Computational Statistics Exercise session 2

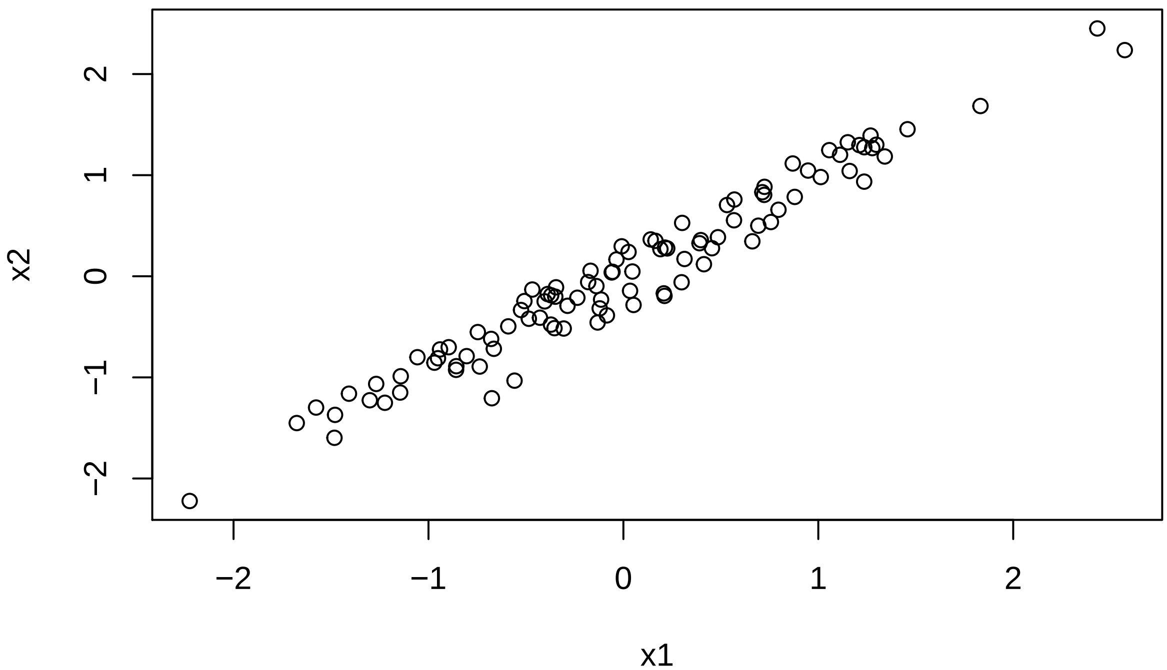
Heiko Kromer

2019-03-15

Exercise 1

a) Create a plot of the observations of the two predictor variables x1 and x2.

```
plot(x1,x2)
```



b) Fit a linear model fit1<-lm(y~x1+x2) and print the summary using summary(fit1).

```
fit1 <- lm(y~x1+x2)
(s1 <- summary(fit1))
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.89540 -0.73467 -0.01828  0.58897  2.43687
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.0645     0.1062  47.678   <2e-16 ***
```

```
## x1            0.4440     0.5521   0.804    0.423
## x2           -0.8638     0.5674  -1.522    0.131
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.061 on 97 degrees of freedom
## Multiple R-squared:  0.1137, Adjusted R-squared:  0.09542
## F-statistic: 6.222 on 2 and 97 DF,  p-value: 0.002869
```

c) Recompute the t-value corresponding to betahat1 by hand using the estimate betahat1 and its estimated standard error se(betahat1).

t value by hand

```r
beta1.hat <- s1$coefficients["x1", "Estimate"]
se.beta1.hat <- s1$coefficients["x1", "Std. Error"]
(tval.beta1 <- beta1.hat / se.beta1.hat)
```

```
## [1] 0.8041907
```

t value from R

```r
(tval.beta1.fromR <- s1$coefficients["x1", "t value"])
```

```
## [1] 0.8041907
```

Check if the values are different.

```r
abs(tval.beta1 -  tval.beta1.fromR)
```

```
## [1] 0
```

d) Give the definition of a p-value. Then compute the p-value corresponding to betahat1 using the t-value from part c) and the quantile function of the t-distribution pt().

Note: You need to provide the correct number of degrees of freedom.

Definition of a p-value

The p-value is the probability of observing any value equal to |t| or larger, where $t = \frac{(\hat{\beta_1}-0)}{SE(\hat{\beta_1})}$ under the null hypothesis which tests $\beta_1 = 0$ (there is no relationship between X and Y) versus $\beta_1 \neq 0$ (there is a relationship between X and Y).

```r
# n are the number of observations, already defined
# p is the intercept and two variables x1 and x2
p <- 2
(pval.beta1 <- 2*pt(abs(tval.beta1), df=n-p, lower=FALSE))
```

```
## [1] 0.4232334
```

```r
(pval.beta1.fromR <- s1$coefficients["x1", "Pr(>|t|)"])
```

```
## [1] 0.4232534
```

check difference

```r
(abs(tval.beta1 -  tval.beta1.fromR))
```

```
## [1] 0
```

