

# Lab 3 - Computational Statistics

*Gustav Sternelöv*

*February, 18, 2016*

## Assignment 1

### 1.1-1.2

The code below shows the function written for selecting one city from the list of all cities. The probability for the respective city to be chosen is connected to the number of inhabitants in the city. Returned by the function is a number, *cityNum*, which is the row number for the selected city.

```
cityFunc <- function(data){
  data$prob <- data$Population / sum(data$Population)
  data$cumProbs <- cumsum(data$prob)
  unifV <- runif(1,0,1)
  dataNew <- subset(data, data$cumProbs >= unifV)
  cityNum <- which(data$cumProbs==min(dataNew$cumProbs))
  return(cityNum)
}
```

### 1.3-1.4

The following program is used for selecting 20 cities from the list. After each evaluation of the function is one city selected and this city is added to the new data frame *newData*. In the next step is the selected city erased from the original list of cities so it cannot be chosen two times. This procedure is repeated until the new data frame contains 20 cities.

```
selectData <- data1
newData <- as.data.frame(matrix(seq(20),nrow=20,ncol=3))
for(i in 1:20){
  set.seed(i)
  cityNum <- cityFunc(selectData)
  newData[i, ] <- selectData[cityNum, 1:3]
  selectData <- selectData[-cityNum, ]
}
```

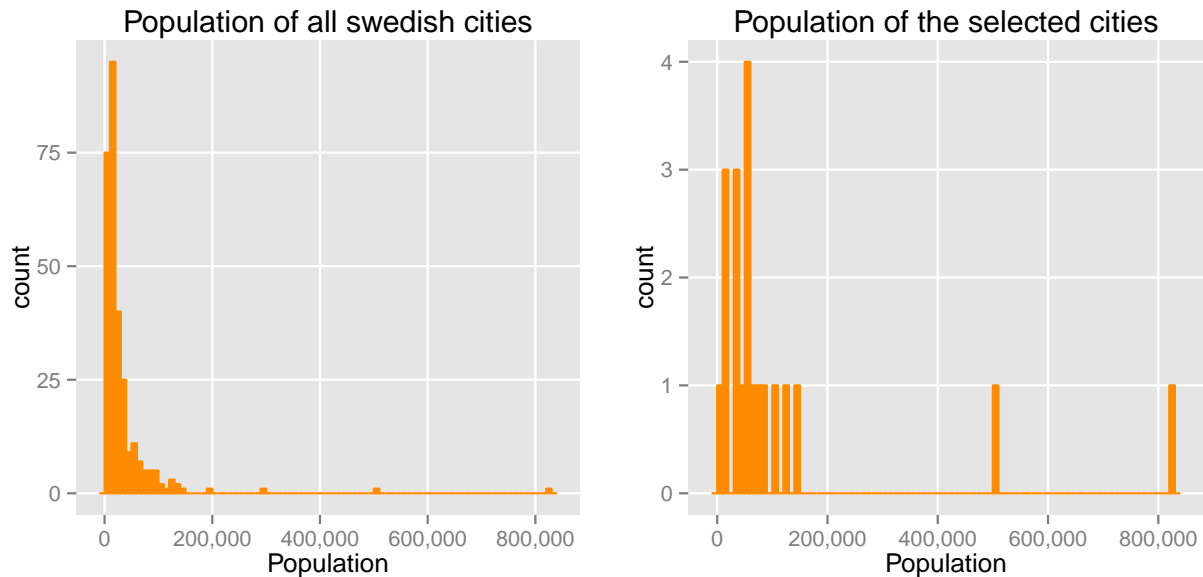
The names of the selected cities and the number of inhabitants in each city is presented in the table below. Among the cities that has been selected it can be noted that both Stockholm and Gothenburg, the two most populated swedish cities, are included. Of the ten cities with highest population are four included in the list. In general do the selected cities have a rather high number of inhabitants with a median population of 53601.5. The median population for all swedish cities is 15282, so the median population is 3.5 times as high for the selected cities.

##	City	Population
## 1	Katrineholm	32303
## 2	Södertälje	85270
## 3	Stockholm	829417

## 4	Göteborg	507330
## 5	Nyköping	51209
## 6	Falköping	31419
## 7	Luleå	73950
## 8	Lund	109147
## 9	Linköping	144690
## 10	Svedala	19625
## 11	Jönköping	126331
## 12	Norrtälje	55927
## 13	Forshaga	11401
## 14	Ödeshög	5314
## 15	Kungälv	40727
## 16	Uddevalla	51518
## 17	Håbo	19452
## 18	Falun	55685
## 19	Täby	63014
## 20	Sandviken	36978

## 1.5

The number of inhabitants in all swedish cities and the number of inhabitants in the selected cities are compared with the following histograms.



