or indices) correspond to the vertical, horizontal and depth axis. Multi-way data analysis concerns the cases in which the number of indices is higher than two (three-way data analysis when the number of indices is three). In this work we limit our attention to the three-way case. For more details on multi-way (and three-way) analysis, refer to, *e.g.*, [4, 6].

In order to summarize $\mathbf{X}$ classical Principal Component Analysis (PCA) can be applied. In the three-way framework, PCA is no longer a valuable choice. More specifically, PCA could still be applied for exploring $\underline{\mathbf{X}}$ either by rearranging it into a matrix (aggregating over one of the three ways) or analyzing all the two-way data matrices contained in the three-way array separately. Nonetheless, such strategies fail to discover the existing three-way interaction in the data and, therefore, produce incomplete or even misleading results. In the literature there exist several three-way extensions of PCA. The two most popular techniques are the Tucker3 (T3) method [7] and the Candecomp/Parafac (CP) method [1, 3].

The aim of this work is to illustrate the main features of the *R* [5] package **ThreeWay** [2] . **ThreeWay** offers a suit of functions for performing three-way component analysis. In particular, the most relevant functions are `T3` and `CP`, which implement, respectively, T3 and CP. These and other functions will be described through examples using data sets available in the package.

References

[1] Carroll, J. D. and J. J. Chang (1970). Analysis of individual differences in multidimensional scaling via an n-way generalization of eckart-young decomposition. *Psychometrika 35*, 283–319.

[2] Del Ferraro, M. A., H. A. L. Kiers, and P. Giordani (2013). **ThreeWay***: Three-Way Component Analysis*. *R* package version 1.1.1.

[3] Harshman, R. A. (1970). Foundations of the parafac procedure: Models and conditions for an "explanatory" multimodal factor analysis. *UCLA Working Papers in Phonetics 16*, 1–84.

[4] Kroonenberg, P. M. (2008). *Applied Multiway Data Analysis*. Hoboken, NJ: John Wiley & Sons.

[5] *R* Core Team (2013). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: *R* Foundation for Statistical Computing. ISBN 3-900051-07-0.

[6] Smilde, A., R. Bro, and P. Geladi (2004). *Multi-way Analysis: Applications in the Chemical Sciences*. Chichester, UK: John Wiley & Sons.

[7] Tucker, L. R. (1966). Some mathematical notes on three-mode factor analysis. *Psychometrika 31*, 279–311.

Reproducible Research in Public Health

Jinseob Kim[1], **Joohon Sung**[1,⋆]

1. Complex Disease and Genetic Epidemiology Branch, Department of Epidemiology and Institute of Environment and Health, School of Public Health, Seoul National University
⋆Contact author: jsung@snu.ac.kr

Keywords: Reprducible research, LaTeX, Public health, knitr

Despite rapid quantitative expansion of health-related publications, reproducibility of the study is sometimes argues. Inappropriate use of statistical methods is not a rare cause underlying the lack of replications across studies, but the importance of it is usually underestimated[2]. For most health reasearchers who learned and applied the statistical methods properly had spent long time for learning statistics. Addtionally, statistical methods are ever evolving and updating their knowledge and analytic skills will require the most precious resources the time. In this study, we aimed to construct pipelines of reproducible statistical analysis in health research. The development of pipelines in this study consists of 1) automatic suggestions of a summary table describing the general characteristics of the study, 2) univariate analysis of both explanatory and outcome variables of a study, 3) graphical presentations of summary and univariate analyses, 4) automatic analysis and tabulations of main results based on frequently used analytical methods in the health research area (e.g., multiple regression, logistic regression, survival analysis, multilevel analysis, genome-wide association study(GWAS)). For example, researchers can obtain tables and figures if they select data set and dependent variables of interest, and define the nature of each variables (e.g. continuous, binomial, count), explanatory variables, and group variable (e.g., sex, region, or unit of random effects).Using *R* package **knitr**, LaTeX and **tex4ht** package in LaTeX with various statistical packages in *R*, we developed a automatic words describing the result tables and figures with PDF or opendocument format directly[1, 3]. This automated statistical pipeline tools will help individual researcher in health-related or broader arena to help to reduce their analytical burdens, as well as to conduct appropriate statistical analysis much faster and reliable manner.

