

R Reference Card

Business Intelligence

July 16, 2014

This table provides a short overview of (most) R commands presented during the exercise sessions. More information on both the commands and their arguments can be retrieved from the R help pages. These are accessed via `?command` or `help(command)`. If you the corresponding package is not loaded, use instead `??command`. A more comprehensive overview is offered by the R reference card, found at <http://cran.r-project.org/doc/contrib/Short-refcard.pdf>.

Command	Syntax/Example	Description
Basic Operations		
library	<code>library(x)</code>	Loads a package with additional functions
setwd	<code>setwd("path")</code>	Changes working directory to a path
c	<code>c(x, y)</code>	Concatenate objects to a vector
seq	<code>seq(from=start, to=end, by=steps)</code>	Generates a sequence with defined steps
:	<code>start:end</code>	Sequence from start to end, possibly descending
length	<code>length(x)</code>	Length of a vector
sum	<code>sum(x)</code>	Sum of all values in x
min, max	<code>min(x), max(x)</code>	Minimum/maximum of vector x
mean	<code>mean(x)</code>	Average value of vector x
sqrt	<code>sqrt(x)</code>	Square root of x
round	<code>round(x, digits=0)</code>	Rounds x to the chosen number of significant digits
log10	<code>log10(x)</code>	Logarithm to base 10
sin, cos, tan	<code>sin(x), cos(x), tan(x)</code>	Sine/cosine/tangent of x
abs	<code>abs(x)</code>	Absolute value of x
rep	<code>rep(x, times=n)</code>	Replicates number/vector n times
"	<code>"text"</code>	Denote a string (i. e. a sequence of characters)
TRUE, FALSE	<code>TRUE, FALSE</code>	Literals denoting boolean values
Data Collection		
names	<code>names(DataFrame)</code>	Column names of a data frame
colnames	<code>colnames(DataFrame)</code>	Column names of a data frame
head	<code>head(DataFrame)</code>	Shows the first 6 rows of a data frame
\$	<code>DataFrame\$Columnname</code>	Accessing specific columns of a data frame
cbind, rbind	<code>cbind(x), rbind(x)</code>	Binding vectors column-wise or row-wise to form matrix
as.data.frame	<code>as.data.frame(x)</code>	Coerces an object to a data frame
read.csv	<code>read.csv("file.csv", sep="," , ← header=TRUE)</code>	Reading a csv file as specified
read.csv	<code>read.csv(file.choose(), sep="," , ← header=TRUE)</code>	Reading a csv file according to manual selection
str	<code>str(x)</code>	Gives the structure of an object
dim	<code>dim(x)</code>	Dimensions of a data frame x
ncol, nrow	<code>ncol(DataFrame), nrow(DataFrame)</code>	Number of columns/rows in data frame or matrix
numeric	<code>numeric(n)</code>	Create vector of length n filled with zeros
attributes	<code>attributes(x)</code>	Access an objects attributes

