

# R Reference Card for Data Mining

by Yanchang Zhao, yanchang@rdatamining.com, January 3, 2013

The latest version is available at <http://www.RDataMining.com>. Click the link also for document *R and Data Mining: Examples and Case Studies*. The package names are in parentheses.

## Association Rules & Frequent Itemsets

### APRIORI Algorithm

a level-wise, breadth-first algorithm which counts transactions to find frequent itemsets

**apriori()** mine associations with APRIORI algorithm (*arules*)

### ECLAT Algorithm

employs equivalence classes, depth-first search and set intersection instead of counting

**eclat()** mine frequent itemsets with the Eclat algorithm (*arules*)

### Packages

**arules** mine frequent itemsets, maximal frequent itemsets, closed frequent itemsets and association rules. It includes two algorithms, Apriori and Eclat.

**arulesViz** visualizing association rules

## Sequential Patterns

### Functions

**cspade()** mining frequent sequential patterns with the cSPADE algorithm (*arulesSequences*)

**seqefsub()** searching for frequent subsequences (*TraMineR*)

### Packages

**arulesSequences** add-on for *arules* to handle and mine frequent sequences

**TraMineR** mining, describing and visualizing sequences of states or events

## Classification & Prediction

### Decision Trees

**ctree()** conditional inference trees, recursive partitioning for continuous, censored, ordered, nominal and multivariate response variables in a conditional inference framework (*party*)

**rpart()** recursive partitioning and regression trees (*rpart*)

**mob()** model-based recursive partitioning, yielding a tree with fitted models associated with each terminal node (*party*)

### Random Forest

**cforest()** random forest and bagging ensemble (*party*)

**randomForest()** random forest (*randomForest*)

**varimp()** variable importance (*party*)

**importance()** variable importance (*randomForest*)

### Neural Networks

**nnet()** fit single-hidden-layer neural network (*nnet*)

### Support Vector Machine (SVM)

**svm()** train a support vector machine for regression, classification or density-estimation (*e1071*)

**ksvm()** support vector machines (*kernlab*)

## Performance Evaluation

**performance()** provide various measures for evaluating performance of prediction and classification models (*ROCR*)

**roc()** build a ROC curve (*pROC*)

**auc()** compute the area under the ROC curve (*pROC*)

**ROC()** draw a ROC curve (*DiagnosisMed*)

**PRcurve()** precision-recall curves (*DMwR*)

**CRchart()** cumulative recall charts (*DMwR*)

### Packages

**rpart** recursive partitioning and regression trees

**party** recursive partitioning

**randomForest** classification and regression based on a forest of trees using random inputs

**rpartOrdinal** ordinal classification trees, deriving a classification tree when the response to be predicted is ordinal

**rpart.plot** plots rpart models with an enhanced version of `plot.rpart` in the *rpart* package

**ROCR** visualize the performance of scoring classifiers

**pROC** display and analyze ROC curves

## Regression

### Functions

**lm()** linear regression

**glm()** generalized linear regression

**nls()** non-linear regression

**predict()** predict with models

**residuals()** residuals, the difference between observed values and fitted values

**glm()** fit a linear model using generalized least squares (*nlme*)

**gnls()** fit a nonlinear model using generalized least squares (*nlme*)

### Packages

**nlme** linear and nonlinear mixed effects models

## Clustering

### Partitioning based Clustering

partition the data into k groups first and then try to improve the quality of clustering by moving objects from one group to another

**kmeans()** perform k-means clustering on a data matrix

**kmeansCBI()** interface function for *kmeans* (*fpc*)

**kmeansruns()** call *kmeans* for the k-means clustering method and includes estimation of the number of clusters and finding an optimal solution from several starting points (*fpc*)

**pam()** the Partitioning Around Medoids (PAM) clustering method (*cluster*)

**pamk()** the Partitioning Around Medoids (PAM) clustering method with estimation of number of clusters (*fpc*)

**cluster.optimal()** search for the optimal k-clustering of the dataset (*bayesclust*)

**clara()** Clustering Large Applications (*cluster*)

**fanny(x, k, ...)** compute a fuzzy clustering of the data into k clusters (*cluster*)

**kcca()** k-centroids clustering (*flexclust*)

**ccfkms()** clustering with Conjugate Convex Functions (*cba*)

**apcluster()** affinity propagation clustering for a given similarity matrix (*apcluster*)

**apclusterK()** affinity propagation clustering to get K clusters (*apcluster*)

**cclust()** Convex Clustering, incl. k-means and two other clustering algorithms (*cclust*)

**KMeansSparseCluster()** sparse k-means clustering (*sparcl*)

**tcclust(x, k, alpha, ...)** trimmed k-means with which a proportion alpha of observations may be trimmed (*tcclust*)

## Hierarchical Clustering

a hierarchical decomposition of data in either bottom-up (agglomerative) or top-down (divisive) way

