Tekniska högskolan, LiU
Matematiska institutionen
Matematisk statistik

TAMS42: Probability and Statistics — computer lab

Instructions (R version)

- The lab is done by **R** which is free. Installation instructions are as follows. Go to

  https://cloud.r-project.org/

  Choose the correct version of **R** according to your operating systems (OS), download and install it. Then in principle you can now be able to interface with **R** using "RGui". To open "RGui", in Microsoft Windows OS one simply clicks "R i386 4.1.1" or "R x64 4.1.1" (not sure the names in other OS but should be similar). For example, in "RGui" R Console, you can write the commands:

  x=1;y=2

  then the command x+y will return you 3. "RGui" is a very basic (sometimes not convenient) interface to work with **R**, and many people prefer to use "RStudio" instead as it has many advantages (see for example the 4 advantages explained here: https://www.theanalysisfactor.com/the-advantages-of-rstudio/). Therefore, let us be stylish and use this interface "RStudio". To use it, one downloads the "Rstudio Desktop" free version from

  https://www.rstudio.com/products/rstudio/download/

- After it has been installed successfully, one can open it by clicking "Rstudio". There are two ways to write commands in "Rstudio" (exactly the same as "Matlab"):

  (i) directly in Console (**Enter** can validate the commands),

  (ii) in an **R Script** file (one select the commands and click **Run**).

  As it is much more convenient and flexible to adjust commands, we will write all commands in **R Script** files (again, exactly the same situation as in "Matlab"). The commands can be then saved as **.R** files.

Part 1 - Probability

Problem 1. Simulation of observations from a discrete distribution

In our lectures, we have often used the experiment of throwing a die to illustrate various calculations of probabilities. In this problem you should now use **R** to simulate such throws. Start **RStudio** and in the top-left corner click + to add an **R Script** file, then save it and name it as problem1.R. Now type the following commands in the R-file (note that anything after # is regarded as comment):

```
rm(list=ls()) # this is to clear all previous data
n=600 # n throws of the die
set.seed(1) # fix random generator seed (same random simulations each time)
throw=sample(1:6, n, replace=TRUE) # simulate n throws of the die
```

In order to compile/run the above codes, select all the codes and click **Run**. NOTE: if you just want to compile/run a part of the codes, then select the part and click **Run**.

Write `help("sample")` in the command window to see how `sample` works in **R**. We have 6 different outcomes: 1, 2, 3, 4, 5, 6. Now we want to know the frequency f_i of each outcome $i = 1, 2, \ldots, 6$. To do this, we write in R-file `sum(throw==i)` for each $i = 1, 2, \ldots, 6$, and **Run** it. What frequencies do you get?

Now we want to find the sample mean and sample standard deviation of these 600 throws. Write the following commands in the R-file and **Run** them:

```
sample_mean = mean(throw) # sample mean
sample_standard_deviation = sd(throw) # sample standard deviation
```

Compare these two (simulated) values with the theoretical values (namely μ and σ which you need to compute by hand).

Now please write all of your answers on the last page "Answer sheet". When you are done, you can close the R-file problem1.R and continue to Problem 2.

Problem 2. Normal approximation to Binomial

Use the same steps as in Problem 1 to create an R-file named problem2.R. In Lecture, we talked about how normal approximations to Binomial (and Poisson) work. In this problem we will use **R** to numerically see these approximations.

(i) For a Binomial random variable $Bin(n, p)$, normal approximation $N(np, np(1-p))$ works if $np \geq 10$ and $n(1-p) \geq 10$. Now we check what happens if these conditions are not satisfied. In problem2.R, write the commands to compute cdf of a Binomial $X \sim Bin(20, 0.3)$ at points $0, 1, \ldots, n$:

```
rm(list=ls()) # this is to clear all previous data
n = 20; p = 0.3 # values of paramters

bino_cdf = 0 # the cdf of a Binomial X
```

```
# commands below give values of the cdf of a Binomial X at 0 to n
for(i in 1:(n+1))
{
  bino_cdf[i] = pbinom(i-1, size=n, prob=p)
}
```

Now let us compute the corresponding Normal cdf at points at points $0, 1, \ldots, n$:

```
norm_cdf = 0 # the corresponding normal cdf
for(i in 1:(n+1))
{
  norm_cdf[i] = pnorm(i-1, n*p, sqrt(n*p*(1-p)))
}
```

What we want to do now is to compare the Binomial cdf and the corresponding Normal cdf at points $0, 1, \ldots, n$, and find the maximal absolute difference of these two cdfs at these points:

```
max(abs(bino_cdf-norm_cdf))
```

What is the maximal absolute difference?