As mentioned in 1.4 it is easily concluded that the selected cities on average are bigger than the average swedish city. This is a result of the selection procedure described in 1.1-1.2. The probability for a city to be among the chosen cities is connected to its population since a higher number of inhabitants gives a higher probability for being selected. As a result of this selection procedure are the obtained histograms for the respective data sets rather dissimilar to each other.

## Assignment 2

### 2.1

For generating values from the double exponential distribution from the uniform[0,1] distribution according to the inverse CDF method, the following steps has to be performed.

To start, the pdf for the double exponential distribution is

$$\frac{\alpha}{2} \exp(-\alpha|x - \mu|)$$

From the pdf are two integrals computed, one when  $x < \mu$  and one when  $x \geq \mu$ . The first then is

$$\begin{aligned} \int_{-\infty}^x \frac{\alpha}{2} \exp(\alpha(x - \mu)) \\ = \frac{1}{2} \exp(\alpha(x - \mu)) \end{aligned}$$

And the second is

$$\begin{aligned} \int_x^{\infty} \frac{\alpha}{2} \exp(-\alpha(x - \mu)) \\ = 1 - \frac{1}{2} \exp(-\alpha(x - \mu)) \end{aligned}$$

Next step is to solve for x in the respective cases.

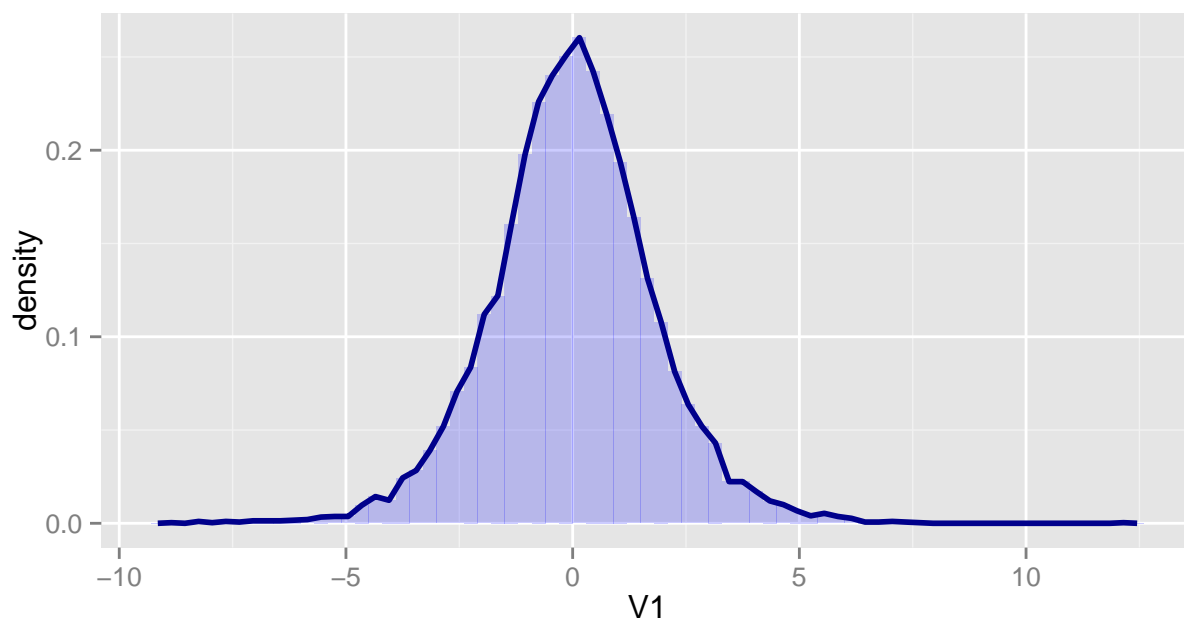
$$x = \ln(2U) + \mu$$

$$x = \mu - \ln(2 - 2U) * \frac{1}{\alpha}$$

The last step is to add together the results.

$$x = \ln(2U) + 2\mu - \ln(2 - 2U) * \frac{1}{\alpha}$$

The equation above can be used for generating values from the double exponential distribution by using values from the uniform[0,1] distribution. As an example is 10000 values generated from DE(0,1) with the inverse CDF method. The result is shown in the histogram below.



I believe this result is fairly reasonable.

It has relatively large tails, compared to a normal distribution, which is typical for values following a DE distribution.

