Applied statistics - R Code

Barcelona School of Management

Prepared for exam and review by Daniëlle Kotter

Table of content

Basic commands:

Basic packages:

```
  "prob",
  "data.table",
  "distrEx",
  "LaplacesDemon",
  "formattable",
  "kableExtra",
  "knitr",
  "TeachingDemos",
  "dplyr",
  "dbplyr",
  "tidyverse",
  "Hmisc",
  "psych",
  "samplingbook",
  "swirl",
  "ggplot2",
  "swirl",
  "snpar",
  "BSDA",
  "actuar",
  "readxl",
  "stargazer"
```

Sample mean, standard deviation

```
mean(variable)
sd(variable)
```

Removes values NA in a data set:

```
mean(variable, na.rm = TRUE)
sd(variable, na.rm = TRUE)
```

Weighted mean & standard deviation

Package: "HMISC".

```
weightedmean <- Xbar = wtd.mean(x,y)
weightedsd <- SQRT(Wtd.var(X,Y))/sqrt(n)
```

Variance

```
var(x)
```

Tables frames & Matrixes

```
matrix(c(1,2,3,4,5,6,7,8), nrow = 4, byrow = TRUE) <- organized by row
matrix(c(1,2,3,4,5,6,7,8), ncol = 4, byrow = FALSE) <- organized by col
data.frame(Column1 = c(1,2,3,4,5), Column2 = c(1,2,3,4,5))
data.table(Column1 = c(1,2,3,4,5), Column2 = c(1,2,3,4,5))

as.table(matrix(c(1,2,3,4,5,6,7,8), nrow = 4))
```

```
rbind(data, newvariable)
cbind(data, newvariable)

rownames(datatable) <- c()
colnames(datatable) <- c()
```

Other:

```
round(value, 2) <- two decimals
as.numeric(value)
rep(5,5) <- repeats 5, 5 times
percent(value) <- presents 0.25 as -> 25.00%
describe(variable)
fivenum(variable)
summary(variable)
str(variable) <- explains the variable
```

Read excel

```
library(readxl)
data <- read.xls("data.xlsx", stringsAsFactors = TRUE)
```

Mathmatical values

```
$\mu$ <- Population mean
$\sigma$ <- Population sd
$\bar{x}$ <- Sample mean
${e}$ <- Standard error
$\pi$ <- pie
$\ge$ <- Bigger than
$\le$ <- Smaller than
```

Charts templates

Pie chart

```
data <- data.frame(
  group = Absolute
  value = Relative
)

blank_theme <- theme_minimal()+
    theme(
        axis.title.x = element_blank(),
        axis.title.y = element_blank(),
        panel.border = element_blank(),
        panel.grid = element_blank(),
        axis.ticks = element_blank(),
        plot.title = element_text(size = 10, face = "bold")
    )

ggplot(data, aes(x="",y = value, fill = group)) +
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y", start = 0) +
  scale_fill_manual("Legendname:",
                    values = paletteDani)+
    blank_theme  +
    theme(axis.text.x=element_blank())+
  labs(title = "Title",
         x = "variablX",
         y = "variableY"
         )
```

Bar chart

```
data <- data.frame(
  Factor = freqtable$Factor,
  Frequency = freqtable$Absolute
)

ggplot(data, aes(x = Frequency, y = LivingSituation)) +
  geom_bar(stat = "identity", fill="#69b3a2", color="#e9ecef") +
  theme(legend.position="none")
```

Histogram

```
ggplot(data = data, aes(variable) ) +
    geom_histogram(fill="#69b3a2", color="#e9ecef", alpha=0.9) +
    ggtitle("Title") +
      xlab("variablex") +
      ylab("variabley") +
    theme(plot.title = element_text(size = 11))
```

Boxplot

```
ggplot(data=Data, aes(x="", y=Variable, fill="")) +
    geom_boxplot(fill="#69b3a2", outlier.colour="red", outlier.shape=8,
    theme_ipsum() +
    theme(
      legend.position="none",
      plot.title = element_text(size=12)
      ) +
    ggtitle("Title") +
    theme(plot.title = element_text(hjust = 0.5)) +
        xlab("")+
        ylab("")
```