(ii) Repeat the whole process described in (i) using $n = 50, n = 100$ and $n = 1000$. As n increases, does the maximal absolute difference decrease? Write down all the answers on the last page "Answer sheet".

Problem 3. Monte Carlo method

Use the same steps as in Problem 1 to create an R-file named problem3.R. Integrals that cannot be computed by hands may be computed approximately using numerical methods. One of these methods is the *Monte Carlo* method, which uses random variables.

Suppose we want to find $|I|$, the area of a region I, which is contained wholly within a rectangle $(a, b) \times (0, c)$ - the rectangle is formed by $x = a, x = b, y = 0$ and $y = c$. It is easy to generate randomly chosen points which are distributed uniformly within the rectangle $(a, b) \times (0, c)$. Suppose we generate n random points. Let f_n denote the number that have landed in the region I. Then one can intuitively see that $\frac{f_n}{n} \approx \frac{\text{area of } I}{\text{area of the rectangle}} = \frac{|I|}{(b-a)\,c}$. It then follows that

$$|I| \approx (b-a)\,c\,\frac{f_n}{n}$$

As n increases, the right hand hand side gives a better approximation of $|I|$. Now we consider a region I formed from the pdf of $N(0, 1)$ restricted on $(0, 0.5)$, and we want to estimate the area I of this region:

$$|I| = \int_0^{0.5} \frac{1}{\sqrt{2\pi}}\, e^{-x^2/2} dx,$$

namely, in the above setting, $a = 0$ and $b = 0.5$. Choose a value for c so that you have a rectangle which contains the region I (for example you can choose $c = 1$, why?)

(i) Once you have chosen c, draw a figure containing both the region I and the rectangle.

(ii) To estimate $|I|$, we use the *Monte Carlo* method. First, let $n = 20$ and we now generate n uniform random variables in the rectangle $(a, b) \times (0, c)$. To do this, write commands in the R-file:

```
rm(list=ls()) # this is to clear all previous data
a = 0; b = 0.5; c = 1; n = 20
set.seed(1) # fix random generator seed (same random simulations each time)
x=runif(n, min=0, max=b) # generate n uniform in the rectangle for x
y=runif(n, min=0, max=c) # generate n uniform in the rectangle for y
```

Now we want to see how many such uniform random points have landed in the region I. To this end, continue to write commands in the R-file:

```
y_curve=(1/sqrt(2*3.14))*exp(-x^2/2) # the normal curve
f_n=sum(y<y_curve) # count how many in the region I
area_I=(b-a)*c*f_n/n # compute the area of I
```

The output `area_I` is the estimated area of I. The (almost) accurate area of I can be found from the Normal table $|I| = P(0 < N(0,1) < 0.5)$. Compare the estimated area and the (almost) accurate area.

(iii) Repeat the whole process for $n = 50, n = 500$ and then for $n = 10000$. When n increases, is the estimated area getting closer to the (almost) accurate area?

Part 2 - Statistics

Problem 4. CI and HT for the difference of two population means

Use the same steps as in Problem 1 to create an R-file named problem4.R. We now generate 10 observations from $N(22, 2^2)$ and 12 observations from $N(16, 2^2)$, using `rnorm`. Write in the command window `help("rnorm")` to see how this works. Write in the R-file:

```
rm(list=ls()) # this is to clear all previous data
n = 10; m = 12; mu1 = 22; mu2 = 16; sigma = 2
set.seed(1) # fix random generator seed (same random simulations each time)
x = rnorm(n,mean=mu1,sd=sigma) # Generate n observations X
y = rnorm(m,mean=mu2,sd=sigma) # Generate m observations Y
```

We may think that these two sets of observations x and y are from two independent populations $X \sim N(\mu_1, \sigma^2)$ and $Y \sim N(\mu_2, \sigma^2)$ respectively (note that the two population variances are the same!). Our goal of this problem is to use **R** to (i) construct a 95% confidence interval of $\mu_1 - \mu_2$, and (ii) perform hypotheses test $H_0 : \mu_1 = \mu_2$ against $H_a : \mu_1 \neq \mu_2$. To this end, write in the R-file:

```
t.test(x,y,var.equal=TRUE)
```

where `var.equal=TRUE` means that two population variances are assumed to be equal (even though unknown). Again, you can write `help("t.test")` to get to know the meaning of it.

