# Mandatory Assignment 1

STK2100

Håkon Berggren Olsen

Spring 2020

## Problem 1.

The dataset, nuclear, contains data regarding the construction of 32 Light-Water Reactors built between 1967 and 1971. The scope of the dataset is to predict the cost of construction for further span given the dataset within this time interval.

The dataset has 32 rows and 11 columns, and has a mix of continous- and categorical variabels. Contionous data such as regarding the cost, construction and net capacity of the a given nuclear powerplant, and categorical such as cooling tower present or absent, if there exists an LWR (Light-Water Reactor) plant at the same site and such.

The description of all the variables are shown in a table below.

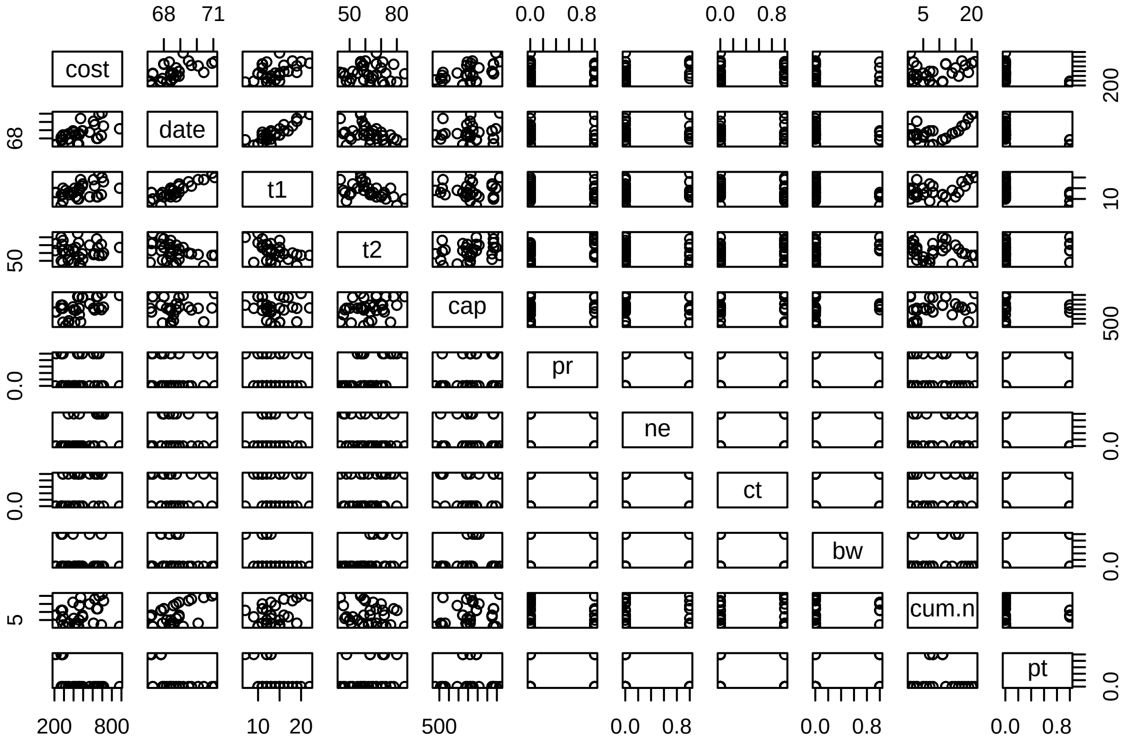
| **Name** | **Continous Variables** |
|---|---|
| cost: | The capital cost of construction in millions of dollars adjusted to 1976 base. |
| date: | The date on which the construction permit was issued. The data are measured in years since January 1 1990 to the nearest month. |
| t1: | The time between application for and issue of the construction permit. |
| t2: | The time between issue of operating license and construction permit. |
| cap: | The net capacity of the power plant (MWe). |
| cum.n: | The cumulative number of power plants constructed by each architect-engineer. |
| **Name** | **Categorical Variables** |
| pr: | A binary variable where 1 indicates the prior existence of a LWR plant at the same site. |
| ne: | A binary variable where 1 indicates that the plant was constructed in the north-east region of the U.S.A. |
| ct: | A binary variable where 1 indicates the use of a cooling tower in the plant. |
| bw: | A binary variable where 1 indicates that the nuclear steam supply system was manufactured by Babcock-Wilcox. |
| pt: | A binary variable where 1 indicates those plants with partial turnkey guarantees. |