Scatter plot

```
 ggplot(data, aes(y = variabley, x = variablex)) +
  geom_point(size=2) +
  geom_smooth(method="lm", fill = NA, color="#69b3a2", fullrange=TRUE, f
   theme(plot.title = element_text(hjust = 0.5)) +
        labs(title = "Title",
         y = "yname",
         x = "xname"
         )
```

Scatter plot with dummies

```
ggplot(data, aes(y = variableY, x = dummy, colour=factor(dummy))) +
  geom_point(size=2) +
  geom_smooth(method="lm", fill = NA, fullrange=TRUE,  formula = y ~ x)
    theme(plot.title = element_text(hjust = 0.5)) +
    scale_colour_manual(name="Legendtitle",
        labels=c("value1", "value2"),values = c("#69b3a2", "#F6726A"))+
    labs(title = "Title",
         y = "Yname",
         x = "Xname"
         )
```

Scatter plot with two variable + dummy

Only difference from above is the first sentence:

```
ggplot(data, aes(y = variableY, x = variableX, colour=factor(dummy))) +
```

Residual plot

```
residual.plot(fitted(model),resid(model),sigma.hat(model), main="Residua
```

Arrange charts next to eachother on a page

```
grid.arrange(chart1, chart2, nrow=1, widths=c(0.9,1))
```

Topic 1 - Probability & Statistical inference

Package = "prob".

```
out <- c(“Red”,“White”,“Black”,“Blue”,“Green”)
freq <- c(1,2,3,4,5)
s <- probspace(out,probs=freq)
```

If you toss two fair coins, what is the probability of two heads?

```
space <- tosscoin(2,makespace=TRUE)
p <- Prob(space, toss1=="H" & toss2=="H")
```

When two dice are thrown, what is the probability of a 3 followed by a 5?

```
space <- rolldie(2, makespace = TRUE)
p <- Prob(space, X1 == 3 & (X2 == 5) )
```

Sampling from an urn with or without replacement. 3 balls and sample size of 2:

```
urnsamples(1:3, size = 2, replace = TRUE, ordered = TRUE)
urnsamples(1:3, size = 2, replace = FALSE, ordered = TRUE)
urnsamples(1:3, size = 2, replace = FALSE, ordered = FALSE)
urnsamples(1:3, size = 2, replace = TRUE, ordered = FALSE)
```

Bayes Theorem

Unconditional probability:

P(S) and P(NS) Success or no success

```
prS <- c(0.4, 0.6)
```

Conditional probability:

P(P I S) and P(P I NS) Predicted given it is successful
Predicted given it is not successful

```
prNS <- c(0.6, 0.2)
```

Bayes prob, posterior probabilities
P(S | P) & P(NS | P)

```
BayesTheorem(prS, prNS)
```

Topic 2 - Discrete probability

Uniform discrete probability distribution

1. Sample space with a set probability. Size = amount of tries
2. Density function: Individual probability. F.E. Getting a 4
3. Cumulatative density: Uniform for a certain value distribution. F.E. 4 or less. 4 or more? 1-punif 3
4. Inverse cumulative density: Uniform for a certain probability (up until a certain value). F.E. up to 25% of the tries

```
1. sample(p, size=n, replace=TRUE)
2. dunif(X, min = a, max = b)
3. punif(X, min=0, max=6)
4. qunif(X, min=0, max=6)
```

Default = # or less. For # or more do: 1-probability of # or less

Binomial distribution

1. Binomial for a specific value for a certain sample. F.E. 2 from the sample are successful.
2. Binomial for a certain distribution of the sample. F.E. At most 2 in the sample are successful. Or 5 or more.
3. Binomial for a certain percentage of the sample. F.E. 25% of the sample has x value or less.
4. Difference between two binomial values. F.E. Prob there are between 4 and 5 of the trials successful.

```
1. dbinom(x, size = n, prob = y)
2. pbinom(x, size = n, prob =y)
3. qbinom(p, size = n, prob =y)
4. diff(pbinom(c(X,Y), size = n, prob =y)
```

