

# Lab 4 732A95 Introduction to machine learning

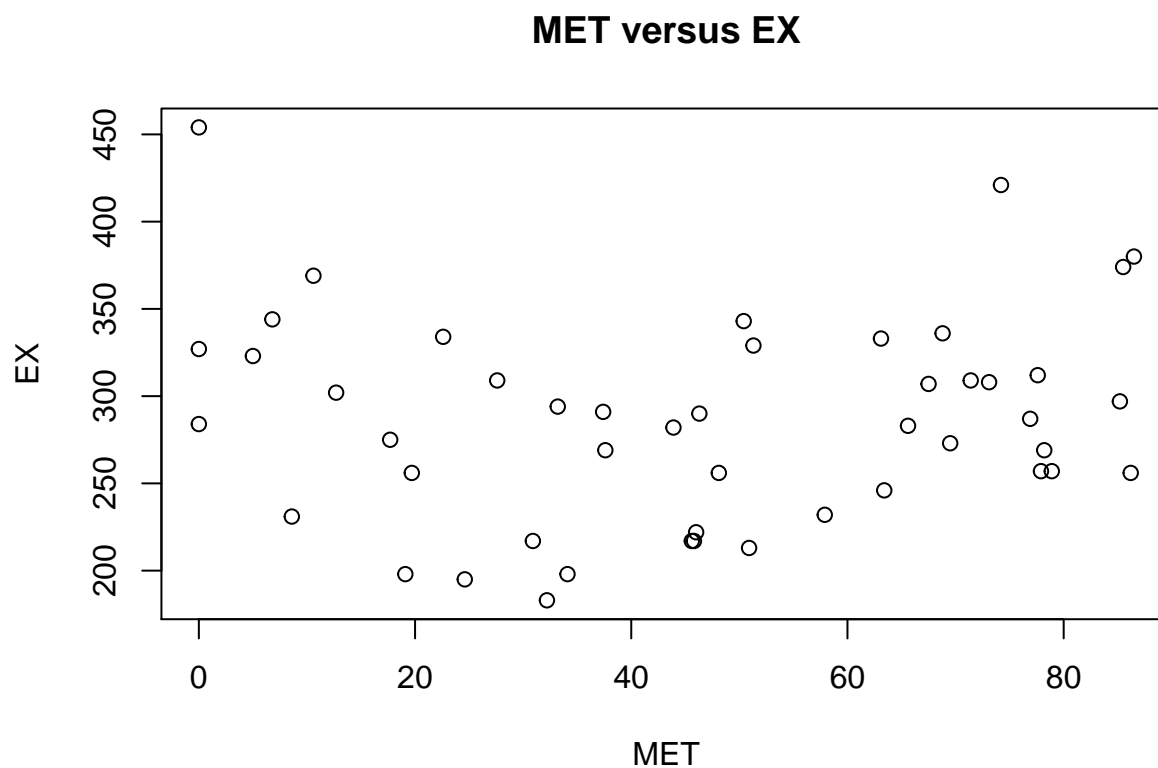
*Anton Persson antpe404*

*30 november 2016*

## Assignment 1 Uncertainty estimation

### Assignment 1.1

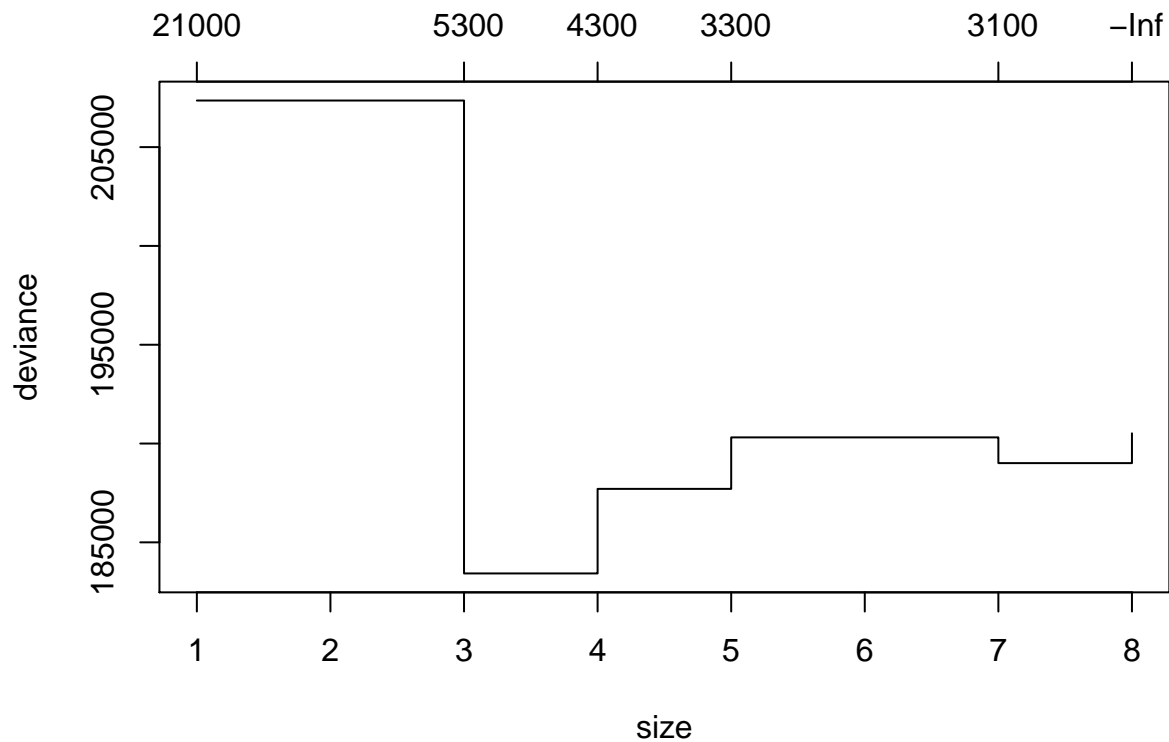
The requested plot of variable *EX* versus *MET* is shown below.



