STK2100 Oblig 1

Håkon Kvernmoen

2/4/2021

Problem 1

a)

First we need to load the data. The code sample for loading did not work for me (got a 400 bad request error). Assuming the datafile "nuclear.dat" is located in the same folder as this file, we load the data and attach it for easier use.

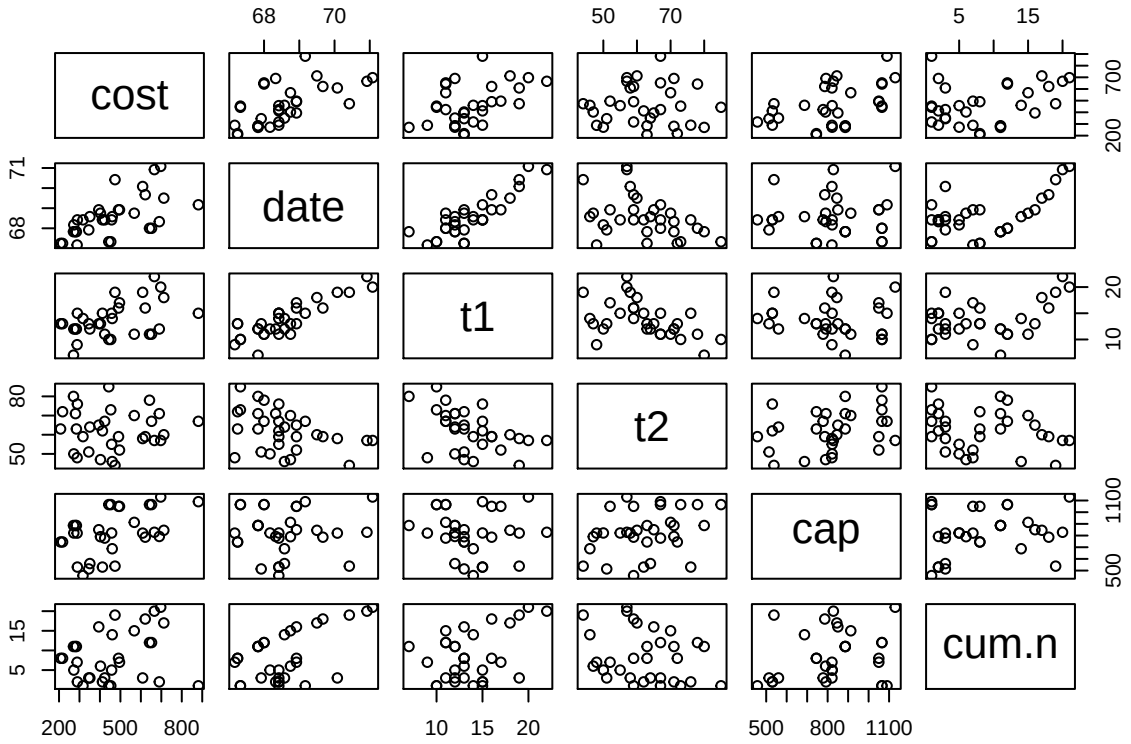
```
nuclear = read.table("nuclear.dat", sep="\t", header=T)
attach(nuclear)
```

We notice that `pr`, `ne`, `ct`, `bw` and `pt` are binary variables, so we set them as factors.

```
nuclear$pr = as.factor(nuclear$pr)
nuclear$ne = as.factor(nuclear$ne)
nuclear$ct = as.factor(nuclear$ct)
nuclear$bw = as.factor(nuclear$bw)
nuclear$pt = as.factor(nuclear$pt)
```

To investigate the data we plot the numerical features against each other. There seems to be some correlation between `date` and `t1`

```
plot(nuclear[,sapply(nuclear, is.numeric)])
```



b)

The standard assumption on the noise terms ϵ_i are.

1. The error terms are normally distributed with a mean of 0
2. The variance σ^2 of this normal distribution is constant
3. The error terms are independent, ϵ_i does not influence ϵ_j

Important?

We will now try to fit the model using all the features. As cost is always positive, we fit the log of the cost as a response variable. With y_i being the i'th observation of the cost, we will try to fit the model.

$$log(y_i) = \beta_0 + \sum_{j=1}^{p} x_{i,j} + \epsilon_i$$

```
all.fit = lm(log(cost)~., data = nuclear)
summary(all.fit)
```

```
##
## 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 ***
## pr1         -1.081e-01  8.351e-02  -1.295  0.20943
## ne1          2.595e-01  7.925e-02   3.274  0.00362 **
## ct1          1.155e-01  7.027e-02   1.644  0.11503
## bw1          3.680e-02  1.063e-01   0.346  0.73261
## cum.n       -1.203e-02  7.828e-03  -1.536  0.13944
## pt1         -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
```

c)

We will now remove the term with the largest P-value. Observing the summary of the linear model, we see that `t1` has the largest P-value at 0.81610. This is sensible to do since the P-value is a measure of the correctness of the null-hypothesis (H_0). A large P-value as in this case indicates that there is a very little statistical basis for `t1` to be a good predictor for log(`cost`) and is thus neglected.

```
all_no_t1.fit = lm(log(cost)~.-t1, data= nuclear)
summary(all_no_t1.fit)
```

```
##
## 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 ***
## pr1         -1.035e-01  7.944e-02  -1.303 0.205922
## ne1          2.607e-01  7.738e-02   3.368 0.002772 **
## ct1          1.142e-01  6.853e-02   1.667 0.109715
## bw1          2.622e-02  9.423e-02   0.278 0.783401
## cum.n       -1.220e-02  7.626e-03  -1.599 0.124034
```

```
## pt1         -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
```

We observe that there are some change in the P-values for a lot of the features after we excluded `t1`. This is probably due to correlation between the features. We would ideally have linearly independent explanatory variables. In example a change in `cap` should not influence any of the other explanatory variables, but this is not the case. On the other hand, the changes in P-values are not huge and the coefficients estimates seems relatively unchanged. In addition the standard error for the coefficients seems to decrease and we continue these modifications.

d)

We now want to fit our model, remove the explanatory variable with a P-value larger than 0.05 and repeat this until we have a model where all explanatory variables have P-values smaller than 0.05. We then implement a backward substitution algorithm. We note that we do not want to remove the intercept even tough its P-value can be larger than 0.05.

```
nuclear_backwards_sub <- data.frame(nuclear)
for (i in 1:ncol(nuclear)) {
    fit <- lm(log(cost)~., data=nuclear_backwards_sub)
    # -1 since we don't want to remove intercept
    p_vals <- summary(fit)$coefficients[-1,4]
    max_idx <- as.integer(which.max(p_vals))

    if(p_vals[max_idx] < 0.05) {
      break
    }
    else {
      # Add one since we don't want to remove targert variable (cost).
      nuclear_backwards_sub <- nuclear_backwards_sub[,-(max_idx+1)]
    }
}

summary(fit)
```

```
##
## Call:
## lm(formula = log(cost) ~ ., data = nuclear_backwards_sub)
##
## 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 ***
## ne1          0.2024364  0.0751953   2.692 0.012042 *
## pt1         -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
```

We are then left with 4 explanatory variables. Two of them continues (`date, cap`) and two binary (`ne, pt`). MAKE SOME PLOTS

e)

The final