4th February, 2021

STK2100 - Machine learning and statistical methods for prediction and classification

Mandatory assignment 1 of 2

Submission deadline

Thursday 18th February 2021, 14:30 in Canvas (canvas.uio.no).

Instructions

You can choose between scanning handwritten notes or typing the solution directly on a computer (for instance with LaTeX). The assignment must be submitted as a single PDF file. Scanned pages must be clearly legible. The submission must contain your name, course and assignment number.

It is expected that you give a clear presentation with all necessary explanations. Remember to include all relevant plots and figures. Students who fail the assignment, but have made a genuine effort at solving the exercises, are given a second attempt at revising their answers. All aids, including collaboration, are allowed, but the submission must be written by you and reflect your understanding of the subject. If we doubt that you have understood the content you have handed in, we may request that you give an oral account.

In exercises where you are asked to write a computer program, you need to hand in the code along with the rest of the assignment. It is important that the submitted program contains a trial run, so that it is easy to see the result of the code.

Application for postponed delivery

If you need to apply for a postponement of the submission deadline due to illness or other reasons, you have to contact the Student Administration at the Department of Mathematics (e-mail: studieinfo@math.uio.no) well before the deadline.

All mandatory assignments in this course must be approved in the same semester, before you are allowed to take the final examination.

Complete guidelines about delivery of mandatory assignments:

uio.no/english/studies/admin/compulsory-activities/mn-math-mandatory.html

GOOD LUCK!

Spesific requirements to this assignment:

In order to get the assignment **accepted** you need to fullfil the following requirements:

- You have done a real attempt on **all** (sub-)questions. **This applies to the first submission**!
- There is a satisfactory answer in at least 2/3 of the (sub-)questions.

Remember that it is allowed to ask for help!
Within the exercises several commands that can be used in **R** are included. If some libraries are not available at your computer, you need to install them, for example by

```
install.packages("MASS")
```

All the command listed in the assignment are aslo available in a separate .R file on the course webpage.

It is allowed to use other programs, but there will then be extra requirements to good documentation on what you have done and you can not expect to obtain help with respect to implementional details.

Problem 1. We will in this exercise look at a dataset `nuclear` and see how we can use regression for prediction of costs of light-water reactor. The data is available in the file `nuclear.dat` while a description of the data is available at `nuclear.txt`, both availble from the course data webpage https://www.uio.no/studier/emner/matnat/math/STK2100/data/.

Our interest will be in the `cost` variabele while the other variables will be explanatory variables. Since `cost` is always positive, we will model this variable at the log-scale.

(a) Make the data available through the commands

```
datadir = "http://www.uio.no/studier/emner/matnat/math/STK2100/data/"
nuclear = read.table(paste(datadir,"nuclear.dat",sep=""),header=T)
n = nrow(nuclear)
```

Make also different plots in order to get some understanding of the data.

(b) We will first look at a model

$$Y_i = \beta_0 + \beta_1 x_{i,1} + \cdots + \beta_p x_{i,p} + \varepsilon_i$$

where Y_i is `cost` at log scale for observation i.

What are the standard assumptions about the noise terms ε_i? Discuss also which of these assumptions that are most important.

Fit this model including all the observations with `log(cost)` as response and all the other variables as covariates.

Discuss the results.

(c) Now remove the variable with the highest corresponding P-value and fit the new model.

Why is this a reasonable procedure?

Discuss potential changes of the P-values for the remaining variables. You can relate this to correlations between the explanatory variables.

(d) Continue to remove explanatory variables until all P-values are less than 0.05. What is your final model?

Make different plots in order to evaluate whether the model is reasonable.

(e) Use the final model to predict response and make a model based on the average quadratic error ($\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$) in order to evaluate how good the model is. Discuss weaknesses with such a procedure.

An automatic routine,`stepAIC` for sequential model selection is available in the **R**-library `MASS`. This procedure do not use the P-values for selection of models, but instead

$$\text{AIC} = -2\log(L(\hat{\boldsymbol{\theta}})) + 2(q+2)$$

where $L(\hat{\boldsymbol{\theta}})$ is the likelihood value evaluated at the maximum likelhood estimate $\hat{\boldsymbol{\theta}}$ while q is the number of parameters in the model. One wants low AIC-values.

For a linear regression model, we get

$$\text{AIC} = \text{Const} + n\log(\hat{\sigma}^2) + 2(q+2) \qquad \text{(LM.BIC)}$$

where $\hat{\sigma}^2$ is an estimate of σ^2 *within the model considered.*

The function `stepAIC` can do both forward and backward selection. We will concentrate on backward selection which can be obtained by the option `direction="backward"`.

Using backward selection, at the step with q variables included in the current model, q different models, each with one of the current variables removed, will be fitted. It then chooses the reduced model with largest likelihood value. It then calculate the AIC value for this new model with

$q-1$ variables and compare it with the current model containing q variables. The procedure continues as long as a reduction in the AIC value is achieved.