**hclust(d, method, ...)** hierarchical cluster analysis on a set of dissimilarities d using the *method* for agglomeration

**birch()** the BIRCH algorithm that clusters very large data with a CF-tree (*birch*)

**pvcclust()** hierarchical clustering with p-values via multi-scale bootstrap resampling (*pvcclust*)

**agnes()** agglomerative hierarchical clustering (*cluster*)

**diana()** divisive hierarchical clustering (*cluster*)

**mona()** divisive hierarchical clustering of a dataset with binary variables only (*cluster*)

**rockCluster()** cluster a data matrix using the Rock algorithm (*cba*)

**proximus()** cluster the rows of a logical matrix using the Proximus algorithm (*cba*)

**isopam()** Isopam clustering algorithm (*isopam*)

**LLAhclust()** hierarchical clustering based on likelihood linkage analysis (*LLAhclust*)

**flashClust()** optimal hierarchical clustering (*flashClust*)

**fastcluster()** fast hierarchical clustering (*fastcluster*)

**cutreeDynamic()**, **cutreeHybrid()** detection of clusters in hierarchical clustering dendrograms (*dynamicTreeCut*)

**HierarchicalSparseCluster()** hierarchical sparse clustering (*sparcl*)

## Model based Clustering

**Mclust()** model-based clustering (*mclust*)

**HDDC()** a model-based method for high dimensional data clustering (*HDDCsif*)

**fixmahal()** Mahalanobis Fixed Point Clustering (*fpc*)

**fixreg()** Regression Fixed Point Clustering (*fpc*)

**mergenormals()** clustering by merging Gaussian mixture components (*fpc*)

## Density based Clustering

generate clusters by connecting dense regions

**dbscan(data, eps, MinPts, ...)** generate a density based clustering of arbitrary shapes, with neighborhood radius set as *eps* and density threshold as *MinPts* (*fpc*)

**pdfCluster()** clustering via kernel density estimation (*pdfCluster*)

## Other Clustering Techniques

**mixer()** random graph clustering (*mixer*)

**nncluster()** fast clustering with restarted minimum spanning tree (*nncluster*)

**orclus()** ORCLUS subspace clustering (*orclus*)

## Plotting Clustering Solutions

**plotcluster()** visualisation of a clustering or grouping in data (*fpc*)

**bannerplot()** a horizontal barplot visualizing a hierarchical clustering (*cluster*)

## Cluster Validation

**silhouette()** compute or extract silhouette information (*cluster*)  
**cluster.stats()** compute several cluster validity statistics from a clustering and a dissimilarity matrix (*fpc*)  
**clValid()** calculate validation measures for a given set of clustering algorithms and number of clusters (*clValid*)  
**clustIndex()** calculate the values of several clustering indexes, which can be independently used to determine the number of clusters existing in a data set (*cclust*)  
**NbClust()** provide 30 indices for cluster validation and determining the number of clusters (*NbClust*)