(i) The output of **R** gives the 95% confidence interval of $\mu_1 - \mu_2$. Can you also construct the 95% confidence interval of $\mu_1 - \mu_2$ by hand? (which should coincide with the output). You should need the two sample means and standard deviations which can be obtained using the commands in Problem 1. NOTE: When construct by hand, you need to pretend that μ_1, μ_2 and σ are unknown (the given values were used only for generation of data).

(ii) The output of **R** gives an answer whether or not H_0 should be rejected. Namely, one can see the value of the test statistic TS and the p-value. So, with $\alpha = 5\%$, what is your conclusion (reject H_0 or not)? Can you reach the same conclusion based on TS and C by hand? (TS has been given already from the output, and one needs to compute the rejection region C by hand).

Problem 5. CI using normal approximations

Now create an R-file named problem5.R. In this problem we will construct confidence intervals when the population is a Binomial random variable $Bin(n, p)$. In order to use normal approximation, it is required that n is large enough such that $np \geq 10$ and $n(1-p) \geq 10$. It is interesting to see what happens if n is not that large, so we will construct 95% confidence interval of p using normal approximations for both large and not so large n.

(i) We first generate 1000 samples (each sample size is 1) from $Bin(16, 0.3)$. Write in the R-file:

```
rm(list=ls()) # this is to clear all previous data
n = 16; p = 0.3
set.seed(1) # fix random generator seed (same random simulations each time)
x = rbinom(1000,n,p) # one can think of x as 1000 samples (each sample size is 1)
```

For each sample one can estimate p by using $\hat{p}$. So 1000 samples give us 1000 estimated $\hat{p}$. Write in R-file:

```
phat = x/n # these are 1000 estimated values of p
```

The corresponding 1000 (95%) confidence intervals are: write in the R-file

```
lower_lim = phat - 1.96*sqrt(phat*(1-phat)/n) # this is a vector with 1000 values
upper_lim = phat + 1.96*sqrt(phat*(1-phat)/n) # this is a vector with 1000 values
```

Now we have 1000 such 95% confidence intervals I_p=(`lower_lim`, `upper_lim`). Since it is 95%, there should be around 950 such intervals containing the real value $p = 0.3$, and around 50 not containing the real value. Now we count how many such intervals not containing $p = 0.3$: write in R-file:

```
missing = sum(lower_lim > p) + sum(upper_lim < p)
```

where '`missing`' gives you the number of intervals not containing the real value $p = 0.3$. Compare this number with the expected number 50 (is it far away from 50 or very close to 50? why?)

(ii) Repeat these procedures in (i) for a new population $Bin(80, 0.3)$, and compare again.

Problem 6. Simple linear regression

Now create an R-file named problem6.R. A geyser is a hot spring, which more or less regularly erupts. During an eruption, the water can spray high into the air. Old Faithful Geyser in Wyoming is one such source which has become a tourist attraction. Time between two consecutive eruptions is usually long, it is therefore interested in being able to predict the time until the next eruption. It is believed that this

time depends on the length of the previous eruption. To construct such a model we put

$$x = \text{ the length of the last eruption (unit:min) and}$$
$$y = \text{ time till the next eruption (unit:min).}$$

We will use the model

$$Y_j = \beta_0 + \beta_1 x_j + \varepsilon_j$$

where $\varepsilon_1, ..., \varepsilon_n$ are independent $N(0, \sigma^2)$. The sample is given as follows (copy x and y into the R-file)

```
x = c(4.4, 3.9, 4.0, 4.0, 3.5, 4.1, 2.3, 4.7, 1.7, 4.9, 1.7, 4.6, 3.4, 4.3, 1.7, 3.9,
      3.7, 3.1, 4.0, 1.8, 4.1, 1.8, 3.2, 1.9, 4.6, 2.0, 4.5, 3.9, 4.3, 2.3, 3.8, 1.9,
      4.6, 1.8, 4.7, 1.8, 4.6, 1.9, 3.5, 4.0, 3.7, 3.7, 4.3, 3.6, 3.8, 3.8, 3.8, 2.5,
      4.5, 4.1, 3.7, 3.8, 3.4, 4.0, 2.3, 4.4, 4.1, 4.3, 3.3, 2.0, 4.3, 2.9, 4.6, 1.9,
      3.6, 3.7, 3.7, 1.8, 4.6, 3.5, 4.0, 3.7, 1.7, 4.6, 1.7, 4.0, 1.8, 4.4, 1.9, 4.6,
      2.9, 3.5, 2.0, 4.3, 1.8, 4.1, 1.8, 4.7, 4.2, 3.9, 4.3, 1.8, 4.5, 2.0, 4.2, 4.4,
      4.1, 4.1, 4.0, 4.1, 2.7, 4.6, 1.9, 4.5, 2.0, 4.8, 4.1)

y = c(78, 74, 68, 76, 80, 84, 50, 93, 55, 76, 58, 74, 75, 80, 56, 80, 69, 57, 90, 42,
      91, 51, 79, 53, 82, 51, 76, 82, 84, 53, 86, 51, 85, 45, 88, 51, 80, 49, 82, 75,
      73, 67, 68, 86, 72, 75, 75, 66, 84, 70, 79, 60, 86, 71, 67, 81, 76, 83, 76, 55,
      73, 56, 83, 57, 71, 72, 77, 55, 75, 73, 70, 83, 50, 95, 51, 82, 54, 83, 51, 80,
      78, 81, 53, 89, 44, 78, 61, 73, 75, 73, 76, 55, 86, 48, 77, 73, 70, 88, 75, 83,
      61, 78, 61, 81, 51, 80, 79)
```