References

[1] Andrew, A., A. Zvoleff, B. Diggs, C. Pereira, H. Wickham, H. Jeon, J. Arnold, J. Stephens, J. Hester, J. Cheng, J. Keane, J. Allaire, J. Toloe, K. Takahashi, M. Kuhlmann, N. Caballero, N. Salkowski, N. Ross, R. Vaidyanathan, R. Cotton, R. Francois, S. Brouwer, S. de Bernard, T. Wei, T. Lamadon, T. Torsney-Weir, T. Davis, W. Zhu, W. Wu, and Y. Xie (2013). *knitr: A general-purpose package for dynamic report generation in R*. R package version 1.5.

[2] Baker, D., K. Lidster, A. Sottomayor, and S. Amor (2012). Research-reporting standards fall short. *Nature 491*, 672.

[3] Cliffe, E. (2012). Methods to produce flexible and accessible learning resources in mathematics: overview document.

The identification of combined genomic expressions as a diagnostic factor for oral squamous cell carcinoma

Ki-Yeol Kim, phD [1,*]

[1] Oral Cancer Research Institute, College of Dentistry, Yonsei University, SEOUL, 120-752, Korea

*Contact author: kky1004@yuhs.ac

Keywords: Oral squamous cell carcinoma, combined biomarker, microarray dataset

Trends in genetics are transforming in order to identify differential coexpressions of correlated gene expression rather than the significant individual gene. Moreover, it is known that a combined biomarker pattern improves the discrimination of a specific cancer. The identification of the combined biomarker is also necessary for the early detection of invasive oral squamous cell carcinoma (OSCC). To identify the combined biomarker that could improve the discrimination of OSCC, we explored an appropriate number of genes in a combined gene set in order to attain the highest level of accuracy. After detecting a significant gene set, including the pre-defined number of genes, a combined expression was identified using the weights of genes in a gene set. We used the Principal Component Analysis (PCA) for the weight calculation. In this process, we used three public microarray datasets. One dataset was used for identifying the combined biomarker, and the other two datasets were used for validation. The discrimination accuracy was measured by the out-of-bag (OOB) error. There was no relation between the significance and the discrimination accuracy in each individual gene. The identified gene set included both significant and insignificant genes. One of the most significant gene sets in the classification of normal and OSCC included *MMP1, SOCS3* and *ACOX1*. Furthermore, in the case of oral dysplasia and OSCC discrimination, two combined biomarkers were identified. The combined genomic expression achieved better performance in the discrimination of different conditions than in a single significant gene. Therefore, it could be expected that accurate diagnosis for cancer could be possible with a combined biomarker.

References

[1] A. de la Fuente (2010), From 'differential expression' to 'differential networking' - identification of dysfunctional regulatory networks in diseases, Trends Genet 26, 326-333.

[2] J.K. Choi, U. Yu, O.J. Yoo, S. Kim (2005), Differential coexpression analysis using microarray data and its application to human cancer, Bioinformatics 21, 4348-4355.

Reproducible Research in Public Health

Jinseob Kim[1], **Joohon Sung**[1,⋆]

1. Complex Disease and Genetic Epidemiology Branch, Department of Epidemiology and Institute of Environment and Health, School of Public Health, Seoul National University
⋆Contact author: jsung@snu.ac.kr

Keywords: Reprducible research, LaTeX, Public health, knitr

For most health reasearchers who learned and applied the statistical methods properly had spent long time for learning statistics. Addtionally, statistical methods are ever evolving and updating their knowledge and analytic skills will require the most precious resources-the time. In this study, we aimed to construct pipelines of reproducible statistical analysis in health research. The development of pipelines in this study consists of 1) automatic suggestions of a summary table describing the general characteristics of the study, 2) univariate analysis of both explanatory and outcome variables of a study, 3) graphical presentations of summary and univariate analyses, 4) automatic analysis and tabulations of main results based on frequently used analytical methods in the health research area (e.g., multiple regression, logistic regression, survival analysis, multilevel analysis, genome-wide association study(GWAS)). For example, researchers can obtain tables and figures if they select data set and dependent variables of interest, and define the nature of each variables (e.g. continuous, binomial, count), explanatory variables, and group variable (e.g., sex, region, or unit of random effects).Using *R* package **knitr**, LaTeX and **tex4ht** package in LaTeX with various statistical packages in *R*, we developed a automatic words describing the result tables and figures with PDF or open-document format directly[1, 2]. This automated statistical pipeline tools will help individual researcher in health-related or broader arena to help to reduce their analytical burdens, as well as to conduct appropriate statistical analysis much faster and reliable manner.