## Packages

**cluster** cluster analysis  
**fpc** various methods for clustering and cluster validation  
**mclust** model-based clustering and normal mixture modeling  
**birch** clustering very large datasets using the BIRCH algorithm  
**pvcust** hierarchical clustering with p-values  
**apcluster** Affinity Propagation Clustering  
**cclust** Convex Clustering methods, including k-means algorithm, On-line Update algorithm and Neural Gas algorithm and calculation of indexes for finding the number of clusters in a data set  
**cba** Clustering for Business Analytics, including clustering techniques such as Proximus and Rock  
**bclust** Bayesian clustering using spike-and-slab hierarchical model, suitable for clustering high-dimensional data  
**biclust** algorithms to find bi-clusters in two-dimensional data  
**clue** cluster ensembles  
**clues** clustering method based on local shrinking  
**clValid** validation of clustering results  
**clv** cluster validation techniques, contains popular internal and external cluster validation methods for outputs produced by package *cluster*  
**bayesclust** tests/searches for significant clusters in genetic data  
**clustvarsel** variable selection for model-based clustering  
**clustsig** significant cluster analysis, tests to see which (if any) clusters are statistically different  
**clusterfly** explore clustering interactively  
**clusterSim** search for optimal clustering procedure for a data set  
**clusterGeneration** random cluster generation  
**clusterCons** calculate the consensus clustering result from re-sampled clustering experiments with the option of using multiple algorithms and parameter  
**gcExplorer** graphical cluster explorer  
**hybridHclust** hybrid hierarchical clustering via mutual clusters  
**Modalclust** hierarchical modal Clustering  
**iCluster** integrative clustering of multiple genomic data types  
**EMCC** evolutionary Monte Carlo (EMC) methods for clustering  
**rEMM** extensible Markov Model (EMM) for data stream clustering

## Outlier Detection

### Functions

**boxplot.stats()**\$out list data points lying beyond the extremes of the whiskers  
**lofactor()** calculate local outlier factors using the LOF algorithm (*DMwR* or *dprep*)  
**lof()** a parallel implementation of the LOF algorithm (*Rlof*)

## Packages

**extremevalues** detect extreme values in one-dimensional data  
**mvoutlier** multivariate outlier detection based on robust methods  
**outliers** some tests commonly used for identifying outliers  
**Rlof** a parallel implementation of the LOF algorithm

## Time Series Analysis

### Construction & Plot

**ts()** create time-series objects (*stats*)  
**plot.ts()** plot time-series objects (*stats*)  
**smoothts()** time series smoothing (*ast*)  
**sfilter()** remove seasonal fluctuation using moving average (*ast*)

### Decomposition

**decomp()** time series decomposition by square-root filter (*timsac*)  
**decompose()** classical seasonal decomposition by moving averages (*stats*)  
**stl()** seasonal decomposition of time series by loess (*stats*)  
**tsr()** time series decomposition (*ast*)  
**ardec()** time series autoregressive decomposition (*ArDec*)

### Forecasting

**arima()** fit an ARIMA model to a univariate time series (*stats*)  
**predict.Arima()** forecast from models fitted by *arima* (*stats*)  
**auto.arima()** fit best ARIMA model to univariate time series (*forecast*)  
**forecast.stl()**, **forecast.ets()**, **forecast.Arima()**  
forecast time series using *stl*, *ets* and *arima* models (*forecast*)

## Packages

**forecast** displaying and analysing univariate time series forecasts  
**timsac** time series analysis and control program  
**ast** time series analysis  
**ArDec** time series autoregressive-based decomposition  
**ares** a toolbox for time series analyses using generalized additive models  
**dse** tools for multivariate, linear, time-invariant, time series models

## Text Mining

### Functions

**Corpus()** build a corpus, which is a collection of text documents (*tm*)  
**tm.map()** transform text documents, e.g., stemming, stopword removal (*tm*)  
**tm.filter()** filtering out documents (*tm*)  
**TermDocumentMatrix()**, **DocumentTermMatrix()** construct a term-document matrix or a document-term matrix (*tm*)  
**Dictionary()** construct a dictionary from a character vector or a term-document matrix (*tm*)  
**findAssocs()** find associations in a term-document matrix (*tm*)  
**findFreqTerms()** find frequent terms in a term-document matrix (*tm*)  
**stemDocument()** stem words in a text document (*tm*)  
**stemCompletion()** complete stemmed words (*tm*)  
**termFreq()** generate a term frequency vector from a text document (*tm*)  
**stopwords(language)** return stopwords in different languages (*tm*)  
**removeNumbers()**, **removePunctuation()**, **removeWords()** remove numbers, punctuation marks, or a set of words from a text document (*tm*)  
**removeSparseTerms()** remove sparse terms from a term-document matrix (*tm*)  
**textcat()** n-gram based text categorization (*textcat*)

**SnowballStemmer()** Snowball word stemmers (*Snowball*)  
**LDA()** fit a LDA (latent Dirichlet allocation) model (*topicmodels*)  
**CTM()** fit a CTM (correlated topics model) model (*topicmodels*)  
**terms()** extract the most likely terms for each topic (*topicmodels*)  
**topics()** extract the most likely topics for each document (*topicmodels*)

