Exam

Linköpings Universitet, Institutionen för datavetenskap, Statistik

---

Course:	732A44 Advanced Programming in R
Date and time:	2014-05-08, 8-12.
Teacher:	Patrik Waldmann
Material:	The Art of R Programming (Matloff). The book should be free from notes and comments, but can contain underlined lines and indicators for quickly finding chapters/sections etc. Slides from all lectures, the lab reports as well as the four papers on data mining in R can be included in a personal folder.
Grades:	Maximum is 20 points. A=19-20 points. B=17-18 points. C=12-16 points. D=10-11 points. E=8-9 points. F=0-7 points.

Solutions should be written in a Word file, do not include the questions.

The R code should be complete and readable code, possible to run by copying directly into a script.

Comment directly in the code whenever something needs to be explained or discussed.

Follow the instructions carefully.

1 Data and programming structure (5p)

a). Below follows explanations but the code is missing. Your task is to fill in code that performs according to explanations.

Creates a sample matrix named mymat with 100 columns and 200 rows. The entries are filled with random values from Beta(1,1) distribution.

Computes the mean and standard deviation of each row in mymat using a for loop. Stores the results in a list called myresfor with components mean and sd.

Computes the mean and standard deviation of each row in mymat using the apply function. Stores the results in a list called myresappl with components mean and sd.

Confirms that the values from both calculations are identical for the mean and standard deviation, respectively. (2p)

b). Use the while function to construct a loop that evaluates if a random number from the Unif(0,1) distribution is larger than 0.8, and if the difference between the current value and the prior value is larger than 0.6. The loop should evaluate new random numbers until both of the criteria is met (the random values should be generated within the loop). The current value and the difference should be printed to the screen at each iteration together with the text: **'The current value is'** the current value **'and the difference is'** the difference. The values should be printed with 4 decimals. Only present the code. (3p)

2 Simulation and graphics (3p)

a). Define two trivariate (X,Y and Z) covariance matrices: the first with variance 1 and covariance 0.8, and the second with variance 1 and covariance 0.1. Simulate 1000 draws from the trivariate normal distribution based on Cholesky decomposition of the covariance matrices and univariate normal random deviates for each of the covariance matrices. Plot the results as two 3D scatter plots factored on each of the data sets (i.e Trellis plot). The sub-plots should have titles 'High covariance' and 'Low covariance', and correct labels on the axes. (3p)

3 Strings and input/output (3p)

a). Write a function that reads a letter "`myletter''` between A and Z from the keyboard, evaluates if the letter is a DNA nucleotide (i.e. either A,C,G or T), and finally prints the results to the screen saying "`myletter''` is a DNA nucleotide", or "`myletter` is NOT a DNA nucleotide". `myletter` should be replaced with its letter in the output. (3p)

4 Interfacing and parallel computations (4p)

a). Write code that reads two matrices (both of dimension n x p) and performs p two-sided t-tests (i.e. tests if the column means are different between the two matrices) in parallel. The code should be so flexible that it detects the number of cores automatically, and the output should be a vector with p-values. (2p)

b). Compare the computation time of your code in a) on a matrix of size n=1000 and p=1000 with the timing from the ordinary apply function (performing t-tests). The timing should be over the complete parallel code. Provide code and timing output. (2p)

5 Data mining (5p)

a). Your task is to employ `mboost` to fit a glm model to the data in `baseball.dat`. Salary should be set up as response variable and all other variables as predictors (no transformation needed). Center the variables and use 10-fold cross validation to find the optimal stopping criterion. Provide a table with the coefficients of all the selected variables as well as a plot from the cross validation. (2p)

b). The same data set should now be used for fitting a generalized additive model with `mboost`. Use 10-fold cross validation also here and provide a table of the selected variables as well as function plots of each selected variable. (3p)