e) Report the p-value of the overall F-test and reproduce it using anova().

```r
options(digits=10)
s1
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##         Min          1Q      Median          3Q         Max
## -2.89539537 -0.73467157 -0.01827616  0.58897411  2.43686760
##
## Coefficients:
##                Estimate Std. Error  t value Pr(>|t|)
## (Intercept)  5.0645272  0.1062239 47.67785  < 2e-16 ***
## x1           0.4439596  0.5520576  0.80419  0.42325
## x2          -0.8637536  0.5674145 -1.52226  0.13120
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.060502 on 97 degrees of freedom
## Multiple R-squared:  0.1136987,   Adjusted R-squared:  0.09542449
## F-statistic:   6.2218 on 2 and 97 DF,  p-value: 0.002868773
```

```r
pOverall.s1 <- 0.002868773
# overall p-value is 0.002869


# We can also reproduce the p-value by comparing two models:
fit1.small <- lm(y~1)
# and conducting a partial F-test:
s1.anova <- anova(fit1.small, fit1)
(pOverall.s1.anova <- s1.anova$`Pr(>F)`[2])
```

```
## [1] 0.002868772659
```

```r
(abs(pOverall.s1.anova-pOverall.s1))
```

```
## [1] 3.405067079e-10
```

f) The overall F-test is significant. However, the p-values for x1 and x2 are not significant. Explain how this can be true.

The overall F-test compares two models: the constant model and a model with both predictors x1 and x2 present. The p-values in the table compare a model with one of the predictors versus a model without the predictor (i.e. model one is $y \sim x1$, model two is $y \sim x1 + x2$). Only one predictor is enough to make a significant prediction. This is not very surprising since we saw that the x1 and x2 are highly correlated, i.e. low x1 correspond to low values of x2 and high values of x1 correspond to high values of x2.

g) Report the residual standard error, interpret it, and recompute it based on residuals(fit1).

```r
options(digits=5)
res.fromR <- residuals(fit1)
```

Re-compute summary statistics of residuals

summary(res.fromR, digits=3)

```r
# Compare to summary(fit1)
summary(fit1)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8954 -0.7347 -0.0183  0.5890  2.4369
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.065      0.106   47.68   <2e-16 ***
## x1             0.444      0.552    0.80     0.42
## x2            -0.864      0.567   -1.52     0.13
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.06 on 97 degrees of freedom
## Multiple R-squared:  0.114,  Adjusted R-squared:  0.0954
## F-statistic: 6.22 on 2 and 97 DF,  p-value: 0.00287
```

The residual standard error is 1.06.

It is

$$\hat{\sigma}^2 = \frac{RSS}{n-p}$$

where RSS is the residual sum of squares

$$RSS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} e_i^2$$

n is the number of observations (rows) and p the number of parameters (3 in this exercise).

The residual standard error (RSE) is given as:

$$RSE = \hat{\sigma} = \sqrt{\frac{RSS}{n-p}}$$

$\hat{\sigma}^2$ is a measure of goodness of fit. It is an estimate of σ^2, the variance of the statistical errors. The smaller the number, the better the fit (points closer to the line). The smaller the better in relation to the scale of the dependent variable. The RSE is measured in the same units as the dependent variable.