## Packages

**tm** a framework for text mining applications  
**lda** fit topic models with LDA  
**topicmodels** fit topic models with LDA and CTM  
**RTextTools** automatic text classification via supervised learning  
**tm.plugin.dc** a plug-in for package *tm* to support distributed text mining  
**tm.plugin.mail** a plug-in for package *tm* to handle mail  
**RcmdrPlugin.TextMining** GUI for demonstration of text mining concepts and *tm* package

**textir** a suite of tools for inference about text documents and associated sentiment  
**tau** utilities for text analysis  
**textcat** n-gram based text categorization  
**YjdnJlp** Japanese text analysis by Yahoo! Japan Developer Network

## Social Network Analysis and Graph Mining

### Functions

**graph()**, **graph.edgelist()**, **graph.adjacency()**,  
**graph.incidence()** create graph objects respectively from edges, an edge list, an adjacency matrix and an incidence matrix (*igraph*)  
**plot()**, **tkplot()** static and interactive plotting of graphs (*igraph*)  
**gplot()**, **gplot3d()** plot graphs (*sna*)  
**V()**, **E()** vertex/edge sequence of *igraph* (*igraph*)  
**are.connected()** check whether two nodes are connected (*igraph*)  
**degree()**, **betweenness()**, **closeness()** various centrality scores (*igraph*, *sna*)  
**add.edges()**, **add.vertices()**, **delete.edges()**,  
**delete.vertices()** add and delete edges and vertices (*igraph*)  
**neighborhood()** neighborhood of graph vertices (*igraph*, *sna*)  
**get.adjlist()** adjacency lists for edges or vertices (*igraph*)  
**nei()**, **adj()**, **from()**, **to()** vertex/edge sequence indexing (*igraph*)  
**cliques()** find cliques, ie. complete subgraphs (*igraph*)  
**clusters()** maximal connected components of a graph (*igraph*)  
**%->%**, **%<-%**, **%--%** edge sequence indexing (*igraph*)  
**get.edgelist()** return an edge list in a two-column matrix (*igraph*)  
**read.graph()**, **write.graph()** read and writ graphs from and to files (*igraph*)

## Packages

**sna** social network analysis  
**igraph** network analysis and visualization  
**statnet** a set of tools for the representation, visualization, analysis and simulation of network data  
**egonet** ego-centric measures in social network analysis  
**snort** social network-analysis on relational tables  
**network** tools to create and modify network objects  
**bipartite** visualising bipartite networks and calculating some (ecological) indices  
**blockmodeling** generalized and classical blockmodeling of valued networks  
**diagram** visualising simple graphs (networks), plotting flow diagrams  
**NetCluster** clustering for networks  
**NetData** network data for McFarland's SNA R labs