Alternatively on can use `stepAIC` for the BIC criterion

$$\text{BIC} = -2\log(L(\hat{\boldsymbol{\theta}})) + \log(n)(q+2)$$

which for linear models becomes

$$\text{BIC} = \text{Const} + n\log(\hat{\sigma}^2) + \log(n)(q+2). \qquad \text{(LM.BIC)}$$

BIC can be obtained by the option `k=log(n)`.

(f) Show that AIC and BIC are equal to (LM.BIC) and (LM.BIC) for the linear regression model with Gaussian noise.

(g) Try out the automatic model selection command on the `nuclear` dataset both using AIC and BIC. Which models do you end up with then?

(h) Have a look on the order of which the two different methods select models. Explain why the order should be the same for AIC and BIC.

(i) Discuss potential differences compared with the manual selection based on the P-values (both with respect to the order they are selected and the number of variables that are included in the end).

Assume now we want to predict `cost` for a new data point. More specifically we are interested in $\theta = E[Y|\boldsymbol{x}^*]$ as well as $\eta = E[\exp(Y)|\boldsymbol{x}^*]$ where $\boldsymbol{x}^*$ is defined by

```
d.new = data.frame(date=70.0,t1=13,t2=50,cap=800,pr=1,
                   ne=0,ct=0,bw=1,cum.n=8,pt=1)
```

Prediction can be performed by the command `predict.lm`.

The command below gives predictions for a fitted model for θ [1]

```
predict(fit,d.new,se.fit)
```

where `fit` is the fitted model.

[1] Note that it is enough to use the generic function `predict` here due to that **R** understands that it is the underlying command `predict.lm` which do predictions for linear models is to be used. If you, however, want to see how the function works, you need to use `help(predict.lm)`. There you also can see how to include uncertainties in the predictions.

(j) Show that if $Z \sim N(\mu, \sigma^2)$, then $E[\exp(Z)] = \exp(\mu + 0.5\sigma^2)$. Use this to argue that we have two possible estimates of η:

$$\hat{\eta}_1 = \exp(\hat{\theta})$$
$$\hat{\eta}_2 = \exp(\hat{\theta} + 0.5\hat{\sigma}^2)$$

Argue for one of these estimates and use that in the following.

(k) Perform predictions based on the three choosen modes you obtained earlier (model choice based on P-values, AIC and BIC). Comment on the results.

Problem 2. This exercise is a continuation of the previous exercise where we will use alternative methods for evaluating different models.

(a) Consider the following commands using `stepAIC` to obtain a range of models and simultaneously estimating prediction errors by splitting into training and test data.

```
n = nrow(nuclear)
ind = sample(1:n,n/2,replace=FALSE)
RMSE.test1=rep(NA,11)
RMSE.test2=rep(NA,11)
model_narrow=lm(log(cost)~1,data=nuclear)
model_wide=lm(log(cost)~.,data=nuclear)
for(i in 0:10)
{
 fit=stepAIC(model_narrow,direction="forward",
   steps=i,data=nuclear[ind,],k=0,
   scope=list(lower=model_narrow,upper=model_wide))
 pred=predict(fit,nuclear[-ind,])
 RMSE.test1[i+1]=sqrt(mean((log(nuclear$cost)-pred)^2))
 RMSE.test2[i+1]=sqrt(mean((nuclear$cost-exp(pred))^2))
}
```

You may plot the results by e.g the following commands:

```
par(mar=c(5,4,4,4)+0.3)
plot(0:10,RMSE.test1,type="l",
      xlab="Complexity",ylab="RMSE1")
par(new=TRUE)
plot(0:10,RMSE.test2,type = "l",axes=FALSE,bty="n",
      xlab="",ylab="",col=2)
axis(side=4,at=pretty(range(RMSE.test2)))
mtext("RMSE2",side=4,line=3)
```

Try to explain what is going on here. Also discuss the results (within the plot) you get.

Repeat the set of commands a few times. How robust are the results?

Discuss some weaknesses with this approach.

(b) An alternative to splitting the data into two is to use cross-validation. The following commands can then be useful:

```
library(lmvar)
RMSE.cv1=rep(0,10)
RMSE.cv2=rep(0,10)
for(i in 0:10)
{
fit=stepAIC(model_narrow,direction="forward",steps=i,k=0,
        scope=list(lower=model_narrow,upper=model_wide),trace=0)
 fit=lm(formula(fit),data=nuclear,x=TRUE,y=TRUE)
 #Note: k below has a different meaning than k above!!!
 RMSE.cv1[i+1]=cv.lm(fit,k=10)$MSE$mean
 RMSE.cv2[i+1]=cv.lm(fit,k=10,log=TRUE)$MSE$mean
}
```

Look at the help page for the `cv.lm` routine to understand that this command is doing.

Make a similar plot as in the previous exercise and comment on the results.

Again repeat a few times to see the robustness of this approach.

(c) Modify the previous commands to perform leave-one-out cross validation. Run the commands and discuss the results in this case.