### a) Loading the dataset

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

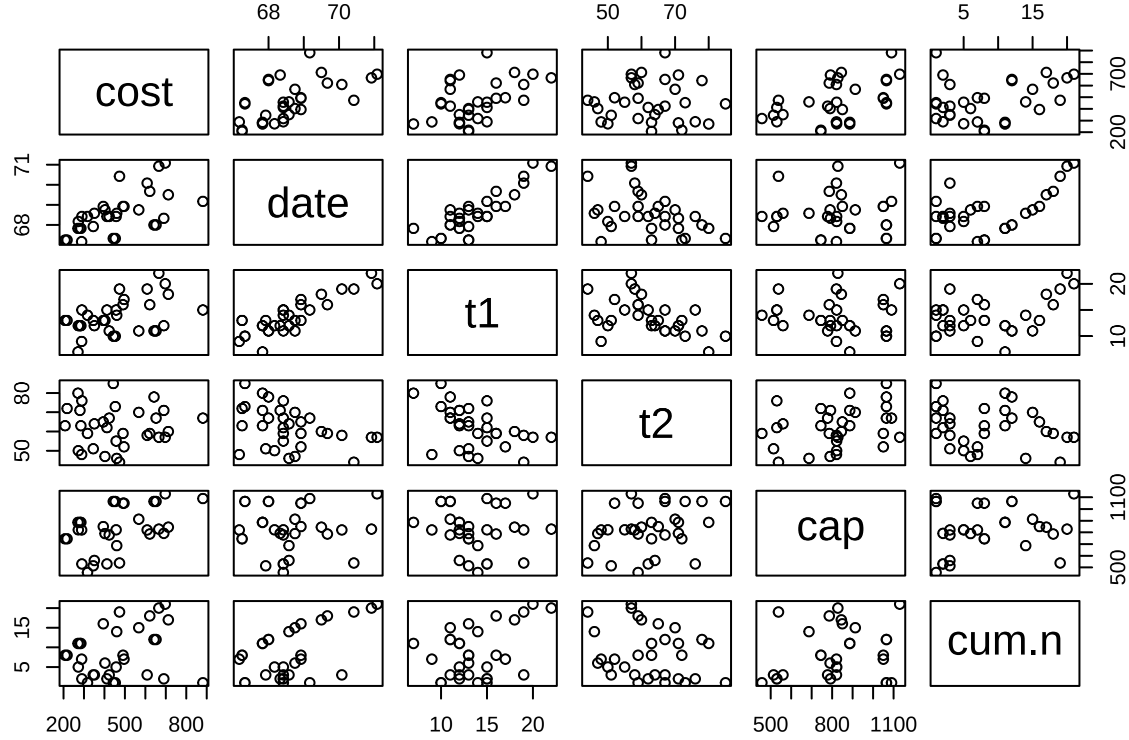
Generating some plots to gain an intuition for the dataset.

```
plot(nuclear)
```



This is very cluttered and hard to grasp, much due to categorical variables not giving any immeadiate insight. Therefore we will plot the dataset only regarding the continous variables.

```
cont_var <- c("cost", "date", "t1", "t2", "cap", "cum.n")
cat_var <- c("pr", "ne", "ct", "bw", "pt")
plot(nuclear[cont_var])
```



From this figure we can see some clear trends. We can see the cost of the LWRs are increasing over time, which can be attributed to inflation. A very positive correlation is also spotted in the t1 vs date indicating that the time between application and issue of the construction permits is steadily increasing and almost doubles at the end of the dataset. t2 vs date shows however a negative correlation, indicating that it takes shorter and shorter time from obtaining operating license to gaining construction permit.

The net capacity of the LWR, cap, does show a slight increase with cost. This is expected as the capital cost of the investment should match it's returns.