**NetIndices** estimating network indices, including trophic structure of foodwebs in R

**NetworkAnalysis** statistical inference on populations of weighted or unweighted networks

**tnet** analysis of weighted, two-mode, and longitudinal networks

**triads** triad census for networks

## Spatial Data Analysis

### Functions

**geocode()** geocodes a location using Google Maps (*ggmap*)

**qmap()** quick map plot (*ggmap*)

**get\_map()** queries the Google Maps, OpenStreetMap, or Stamen Maps server for a map at a certain location (*ggmap*)

**gvisGeoChart()**, **gvisGeoMap()**, **gvisIntensityMap()**,  
**gvisMap()** Google geo charts and maps (*googleVis*)

**GetMap()** download a static map from the Google server (*RgoogleMaps*)

**ColorMap()** plot levels of a variable in a colour-coded map (*RgoogleMaps*)

**PlotOnStaticMap()** overlay plot on background image of map tile (*RgoogleMaps*)

**TextOnStaticMap()** plot text on map (*RgoogleMaps*)

### Packages

**plotGoogleMaps** plot spatial data as HTML map mashup over Google Maps

**RgoogleMaps** overlay on Google map tiles in R

**plotKML** visualization of spatial and spatio-temporal objects in Google Earth

**ggmap** Spatial visualization with Google Maps and OpenStreetMap

**clustTool** GUI for clustering data with spatial information

**SGCS** Spatial Graph based Clustering Summaries for spatial point patterns

**spdep** spatial dependence: weighting schemes, statistics and models

## Statistics

### Summarization

**summary()** summarize data

**describe()** concise statistical description of data (*Hmisc*)

**boxplot.stats()** box plot statistics

### Analysis of Variance

**aov()** fit an analysis of variance model (*stats*)

**anova()** compute analysis of variance (or deviance) tables for one or more fitted model objects (*stats*)

### Statistical Test

**t.test()** student's t-test (*stats*)

**prop.test()** test of equal or given proportions (*stats*)

**binom.test()** exact binomial test (*stats*)

### Mixed Effects Models

**lme()** fit a linear mixed-effects model (*nlme*)

**nlme()** fit a nonlinear mixed-effects model (*nlme*)

### Principal Components and Factor Analysis

**princomp()** principal components analysis (*stats*)

**prcomp()** principal components analysis (*stats*)

### Other Functions

**var()**, **cov()**, **cor()** variance, covariance, and correlation (*stats*)

**density()** compute kernel density estimates (*stats*)

## Packages

**nlme** linear and nonlinear mixed effects models

## Graphics

### Functions

**plot()** generic function for plotting (*graphics*)

**barplot()**, **pie()**, **hist()** bar chart, pie chart and histogram (*graphics*)

**boxplot()** box-and-whisker plot (*graphics*)

**stripchart()** one dimensional scatter plot (*graphics*)

**dotchart()** Cleveland dot plot (*graphics*)

**qqnorm()**, **qqplot()**, **qqline()** QQ (quantile-quantile) plot (*stats*)

**coplot()** conditioning plot (*graphics*)

**sploM()** conditional scatter plot matrices (*lattice*)

**pairs()** a matrix of scatterplots (*graphics*)

**cpairs()** enhanced scatterplot matrix (*gclus*)

**parcoord()** parallel coordinate plot (*MASS*)

**cparcoord()** enhanced parallel coordinate plot (*gclus*)

**paracoor()** parallel coordinates plot (*denpro*)

**parallelplot()** parallel coordinates plot (*lattice*)

**densityplot()** kernel density plot (*lattice*)

**contour()**, **filled.contour()** contour plot (*graphics*)

**levelplot()**, **contourplot()** level plots and contour plots (*lattice*)

**smoothScatter()** scatterplots with smoothed densities color representation; capable of visualizing large datasets (*graphics*)

**sunflowerplot()** a sunflower scatter plot (*graphics*)

**assocplot()** association plot (*graphics*)

**mosaicplot()** mosaic plot (*graphics*)

**matplot()** plot the columns of one matrix against the columns of another (*graphics*)

**fourfoldplot()** a fourfold display of a  $2 \times 2 \times k$  contingency table (*graphics*)

**persp()** perspective plots of surfaces over the x?y plane (*graphics*)

**cloud()**, **wireframe()** 3d scatter plots and surfaces (*lattice*)

**interaction.plot()** two-way interaction plot (*stats*)

**iplot()**, **ihist()**, **ibar()**, **ipcp()** interactive scatter plot, histogram, bar plot, and parallel coordinates plot (*iplots*)

**pdf()**, **postscript()**, **win.metafile()**, **jpeg()**, **bmp()**,  
**png()**, **tiff()** save graphs into files of various formats

**gvisAnnotatedTimeLine()**, **gvisAreaChart()**,  
**gvisBarChart()**, **gvisBubbleChart()**,  
**gvisCandlestickChart()**, **gvisColumnChart()**,  
**gvisComboChart()**, **gvisGauge()**, **gvisGeoChart()**,  
**gvisGeoMap()**, **gvisIntensityMap()**,  
**gvisLineChart()**, **gvisMap()**, **gvisMerge()**,  
**gvisMotionChart()**, **gvisOrgChart()**,  
**gvisPieChart()**, **gvisScatterChart()**,  
**gvisSteppedAreaChart()**, **gvisTable()**,  
**gvisTreeMap()** various interactive charts produced with the Google Visualisation API (*googleVis*)