(i) We first plot y against x in order to see if there is a linear relation between them, write in the R-file:

```
plot(x,y)
```

We can even find the correlation coefficient $\rho_{X,Y}$ using the code in the R-file

```
correlation=cor(x,y)
```

Note that if $|\rho_{X,Y}| \approx 1$ then it means that x and y have linear relation.

(ii) Now we do linear regression using `lm`, write in the R-file:

```
regre=lm(y~x)
summary(regre)
```

What is the estimated regression line?

(iii) In the output of **R**, find the standard errors of the coefficients, namely, $s_{\hat{\beta}_0}$ and $s_{\hat{\beta}_1}$. (Note that in Lecture, we also use the notations $s\sqrt{h_{00}} = d(\hat{\beta}_0)$ and $s\sqrt{h_{11}} = d(\hat{\beta}_1)$ for $s_{\hat{\beta}_0}$ and $s_{\hat{\beta}_1}$).

(iv) With a significance level $\alpha = 0.01$, test the hypotheses

$$H_0 : \beta_1 = 0 \text{ against } H_1 : \beta_1 \neq 0$$

Is H_0 rejected? (Use TS and C to answer).

(v) It is assumed that the error terms $\varepsilon_j \sim N(0, \sigma^2)$, but is this really true? We will study this by looking at the residual plot and histogram. Write in the R-file:

```
res=regre$residuals # to get the residuals

plot(x,res) # plot of the residuals vs x

hist(res) # histogram of the residuals
```

The idea is: (a) In the plot of the residuals vs x, if there is no obvious pattern, then it is reasonable to say $\varepsilon_j \sim N(0, \sigma^2)$; but if there is an obvious pattern, then the error terms are not normal. (b) In the histogram, if the shape looks like a normal curve, then $\varepsilon_j \sim N(0, \sigma^2)$, otherwise the errors are not normal. Based on the residual plot and histogram, do you think $\varepsilon_j \sim N(0, \sigma^2)$?

Problem 7. Logistic regression

Now create an R-file named problem7.R. Copy the following x and y into the R-file:

```
x = c(41, 41, 42, 43, 54, 53, 57, 58, 63, 66, 67, 67, 67, 68, 69, 70, 70, 70, 70,
      72, 73, 75, 75, 76, 76, 78, 79, 81, 85, 86, 86, 88)

y = c(1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0,
      0, 0, 0, 0, 0, 0, 0)
```

The variable x represents the launch temperature, and y denotes incidence of failure of O-rings in 32 space shuttle launches prior to the Challenger disaster of 1986. It is noted that y can only take two values: $y = 1$ (failure) and $y = 0$ (success). From the above data it is suspected that space shuttle launch will be likely to fail ($y = 1$) if launch temperature is low (say $x \leq c$ for some threshold c). We now use **logistic regression to model such relation**. Namely let Y be a Bernoulli random variable with

$$P(Y = 1) = p(x), \quad P(Y = 0) = 1 - p(x),$$

where the failure probability $p = p(x)$ depends on the launch temperature x. For logistic regression, we assume that $p(x)$ depends on x via the logit function

$$p(x) = \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}}.$$

(i) Use **R** to find the estimated logit function

$$\hat{p}(x) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 x}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 x}}.$$

To do this, write in the R-file:

```
logregre=glm(y~x,family=binomial())

summary(logregre)
```

After **Run**, the output in **R** gives the estimated parameters $\hat{\beta}_0$ and $\hat{\beta}_1$: namely in the column **Estimate** the values of $\hat{\beta}_0$ and $\hat{\beta}_1$ are given. What is the estimated logit function?

