Examination Advanced R Programming

Linköpings Universitet, IDA, Statistik

Course code and name:	732A94 Advanced R Programming
Date:	2019/02/28, 8–12
Teacher:	Krzysztof Bartoszek
Allowed aids:	The extra material is included in the zip file **exam_material.zip**
Grades:	A= $[18-20]$ points B= $[16-18)$ points C= $[12-16)$ points D= $[9-12)$ points E= $[8-9)$ points F= $[0-8)$ points
Instructions:	Write your answers in an R script file named **[your exam account].R** 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. There are **THREE** problems (with sub–questions) to solve.

Problem 1 (5p)

a) (3p) What R package is required to run code in parallel? Provide an example of a function from that package that is used to run parallel code and discuss its most important parameters.
b) (2p) What sort of R functions are directly parallelizable, that is with minimum code changes?

Problem 2 (10p)

READ THE WHOLE QUESTION BEFORE STARTING TO IMPLEMENT! Remember that your functions should **ALWAYS** check for correctness of user input!

a) (3p) In this task you should use object oriented programming in S3 or RC to write code that stores clusters of points.

The first task is to implement a function called `build_data_object()` that returns an object corresponding to the collection of observations. Each observation is just a single number. The `build_data_object()` constructor function should take one numeric argument defining a parameter of the distance function: `p`. The object should have data structures that contain the clusters, the center of the cluster, number of points in the cluster, the observations, to which cluster does each one belong to and the distance of the observation from the center of its cluster. Each observation and cluster has to possess a **unique ID** that you have assign **yourself**. The object should also have field/slot that contains the number of clusters and number of points. The object should have a L_p distance function that calculates the distance between two points according to the formula:

$$d(x, y) = |x - y|^p.$$

```
## S3 and RC call to build_data_object() function
data_obj <- build_data_object(p=2
```

b) (3p) Now implement a function called `create_cluster()` to add a cluster to your data object. The function should take one arguments, the center of the cluster (a single number). This user provided data should then be remembered in the data structure that stores the clusters. Run the function a number of times (say 10 times) to add clusters. Remember to update the field counting the number of clusters. The centre should be unique for each cluster, i.e. two different clusters cannot have the same centre. If a user creates such a situation the function should react appropriately. Choose some way of generating the clusters' centres.

```
## S3 and RC call
data_obj <-create_cluster(data_obj,1)
## if using RC you may also call in this way
data_obj$(1)
```

c) (3p) Now implement a functions called `add_observation()` that adds an observation to your object. The function should take as an argument the numeric value corresponding to the observation. The function should check to which cluster's centre the observation is closest to, assign the observation to that cluster and appropriately modify all the fields and data structures in the object. Run the function a number of times (say 50 times) to add points. Use a random number generator from some distribution of your choice to generate the observations. Provide some example calls to your code.

d) (1p) Implement a plot **OR (NO NEED TO DO BOTH!)** print function to present your stored data.

```
## two possibilities of plotting calls
plot(data_obj); data_obj$plot()
## two possibilities of printing calls
print(data_obj); data_obj$print()
```

Problem 3 (5p)

a) (3p) In probability, statistics and combinatorics a key value to calculate is the binomial coefficient, $b : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ defined as

$$b(n,k) = \frac{n!}{(n-k)!k!} = \frac{1 \cdot 2 \cdot 3 \ldots n}{(1 \cdot 2 \cdot 3 \ldots (n-k))\,(1 \cdot 2 \cdot 3 \ldots k)}$$

for $n > k > 0$ and $b(n,0) = b(n,n) = 1$.

Write your own function that takes as its input two integers and returns the value of the binomial coefficient.. Do not forget that your function should check for correctness of input and react appropriately.

b) (1p) What is the complexity of your solution in terms of the number of required multiplication operations?

c) (1p) The same can be achieved using R's `choose()` function, i.e. the value of the binomial coefficient will be calculated as `choose(n,k)`. Implement a unit test that compares your implementation with direct R calculation.