**gvisMerge()** merge two *googleVis* charts into one (*googleVis*)

### Packages

**ggplot2** an implementation of the Grammar of Graphics

**googleVis** an interface between R and the Google Visualisation API to create interactive charts

**lattice** a powerful high-level data visualization system, with an emphasis on multivariate data

**vcd** visualizing categorical data

**denpro** visualization of multivariate, functions, sets, and data

**iplots** interactive graphics

## Data Manipulation

### Functions

**transform()** transform a data frame

**scale()** scaling and centering of matrix-like objects

**t()** matrix transpose

**aperm()** array transpose

**sample()** sampling

**table()**, **tabulate()**, **xtabs()** cross tabulation (*stats*)

**stack()**, **unstack()** stacking vectors

**split()**, **unsplit()** divide data into groups and reassemble

**reshape()** reshape a data frame between “wide” and “long” format (*stats*)

**merge()** merge two data frames; similar to database *join* operations

**aggregate()** compute summary statistics of data subsets (*stats*)

**by()** apply a function to a data frame split by factors

**melt()**, **cast()** melt and then cast data into the reshaped or aggregated form you want (*reshape*)

**complete.cases()** find complete cases, i.e., cases without missing values

**na.fail**, **na.omit**, **na.exclude**, **na.pass** handle missing values

### Packages

**reshape** flexibly restructure and aggregate data

**data.table** extension of data.frame for fast indexing, ordered joins, assignment, and grouping and list columns

**gdata** various tools for data manipulation

## Data Access

### Functions

**save()**, **load()** save and load R data objects

**read.csv()**, **write.csv()** import from and export to .CSV files

**read.table()**, **write.table()**, **scan()**, **write()** read and write data

**write.matrix()** write a matrix or data frame (*MASS*)

**readLines()**, **writeLines()** read/write text lines from/to a connection, such as a text file

**sqlQuery()** submit an SQL query to an ODBC database (*RODBC*)

**sqlFetch()** read a table from an ODBC database (*RODBC*)

**sqlSave()**, **sqlUpdate()** write or update a table in an ODBC database (*RODBC*)

**sqlColumns()** enquire about the column structure of tables (*RODBC*)

**sqlTables()** list tables on an ODBC connection (*RODBC*)

**odbcConnect()**, **odbcClose()**, **odbcCloseAll()** open/close connections to ODBC databases (*RODBC*)

**dbSendQuery** execute an SQL statement on a given database connection (*DBI*)

**dbConnect()**, **dbDisconnect()** create/close a connection to a DBMS (*DBI*)



## Packages

**RODBC** ODBC database access

**DBI** a database interface (DBI) between R and relational DBMS

**RMySQL** interface to the MySQL database

**RJDBC** access to databases through the JDBC interface

**RSQLite** SQLite interface for R

**ROracle** Oracle database interface (DBI) driver

**RpgSQL** DBI/RJDBC interface to PostgreSQL database

**RODM** interface to Oracle Data Mining

**xlsReadWrite** read and write Excel files

**WriteXLS** create Excel 2003 (XLS) files from data frames

## Big Data

### Functions

**as.ffdf()** coerce a dataframe to an *ffdf* (*ff*)

**read.table.ffdf()**, **read.csv.ffdf()** read data from a flat file to an *ffdf* object (*ff*)

**write.table.ffdf()**, **write.csv.ffdf()** write an *ffdf* object to a flat file (*ff*)

**ffdfappend()** append a dataframe or an *ffdf* to an existing *ffdf* (*ffdf*)

**big.matrix()** create a standard *big.matrix*, which is constrained to available RAM (*bimemory*)

**read.big.matrix()** create a *big.matrix* by reading from an ASCII file (*bimemory*)

**write.big.matrix()** write a *big.matrix* to a file (*bimemory*)

**filebacked.big.matrix()** create a file-backed *big.matrix*, which may exceed available RAM by using hard drive space (*bimemory*)

**mwhich()** expanded “which”-like functionality (*bimemory*)

### Packages

**ff** memory-efficient storage of large data on disk and fast access functions

**ffbase** basic statistical functions for package *ff*

**filehash** a simple key-value database for handling large data

**g.data** create and maintain delayed-data packages

**BufferedMatrix** a matrix data storage object held in temporary files

**biglm** regression for data too large to fit in memory

**bimemory** manage massive matrices with shared memory and memory-mapped files

**biganalytics** extend the *bimemory* package with various analytics

**bigtabulate** table-, tapply-, and split-like functionality for matrix and *big.matrix* objects

## Parallel Computing

### Functions

**foreach(...)** %dopar% looping in parallel (*foreach*)

**registerDoSEQ()**, **registerDoSNOW()**, **registerDoMC()** register respectively the sequential, SNOW and multicore parallel backend with the *foreach* package (*foreach*, *doSNOW*, *doMC*)