I do not find any total obvious modelation for the data shown above. A cubic spline might work out okay. I piecewise linear model with one knot would maybe be decent as well.

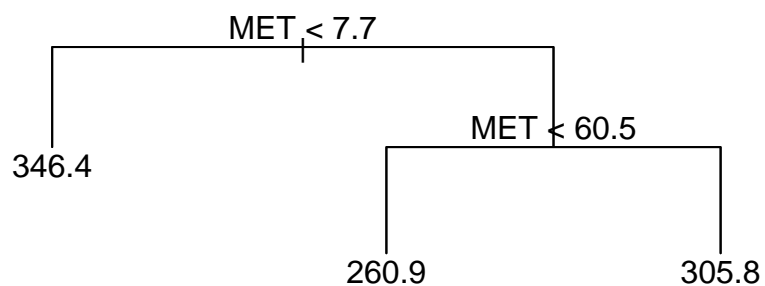
## Assignment 1.2

I set the options as in the instruction and ran the tree. Since the task was about to select the number of leaves by cross validation, I start off by showing the result of that procedure.

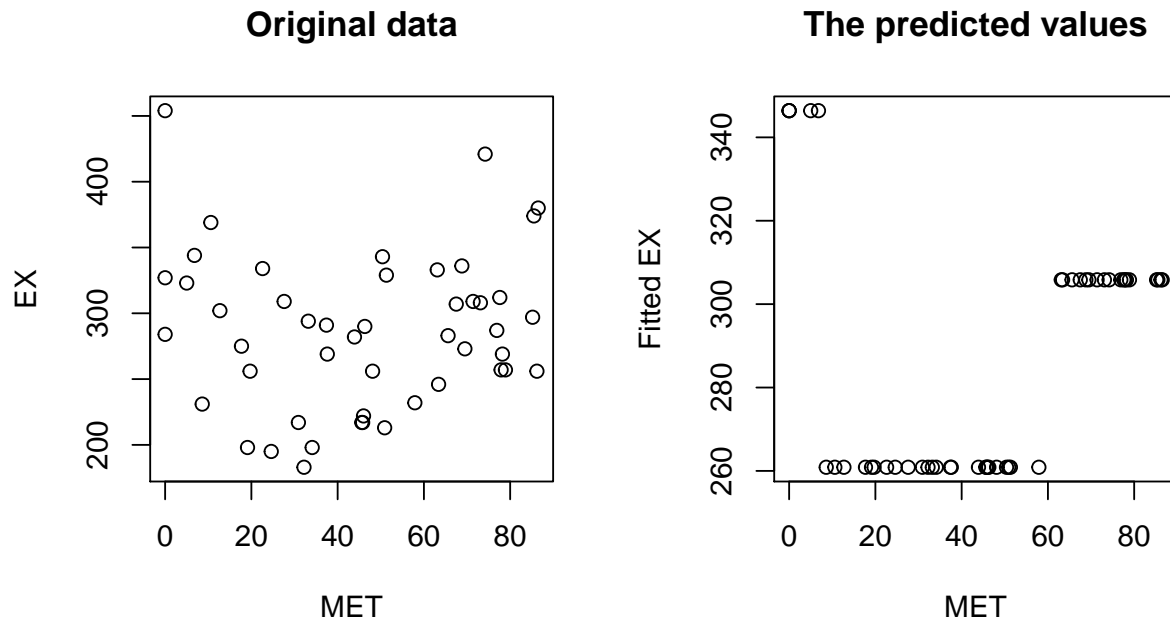


From the plot above I do conclude that three or four leaves seems to minimize the deviance. Since it's reasonable to also minimize the complexity, I therefore decide to select a tree with three leaves. The tree is shown below.