References

[1] Andrew, A., A. Zvoleff, B. Diggs, C. Pereira, H. Wickham, H. Jeon, J. Arnold, J. Stephens, J. Hester, J. Cheng, J. Keane, J. Allaire, J. Toloe, K. Takahashi, M. Kuhlmann, N. Caballero, N. Salkowski, N. Ross, R. Vaidyanathan, R. Cotton, R. Francois, S. Brouwer, S. de Bernard, T. Wei, T. Lamadon, T. Torsney-Weir, T. Davis, W. Zhu, W. Wu, and Y. Xie (2013). *knitr: A general-purpose package for dynamic report generation in R*. R package version 1.5.

[2] Cliffe, E. (2012). Methods to produce flexible and accessible learning resources in mathematics: overview document.

Visualization and Statistical Modeling of Financial Data with R

Masayuki Jimichi[1*], **Shinnosuke Maeda**[2]

1. Prof. School of Business Administration, Kwansei Gakuin University, Japan
2. M.B.A. candidate, Graduate School of Business Administration, Kwansei Gakuin University, Japan
*Contact author: jimichi@kwansei.ac.jp

Keywords: Data Visualization, Statistical Modeling, Exploratory Data Analysis, Financial Data

We treat visualization (*e.g.*, [6]) and statistical modeling (*e.g.*, [2]) for financial longitudinal data (*e.g.*, sales, employee, assets) of Japanese firms which belong to the first section market of the Tokyo Stock Exchange based on exploratory data analysis [7] with *R*. They are extracted from a database system of Nikkei NEEDS financial data. (See [5].) As a result of data visualization from temporal and cross-sectional aspects, we know the joint distribution of the data sets at each closing day is a multivariate lognormal (*e.g.*, [3]) by using *R* packages **ggplot2** [8], and **googleVis** [4]. We build a statistical model based on the result. Under the condition of fixed time, a lognormal linear model (*e.g.* [1]) with dummy variables which denote the middle classification of industries is very useful for explaining sales by employee and assets. Furthermore, this result is valid in terms of time variation. Note that we can fit the model to the dataset by using the basic *R* function `lm` only.

References

[1] Bradu, D. and Y. Mundlak (1970). Estimation in lognormal linear models. *Journal of the American Statistical Association 65*(329), 198–211.

[2] Chambers, J. M. and T. J. Hastie (Eds.) (1991). *Statistical Models in S*. Chapman and Hall/CRC.

[3] Crow, E. L. and K. Shimizu (Eds.) (1988). *Lognormal Distributions: Theory and Applications*. Marcel Dekker.

[4] Gesmann, M. and D. de Castillo (2011). Using the Google visualization API with R. *The R Journal 3*(2), 40–44.

[5] Jimichi, M. (2010). Building of financial database servers. Technical report. http://kgur.kwansei.ac.jp/dspace/handle/10236/6013.

[6] Tufte, E. R. (2001). *The Visual Display of Quantitative Information*. Graphics Press, Cheshire, Connecticut.

[7] Tukey, J. W. (1977). *Exploratory Data Analysis*. Addison-Wesley Publishing Co.

[8] Wickham, H. (2009). *ggplot2: Elegant Graphics for Data Analysis*. Springer Verlag.

Robust Linear Modeling using the Hyperbolic Distribution

Xinxing Li[1,2,⋆], **David Scott**[1,3]

1. Department of Statistics, The University of Auckland
⋆Contact author: joyce.li@auckland.ac.nz

Keywords: Hyperbolic, Regression, Robust

Linear regression is a very important statistical tool in many fields. However the analysis results can be misleading when the data violates the assumptions behind. One of the key assumptions of linear regression is the error terms are normally distributed which the financial assets return data often violates. The generalized hyperbolic distribution family, introduced by Barndorff-Nielsen ([1]), possesses the non-Gaussian characters which typically are present in financial assets return data. As a special case of the family, the hyperbolic distribution is also featured semi-heavy tails and exhibits skewness for certain parameter values. We propose an approach to fit linear regression models with hyperbolic distributed error to analyze data with heavy-tail and skewness characters. In this work, we developed a set of *R* functions, including `hyperblm` and `summary.hyperblm` function, to implement this approach and provide the result in an appropriate format. These functions are included in the **GeneralizedHyperbolic** package ([2]).

References

[1] Barndorff-Nielsen, O. (1977, 03). Exponentially decreasing distributions for the logarithm of particle size. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences 353*, 401–419.

[2] Scott, D. J. (2010). *GeneralizedHyperbolic*. R package version 0.8-1.