(ii) Now we want to test, with a significance level $\alpha = 0.05$,

$$H_0 : \beta_1 = 0 \quad \text{against} \quad H_a : \beta_1 \neq 0.$$

If we reject H_0, then it means that $\beta_1 \neq 0$ which suggests that the launch temperature indeed affects the launch failure. Based on the data, do we reject H_0 ? Use TS and C to answer the question: since $n = 32$ is large, we have the random variable $\frac{\hat{\beta}_1 - \beta_1}{S_{\hat{\beta}_1}} \approx N(0,1)$, therefore $TS = \frac{\hat{\beta}_1 - 0}{s_{\hat{\beta}_1}}$ (in the output the column **Std. Error** gives us both $s_{\hat{\beta}_0}$ and $s_{\hat{\beta}_1}$), and $C = (-\infty, -z_{\alpha/2}) \cup (z_{\alpha/2}, +\infty)$.

(iii) Now if a new space shuttle is going to be launched at a temperature $x = 65$ (which is not a temperature in the data), then what is the estimated probability $\hat{p}(65)$ that the launch will fail? If $\hat{p}(65) \geq 0.5$, then we classify $y(65) = 1$ (failure), otherwise we classify it as success.

Answer sheet. Write down your name and personal number.

1) Daniel Andersson danan623

.....

2)

.....

3)

.....

Problem 1 **OK** .....

Frequencies are $f_1 =$ 106 $f_2 =$ 95 $f_3 =$ 90 $f_4 =$ 102 $f_5 =$ 109 $f_6 =$ 98

Sample mean $\bar{x} =$ 3,511667 theoretical mean $E(X_i) =$ 3,5 . $\mu = \sum_{i=1}^{N} x_i / N = 3,5$

Sample standard deviation $s =$ 1,724767; theoretical standard deviation $\sigma = \left(\sum_{i=1}^{N} (x_i - \mu)^2 / N\right)^{1/2}$ $\approx 1,7078$

Problem 2 **OK** .....

(i) $n = 20$: what is the maximal absolute difference .....

(ii) $n = 50$: what is the maximal absolute difference .....

$n = 100$: what is the maximal absolute difference .....

$n = 1000$: what is the maximal absolute difference .....

Problem 3 **OK** .....

(i) Draw a figure below containing both the region I and the rectangle:

(ii) $n = 20$: $f_n =$ ..... and the estimated area of I is .....

The (almost) accurate area of $|I| = P(0 < N(0,1) < 0.5)$ from Normal table is .....

(iii) $n = 50$: $f_n =$ ..... and the estimated area of I is .....

$n = 500$: $f_n =$ ..... and the estimated area of I is .....

$n = 10000$: $f_n =$ ..... and the estimated area of I is .....

Problem 4 **OK** .....

(i) **R**: output of 95% confidence interval of $\mu_1 - \mu_2$: .....

By hand: 95% confidence interval of $\mu_1 - \mu_2$: .....

which formula did you use : .....

(ii) **R**: reject H_0? : .....

By hand: $TS =$ ..... and $C =$ ....., reject H_0? : .....

Problem 5 **OK** .....

With such 1000 (95%) confidence intervals, around 50 such intervals do not containing the real value p.

(i) For $Bin(16, 0.3)$, find $np =$ ..... $n(1-p) =$ .....

The number of intervals not containing the real value p : .....

(ii) For $Bin(80, 0.3)$, find $np =$ ..... $n(1-p) =$ .....

The number of intervals not containing the real value p : .....

Problem 6 **OK** .....

(i) Does the plot of x and y look like a line?..... Correlation coefficient $\rho_{X,Y} =$ .....

(ii) What is the estimated regression line?.....

(iii) Standard errors of the coefficients: $s_{\hat{\beta}_0} =$ ..... and $s_{\hat{\beta}_1} =$ .....

(iv) $TS =$ ..... and $C =$ ....., reject H_0? : .....

(v) Do you think $\varepsilon_j \sim N(0, \sigma^2)$?.....

Problem 7 **OK** .....

(i) What is the estimated logit function?.....

(ii) $TS =$ ..... and $C =$ ....., reject H_0? : .....

(iii) $\hat{p}(65) =$ .....