Lund University
Mathematical Sciences HT2018

Computer Lab 2
Inference Theory

1 Introduction

This computer lab consists of five main parts:

1. Confidence intervals
2. Hypothesis testing
3. Use of normal approximation/limit distribution results, for the construction of tests and confidence intervals.
4. Parametric and nonparametric tests
5. Regression analysis.

Try to answer all questions. Ask when you do not understand. In the the first three parts (1-3) you will analyse only simulated/computer generated data, while in the last two parts (4-5) you will analyse real data.

2 Confidence intervals

This part of the computer lab consists of learning how to construct confidence intervals and to illustrate the covering probability.

1. We want to construct a confidence interval for the expectation μ in a Normal distribution with known variance. We therefore generate a sample of size $n = 200$, from a $N(\mu, 1)$ distribution. Choose a suitable value of μ yourself. (In the code below nedan we have chosen $\mu = 2$, you may choose another value, it is of no importance for the results of the lab).

```
n<-200
mu<-2
n<-200
x<-rnorm(n,mean=mu,sd=1)
```

The vector x now contains all data (the observations $x_1, \ldots, x_{200}$), and we will analyse this vector. We construct a 95% two-sided confidence interval.

```
mu.hat<-mean(x)
error <- qnorm(0.975)*1/sqrt(n)
left <- mu.hat-error
right <- mu.hat+error
```

Then the interval $(left, right)$ is 95% is a two-sided confidence interval for μ.

To check, and illustrate, if the theoretical covering probability 0.95 is correct we will do many times (N times) the above, so we will construct N confidence intervals, and see how many times the corresponding constructed confidence interval $(left, right)$ contains the correct value μ.

```
N<-1000
count<-0
for (i in (1:1000)){
       x<-rnorm(n,mean=mu,sd=1)
       mu.hat<-mean(x)
       error <- qnorm(0.975)*1/sqrt(n)
       left <- mu.hat-error
       right <- mu.hat+error
       count<-count+as.double(left<=mu & mu<=right)
       }
```

Then the variable *count* contains the number of times the interval contains the correct value of μ, and

```
       count/N
```

is the proportion.

(a) Does it seem as the covering proportion 0.95 is correct?

(b) Use the LLN to motivate what $count/N$ converges to, when some number (which?) converges to infinity.

Note that this is not typically not possible to do for real data situations: We can construct 1000 confidence intervals since we have simulated data. In real life we have *one single* data sample that we need to analyse.

2. We want to construct a confidence interval for the variance in a $N(\mu, \sigma^2)$-distribution. We generate 300 data points from a $N(\mu, \sigma^2)$-distribution.

```
n<-300
mu<-2
sigma<-2
x<-rnorm(n,mean=mu,sd=sigma)
```

The vector x now contains the data (the observations $x_1, \ldots, x_{300}$), and we will analyse this vector. We construct a 95% two-sided confidence interval.

```
sigma.hat<-sd(x)
left <- (n-1)*sigma.hat^2/qchisq(0.975,df=n-1)
right <- (n-1)*sigma.hat^2/qchisq(0.025,df=n-1)
```

Then the interval $(left, right)$ is 95% is a two-sided confidence interval for σ^2. To check if the theoretical covering probability is 0.95, we will construct N confidence intervals, and see how many times the corresponding constructed confidence interval $(left, right)$ contains the correct value σ^2.

```
N<-1000
count<-0
for (i in (1:1000)){
```

```
x<-rnorm(n,mean=mu,sd=sigma)
sigma2.hat<-sd(x)^2
left <- (n-1)*sigma2.hat/qchisq(0.975,df=n-1)
right <- (n-1)*sigma2.hat/qchisq(0.025,df=n-1)
count<-count+as.double(left<= sigma^2 & sigma^2<=right)
}
```

The variable *count* contains the number of times the interval contains the correct value of σ^2, and

```
count/N
```

is the proportion of times it does.

(a) Does it seem as the covering proportion 0.95 is correct?

(b) Use the LLN to motivate what $count/N$ converges to, when some number (which?) converges to infinity.