Extending the useR community: developing Shiny applications and interactive graphics

(within an enterprise framework)

Marie Vendettuoli

USDA APHIS Center for Veterinary Biologics
marie.c.vendettuoli@aphis.usda.gov

Keywords: shiny, ggvis, interactive graphics, usability, enterprise architecture

A common challenge when using *R* in an enterprise environment is one of *accessibility*, for both the developer and their clients. Clients requesting analysis may not have the resources to write or run *R* code or scripts. It may be beyond the limits of local IT resources to deploy, maintain, and troubleshoot *R* on more than a handful of computers, and even then only at the most superficial level. The packages **shiny**, **ggvis**, and **gridSVG** attempt to address the first issue by providing a rapid approach to creating custom web interfaces for the *R* functions of interest and embedding interactive, data-driven, graphics. However, deploying **shiny** applications over the web may still be beyond a developer's access due to privacy, IT or budgetary constraints.

We present a case study at USDA APHIS that makes use of virtual desktops to deploy **shiny** applications with minimal resource demands on both IT and individual developers. We present usage statistics from a pilot application that demonstrates measurable value-added impact for both client and developer and allows more users to interact with *R*. We describe strategies for scaling for multiple users, responding to changes in developmental packages and tracking usage. We highlight the importance of studies for enhancing the user experience.

The Compatibility Challenge: Examining R and Developing TERR

Michael Sannella[1,*]

1. TIBCO Software Inc.
*Contact author: msannell@tibco.com

Keywords: R, TERR, implementation, semantics, performance

My group has been working for several years to develop TIBCO® Enterprise Runtime for R (TERR) [1], a new R-compatible engine. We wanted the TERR software to be completely independent of the open-source R engine, so we redesigned and rebuilt the engine from scratch. However, we also wanted TERR to be compatible with R so that we could load and execute existing R software from CRAN and other repositories.

This talk will discuss the challenges we encountered trying to make TERR compatible with R. In order to duplicate the functionality of R, we closely examined its behavior, exposing some interesting features that may not be evident to most users. For some features, we took a different implementation approach than R, though the result was still compatible. In other cases, we deliberately chose not to make TERR 100% compatible.

References

[1] TIBCO (2014). TIBCO Enterprise Runtime for R, http://spotfire.tibco.com/terr/.

Interactive Prototyping of Statistical Graphics with WeBIPP

Jimmy Oh[1,*]

1. Department of Statistics, University of Auckland
*Contact author: joh024@aucklanduni.ac.nz

Keywords: Statistical Graphics, Interactive Graphics, Web Graphics, Visual Programming Environment

This talk presents an R interface to WeBIPP, an interactive web-based tool for creating data-based graphics.

WeBIPP possesses a GUI front-end that is easy to use, allowing the user to rapidly build new statistical graphics from scratch with a few clicks of the mouse. More, the underlying mechanism for WeBIPP ensures that the GUI front-end does not limit any programming input, enabling any programming knowledge in the relevant fields (*HTML*, *CSS*, *SVG*, *JavaScript*, *D3js*[1]) to easily be employed to further enhance the graphic. WeBIPP caters to both the novice GUI user and the expert while enabling the creation of a variety of statistical graphics, from familiar mainstream plots like a barplot, to completely new and innovative plots tailored to presenting a specific feature of a dataset.

References

[1] Bostock, M., V. Ogievetsky, and J. Heer (2011). D3: Data-driven documents. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)*.

How to load and what to do with the PISA data (Program for International Student Assessment)

Przemyslaw Biecek[1,2,*]

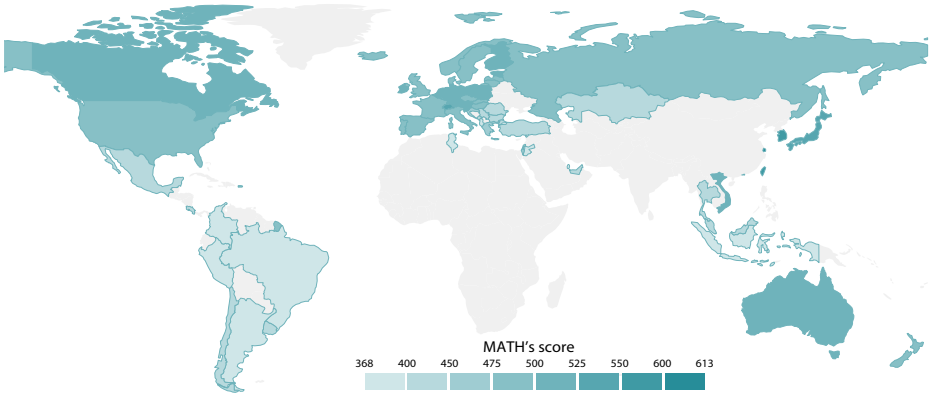
1. Ovali, Iqor Polska
2. Interdisciplinary Centre for Mathematical and Computational Modelling, University of Warsaw
*Contact author: przemyslaw.biecek@gmail.com