Default = # or less (left area of the distribution). For # or more do: 1-probability of # or less

Poisson distribution

Expected value = n * p = LAMDA

1. Poisson for a certain value. Lambda = n*p. F.E. Prob of having a 5
2. Poisson for a certain value distribution. F.E. Prob of having less than 5. More than 5? = 1- Ppois(4, lambda)
3. Poisson for a certain probability to capture a certain value. F.E. Poisson value for 25%.

```
1. dpois(x,lambda)
2. ppois(x, lambda)
3. qpois(x,lambda)
```

Default = # or less (left area of the distribution). For # or more do: 1-probability of # or less

Topic 3 - The normal distribution

Empirical rule

For all normal distributions: 68-95-99.7 rule

99.7% of observations are located between: -3 mu and 3
95% of observations are located between: -2 mu 2
68% of observations are located between: -1 mu 1

Normal distribution

Z-value

```
z <- (x-mean)/sd
```

1. Normal distribution for a certain proportion. Pi = population proportion mean%.
2. Normal distribution for a certain value distribution. F.E. Prob of value above 5. FALSE Prob less than 9. TRUE
3. Normal distribution for a certain probability to capture a certain value. F.E. Value that is given at 25% point.
4. Difference between two values on the normal distribution. F.E. between 5 and 10.

```
1. pnorm(X, pi, sd, lower.tail = FALSE)
2. pnorm(X, mean = mean, sd = sd, lower.tail = FALSE)
3. qnorm(p, mean = mean, sd = sd, lower.tail = FALSE)
4. diff(pnorm(c(X,Y), mean = mean, sd = sd, lower.tail = FALSE)
```

lower.tail = TRUE: The area of the left side of the slope
lower.tail = FALSE: The area of the right side of the slope **Confidence interval for normal distribution**

```
z.test(x, sd=sigma)
binconf(x = x, n = n) <- proportions
t.test(variable) <- t-distribution for conf.inv
```

Plotting the normal distribution

"With mean = 3 and standard deviation = 7
Limits: mean +/- 3 * standard deviation = 3*7 = 21 Lower limit = 3 – 21 = -18
Upper limit = 3 + 21 = 24"

Example:

```
x <- seq(15, 45, length=50)
y <- dnorm(x, 30, 5)
plot(x,y,type="l",lwd=2,col="black")

x <- seq(15,35,length=100)
y <- dnorm(x, 30,5 )
polygon(c(15,x,35),c(0,y,0), density = c(15, 35), col = "black")

p <- pnorm(35, mean = 30, sd = 5,lower.tail = TRUE)
text(0,0.15,"68%")
```

Binomial

It will be possible to use the Normal distribution as an approximation to the Binomial if: n is large and p > 0.1

1. Density function (individual probability).
2. Cumulative density (between certain values).
3. Difference between two binomial values
4. Inverse cumulative density. For a certain prob.

```
1. dbinom(x, mean, sd, lower.tail = FALSE)
2. pbinom(x, mean, sd, lower.tail = FALSE)
3. diff(pbinom(c(X,Y), mean = mean, sd = sd, lower.tail = FALSE)
4. qbinom(p, mean, sd, lower.tail = FALSE)
```

Topic 4 - Samples, estimation & confidence intervals

The standard error of the sampling distribution of the mean

```
se <- sigma / sqrt(n)
```

Probability sample

1. To find the probability that X is larger than mu
2. To find the probability that X is smaller than mu

```
p <- pnorm(X, mu, se, lower.tail = TRUE)
p <- pnorm(X, mu, se, lower.tail = FALSE)
```

Probability proportions sample

```
sd <- sqrt((pi*(n-pi))/n)
z <- (p - pi)/sd

pnorm(X, pi, se, lower.tail =FALSE)
```

Sample size

Package = "samplingbook".