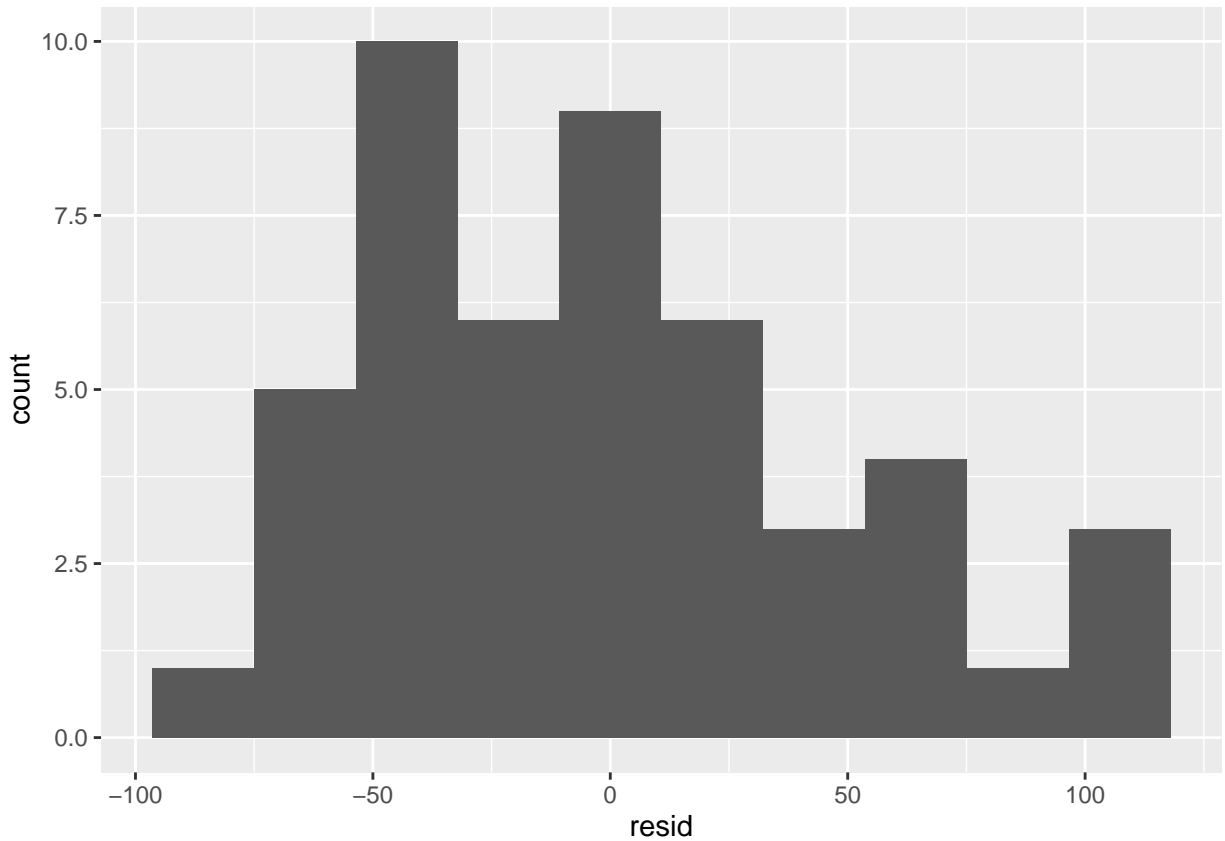
### Regression tree selected by CV, EX~MET



The tree above has three leaves, depending on the levels of  $MET$  on the points 7.7 and 60.5. The original data and the fitted data are plotted below.



The quality of the fit obviously doesn't look super good, which is intuitive just by looking at the original data. But at least I can see the pattern in the prediction the with high values to the left in the plot, lower in the middle and medium to the right. It's a underfit model however, its' confident bands wouldn't look good. The histogram of the residuals is presented below.