Keywords: PISA, PIAAC, OECD

OECD (Organisation for Economic Co-operation and Development) runs PISA (Program for International Student Assessment) and PIAAC (Programme for the International Assessment of Adult Competencies) programs to find data-driven relations between people skills and other factors like occupation, education, school or policy factors.

Each of these programs results in a large and complex datasets, which combine skills of students and adults from different countries together with other characteristics like education, occupation, wealth, school and more than 1000 other factors. All together we have survey results with over 1000 features for over 2 000 000 people from more than 60 countries. Quite large dataset!

During the presentation I will show how to use packages **PIAAC** and **PISA2012lite** to load this data set into *R*, discuss some technical issues and present some exemplary results and visualisations obtained with this dataset.



References

[1] Biecek, P. (2014). Occupations@pisa2012. http://beta.icm.edu.pl/PISAoccupations2012/.

[2] OECD (2014). Piaac and Pisa2012lite packages. https://github.com/pbiecek/PIAAC, https://github.com/pbiecek/PISA2012lite.

Use of Classification Trees for Prediction of Violence

Charles Broderick[1]*, Benjamin Rose[1] , Marie C. Schur[1], Katherine Warburton[1]

1. California Department of State Hospitals
*Contact author: charles.broderick@dsh.ca.gov

Keywords: classification trees, prediction, survival analysis, violence risk assessment

Prediction of future violence, or dangerousness, in mentally ill offenders who have committed previous violent acts is a serious endeavor with both public safety and individual rights issues. The seminal work on violence risk assessment was the MacArthur study [1], which used an early commercial implementation of classification trees [2]. While the MacArthur study was groundbreaking, the findings are not always applicable to all settings. With such a sensitive topic as prediction of future violence risk, it is incumbent upon professionals and organizations to act as responsibly as possible, and ensure that prediction findings are valid and accurate for their setting. The use of *R* [3], along with specialized packages such as **party** [4] and **caret** [5] has enabled our organization to conduct analyses within our system, using our own patient data, to ensure applicability to our setting.

Typically, in any given year, about 30% of our patients have one or more episodes of physically violent behavior. Starting with archived data, results of our pilot studies with *R* and the **party** package have produced classification tree models in a patient group (n=1277) based on only five demographic variables available pre-admission with adequate predictive ability (AUC = 0.67), when evaluating prediction models derived from a training set (70%) applied to a separate test set (30%). Use of these tools has enabled us to go beyond simple "yes/no" categorizations, i.e., we can predict groups of patients at higher risk of multiple aggressive/violent incidents, and by using conditional inference trees with a survival function [6] we can also evaluate patients for time to first violent act. In this manner, our hospitals can potentially better allocate resources in an attempt to prevent violent acts among higher-risk groups before violence even occurs. With the success of initial pilot models, our focus now is turning to identifying clinical variables that can be added to our model, to enhance prediction as well as to make the model more relevant to our clinicians and treatment teams working with these patients. In summary, access to free, open source tools such as *R*, with packages such as **party** and **caret,** has enabled our organization to undertake analyses aimed at predicting violent behavior before the first displays of violence occur. Our goal is to direct enhanced treatments towards individuals identified as high risk, to mitigate risk, before any such violent behaviors even occur.

References

[1] John Monahan, Henry J. Steadman, Eric Silver, .et.al., (2001). Rethinking Risk Assessment. New York: Oxford University Press.
[2] SPSS, Inc. (1993). SPSS for Windows CHAID (Release 6.0). Chicago: SPSS.
[3] R Core Team (2014). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.
[4] Torsten Hothorn, Kurt Hornik and Achim Zeileis (2006). Unbiased Recursive Partitioning: A Conditional Inference Framework. Journal of Computational and Graphical Statistics, 15(3), 651--674.
[5] Max Kuhn. Contributions from Jed Wing, Steve Weston, Andre Williams, Chris Keefer, Allan Engelhardt, Tony Cooper, Zachary Mayer and the R Core Team (2014). caret: Classification and Regression Training. R package version 6.0-24. http://CRAN.R-project.org/package=caret
[6] Brian S. Everitt and Torsten Hothorn (2013). HSAUR: A Handbook of Statistical Analyses Using R. R package version 1.3-3. http://CRAN.R-project.org/package=HSAUR

Using *R* for official statistics: Census of Foreign Capital in Brazil.