3. We want to construct a confidence interval for θ in a $Un(0, \theta)$-distribution. We generate 150 data points from a $Un(0, \theta)$ fördelning.

```
n<-300
teta<-2.5
x<-runif(n,min=0, max=teta)
```

The vector x now contains the data (the observations $x_1, \ldots, x_{150}$), and we will analyse this vector. We construct a 95% two-sided confidence interval for θ (recall Example 15.5 in the course book.)

```
teta.hat<-max(x)
left <- teta.hat/(0.975)^(1/n)
right <- teta.hat/(0.025)^(1/n)
```

Then $(left, right)$ is 95% is a two-sided confidence interval for θ. We will construct N confidence intervals, and see how many times the corresponding constructed confidence interval $(left, right)$ contains the correct value of θ.

```
N<-1000
count<-0
for (i in (1:1000)){
       x<-runif(n,min=0, max=teta)
       teta.hat<-max(x)
       left <- teta.hat/(0.975)^(1/n)
       right <- teta.hat/(0.025)^(1/n)
       count<-count+as.double(left<= teta & teta<=right)
       }
```

The variable *count* contains the number of times the interval contains the correct value of θ, and

```
count/N
```

is the proportion of times it does.

(a) Does it seem as the covering proportion 0.95 is correct?

(b) Use the LLN to motivate what $count/N$ converges to, when some number (which?) converges to infinity.

3 Hypothesis testing

This part of the computer lab consists of constructung tests for simple and composite null hypotheses, to illustrate the significance level and determine and plot the power function.

1. We test the hypotheses

$$\begin{aligned} H_0 : &\quad \theta = 0, \\ H_1 : &\quad \theta \neq 0, \end{aligned}$$

in a $N(\theta, 1)$ distribution. We generate a sample of 200 data points from the $N(0, 1)$ distribution.

```
n<-200
mu<-0
n<-200
x<-rnorm(n,mean=mu,sd=1)
```

The test statistic and the tests result are, for a test on level 0.05,

```
teta.hat<-mean(x)
test.stat<-teta.hat/(1/sqrt(n))
reject<-(abs(test.stat)>qnorm(0.975))
```

Then *reject* is a logical variable taking the value $TRUE$ or $FALSE$ depending on the result of the test. We illustrate the significance level by counting the number of times that we reject the null hypothesis when we do $N = 1000$ tests.

```
N<-1000
count<-0
for (i in (1:1000)){
       x<-rnorm(n,mean=mu,sd=1)
       teta.hat<-mean(x)
       test.stat<-teta.hat/(1/sqrt(n))
       teta.hat<-mean(x)
       test.stat<-teta.hat/(1/sqrt(n))
       reject<-(abs(test.stat)>qnorm(0.975))
       count<-count+as.double(reject)
       }
       count/N
```

(a) Does a significance value of 0.05 seem to be correct?

(b) Use the LLN to motivate what $count/N$ converges to, when some number (which?) converges to infinity.

We next will plot the power function $\pi(\theta)$ for the above test, on the interval $(-2, 2)$. Recall the derivation of the power function for a two-sided test of θ in $N(\theta, 1)$, in the lecture notes.

(a) We first use $n = 200$ as above

```
x.points<-(1:1000)/250-2
power<-1-(pnorm(qnorm(0.975)-x.points/(1/sqrt(n)))
         -pnorm(qnorm(0.025)-x.points/(1/sqrt(n))))
plot(x.points,power,type="l")
```

(b) How does the power of he test change when you have fewer data points, for instance $n = 25$?

```
n<-25
x.points<-(1:1000)/250-2
power<-1-(pnorm(qnorm(0.975)-x.points/(1/sqrt(n)))
          -pnorm(qnorm(0.025)-x.points/(1/sqrt(n))))
plot(x.points,power,type="l")
```

(c) What happens when you have many data points, say $n = 1000$?

```
n<-1000
x.points<-(1:1000)/250-2
power<-1-(pnorm(qnorm(0.975)-x.points/(1/sqrt(n)))
         -pnorm(qnorm(0.025)-x.points/(1/sqrt(n))))
plot(x.points,power,type="l")
```

Explain your results.

2. We want to construct a test of the hypotheses

$$\begin{aligned} H_0 : &\quad \theta \leq 2, \\ H_1 : &\quad \theta > 2 \end{aligned}$$

in a $Un(0, \theta)$-distribution. We generate 300 data points from a $Un(0, \theta)$-distribution. Refer to the lecture notes for testing in and the power function of a Uniform distribution.

```
n<-300
teta<-2
x<-runif(n,min=0, max=teta)
```

The test statistic and the result of the test are

```
teta.hat<-max(x)
test.stat<- teta.hat/teta
reject<-(test.stat>(0.95)^(1/n))
```

We count the number of times that we reject the null hypothesis, when we perform $N = 1000$ tests

```
N<-1000
count<-0
for (i in (1:1000)){
       x<-runif(n,min=0, max=teta)
       teta.hat<-max(x)
       test.stat<- teta.hat/teta
       reject<-(test.stat>(0.95)^(1/n))
       count<-count+as.double(reject)
       }
       count/N
```

(a) Does a significance value of 0.05 seem to be correct?