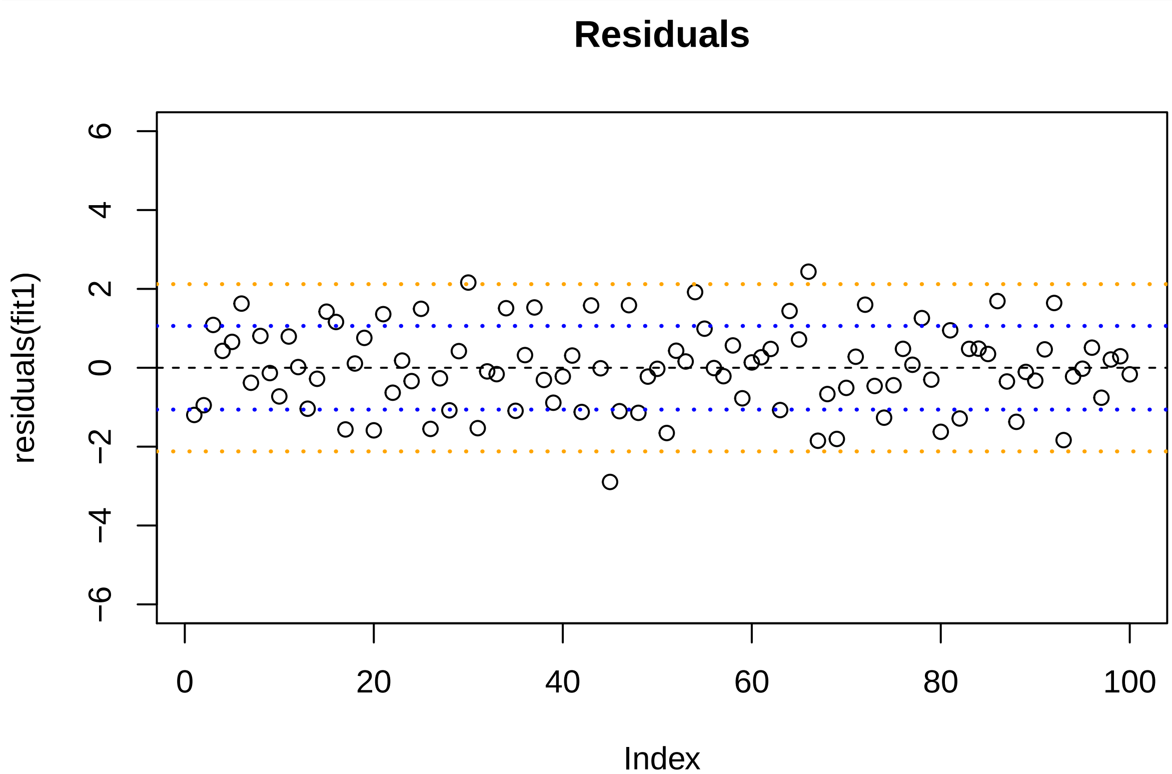
```
# Re-compute residual standard error (RSE):
p <- 3                                  # intercept and two variables
sum(res.fromR^2)/(n-p)                        # sigmahat^2 = RSS/(n-p)
```

```
## [1] 1.1247
```

```
(RSE <- sqrt( sum(res.fromR^2)/(n-p) ))       # RSE = sqrt(RSS/(n-p))
```

```
## [1] 1.0605
```

```
plot(residuals(fit1), ylim=c(-6,6), main="Residuals")
# In case of normally distributed errors, we expect:
#   About 66% of the points are within +/- hat.sigma
#     from the regression plane (blue dotted lines)
#   Aboute 95% of the points are within +/- 2*hat.sigma
#     from the regression plane (orange dotted lines)
abline(h=0, lty=2)
abline(h=RSE, lty=3, col="blue", lwd=2)
abline(h=-RSE, lty=3, col="blue", lwd=2)
abline(h=2*RSE, lty=3, col="orange", lwd=2)
abline(h=-2*RSE, lty=3, col="orange", lwd=2)
```



h) Report the R2 value, interpret it, and recompute it using residuals(fit1).

```r
s1
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8954 -0.7347 -0.0183  0.5890  2.4369
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.065      0.106   47.68   <2e-16 ***
## x1             0.444      0.552    0.80     0.42
## x2            -0.864      0.567   -1.52     0.13
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.06 on 97 degrees of freedom
## Multiple R-squared:  0.114,  Adjusted R-squared:  0.0954
## F-statistic: 6.22 on 2 and 97 DF,  p-value: 0.00287
```

The R2 value is 0.114. The adjusted R2 value is 0.0954.

The R2 value represents the proportion of variance explained by regression model. R2 is the proportion of the variance in y that is explained by the model. If R2 = 0, then the model is useless; if R2 = 1, model explains everything (errors are the smallest).

$$R^2 = \frac{TSS - RSS}{TSS} = 1 - \frac{RSS}{TSS}$$

where TSS is the total sum of squares $TSS = \sum_{i=1}^{n} (y_i - y)^2$.

The adjusted R2 value penalizes larger models. A bigger model with more freedom has a better R2 than a smaller model. When adding variables (not more points, but more columns) to a model, the R2 can only go up. If p goes up and RSS stay the same —> RSS/(n-p) becomes larger —> adjusted R2 becomes smaller. So if the gain in a decrease of RSS when adding another variable outweighs the decrease in (n-p), the model will be better.

$$R^2_{adj} = 1 - \frac{RSS/(n-p)}{TSS/(n-1)}$$

```r
# Re-compute R^2:
RSS <- sum( residuals(fit1)^2 )  # residuals = y - yhat
TSS <- sum( (y-mean(y))^2 )
(Rsquared <- 1 - RSS/TSS)
```

```
## [1] 0.1137
```

```r
# Re-compute adjusted R^2:
(Rsquared.adj <- 1 - (RSS/(n-p))/(TSS/(n-1)))
```

```
## [1] 0.095424
```

```r
# Adjusted for number of variables in the model
```

i) Assume now that we only observed the values for x1 and y whereas x2 is a hidden predictor that we do not observe. Fit the model fit3<-lm(y~x1) and print the summary summary(fit3). Compare the estimated coefficient corresponding to x1 to the one in part b). Interpret the coefficient of x1 in both models.

```r
fit3<-lm(y~x1)
(s3 <- summary(fit3))
```

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0078 -0.6350 -0.0781  0.6853  2.5553
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.056      0.107   47.35   <2e-16 ***
## x1            -0.377      0.119   -3.16   0.0021 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.07 on 98 degrees of freedom
## Multiple R-squared:  0.0925, Adjusted R-squared:  0.0833
## F-statistic: 9.99 on 1 and 98 DF,  p-value: 0.00209
```

In the first model it is $\beta_1 = 0.444$ and in this model (fit3) it is $\beta_1 = -0.377$. The sign of β_1 flipped between the 1st and 3rd model. This shows that the interpretation of each β_k depends on the other variables in the model!