Carlos Cinelli[1,2,*] , Rodrigo Wang[1]

1. Brazilian Central Bank
2. University of Brasilia
*Contact author: carlos.cinelli@bcb.gov.br

Keywords: official statistics, data imputation, census of foreign capital in brazil, international investment position.

The Census of Foreign Capital in Brazil (Census) has been carried out quinquennially since 1996 by the Brazilian Central Bank (BCB). Its major purpose is to measure the stock of Foreign Direct Investments in Brazil (FDI), necessary to compile the International Investment Position (IIP) statistics. As of 2011, the Census was split into two surveys: the 5-year Census and the Annual Census, the latter targeted to large enterprises only. The distribution of the FDI stock is heavy tailed, so large enterprises comprise 80-90% of the total value, but represent only a small fraction (10-20%) of the total number of respondents, thus reducing the cost of the survey without much loss of information.

As for the data of the missing respondents, we mostly replicate their latest survey values and add their flows registered in the Balance of Payments. But this ***is not*** as trivial as it sounds. During the 5-year gap between the complete Censuses a lot can happen: new companies start, some companies will close, other companies will merge with each other, some companies will not have foreign investors anymore or the nationality of the foreign investors may change. To deal with these facts we must gather information from other sources (for example, the Brazilian IRS) and perform some data analysis in order to decide which data will or will not be imputed, and how it will be imputed. The further you are from the latest 5-year Census, the more complicated this task gets.

Microsoft Excel is widely used for data manipulation and data analysis in Central Banks and International Organizations (like the IMF). So, unfortunately, our first choice was to use Excel. This was prone to a lot of operational errors - *Reinhart-Rogoff style*. It required the use of many different spreadsheets and files: a cumbersome process to manage that was hard to find a bug when there was one. It was not easy to immediately reproduce the results, and it was not really clear to an outsider where and when data modification was taking place, because of all the cross-references between worksheets. So we decided to create *R* packages with functions that automate the process.

We have developed three packages (names in Portuguese): **censo.criar.base,** which gathers and combines all information necessary to the imputation and compilation into a new database; **censo.extrapolar,** which automates the imputation; and, **censo.quadros,** with functions to calculate the main statistics and analysis. The publication's process time *reduced from 1-2 weeks to a couple calls in the command line* (that takes only a few minutes). It is now easier to track bugs and errors, because all data comparisons and transformations are clearly stated on the codes. Another advantage is that new kinds of data exploration and visualization, that once were not possible, are now easily available through *R*. This has helped the development of a more structured ***validation*** and ***exploration*** of the data – and those are the new packages we are working on.

The goal of our presentation is to describe the imputation/validation/publication process of the FDI statistic in BCB, focusing on the application of *R* in our workflow and deepening the discussion of the main advantages/disadvantages of using *R* for official statistics.

ALUES: an R package for evaluating land for agricultural use

Arnold R. Salvacion[1], Al-Ahmadgaid B. Asaad[2]

[1] Institute for Governance and Rural Development,College of Public Affairs and Development
University of the Philippines Los Baños, College 4031, Laguna
[2] Mindanao State University – Iligan Institute of Technology, Iligan 9200
Philippines
*Contact author: arsalvacion@gmail.com

Keywords: Agricultural land use, fuzzy logic, R

Agricultural Land Use Evaluation System (ALUES) is an R package that evaluates land suitability for different crop production. The package is based on the Food and Agriculture Organization (FAO) and the International Rice Research Institute (IRRI) methodology for land evaluation. Development of ALUES is inspired by similar tool for land evaluation, Land Use Suitability Evaluation Tool (LUSET). The package uses fuzzy logic approach to evaluate land suitability of a particular area based on inputs such as rainfall, temperature, topography, and soil properties. The membership functions used for fuzzy modeling are the following: Triangular, Trapezoidal, Gaussian, Sigmoidal and custom models with functions that can be defined by the user. The package also aims on complicated methods like considering more than one fuzzy membership function on different suitability class. The methods for computing the overall suitability of a particular area are also included, and these are the Minimum, Maximum, Product, Sum, Average, Exponential and Gamma. Finally, ALUES utilizes the power of Rcpp library for efficient computation.