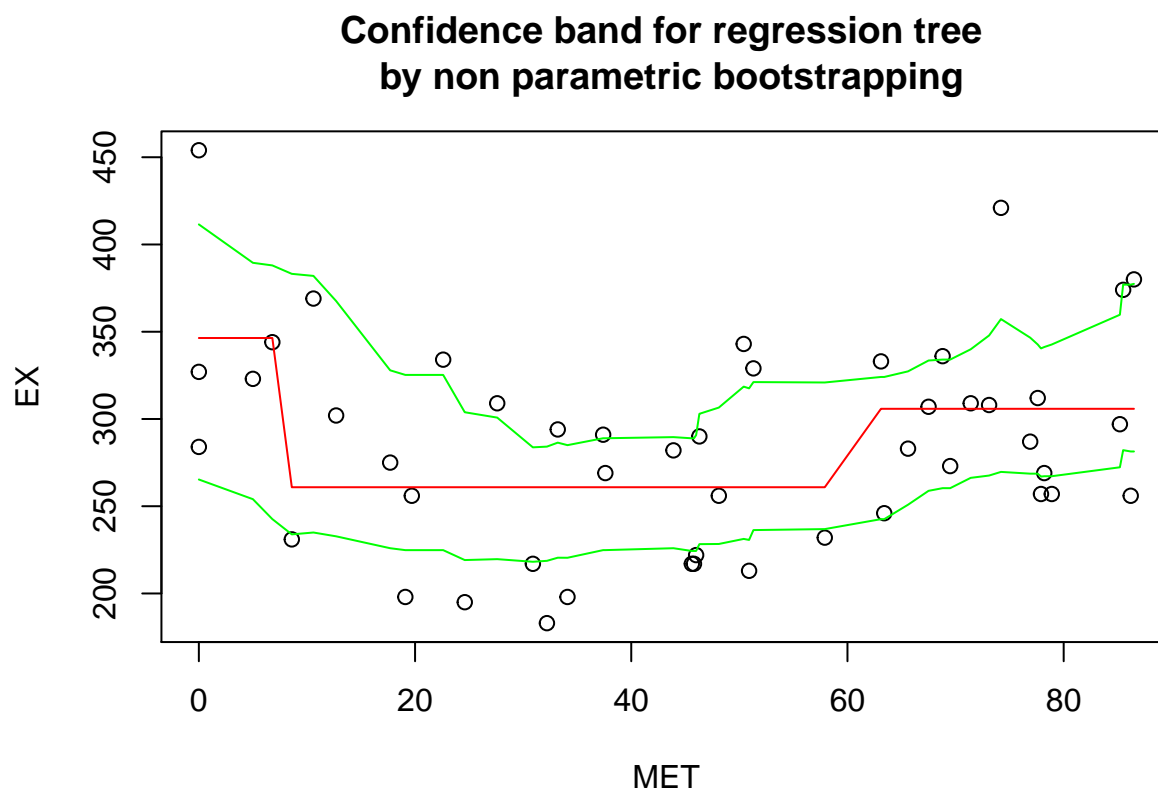


From the histogram above it's hard to tell whether the residuals are normally distributed or not. There are very few observations, which means that I can't expect to see a clear distribution anyhow. It could be a normal distribution, or maybe some gamma distribution.

If I compare the scatterplots above, I can see where the big residuals are coming from. Not all observation with MET values between 8 and 60 has low EX values for example, and not all observations with low values of MET has high EX.

## Assignment 1.3

I used the lectures from this course and some code chunks from the slides to come up with a non parametric bootstrap method. The result, i.e. the plot with the fitted values and the confidence bands, is presented below.