**sfInit()**, **sfStop()** initialize and stop the cluster (*snowfall*)

**sfLapply()**, **sfSapply()**, **sfApply()** parallel versions of *lapply()*, *sapply()*, *apply()* (*snowfall*)

### Packages

**multicore** parallel processing of R code on machines with multiple cores or CPUs

**snow** simple parallel computing in R

**snowfall** usability wrapper around *snow* for easier development of parallel R programs

**snowFT** extension of *snow* supporting fault tolerant and reproducible applications, and easy-to-use parallel programming

**Rmpi** interface (Wrapper) to MPI (Message-Passing Interface)

**rpvm** R interface to PVM (Parallel Virtual Machine)

**nws** provide coordination and parallel execution facilities

**foreach** foreach looping construct for R

**doMC** foreach parallel adaptor for the *multicore* package

**doSNOW** foreach parallel adaptor for the *snow* package

**doMPI** foreach parallel adaptor for the *Rmpi* package

**doParallel** foreach parallel adaptor for the *multicore* package

**doRNG** generic reproducible parallel backend for foreach Loops

**GridR** execute functions on remote hosts, clusters or grids

**fork** R functions for handling multiple processes

## Generating Reports

**Sweave()** mixing text and R/S code for automatic report generation (*utils*)

**knitr** a general-purpose package for dynamic report generation in R

**R2HTML** making HTML reports

**R2PPT** generating Microsoft PowerPoint presentations

## Interface to Weka

Package **RWeka** is an R interface to Weka, and enables to use the following Weka functions in R.

Association rules:

**Apriori()**, **Tertius()**

Regression and classification:

**LinearRegression()**, **Logistic()**, **SMO()**

Lazy classifiers:

**IBk()**, **LBR()**

Meta classifiers:

**AdaBoostM1()**, **Bagging()**, **LogitBoost()**,

**MultiBoostAB()**, **Stacking()**,

**CostSensitiveClassifier()**

Rule classifiers:

**JRip()**, **M5Rules()**, **OneR()**, **PART()**

Regression and classification trees:

**J48()**, **LMT()**, **M5P()**, **DecisionStump()**

Clustering:

**Cobweb()**, **FarthestFirst()**, **SimpleKMeans()**,

**XMeans()**, **DBScan()**

Filters:

**Normalize()**, **Discretize()**

Word stemmers:

**IteratedLovinsStemmer()**, **LovinsStemmer()**

Tokenizers:

**AlphabeticTokenizer()**, **NGramTokenizer()**,

**WordTokenizer()**

## Editors/GUIs

**Tinn-R** a free GUI for R language and environment

**RStudio** a free integrated development environment (IDE) for R

**rattle** graphical user interface for data mining in R

**Rpad** workbook-style, web-based interface to R

**RPNG** graphical user interface (GUI) for interactive R analysis sessions

**gWidgets** a toolkit-independent API for building interactive GUIs

**Red-R** An open source visual programming GUI interface for R

**R AnalyticFlow** a software which enables data analysis by drawing analysis flowcharts

**lattice** a graphical user interface for exploratory visualisation

## Other R Reference Cards

*R Reference Card*, by Tom Short

[http://rpad.googlecode.com/svn-history/r76/Rpad\\_homepage/](http://rpad.googlecode.com/svn-history/r76/Rpad_homepage/R-refcard.pdf)

[R-refcard.pdf](#) or

<http://cran.r-project.org/doc/contrib/Short-refcard.pdf>

*R Reference Card*, by Jonathan Baron

<http://cran.r-project.org/doc/contrib/refcard.pdf>

*R Functions for Regression Analysis*, by Vito Ricci

<http://cran.r-project.org/doc/contrib/Ricci-refcard-regression.pdf>

*R Functions for Time Series Analysis*, by Vito Ricci

<http://cran.r-project.org/doc/contrib/Ricci-refcard-ts.pdf>

## RDataMining Website, Package, Twitter & Groups

RDataMining Website: <http://www.rdatamining.com>

Group on LinkedIn: <http://group.rdatamining.com>

Group on Google: <http://group2.rdatamining.com>

Twitter: <http://twitter.com/rdatamining>

RDataMining Package: <http://www.rdatamining.com/package>  
<http://package.rdatamining.com>