(b) Use the LLN to motivate what $count/N$ converges to, when some number (which?) converges to infinity.

Determine (the theoretical) power function for this test (and check that the code below is correct). We would like to plot the power function. Do this for a a some values of n, choose yourself interesting values for n.

```
x.points<-(1:1000)/200
n<-10
power<-1-(punif((2/x.points)*(0.95)^(1/n)))^n
plot(x.points,power,type="l")
```

(a) How does the power function change when the number of observations change?

(b) What is the power functions value for $\theta = 2$?

(c) What is the power functions value for $\theta < 2$? Explain why.

4 The use of Normal approximation in testing and confidence interval construction

We will in this part construct a confidence interval for a functional $E(g(X))$ which is linear in F, when data come from the (unknown) distribution F. For this we will use the CLT and Slutsky's theorem. Convince yourself of how the statements of these two results are used in the code below

1. We generate 200 data points from some distribution, below we have used the $Exp(1)$ distribution, you may use this or choose any other distribution.

```
n<-200
x<-rexp(n)
```

We want to construct a confidence interval for the functional $\theta = E(e^X)$ (this is the moment generating function $\psi(t) = E(e^{tX})$ evaluated in $t = 1$). Note that this is a suggestion of a linear functional, where we have put $g(u) = e^u$, you may use another functional if you wish. The code below uses the plug-in estimators of θ and of $Var(\theta)$. Convince yourself that you understand how.

```
teta.hat<-mean(exp(x))
sigma.hat<-sd(exp(x))
error<-qnorm(0.975)*sigma.hat/sqrt(n)
left<-teta.hat-error
right<-teta.hat+error
```

Then $(left, right)$ is a confidence interval for θ with the approximate confidence grade 0.95.

(a) Calculate the theoretical value of θ for your chosen distribution and functional. (Note that this might not be possible for all choices, if that happens, choose some other distribution and/or functional).

(b) Make a simulation study with $N = 1000$ data samples, analogously to previously, to check what the confidence intervals covering probability is.

(c) For a fixed number $n = 200$, what does $count/N$ converge to? Use the LLN to motivate your statement.

(d) (Harder question!) If we let n instead vary: what does the theoretical covering probability of a test such as above converge to, when $n \to \infty$? Use the CLT and Slutsky's theorem to motivate your statement.

5 Real data

The data that you are supposed to analyse in this part of the lab come from the Clinical Research Center (CRC) at the University Hospital in Malmö (UMAS). The data are real data and are data that researchers at CRC have done (part of) their research on. Thus a main goal of this lab is to

illustrate the types of analyses that one can do as a research scientist at a high quality research institute.

Note that there are no "right answers" to the questions below. You are supposed to analyse the material. You can of course do this in a more or less clever way. The lab instructions tell you what you should think of when you perform the analyses, but again there are no "right answers" to the research questions. And you can possibly find a new connection or result that previously was not known!

5.1 The data material

Data consists of $n = 4547$ individuals that have been followed until either (i) they die (in a heart disease) or until (ii) they leave the study, for some reason. For each individual there are measurements on a number of phenotyopes

```
T2D
FASTINGGLUCOSE
CHOL
TRIGL
HDLCHOL
LDL
FASTINGINSULIN
BMI
WH
SEX
SMOKING
PHYSACTIVITY
```

There are also, for each individual, measurements on 7 genotypes. These genes have actual names, known to the researchers at CRC, but for ethical purposes they are for us coded as gene 1, gene 2, ..., gene 7. In the data material we have they are labeled as

```
g1, g2, g3, g4, g5, g6, g7
```

The individuals are identified by two identifiers:

```
PATIENT
FAMILY
```

The identifier PATIENT is a unique identifier for each individual, while the identifier FAMILY can be shared by several individuals (that belong to the family identified by the variable FAMILY).