Provides the sample size needed to have a 95% confidence to estimate the population mean. Level = confidence level. Se is required standard error.

```
sample.size.mean(se, sigma, level=0.95)
```

Topic 5 - Significance testing

Critical values

Critical value for normal distribution, sample > 30

1. Two-sided: Critical value, 5% significance level = 1.96
2. Two-sided: Critical value, 1% significance level = 2.58
3. Two-sided: Critical value, 10% significance level = 1.96
4. One-sided: Critical value, 5% significance level = 1.64
5. One-sided: Critical value, 1% significance level = 2.33
6. One-sided: Critical value, 10% significance level = 1.28

```
cv <- qnorm(0.975)
cv <- qnorm(0.995)
cv <- qnorm(0.95)

cv <- qnorm(0.95)
cv <- qnorm(0.99)
cv <- qnorm(0.90)
```

Critical values t-distribution

1. One-sided: critical value at a 5% significance level
2. One-sided: critical value at a 10% significance level
3. One-sided: critical value at a 1% significance level
4. Two-sided: critical value at a 5% significance level
5. Two-sided: critical value at a 10% significance level
6. Two-sided: critical value at a 1% significance level

```
cv <- qt(0.95, df)
cv <- qt(0.90, df)
cv <- qt(0.99, df)

cv <- qt(0.975, df)
cv <- qt(0.95, df)
cv <- qt(0.995, df)
```

Confidence interval

```
cv <- cv
mu <- mu
sd <- sd
se <- sd / (sqrt(n))
n <- n

conf_int95 <- cv * sd / (sqrt(n))
mu_plus <- mu + conf_int95
mu_min <- mu - conf_int95
```

Hypothesis testing

Step	Example
1 State hypotheses.	$H_0: \mu \neq \mu_0$ $H_1: \mu \neq \mu_0$
2 Decide on the appropriate statistical distribution for testing H_0.	For testing the mean, assuming a large sample, use the Normal distribution.
3 State the significance level (α).	5%
4 State critical (cut-off) values associated with the sampling distribution of the test	For a Normal distribution these are –1.96 and +1.96.
5 Calculate the test statistic (e.g. z).	Answer varies for each test, but say 2.5 for example.
6 Compare the value of the test statistic to the critical values.	In this case it is above +1.96.
7 Come to a conclusion.	Here we would *reject* H_0.
8 Put your conclusion into English.	The sample evidence does not support the original claim that the population mean was the specified value.

Large sample significance testing

Package: "BSDA".

1. Two-sided
2. One-sided: X is greater than the population mean
3. One-sided: X is less than the population mean

```
1. tsum.test(mean.x = X, s.x = sd, n.x = n, mu = mu, alternative = "two.

2. tsum.test(mean.x = X, s.x = sd, n.x = n, mu = mu, alternative = "grea

3. tsum.test(mean.x = X, s.x = sd, n.x = n, mu = mu, alternative = "less
```

For proportions:

```
prop.test(x= x,n = n,p = p,correct=TRUE,alternative="two.sided")
```

Same goes for above: two.sided, greater, less

Test of equality - two samples

H*0* <- $\mu 1 = \mu 2$ *or* $(\mu 1 - \mu 2) = 0$
*H*a <- $\mu 1 \neq \mu 2$ or $\mu 1 - \mu 2 \neq 0$

Difference in two means with a certain confidence level confidence interval. Default = 95%

```
tsum.test(mean.x = X, s.x = sd, n.x = n, mean.y = X, s.y = sd, n.y = n,
```

2-sample test for equality of proportions without continuity correction.

```
data <- matrix(c(values), byrow=TRUE, nrow=2)
prop.test(data, correct=FALSE, alternative="greater")
```

Topic 5 - Non-Parametric testing

Contengency table / frequencies

Obtain contingency table

```
dist <- table(variable)
```

Chi-square

1. Chi-square test
2. Get the expected value
3. Probability for chi-square

```
data <- matrix(c(27,373,33,567),byrow=TRUE,nrow=2)
chisq.test(data,correct=FALSE)

chisq.test(data,correct=FALSE)$expected

prop.table(chisq.test(data,correct=FALSE)$expected,1)
prop.table(chisq.test(data,correct=FALSE)$expected,2)
```

Degree of freedom = # of row - 1 * # of columns = fixed

All expected frequencies must be above five! If not, categories must be combined!