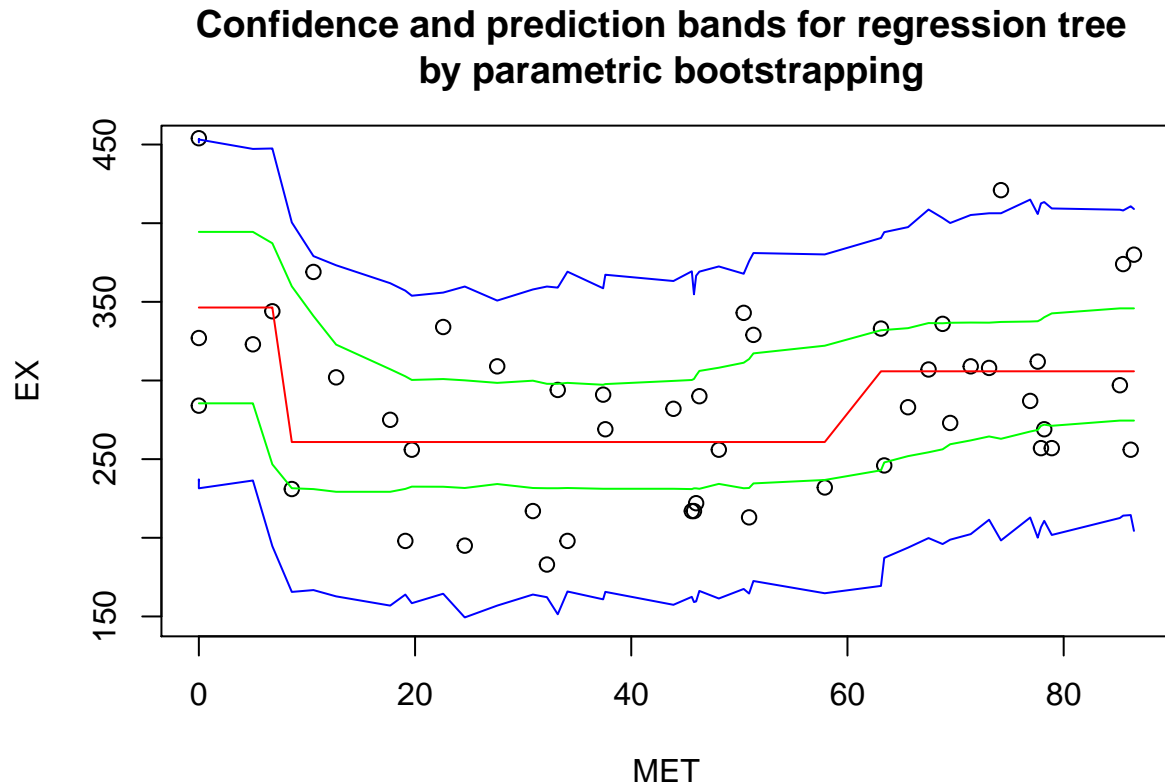


The band is quite bumpy. That is probably because some of the big outliers doesn't appear in all models created by bootstrapping. Most of the trees seem to fit observations with MET  $[30, 50]$  in the same way, that's why it has a more narrow band there, while it's much broader for example in the beginning of the series where we've got one obvious outlier.

The result from assignment 1.2, which is the red line in the plot above, doesn't seem very reliable. The confidence band is quite broad, and the confidence band is supposed to show the true value of EX given MET. It's not supposed to cover 95 % of the observations.

## Assignment 1.4

I did the parametric bootstrapping procedure to get both the confidence- and prediction band by using the code shells available in the slide for the course. The results for the parametric bootstrap are presented below.



In the plot above, the green lines represent the confidence band and the blue lines represents the prediction bands. The confidence bands is clearly more narrow here, when I've used parametric bootstrapping. Regarding this plot, the regression tree in assignment 2 (still red line in the plot above) looks more reliable. It does look like two of the observations aren't covered by the prediction band. That's as close as I can come to 5 % of the data, since the dataset only consists of 48 observations. The prediction band is practically saying that a new observation with a given MET value will be within this span, why it's reasonable to find out that two observations are outside in this case. If all observations would be inside the prediction band by a wide margin, or ten observation would be outside, I couldn't say it's definitely wrong but it would be worrying.

## Assignment 1.5

Parametric bootstrapping has the advantage of working even for small samples, which I do have in this case. The non parametric bootstrapping method doesn't require that I can tell which distribution I'm working with. When I did the parametric bootstrapping in assignment 1.4 I assumed