It is centered around zero, which makes sense since the value of  $\mu$  specifies the median for the distribution.

.

.

. Should it not be even more narrow around zero? A higher peak around zero.

## 2.2

The following steps are performed in order to use the acceptance/rejection method with DE(0,1) as majorizing density to generate N(0,1) variables.

✕ First, the value of c is calculated.

This is done by using the Laplace distributed values from 2.1. The value of c must be higher than all ratios of  $f_y(x)$  and  $f_x(x)$  where  $f_y$  is the pdf for the double exponential distribution and  $f_x$  is the pdf for the normal distribution.

$$c > \frac{f_x(x)}{f_y(x)}$$

Then, the value of c is determined to be 1.315489.

✕ The first step of the acceptance/rejection is to generate Y.

This is done with the inverse CDF method explained in 2.1

✕ Then the generated Y values is inserted into  $f_y(Y)$  and  $f_x(Y)$ .

This is done by using the probability density function for  $f_y$ , the double exponential distribution, and  $f_x$ , the normal distribution.

✕ Generate a U[0,1] value.

✕ Test if Y should be accepted or rejected.

If the Y should be kept or not is decided with the following rule. If

$$U \leq \frac{f_x(Y)}{f_y(Y)}$$

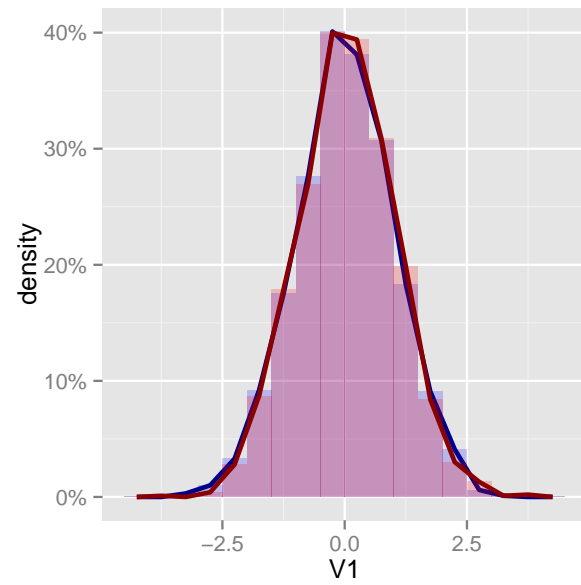
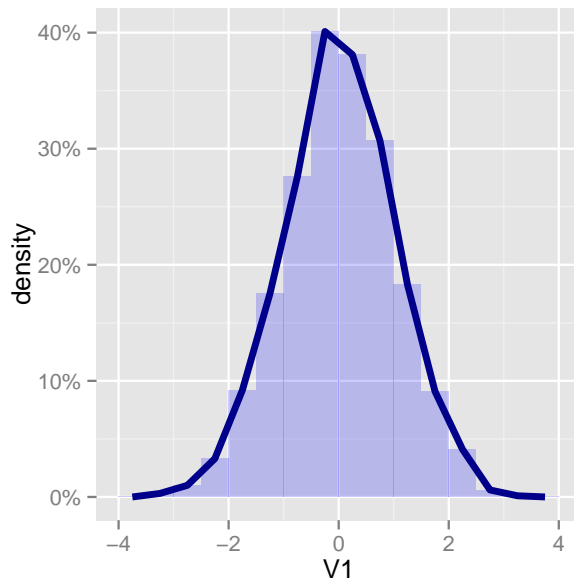
keep Y, else return to step one and generate a new Y value.

The implemented acception/rejection algorithm is tested and 2000 N(0,1) values are generated. In the histogram below are these values illustrated with the blue line and the red line is 2000 N(0,1) values generated with *rnorm* function.

The acceptance rate is  $\frac{1}{c}$ , hence the rejection rate is  $1 - \frac{1}{c}$ . The expected rejection rate then is

$$1 - \frac{1}{1.315489} = 0.24$$

The 2000 N(0,1) values generated with the acceptance/rejection method required 2604 Y values to be generated. So 604 values were rejected, resulting in an rejection rate of 0.23. In other words are the obtained rejection rate very close to the expected rejection rate. Hence, the chosen value for c seem to be reasonable. In the histogram below to the left are the values generated from the acception/rejection method plotted. In the histogram to the right are these values, the blue line, compared to 2000 N(0,1) values, the red line, generated with the *rnorm* function.



As can be seen in the histogram to the right are the obtained histograms very similar to each other. Both the acceptance/rejection algorithm and the *rmorm* function produces values that seem to be  $N(0,1)$  distributed.