Goodness of fit

Uniform:

Degree of freedom = number of categories - number of parameters - 1.

```
x <- c(frquencies)
p <- c(rep(1/5,n))
chisq.test(x,p=p)
```

All expected frequencies must be above five! If not, categories must be combined!

Binomial:

Package "actuar".

dbinom(x, size = n, prob = y)

For example:

```
cj <- c(-0.5, 0.5, 1.5, 2.5, 3.5, 4.5, 5.5)

#or

cj <- seq(from = -0.5, to=5, by=1)

nj <- c(15,20,20,18,13,10)
data <- grouped.data(Group = cj, Frequency = nj)
p <- mean(data)/5
pr <-c(dbinom(0,5,p),dbinom(1,5,p),dbinom(2,5,p),dbinom(3,5,p),dbinom(4,

nj2 <- c(35,20,18,23)
pr2 <- c(dbinom(0,5,p)+dbinom(1,5,p),dbinom(2,5,p),dbinom(3,5,p),dbinom(

chisq.test(nj2,p=pr2)
```

All expected frequencies must be above five! If not, categories must be combined!

Poisson

Degree of freedom = number of categories - number of parameters - 1.

NOTE! Distribution goes to infinity. Counter for one value that is X or more. 1 - until X.

Example:

```
cj <- c(-0.5, 0.5, 1.5, 2.5, 3.5, 4.5, 5.5)

or

cj <- seq(from = -0.5, to=6, by=1)
nj <- c(16, 30, 37, 7, 10)
data <- grouped.data(Group = cj, Frequency = nj)
m <- mean(data)

pr <- c(dpois(0, m),dpois(1,m),dpois(2, m), dpois(3, m), dpois(4, m), +

chisq.test(nj, p = pr)
```

Normal distribution

Example:

```
cv <- qchisq(0.90, 2)

cj <- c(0, 1, 3, 10, 15, 30)
nj <- c(16, 30, 37, 7, 10)
data <- grouped.data(Group = cj, Frequency = nj)
m <- mean(data)
s <- sqrt(emm(data,2))

pr <- c(pnorm(1,m,s), diff(pnorm(c(1,3),m,s)), diff(pnorm(c(3,10),m,s)),

chisq.test(nj,p=pr)
```

Mann-whitney test

N = Number of pairs - number of draws

For small tests

c1 values sample 1
c2 values sample 2

```
wilcox.text(c1,c2)
```

Larger sample test > 10

You can use a approximation based on the normal distribution. Therefore critical values will be 1.96 for this two sided test.

Wilcoxon test

Two options
- Do not predict direction --> two sided
- Predict direction --> one sided

```
wilcox.test(w1,w2,paired=TRUE,correct=FALSE)
```

Run test

Package "randtests".

```
pers <- c(0,1,1,0,0,0,0,1,1,0,1)
pers.f <- factor(pers,labels=c("Male","Female"))
runs.test(pers)
```

P-value

Find p value: Probability of getting this test statistic or more:

```
pchisq(ts,df,lower.tail=FALSE)
```

Topic 6 - Regressions, correlation and dummy's

Y = Dependent
X = Explanatory

Correlation

```
cor(data$X, data$Y)
```

R-Squared

Package: "stargazer".

```
Stargazer package =
stargazer(lm(Y~X, data=data), type="text")
```

Regressions

Plotting regression

```
plot(y~x,data=data, main="Title",ylab="Selling price",xlab="Size")
```

Regression line:

```
abline(lm(y~x,data=data),col="blue")
```

Creating the regression:

1. To plot the regression model
2. Evaluates the coefficient of the model
3. Only the first colum estimattion

```
model <- lm(y~x, data = data)
summary(model)$coef
est <- summary(model)$coef[,1]
```

Confidence interval around slope

```
confint(lm(variableY~variableX), level=0.95)
```

Subsampling regression

Specify dimentions [,]. First is row. Column, second.