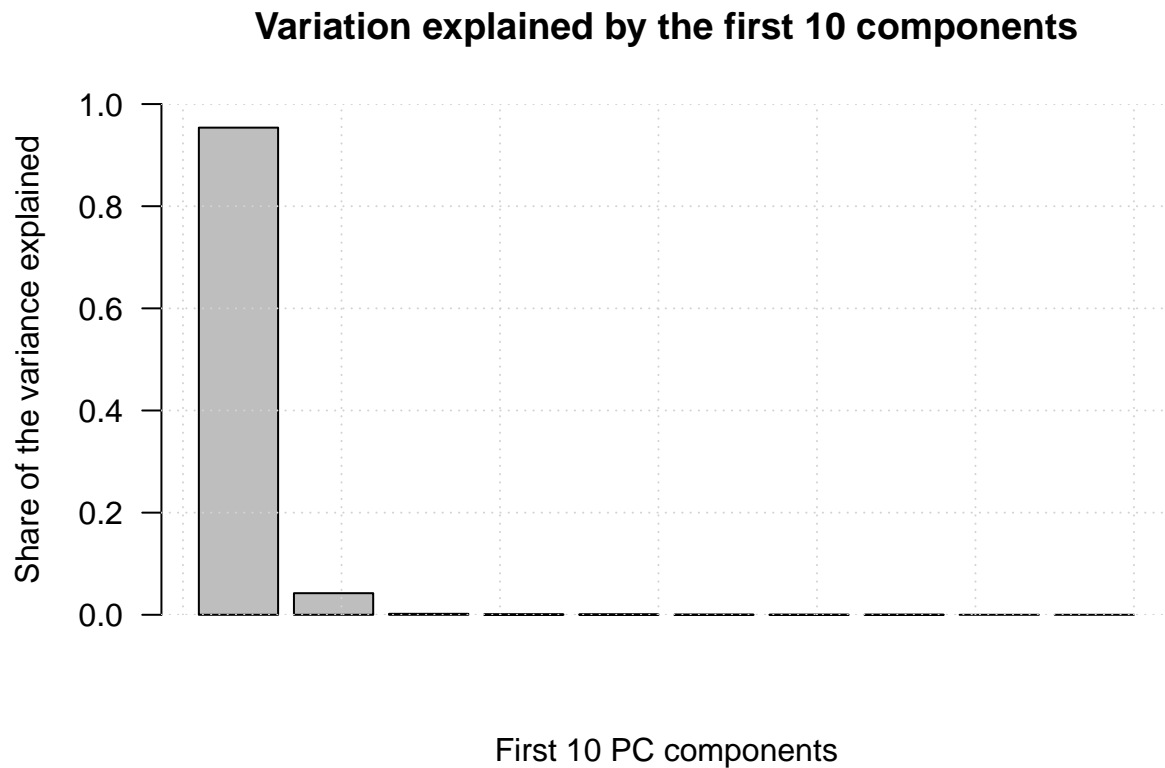
$$Y \sim N(\mu_i, \sigma^2)$$

but as I commented on the histogram in assignment 1.2, I'm not sure of that. The parametric bootstrap is shown to be more accurate if the assumed (or known of course) distribution is correct. In my case the parametric bootstrapping in 1.4 gives better results, which I suppose might indicate that I did the right thing to say that the residuals shown in the histogram in assignment 1.2 might come from a normal distribution. Overall I'd say that the parametric bootstrapping, i.e. what's done in assignment 1.4, is more appropriate in this case.

## Assignment 2 Principal components

### Assignment 2.1

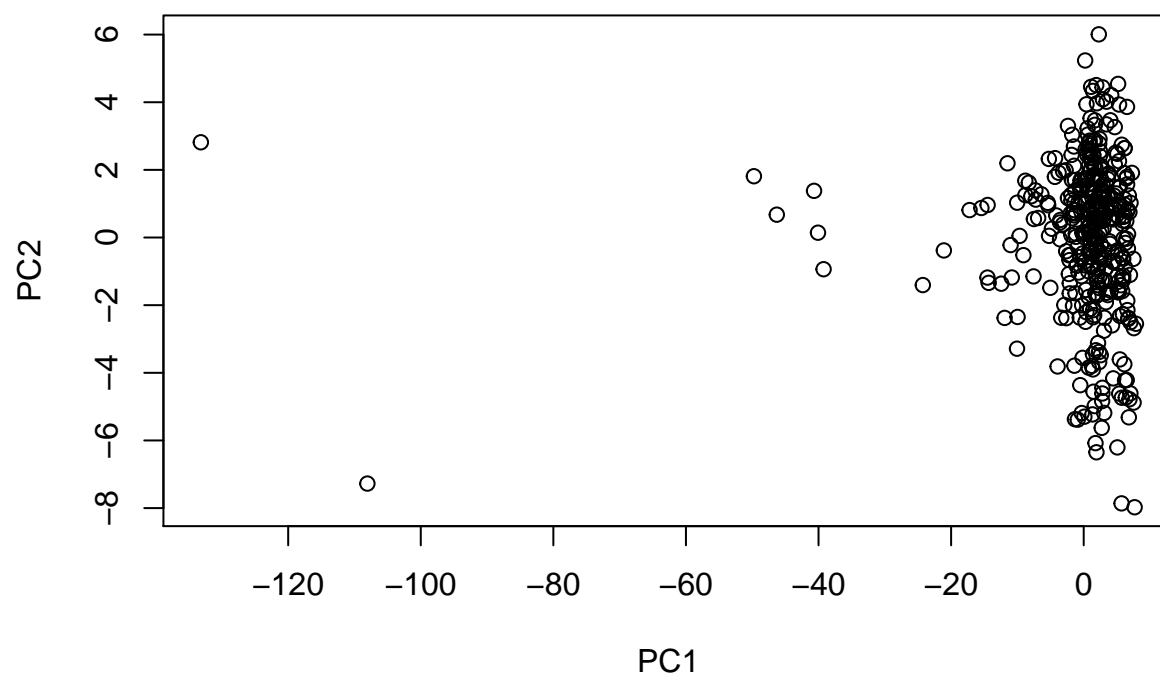
I didn't use the `prcomp()` function but did the calculations myself. The plot explaining how much of the variation the first ten principal components explains is shown below.



The plot above tells me that the first component explains very much, about 95 %. The plot tells me that one or two PC:s should be extracted, depending on how much variance I need to explain. The instructions tell me to explain at least 99 %, which means that I choose 2 components. That is, the first two bars in the plot above explains at least 99 %. The requested plot of the scores for the two PC:s is shown below.



