Lunds University
Mathematical Sciences/Mathematical Statistics 2015

Computer lab 1

1 Introduction to the lab

The computer lab consists of the parts

- General introduction to the package R
- The law of large numbers (LLN).
- The central limit theorem (CLT).

Try to answer all questions. Do not pass a part until you understand it.

2 The computer package R

R is a computer package for statistical calculations, and is the most used package in Mathematical Statistics. Newly developed methods are often implemented with short delay in R, and uploaded on the R server. R is free (open source and GNU licence), so there is no licence fee, and is available for download and installation to UNIX/Linux machines, Windows and MacOS on http://www.r-project.org/. R is related to the package S-plus, which however there is a licence fee for.

Start up R. This is done on command line in UNIX environment by typing R, or in Windows or MacOS by starting the GUI'n for R (click on the appropriate icon)

An excellent introduction to R can be found the homepage http://www.r-project.org/ under "Manuals" (on the left) and under the links under "An introduction to R", try for instance the html-version. Note that this is very thorough and a lot of the commands assume a good knowledge of the basics in Mathematical Statistics; however Sections 2 and 5 are general and good to have a look at.

A useful command in R is "?" or "help". Do

```
help(sum)
```

or

```
?sum
```

Om you do not know what your are looking for in detail you can do an extended search with the command "help.search" on what your are looking for as a string (so as a sequence of characters). Do

```
help.search("sum")
help.search("normal")
```

You can use R as a usual calculator. Do

```
1+2
34.7/23
```

Variables are defined as:

```
a<-2
```

Elementary calculations can be done as

```
a<-2
b<-13
c<-(a+b)/(1-2*b)
```

There is a large number of mathematical functions built in into R. Try

```
log(10)
cos(0)
```

Vectors are defined as

```
x <- c(1, 2, 7.1, 4.4, -23.7)
```

Elementwise operations are done as

```
x*x
2*x+1
```

and unitary operations on vectors are done as

```
sin(x)
exp(x)
sum(x)
```

The scalar product between two vectors can be calculated as either of

```
y<-c(2,3,2,1,5)
sum(x*y)
x%*%y
```

logical operations are defined elementwise, using the same rules as above, for instance

```
x>=y
x[x>=y]
```

Sometimes one would like to transform the logical values TRUE and FALSE to "1" and "0". This is done with the command "as.double", for instance

```
ind<-as.double(x>=y)
```

Note that this can be used to define indicator variables, and also random such.

Matrices can be defined with the function "array", try

```
array(c(x,y),dim=c(5,2))
array(c(x,y),dim=c(2,5))
array(1:10,dim=c(2,5))
array(1:5,dim=c(2,5))
array(1:5,dim=c(5,2))
```

Indexing in matrices can be done for instance by

```
z<-array(c(x,y),dim=c(5,2))
z[2,1]
z[,1]
z[,2]
z[1,]
```

The transpose of a matrix can be done with the function "t()"

```
z
t(z)
```

In R one can apply functions on matrices just as for regular numbers

```
sin(z)
exp(z)
```

Operations on matrices are element wise operations, for instance

```
z*z
z^2
```

Matrix multiplication is done with the command "$\%*\%$", for instance

```
z%*%t(z)
t(z)%*%z
```

and multiplication between matrices and vectors as

```
x%*%z
t(z)%*%x
```

Matrix inversion is done with the function "solve" (see also the help function for solution of linear system of equations):

```
u<-t(z)%*%z
solve(u)
```

3 The Law of Large Numbers (LLN)

The LLN is one of the most important theoretical results in Mathematical Statistics. The LLN says that if $X_1, X_2, \ldots$ is a sequence of independent identically distributed random variables with expectation $E(X_i) = m$, then for every intervall $(m-\epsilon, m+\epsilon)$ the average $\bar{X}_n$ is lying in this interval with high probability if n is large. This part of the computer lab will illustrate this.

1. We generate 10 000 $Un(0,1)$ distributed r.v.'s and plot the sequence of averages $\bar{X}_n = n^{-1}\sum_{i=1}^n X_i$ as a function of n.

   ```
   n<-10000
   x<-runif(n)
   x_n<-cumsum(x)/(1:n)
   plot(1:n,x_n)
   ```

   What does the graph seem to converge towards? Calculate (theoretically, with paper and pen) $E(X_1)$. Use the LLN to explain what you see.

2. We generate 10 000 $Un(0,1)$ distributed r.v.'s and form $Z_n = n^{-1}\sum_{i=1}^n X_i^2$, and plot Z_n against n.

   ```
   n<-10000
   x<-runif(n,0,1)
   z_n<-cumsum(x^2)/(1:n)
   plot((1:n),z_n)
   ```

   What does the graph seem to converge towards. Calculate (theoretically, with paper and pen) $E(X_1^2)$. Use the LLN to explain what you see. note that this gives us a way to calculate a (stochastic) approximation to $E(g(X))$ for many different functions g.

4 The Central Limit Theorem (CLT)

The CLT is another very important theorem in Mathematical Statistics. The CLT says that if $X_1, X_2, \ldots$ is a sequence of independent identically distributed r.v.'s, then their sum $S_n = \sum_{i=1}^n X_i$ and average $\bar{X}_n = n^{-1}S_n$ are approximately distributed as a Normal r.v. if n is large. We will next illustrate this. To do this we will use an estimate of the distribution function F of a random variable X which is called the empirical distribution function

$$F_n(x) \;=\; \frac{1}{n}\sum_{i=1}^n 1\{X_i \le x\}.$$

Calculate theoretically $E(F_n(x))$? Use the LLN to motivate why F_n can be used as an approximation of F. In what sense does (using the LLN) "F_n converge towards F"?

Do "help" on "qqnorm" before you proceed.

1. Generate $n = 100$ r.v.'s $X_1, \ldots, X_{100}$ that are exponentially distributed with expectation $E(X_1) = 1$ (use the function "rexp") and form their average $\bar{X}_n$. Do this $m = 200$ times so that you get 200 averages $\bar{X}_n^1, \ldots, \bar{X}_n^{200}$. Do a normal probability plot of these:

```
n<-100
m<-200
x<-rexp(n*m)
x2<-matrix(x,m,n)
x_n<-apply(x2,1,mean)
qqnorm(x_n)
qqline(x_n)
```

   Do this with the values $n = 10, 20, 200, 1000$, what is the result? Vary m, what is the result? Are your results in line with what the CLT says?

2. Do the same thing with $Un(0, 1)$ distributed r.v.'s, vary $n = 5, 10, 50$.

```
n<-100
m<-200
x<-runif(n*m)
x2<-matrix(x,m,n)
x_n<-apply(x2,1,mean)
qqnorm(x_n)
qqline(x_n)
```

   What is the result? Do you see a difference against the previous exercise?

3. Generate $n = 100$ $Bin(nn, pp)$ distributed r.v.'s with $nn = 3$ and $pp = 0.5$ and form their average $\bar{X}_n$. Do this $m = 200$ times. Make a normal probability plot of these

```
n<-100
m<-200
nn<-3
pp<-0.5
x<-rbinom(n*m,nn,pp)
x2<-matrix(x,m,n)
x_n<-apply(x2,1,mean)
qqnorm(x_n)
qqline(x_n)
```

   Vary $n = 10, 50, 200$, what is the result? What happens if you change $nn = 100$, and vary $n = 10, 50, 200$? Can you use the CLT to describe the difference between $Bin(3, 0.5)$ and $Bin(100, 0.5)$?

4. Think of what would happen if you change the distribution in the previous exercise to $Bin(100, 0.1)$. Think first and make a computer run then.

End of Lab