NOTE: still not sure how the cum.n "The cumulative number of power plants constructed by each architect-engineer." — what architect-engineer? – is construed.

## b) Constructing a model

Look at the model

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

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

- What are the standard assumptions about the noise term $\epsilon_i$? Discuss also which of these assumptions that are the most important?
  - The variance of the noise, $\epsilon_i$ are constant.
  - The expectance of the noise is $\mathbb{E}[\epsilon_i] = 0$
  - The noise is uncorrelated.
  - "The indepenedent variables are measured with no error"

  - “The sample is representative of the population at large.”
- Fit this model including all the observations with log(cost) as response and all the other variables as covariates. Discuss the results

```
#fit with all covariates
model2 = lm(log(cost)~., data=nuclear)
summary(model2)
```

```
##
## Call:
## lm(formula = log(cost) ~ ., data = nuclear)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.284032 -0.081677  0.009502  0.090890  0.266548
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.063e+01  5.710e+00  -1.862  0.07662 .
## date         2.276e-01  8.656e-02   2.629  0.01567 *
## t1           5.252e-03  2.230e-02   0.236  0.81610
## t2           5.606e-03  4.595e-03   1.220  0.23599
## cap          8.837e-04  1.811e-04   4.878 7.99e-05 ***
## pr          -1.081e-01  8.351e-02  -1.295  0.20943
## ne           2.595e-01  7.925e-02   3.274  0.00362 **
## ct           1.155e-01  7.027e-02   1.644  0.11503
## bw           3.680e-02  1.063e-01   0.346  0.73261
## cum.n       -1.203e-02  7.828e-03  -1.536  0.13944
## pt          -2.220e-01  1.304e-01  -1.702  0.10352
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1697 on 21 degrees of freedom
## Multiple R-squared:  0.8635, Adjusted R-squared:  0.7985
## F-statistic: 13.28 on 10 and 21 DF,  p-value: 5.717e-07
```

Discuss the results here

### c) Removing covariates with high P-value

Removing the highest corresponding P-value, i.e. t1 gives a better result as a high P-value means that the outcomes in $log(cost)$ cannot be sufficiently explained by the changes in $t1$. When doing this we get

```
#Fit with t1 taken away.
model2 = lm(log(cost)~.-t1,data=nuclear)
summary(model2)
```

```
##
## Call:
## lm(formula = log(cost) ~ . - t1, data = nuclear)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.28898 -0.07856  0.01272  0.08983  0.26537
##
## Coefficients:
```

```
##                Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -1.161e+01  3.835e+00  -3.027 0.006187 ** 
## date         2.431e-01  5.482e-02   4.435 0.000208 ***
## t2           5.451e-03  4.449e-03   1.225 0.233451    
## cap          8.778e-04  1.755e-04   5.002 5.25e-05 ***
## pr          -1.035e-01  7.944e-02  -1.303 0.205922    
## ne           2.607e-01  7.738e-02   3.368 0.002772 ** 
## ct           1.142e-01  6.853e-02   1.667 0.109715    
## bw           2.622e-02  9.423e-02   0.278 0.783401    
## cum.n       -1.220e-02  7.626e-03  -1.599 0.124034    
## pt          -2.157e-01  1.249e-01  -1.727 0.098181 .  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.166 on 22 degrees of freedom
## Multiple R-squared:  0.8631, Adjusted R-squared:  0.8072 
## F-statistic: 15.42 on 9 and 22 DF,  p-value: 1.424e-07
```

An increase from 13.28 to 15.42 in the F-statistic (explain F-statistic), increase of the multiple R-squared from 0.8635 to 0.8631 and likewise 0.7985 to 0.8072 for the adjusted R-squared.