1. Selects the rows where age is larger than 45.
2. Lower than 45.

```
summary(lm(y-x, data=data[age>=45,]))
summary(lm(y-x, data=data[age<=45,]))
```

Dummy variables, diff in means

Example:

```
Allpack <- c(Package$Pack1,Package$Pack2)
Package$dummy1 <- 0
Package$dummy2 <- 1
dummy <- c(Package$dummy1,Package$dummy2)
newdat <- data.frame(Allpack,dummy)

summary(lm(Allpack~dummy,data=newdat))
```

Regression + dummy

Y = Constant0 + B0 * X - Diffinmeans + B1 * variable1*2

Example:

```
Time <- c(Monterey$Time,Bakersfield$Time)
Boxes <- c(Monterey$Boxes,Bakersfield$Boxes)
Monterey$dummy <- 0
Bakersfield$dummy <- 1
dummy <- c(Monterey$dummy, Bakersfield$dummy)
Monterey$slopedummy <- 0
Bakersfield$slopedummy <- Bakersfield$Boxes
slopedummy <- c(Monterey$slopedummy, Bakersfield$slopedummy)

newdat <- data.frame(Time, Boxes, dummy, slopedummy)
```

Ommiting the intercept:

```
nfit <- lm(var1 ~ var2 - 1, data)
Shows the means seperately and not the difference between means. Tests w
```

Reorders group, to specific value to be first.

```
variable2 <- relevel(variable, "C")
```

Excluding the constant:

-1 excludes the constant. Now we get the means of each variable seperatly. Not the difference in means.

```
summary(lm(Y~dummy1 + dummy2 - 1, data=newdata))
```

Topic 7: Prediction

Prediction

```
xvalues <- data.frame(variablename = c(1,2,3,4,5))
predict(model, newdata = xvalues)
```

Prediction confidence interval:

1. One value
2. Multiple values from a existing data frame

```
predict(model, data.frame(valuename = value), interval = "confidence", l
predict(model, newdata = xvalues, interval = "confidence", level=0.95)
```

Prediction interval

1. One value
2. Multiple values from a existing data frame

```
predict(model,data.frame(valuename = value), interval="predict",level=0.
predict(model, data.frame, interval="predict",level=0.95)
```

Confidence and prediction plotting

Package: "HH".
Adds: observed values, fitted line, conf interval, predicted interval

```
fit <- lm(variable1~variable2, data=data)
ci.plot(fit)
```

Prediction with dummy variables

Prediction = α1+α2Constant Dummy+β1$Size$+β2Slope Dummy

Prediction intervals examples

Prediction

```
fit <- lm(Y ~ X + dummy + dummyslope, data=data)

predict(fit, data.frame(VariableX = c(10), Dummy = c(1),
Slopedummy = c(10)) )
```

Confidence interval prediction

```
fit <- lm(Y ~ X + dummy + dummyslope, data=data)

predict(fit, data.frame(VariableX = c(10), Dummy = c(1),
Slopedummy = c(10), interval="confidence")
```

Prediction interval

```
fit <- lm(Y ~ X + dummy + dummyslope, data=data)

predict(fit, data.frame(VariableX = c(10), Dummy = c(1),
Slopedummy = c(10), interval="predict")
```

Topic 8 - Data problems

Residual plot

```
m1 <- lm(Y~X, data=data)
residual.plot(fitted(m1),resid(m1),sigma.hat(m1), main="Title")
```

Influential measure test

```
influence.measures(m1)
```

Multicollinearity

1. F-test
2. Variance inflation factors greater than 10

```
anova(fit, fitres)
vif(fit)
```

ANOVA

One-way: one value

```
res.aov <- aov(Y ~ X, data = data)
summary(res.aov)
```

Two-way: more than two factors

```
res.aov <- aov(Y ~ X + X2, data = data)
summary(res.aov)
```

With interaction

```
res.aov <- aov(Y ~ X * X2, data = data)
summary(res.aov)
```

MANOVA: Multiple vactors

1. Test in difference
2. Test seperately

```
test_manova <- manova(cbind(Y, Y2) ~ X, data = data)
summary(test_manova)
summary.aov(test_manova)
```

Linear hypothesis test

Example:

```
fit <- lm(MKTDUB~pdub + poscar + pbpreg + pbpbeef,data=Hotdog)
linearHypothesis(fit,c("pbpreg + pbpbeef=0.0005"), test="F")
```