5.2 Load data into R

To load the data into R's data working memory: Do

```
dat<-read.table("datafil.txt")
```

You can list all variables for the data by doing

```
names(dat)
```

6 Nonparametric tests and estimators

In this part you are supposed to get a picture of the distributions for the different phenotypes and do tests for if there is a difference between those distributions.

6.1 Nonparametric estimates

The empirical distribution function (ecdf) is an estimator of the true and unknown distribution function. To calculate the empirical distribution function, for for instance BMI, do

```
plot(ecdf(dat$BMI))
```

If you want to look at the ecdf for BMI for the group that has the riskgenotype $g1$ (corresponding to the value $g1 = 1$), do

```
plot(ecdf(dat$BMI[dat$g1==1]))
```

If you want to see the ecdf for BMI for the two groups that have and that have not the risk genotype (corresponding to $g1 = 1$ and $g1 = 0$) in the same figure, do

```
plot(ecdf(dat$BMI[dat$g1==0]),xlim=c(15,60))
par(new=TRUE,col="red")
plot(ecdf(dat$BMI[dat$g1==1]),xlim=c(15,60))
```

Do this for some interesting groups. Some suggestions for possible group division are and how you divide are: You can for instance divide into gender (variable SEX, 1 means man, 2 means WOMAN), whether the individual is a smoker or not (variable SMOKING), those that have type II diabetes versus those that have not (T2D). Other interesting phenotypes apart from BMI that are continuous and that you can study the distribution for and distributional differences between groups for are CHOL, TRIGL, HDLCHOL, LDL, FASTINGINSULIN.

6.2 Nonparametric tests

The above gives you estimates of the distribution functions and the plotted figures can give you an indication of whether there are differences of the distributions between groups.

To make a formal test of differences between groups you can use a two-sample Kolmogorov-Smirnov test (R code function is *ks.test*). Do

```
plot(ecdf(dat$HDL[dat$SMOKING==0]),xlim=c(0,4))
par(new=TRUE,col=''red")
plot(ecdf(dat$HDL[dat$SMOKING==1]),xlim=c(0,4))
ks.test(dat$HDL[dat$SMOKING==1],dat$HDL[dat$SMOKING==0])
```

Do the figures and the formal test agree?

You can use the ecdf to estimate quantiles, by taking the inverse of the ecdf at a fixed point. More convenient and what amounts to the same thing is to calculate the empirical quantiles. Convince yourself that you understand the equivalence of this and ask the teacher if it is not clear to you. As an example, if you want to estimate the lower 10% quantile för HDL in the group of smokers do

```
quantile(dat$HDL[dat$SMOKING==1], 0.1, na.rm=TRUE)
```

You can use a one-sample Kolmogorov-Smirnov test to check if data are Normal distributed.

```
ks.test(dat$HDL[dat$SMOKING==1],"pnorm")
plot(ecdf(dat$HDL[dat$SMOKING==1]),xlim=c(0,4))
t<-seq(0,4,by=0.01)
m<-mean(dat$HDL[dat$SMOKING==1],na.rm=TRUE)
v<-var(dat$HDL[dat$SMOKING==1],na.rm=TRUE)
par(new=TRUE,col="blue")
plot(t,pnorm(t,m,sqrt(v)),xlim=c(0,4))
```

A better way to graphically see if there is a difference to the Normal distribution is to use a qq-plot.

```
qqnorm(dat$HDL[dat$SMOKING==1])
```

Use the help function on "qqnorm", and convince yourself that you understand what a qq-plot does. Ask if you don't understand.

7 Parametric tests

One can use for instance a t-test to test for whether there is a difference between the expectations for a covariate in different groups. Note that even if we see, as above, that the distributions are not Gaussian, since the estimates of the expectations are averages, and since an average of i.i.d. data is approximately Gaussian by the CLT, the t-test will still give (approximately) correct p-values.

Note however that the parametric estimates are not as informative for the (whole) distribution, since we only estimate the expectation in the distribution, and note also that parametric tests are not as sensitive for differences, since we only test for differences between the expectations in the distributions.

To do a t-test, do

```
t.test(dat$HDL[dat$SMOKING==1],dat$HDL[dat$SMOKING==0])
```

Try to do some more analyses. Do you get the same results as in the previous section, using non-parametric tests?

8 Linear regerssion

8.1 Univariate linear regression

We will use two variables as response variables, BMI and WH, and try to find what other variables might influence them, and try to estimate or describe that influence.