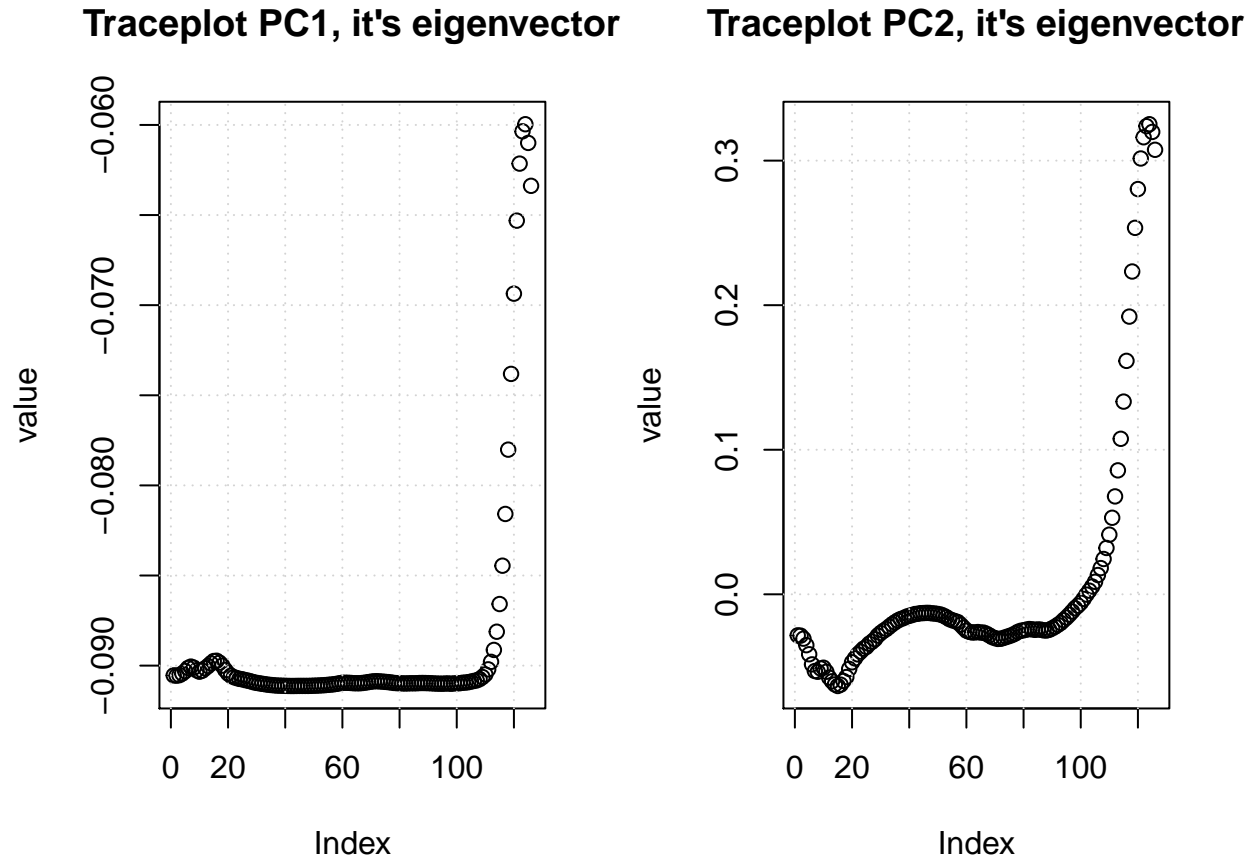
### The scores for PC1 and PC2, PCA



The plot of the scores for PC1 and PC2 shows that there are a few observations of unusual diesel fuels. There are two major outliers according to PC1 especially. Those two seem to be more normal according to PC2, even though the observation on the left bottom of the plot can be seen as an outlier in both PCs. Yes, it seems to a few unusual fuels, especially on the PC1.

## Assignment 2.2

The trace plots for the two selected PCs are shown below.

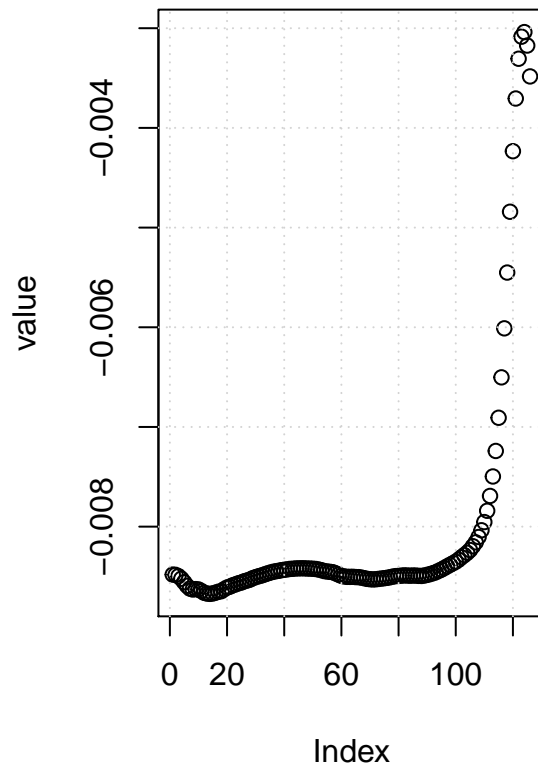


From the left plot above, the loadings for PC1, I conclude that all original variables contribute to PC1. That can be seen since all 126 variable indexes on the X axis has a value separated from zero. For PC2 on the other side, not all original variables contributes to explain PC2. PC2 seems to be explained mainly by the first 25 and the last 20 original variables, i.e. those who are separated from 0 in the plot.