## d)

Continuing to remove all explanatory variables untill all P-values are less than 0.05. What is the final model? Make different plots in order to evaluate whether the model is reasonable.

```
model3 <- lm(log(cost)~. -t1-bw, data=nuclear)
#summary(model3)

model4 <- lm(log(cost)~. -t1-bw-pr, data=nuclear)
#summary(model4)

model5 <- lm(log(cost)~. -t1-bw-pr-t2, data=nuclear)
#summary(model5)

model5 <- lm(log(cost)~. -t1-bw-pr-t2-cum.n, data=nuclear)
#summary(model5)

model6 <- lm(log(cost)~. -t1-bw-pr-t2-cum.n-ct, data=nuclear)
summary(model6)
```

```
## 
## Call:
## lm(formula = log(cost) ~ . - t1 - bw - pr - t2 - cum.n - ct, 
##     data = nuclear)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.42160 -0.10554 -0.00070  0.07247  0.37328 
## 
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -4.5035539  2.5022087  -1.800 0.083072 .  
## date         0.1439104  0.0363320   3.961 0.000491 ***
## cap          0.0008783  0.0001677   5.238 1.61e-05 ***
## ne           0.2024364  0.0751953   2.692 0.012042 *
## pt          -0.3964878  0.0963356  -4.116 0.000326 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1767 on 27 degrees of freedom
## Multiple R-squared:  0.8096, Adjusted R-squared:  0.7814
## F-statistic:  28.7 on 4 and 27 DF,  p-value: 2.255e-09
```

```
# Making plots of the final model from forward selection with the criteria of p values < 0.05
library(ggplot2)

predicted_cost <- data.frame(cost_pred = predict(model6, nuclear), cost=nuclear$cost)

#ggplot(nuclear, aes(x = cap, y = cost)) +
#  geom_point() +
#  geom_line(color="red", data = predicted_cost, aes(x = cap, y = cost))
```

### e) Model based on quadratic error

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 ot evaluate how good the model is. Discuss weaknesses with such a procedure.

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

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

#predict(fit, d.new, interval="confidence")
#predict(fit, d.new, interval="predict")
```

### f) Differences between confidence and predict commands

Run the two commands. Discuss the differences bweteen the two predict commands.

### g) Constructing log intervals on non-logarithmic scale

The intervals given in the previous point is related to *cost* on log-scale. Try to construct intervals for *cost* on the original scale.

### h) Lasso regression

Aslo try out Lasso regression on this data set.

If you use cross-validation for selection of the penalty parameter, which variables are then included in the final model?

Also compare this with the model you obtained earlier.

Hint: Loook at the Hitters_lasso.R script.

## Problem 2.

### a) Reading data

Getting the data

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

options(contrasts=c("contr.treamtent", "contr.treatment"))

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

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

```
#
```

The fitting does not work as the data is not

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

#fit2 <- lm(Fe~form, data=Fe)
#summary(fit2)
```

### b)

From the summary command (after you used the as.factor command) you should get a regression table where there is no row corresponding to $\beta_1$. The specific options command given above actually include the contraint $\beta_1 = \hat{\beta}_1 = 0$.

Why is such a contraint necessary?

What interpretations do the other $\beta_j$ parameters then have?

### c) Alternative Constraint

An alternative constraint is to put $\beta_0 = 0$. This can be obtained by

```
fit2 <- lm(Fe~form+0, data=Fe)
```

```
summary(fit2)
```

### d)

The constraints $\beta_1 = 0$ or $\beta_2 = 0$ are denoted by *contrasts* in the linear regression terminology. The `contr.treatment` used above corresponds to putting the regression coefficient of the first category equal to zero. An alternative is

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

in which case a constraint/contrats $\sum_{j=1}^{K} \beta_j = 0$ is imposed. The `summary`command will still only give 4 rows in the regression table, not including the row corresponding ot $\beta_4$ in this case. How can you obtain $\hat{\beta}_4$?

### e)

Do the results indicate that there are differences between the formations? Which of the fitted models do you find most suitable for answering this questions?

### f)

Now try out the commands

```
newdata = data.frame(form=as.factor(c(1,2,3,4)))  pred1 = predict(fit1, newdata)  pred2 =
predict(fit2, newdata) pred3 = predict(fit3, newdata)
```

Compare the three predictions and comment on the results.

## g)

Based on the summary outputs from the different models, is it possible to simplify the model in some way?

Hint: Not all the different outputs will tell the same story here.