Extending Agriculture Simulator Capabilities with *R*

Bryan Stanfill[1*], **Jody Biggs**[2]

1. CSIRO Comuptational Informatics
2. CSIRO Ecosystem Sciences
*Contact author: bryan.stanfill@csiro.au

Keywords: APSIM, uncertainty of complex computer models, visualization, high performance computing

The Agricultural Production Simulator (APSIM) is a widely used, powerful and highly complex computer program. Based on information about weather, soil properties, farming practices and land use, APSIM can predict crop and environmental outcomes such as crop yield, nitrogen runoff and sediment loss as a function of time and space. Recent increased interest in additionally quantifying and reducing uncertainty about APSIM predictions has made the short comings of the current APSIM interface more apparent. In particular, only basic visualization and summary techniques are available within APSIM; this leads researchers to use a second program, such as *R*, in order to better understand the results. Additionally, running APSIM for a variety of input values is not straight forward and interested researchers need to write their own scripts to automate repeated APSIM runs. We introduce the **apsimr** package, which aims to extend APSIM by adding advanced analytic measures and to ease the pain of learning APSIM by researchers in other fields. The **apsimr** package includes function to allow the user to create, alter and run APSIM simulations individually or in large batches. The results can then be visualized and summarized using standard or advanced analytics. Sensitivity and uncertainty analysis can require several hundred APSIM runs, therefore **apsimr** links to the **APSIMBatch** package to run APSIM on high performance computers. In this talk we will demonstrate the use of **apsimr** and discuss the problems that arose in its creation, most of which stem from the unique structure of the input and output files expected and produced by APSIM.

RIGHT: an *HTML* canvas and *JavaScript*-based interactive data visualization package for linked graphics

ChungHa Sung[1], TaeJoon Song[2], Jae W. Lee[1], and Junghoon Lee[3*]

1. Sungkyunkwan University (SKKU), Suwon, Gyeonggi-Do, 440-746, South Korea
2. Samsung Electronics, Hwaseong-Si, Gyeonggi-Do, 143-130, South Korea
3. Merck Research Laboratories, Rahway, NJ, 08901, U.S.A.
*Contact author: jung_hoon_lee@merck.com

Keywords: interactive data visualization, linked graphics, *HTML* canvas, *JavaScript*

Interactive data visualization has received broad interest in the *R* community due to its obvious benefits over static visualization: more information can be delivered concisely and intuitively by user engagement. As a result, various *R* packages supporting single-layer, multi-layer, and linked graphics have been developed, including **rCharts**, **iPlots**, **cranvas**, **ggvis**, **animint** and **googleVis** [1]. R Interactive Graphics via HTml (**RIGHT**, https://code.google.com/p/r-interactive-graphics-via-html/) is an interactive data visualization package for linked graphics based on *HTML* canvas and *JavaScript*. It provides an *R* API similar to base graphics to easily construct various interactive plots, including scatter, line, bar, pie, and box-whisker plots.

This poster presents an overview of **RIGHT** and the *JavaScript* data structure that enables linked graphics. **RIGHT** is the first package that implements linked graphs using *HTML* canvas and *JavaScript*. Linked graphics help answer obvious questions a collection of plots tend to raise: how one point in the plot is related to another point in another plot. *HTML* canvas and *JavaScript* make it possible to deliver the visualization to various platforms, including mobile devices, since they are standard web technologies supported by most modern web browsers (albeit some remaining compatibility issues). This approach can also benefit from the improvement of *JavaScript* performance every generation, driven by various web applications with ever increasing complexity and sophistication.

References

[1] Toby Dylan Hocking, https://github.com/tdhock/interactive-tutorial (as of March 16, 2014).

Faster FastR through Partial Evaluation and Compilation

Michael Haupt[1,*], **Christian Humer**[2], **Mick Jordan**[1], **Prahlad Joshi**[3], **Jan Vitek**[3], **Adam Welc**[1], **Christian Wirth**[1], **Andreas Wöß**[2], **Mario Wolczko**[1], **Thomas Würthinger**[1]

1. Oracle Labs
2. Johannes Kepler University, Linz, Austria
3. Purdue University, West Lafayette, IN, USA
*Contact author: michael.haupt@oracle.com

Keywords: R Language Runtime, Java, R Performance, AST Interpretation, Partial Evaluation

FastR, first introduced at *useR! 2013* [3], is an implementation of the *R* programming language in *Java* [2]. It uses the concept of self-specialising abstract syntax tree (AST) interpretation [5]. In such interpreters, AST nodes replace themselves with nodes that are specialised for handling the types and data actually occurring during execution. This saves considerable time in the implementation of dynamically typed programming languages.