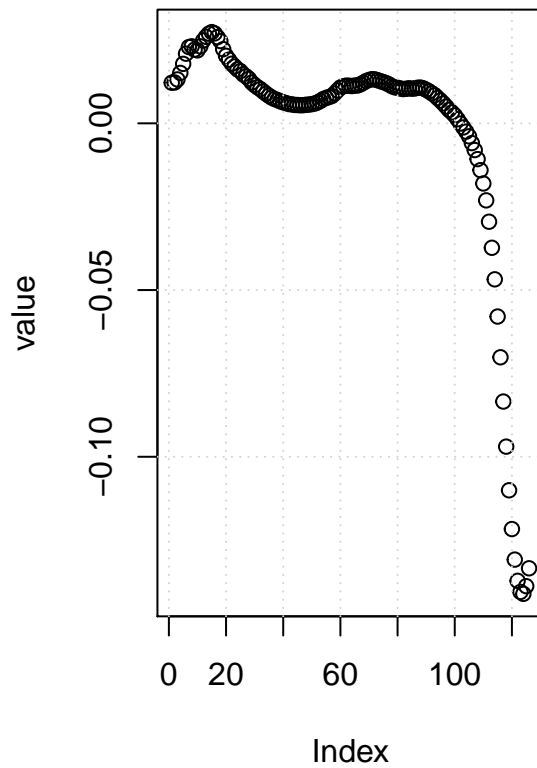
## Assignment 2.3a

I ran the fastICA procedure and did the calculations as instructed, and the resulting traceplots is presented below. To compare them to the PCA-traceplots I decided to plot all four plots next to each other.

**Traceplot IC1, it's eigenvector**



**Traceplot IC2, it's eigenvector**



The traceplots above for the fastICA model look very much the same as the traceplots for the two principal components. I think that

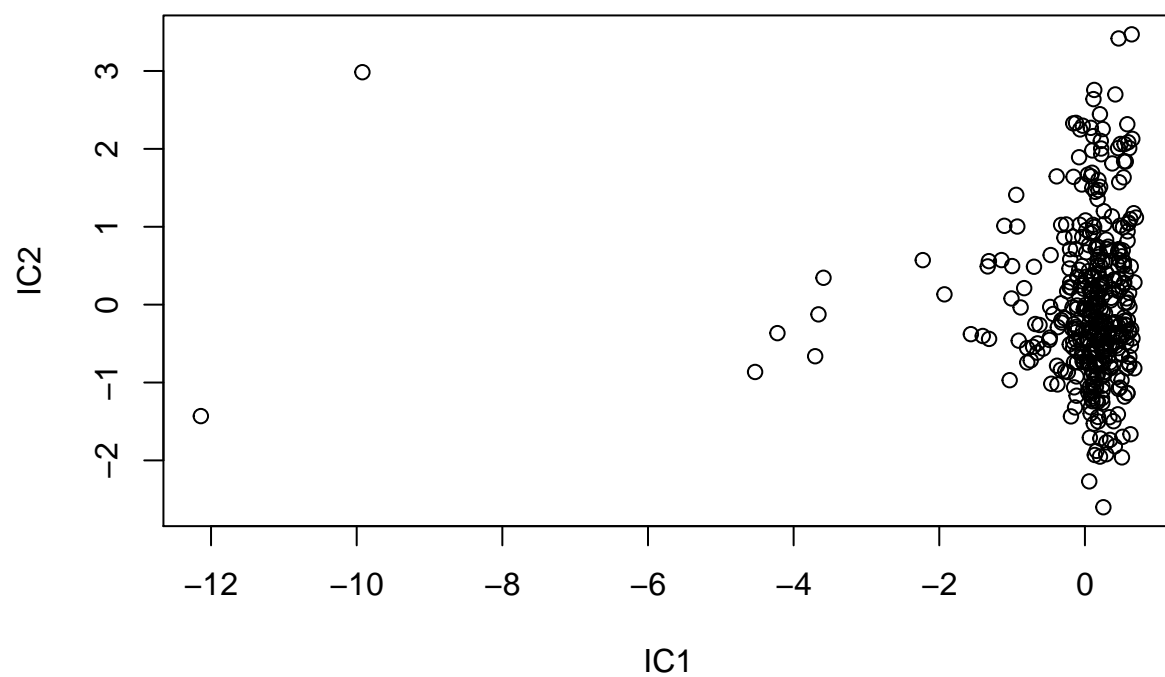
$$W'$$

is the eigenvectors of the ICA-model, corresponding to the eigenvectors in the U matrix in PCA.

FORMELN OCH NEGENTROPY

The scores for the first two latent features from the fastICA model, i.e. IC1 and IC2, are shown below.

### The scores for the two latent features, ICA



Looks the same but mirrored in my case because of the negative PC1 in my PCA.