Indexing		
<p>[]</p> <p>which</p> <p>which.min</p>	<p><code>x[n]</code></p> <p><code>x[-n]</code></p> <p><code>x[1:n]</code></p> <p><code>x[c(1,3,4)]</code></p> <p><code>x[x>3]</code></p> <p><code>mat[i,j]</code></p> <p><code>mat[,j]</code></p> <p><code>DataFrame[["Columnname"]]</code></p> <p><code>which(x>3)</code></p> <p><code>which(d)</code></p>	<p>n-th element of vector x</p> <p>Vector x without the n-th element</p> <p>Elements 1 to n from vector x</p> <p>First, third and fourth element of vector x</p> <p>Selects elements fulfilling condition, i. e. greater than 3</p> <p>Matrix element in i-th row and j-th column</p> <p>j-th column of matrix</p> <p>Accessing specific columns of a data frame</p> <p>Indices of element fulfilling condition</p> <p>Index of element with minimum value</p>
Data Visualization		
<p>barplot</p> <p>pie</p> <p>hist</p> <p>plot</p> <p>text</p> <p>boxplot</p> <p>lines</p> <p>qqnorm</p> <p>qqline</p> <p>abline</p> <p>window</p> <p>ggplot</p>	<p><code>barplot(x)</code></p> <p><code>pie(x)</code></p> <p><code>hist(x)</code></p> <p><code>hist(x, freq=FALSE, breaks=b)</code></p> <p><code>plot(x,y)</code></p> <p><code>plot(x, y, type='l')</code></p> <p><code>text(x, y, data, position)</code></p> <p><code>boxplot(x)</code></p> <p><code>lines(x)</code></p> <p><code>qqnorm(x)</code></p> <p><code>qqline(x)</code></p> <p><code>abline(a=intercept, b=slope)</code></p> <p><code>window(x, start=start, end=end)</code></p> <p><code>gplot(...)</code></p>	<p>Barplot, absolute frequencies</p> <p>Draws a pie chart</p> <p>Frequency histogram with fixed ranges</p> <p>Density histogram with variable ranges</p> <p>Two-dimensional point plot</p> <p>Two-dimensional line plot</p> <p>Adding text to the axes</p> <p>Produces box and whisker plots</p> <p>Connecting points of a plot</p> <p>Gives a quantile-quantile plot of the data</p> <p>Adds a qq line of a normal distribution</p> <p>Adds a straight line to a plot</p> <p>Defines a time window from start to end</p> <p>Fancy graphics with additional options</p>
Data Analysis		
<p>var</p> <p>sd</p> <p>table</p> <p>median</p> <p>quantile</p> <p>summary</p> <p>runif</p> <p>sample</p> <p>rnorm</p> <p>dnorm</p> <p>cor</p> <p>cor.test</p> <p>lm</p> <p>lm</p> <p>bptest</p> <p>acf</p> <p>filter</p> <p>dwtest</p> <p>vif</p> <p>kappa</p>	<p><code>var(x)</code></p> <p><code>sd(x)</code></p> <p><code>table(x)</code></p> <p><code>median(x)</code></p> <p><code>quantile(x)</code></p> <p><code>summary(x)</code></p> <p><code>runif(x)</code></p> <p><code>sample(from:to, n, replace=TRUE)</code></p> <p><code>rnorm(x)</code></p> <p><code>dnorm(x, mean, sigma)</code></p> <p><code>cor(x,y)</code></p> <p><code>cor.test(x,y)</code></p> <p><code>lm(y ~ x)</code></p> <p><code>lm(y ~ ., data=d)</code></p> <p><code>bptest(y ~ x)</code></p> <p><code>acf(x)</code></p> <p><code>filter(x, filter, method)</code></p> <p><code>dwtest(y ~ x)</code></p> <p><code>vif(model)</code></p> <p><code>kappa(matrix)</code></p>	<p>Gives variance of x</p> <p>Gives the standard deviation of x</p> <p>Table of absolute frequencies of factor levels</p> <p>Computes sample median</p> <p>Computes sample quantile</p> <p>Gives descriptive statistics</p> <p>Generates random number from a uniform distribution</p> <p>Takes a sample of the specified size out of an object</p> <p>Generates random numbers from a normal distribution</p> <p>Density of standard normal distribution</p> <p>Pearson correlation coefficient of x and y</p> <p>Test for correlation of x and y</p> <p>Linear regression model</p> <p>Linear regression model using all columns as specified in data object</p> <p>Breusch-Pagan test for homoskedasticity</p> <p>Computes and plots the autocorrelation function</p> <p>Applies a linear filter to a time series</p> <p>Durbin-Watson test for autocorrelation</p> <p>Variance inflation factor of the model</p> <p>Computes condition number of a matrix</p>