The implementation introduced in 2013 was a pure interpreter. We introduce the next version of *FastR*. The current implementation is based on Truffle [4]. Truffle is a framework for the implementation of specialising AST interpreters. Truffle-based language implementations transparently employ partial evaluation of specialised ASTs, and dynamic compilation, to obtain performance competitive with that of dedicated dynamic compilers.

The performance of some development versions of *FastR* running the b25 benchmarks and an *R* version of a subset of the Computer Language Benchmarks Game ("shootout") is, on average, more than an order of magnitude faster than the GNU R byte code interpreter, and significantly faster than the purely interpreted version of FastR. *FastR* is available as an open source project [1] under the terms and conditions of the GNU General Public License 2.

We will describe the status of the implementation and outline our plans for the future. An important long-term goal of the *FastR* project is to dispense with the need for implementing performance-critical parts of *R* applications in lower-level languages.

References

[1] BitBucket (2014). FastR project. http://bitbucket.org/allr/fastr.

[2] Kalibera, T., P. Maj, F. Morandat, and J. Vitek (2014). A Fast Abstract Syntax Tree Interpreter for R. In *Proceedings of the 10th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments*, VEE '14, New York, NY, USA, pp. 89–102. ACM.

[3] Kalibera, T., P. Maj, and J. Vitek (2013). R in Java: Why and How? In *The R User Conference, useR! 2013, Book of Contributed Abstracts*, pp. 111. http://www.edii.uclm.es/~useR-2013/docs/useR2013_abstract_booklet.pdf.

[4] Würthinger, T., C. Wimmer, A. Wöß, L. Stadler, G. Duboscq, C. Humer, G. Richards, D. Simon, and M. Wolczko (2013). One VM to rule them all. In *Proceedings of the 2013 ACM international symposium on New ideas, new paradigms, and reflections on programming & software*, pp. 187–204. ACM.

[5] Würthinger, T., A. Wöß, L. Stadler, G. Duboscq, D. Simon, and C. Wimmer (2012). Self-optimizing AST interpreters. In *Proceedings of the 8th Symposium on Dynamic Languages*, DLS '12, New York, NY, USA, pp. 73–82. ACM.

Distributed Matrix Exponentiation in R

Drew Schmidt[1,*], **Nick Matzke**[2]

1. National Institute for Computational Sciences, University of Tennessee
2. National Institute for Mathematical and Biological Synthesis, University of Tennessee
*Contact author: schmidt@math.utk.edu

Keywords: HPC, linear algebra, ScaLAPACK, MPI, parallel programming

Matrix exponentiation is an important matrix function which is useful in a wide variety of domains and applications. Formally, matrix exponentiation is an easily understood power series; but efficient, numerically stable algorithms for computing this function have been debated for over 30 years. There are several serial implementations of the matrix exponential available to *R*, including those found in the **Matrix** and **rexpokit** packages. We introduce a relatively new algorithm for computing the matrix exponential due to Al-Mohy and Higham, implemented in the **pbdDMAT** package. Our implementation includes both serial and distributed versions of this algorithm, the latter of which fully integrates with the pbdR framework for high performance computing with *R*. Finally, we will conclude by demonstrating the scalability of the implementation with benchmarks on University of Tennessee supercomputing resources.

References

[1] Al-Mohy, A. H. and N. J. Higham (2009). A New Scaling and Squaring Algorithm for the Matrix Exponential.

[2] Blackford, L. S., J. Choi, A. Cleary, E. D'Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker, and R. C. Whaley (1997). *ScaLAPACK Users' Guide*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

[3] Ostrouchov, G., W.-C. Chen, D. Schmidt, and P. Patel. Programming with Big Data in R,.

[4] Schmidt, D., W.-C. Chen, G. Ostrouchov, and P. Patel (2012). pbdDMAT: Distributed Matrix Algebra Computation. R Package, URL http://cran.r-project.org/package=pbdDMAT.

[5] Sidje, R. B. (1998, March). Expokit: A Software Package for Computing Matrix Exponentials. *ACM Trans. Math. Softw. 24*(1), 130–156.