Problem 3. We will in this exercise look at linear regression with quantitative (categorical) explanatory variables. Assume we have data $(c_1, y_1), ..., (c_n, y_n)$ where $c_i \in \{1, ..., K\}$. Define for $j = 1, ..., K$

$$x_{i,j} = \begin{cases} 1 & \text{if } c_i = j; \\ 0 & \text{otherwise.} \end{cases}$$

(a) Show that the two models

$$Y_i = \beta_0 + \beta_2 x_{i,2} + \cdots + \beta_K x_{i,K} + \varepsilon_i \tag{1}$$

and

$$Y_i = \alpha_1 x_{i,1} + \cdots + \alpha_K x_{i,K} + \varepsilon_i \tag{2}$$

are equivalent. Write an explicit relationship between $\boldsymbol{\beta} = (\beta_0, \beta_2, ..., \beta_K)$ and $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_K)$. Use also the two models to give an interpretation of the various parameters involved.

We will in the following concentrate on model (2) since this is version is somewhat simpler mathematically.

(b) Let $\boldsymbol{X}$ be the design matrix for model (2), that is the ith row of $\boldsymbol{X}$ contains the values $x_{i,j}, j = 1, ..., K$. Show that $\boldsymbol{X}^T\boldsymbol{X}$ will be a diagonal matrix with diagonal elements n_j where n_j is the number of observations with $c_i = j$.

Also show that $\boldsymbol{X}^T\boldsymbol{y}$ is a vector where the j-th element is equal to element $\sum_{i:c_i=j} y_i$.

Based on this, derive the least squares estimates for $\alpha_1, .., \alpha_K$. Discuss whether these estimates are reasonable.

(c) Based on the relationship between $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$, construct estimates for $\boldsymbol{\beta}$.

Argue why also these estimates are least squares estimates for $\boldsymbol{\beta}$.

(d) Another alternative model is

$$Y_i = \gamma_0 + \gamma_1 x_{i,1} + \cdots + \gamma_K x_{i,K} + \varepsilon_i \tag{3}$$

where we constain the parameters by assuming $\sum_{j=1}^{K} \gamma_j = 0$.

What values do the γ_j's need to have in order for this model to be equivalent to the previous models?

What interpretation do the γ's have in this case?

Note that the two previous models can be rewritten to

$$Y_i = \beta_0 + \beta_1 x_{i,1} + \cdots + \beta_K x_{i,K} + \varepsilon_i$$

where $\beta_1 = 0$ and

$$Y_i = \alpha_0 + \alpha_1 x_{i,1} + \cdots + \alpha_K x_{i,K} + \varepsilon_i$$

where $\alpha_0 = 0$. That is, all three models are basically expressed in the same way with $K + 1$ parametre but are reduced to K *free* parametres by putting *constraints* on the parameters. We will see below how we can introduce such constraints when performing analysis within **R**.

We will in the rest of the exercise look at a dataset from Devore & Berk (2012, exercise 11.5). The dataset consists of measurements of iron content in four different iron formations (1=carbonate, 2=silicate, 3=magnetite, 4 =hematite). The table below shows the data, with 10 observations within each type of iron formation.

Type	1	2	3	4	5	6	7	8	9	10
1	20.50	28.10	27.80	27.00	28.00	25.20	25.30	27.10	20.50	31.30
2	26.30	24.00	26.20	20.20	23.70	34.00	17.10	26.80	23.70	24.90
3	29.50	34.00	27.50	29.40	27.90	26.20	29.90	29.50	30.00	35.60
4	36.50	44.20	34.10	30.30	31.40	33.10	34.10	32.90	36.30	25.50

We will to test whether there are differences in iron content between the different formations.

(e) Read the data by the command

```
datadir = "http://www.uio.no/studier/emner/matnat/math/STK2100/data/"
Fe <- read.table(paste(datadir,"fe.txt",sep=""),header=T,sep=",")
```

Try out the following command

```
fit1 <- lm(Fe~form+0,data=Fe)
summary(fit1)
```

Why do this go wrong?

Then give the command

```
Fe$form <- as.factor(Fe$form)
```

Now try out the same call to `lm` again. Why do you get a more reasonable result in this case? Which of the three models discussed earlier do this fit belong to? That is, which constraint does it correspond to?

(f) Try now out

```
options()$contrasts
options(contrasts=c("contr.treatment","contr.treatment"))
fit2 <- lm(Fe~form,data=Fe)
summary(fit2)

options(contrasts=c("contr.sum","contr.sum"))
options()$contrasts
fit3 <- lm(Fe~form,data=Fe)
summary(fit3)
```

Which models do these fits relate to?

For all the three models, make a table of *all* the $K+1$ regression parameters.

Do the results match the results you derived earlier?

(g) Assume you want to test the difference between different types of iron formation.

Formulate a reasonable null and alternative hypothesis and explain how you can use *one* of the summaries from the earlier fitted models to perform such a test.

What is the conclusion of this test?

(h) Based on the outputs from the different models, suggest a possible simplification of the model.