A univariate linear regression model can be fitted to the data as

```
fit<-lm(BMI~FASTINGGLUCOSE,data=dat)
summary(fit)
```

Give an interpretation of the results! Do help on *lm*. Try using other explanatory variables. To graphically see if the model fits data one can plot the residuals

```
plot(fit$fit,fit$res)
```

What should the residuals look like? Ask the teacher what you can do if they do not look look like they should. To see if the residuals are Normal distributed, you can do a qq-plot

```
qqnorm(fit$res)
```

Formal test of if the model fits the data is given by the F-statistic in

```
summary(fit)
```

Ask if you do not understand.

8.2 Multivariate linear regression

A multivariate regression model can be fitted to the data (using the least squares method), for instance using the covariates FASTINGGLUCOSE, CHOL, TRIGL, HDLCHOL, LDL and FASTINGINSULIN as explanatory variables, by

```
fit<-lm(BMI~FASTINGGLUCOSE+CHOL+TRIGL+HDLCHOL+LDL+FASTINGINSULIN,data=dat)
summary(fit)
```

Interpret the results! If you compare this model with a model with only FASTINGGLUCOSE as explanatory variable, you will find that FASTINGGLUCOSE is not any more significant in the multivariate regression model. Why?

The question of which covariates you should end up using in a multivariate model, is known as a model selection problem. To do model selection in a regression model it is convenient to use the commands *add*1 and *drop*1. One of the standard methods for that is to do so called stepwise backwards elimination of a "large" model, as long as you can. In the example above one would do

```
fit<-lm(BMI~FASTINGGLUCOSE+CHOL+TRIGL+HDLCHOL+LDL+FASTINGINSULIN,data=dat)
drop1(fit,test="Chisq")
fit<-lm(BMI~CHOL+TRIGL+HDLCHOL+LDL+FASTINGINSULIN,data=dat)
drop1(fit,test="Chisq")
etc
```

1. Use the above method to find a model for BMI. Start with a multivariate model that uses *all* variables as explanatory and eliminate them one at a time, until it is no longer possible.
2. Do the same thing (i.e. find a good model) for WH.
3. Find separate models for BMI (and for WH), for men and for women. What are your conclusions? To do an analysis on a subset of all data, for instance for only males, one can use the code

```
fit<-lm(BMI~FASTINGGLUCOSE+CHOL+TRIGL+HDLCHOL
          +LDL+FASTINGINSULIN,data=dat,subset=SEX==1)
```

A way to incorporate a varible that you believe "should be included" but that was not significant, is to make a (non-linear) transformation of the variable. For instance one could dichotomise a continuous variable. Try to do this for FASTINGGLUCOSE, by dichotomising for instance at it's median

```
fast.m<-quantile(dat$FASTINGGLUCOSE,0.5,na.rm=TRUE)
dat$ny.fast<-as.real(dat$FASTINGGLUCOSE>fast.m)
fit<-lm(BMI~ny.fast+CHOL+TRIGL+HDLCHOL+LDL+FASTINGINSULIN,data=dat)
summary(fit)
```

What is your conclusion?

9 Nonparametric estimation of the probability density function

Load the package *KernSmooth* into R's program memory

```
library(KernSmooth)
```

Do help on the function *bkde*. Choose one of the variables that you could be interested in estimating the density for, and do this, for instance for BMI the code is

```
fit<-bkde(dat$BMI[!is.na(dat$BMI)])
plot(fit,type="l")
```

Try to change the bandwidth to some different values.

```
fit<-bkde(dat$BMI[!is.na(dat$BMI)],band=1)
plot(fit,type="l")
fit<-bkde(dat$BMI[!is.na(dat$BMI)],band=0.5)
plot(fit,type="l")
fit<-bkde(dat$BMI[!is.na(dat$BMI)],band=0.3)
plot(fit,type="l")
fit<-bkde(dat$BMI[!is.na(dat$BMI)],band=0.1)
plot(fit,type="l")
fit<-bkde(dat$BMI[!is.na(dat$BMI)],band=0.05)
plot(fit,type="l")
```

What is your explanation to what you see?

Try to plot the kernel estimator of the density (for some choice of bandwidth) against a Gaussian density (for instance with values for the expectation and the variance estimated from the data), for a comparison. Explain what you see. Is this a formal test, and why not? Explain where in the lab you did the formal test.

The End