AIC, BIC	<code>AIC(model)</code> , <code>BIC(model)</code>	Calculates the Akaike/Bayesian Information Criterion
logLik	<code>logLik(x)[1]</code>	Extracts a log-likelihood value
predict	<code>predict(model, newdata=nd)</code>	Predict values with a fitted model
Data Mining		
knn	<code>knn(train, test, response, k=1)</code>	<i>k</i> -Nearest-Neighbor
rpart	<code>rpart(formula, data, method="class")</code>	Building a decision tree
printcp	<code>printcp(tree)</code>	Prints a table of optimal pruning based on a complexity parameter
plotcp	<code>plotcp(tree)</code>	Plots a complexity parameter table of a tree-object
prune	<code>prune(tree, cp=...)</code>	Prune decision tree by complexity parameter
table	<code>table(pred=p, true=c)</code>	Confusion matrix with predicted values <i>p</i> and actual classes <i>c</i>
kmeans	<code>kmeans(d, k, nstart=...)</code>	<i>k</i> -means clustering on a data matrix (with <i>n</i> runs)
svm	<code>svm(formula, data, type)</code>	Train a support vector machine
svm	<code>svm(formula, data, ← type="C-classification")</code>	Use support vector machine for classification
svm	<code>svm(formula, data, ← type="eps-regression")</code>	Use support vector regression
as.party	<code>as.party(tree)</code>	Coerces fitted tree to the object party, needed for drawing trees nicely
nnet	<code>nnet(formula, data, weights, ← size, rang, decay, maxit)</code>	Train a single-hidden-layer neural network
attributes	<code>attributes(pred)\$decision.values</code>	Extract decision values from predicted values
plot.roc	<code>plot.roc(true.classes, ← decision.values)</code>	Plots a ROC curve (true classes can't be a factor)
Text Mining		
Corpus	<code>Corpus(source, readerControl)</code>	Creates a corpus of a collection of text documents
inspect	<code>inspect(Corpus)</code>	Display detailed information on a corpus or a term-document matrix
tm_map	<code>tm_map(Corpus, transformation)</code> <code>tm_map(corpus, PlainTextDocument)</code> <code>tm_map(corpus, stripWhitespace)</code> <code>tm_map(corpus, tolower)</code> <code>tm_map(corpus, removeNumbers)</code> <code>tm_map(corpus, removePunctuation)</code> <code>tm_map(corpus, removeWords)</code> <code>tm_map(corpus, stemDocument)</code>	Apply transformation functions to corpora Transform into plain text form Eliminate white-spaces Conversion to lower case Remove numbers Remove punctuation Remove words, e. g. stopwords Stemming the document
stopwords	<code>stopwords("language")</code>	Different kind of stopwords in various languages
tm_filter	<code>tm_filter(corpus, pattern="word")</code>	Full text search
tdm	<code>TermDocumentMatrix(corpus)</code>	Create a term document matrix <i>tdm</i>
findFreqTerms	<code>findFreqTerms(tdm, n)</code>	Retrieve words from a <i>tdm</i> that occurs at least <i>n</i> times
findAssocs	<code>findAssocs(tdm, "term", x)</code>	Find associations with at least <i>x</i> correlation for the term
removeSparseTerms	<code>removeSparseTerms(tdm, x)</code>	Remove terms that occur in less than <i>x</i> % of documents
kmeans	<code>kmeans(t(tdm), k, nstart=n)</code>	<i>k</i> -means clustering on texts (with <i>n</i> runs)
Necessary Packages		
car	<code>library(car)</code>	Companion to applied regression
caret	<code>library(caret)</code>	Machine learning interfaces (including datasets)

class	<code>library(class)</code>	Functions for classification
e1071	<code>library(e1071)</code>	Needed for support vector machine
ggplot2	<code>library(ggplot2)</code>	Fancy plots
ISLR	<code>library(ISLR)</code>	Needed for support vector regression
lmtest	<code>library(lmtest)</code>	Testing linear regression models
MASS	<code>library(MASS)</code>	Datasets
nnet	<code>library(nnet)</code>	Training neural networks
party	<code>library(party)</code>	Recursive partitioning
partykit	<code>library(partykit)</code>	Recursive partitioning
rpart	<code>library(rpart)</code>	Recursive partitioning and regression trees
pROC	<code>library(pROC)</code>	ROC Curve
tm	<code>library(tm)</code>	Text mining