Lab 3

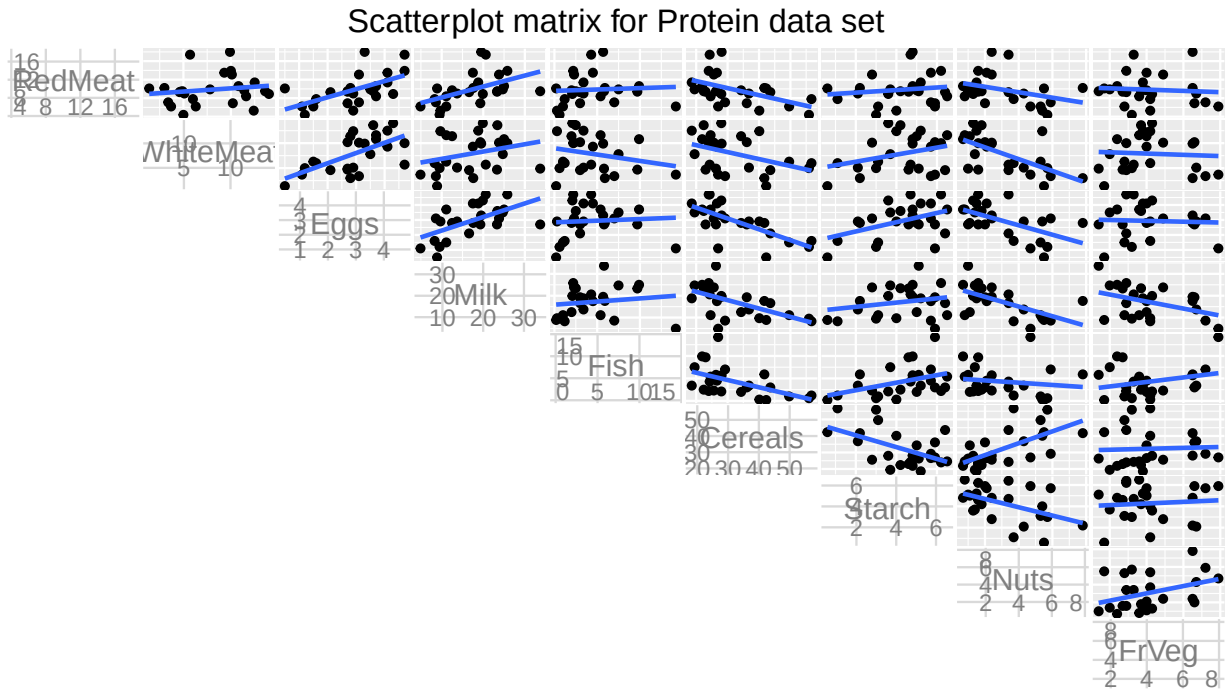
Gustav Sternelöv

September 22, 2016

Assignment 1

1.

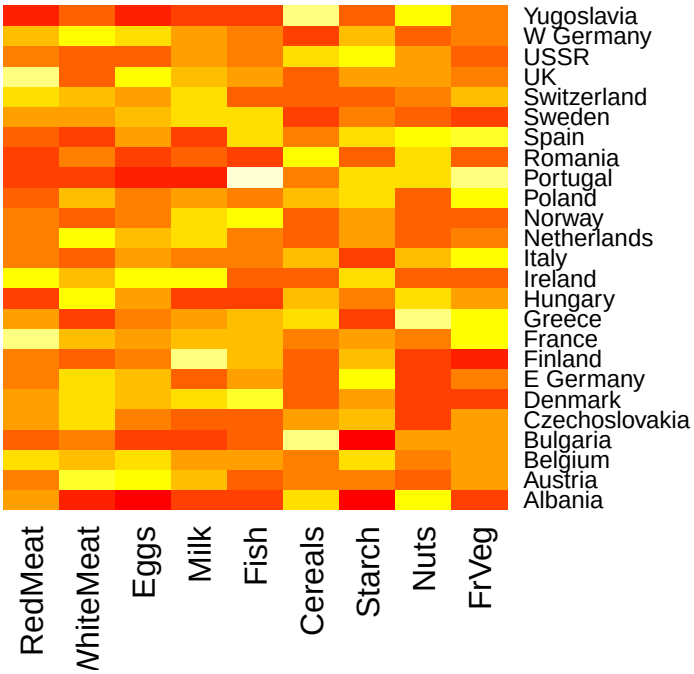
The scatterplot matrix with the variables from the data set *protein* is presented below. For each pair of variables is a linear regression line plotted.



The relationship between the variables is interpreted by looking at the slope of the regression curves. Some relatively strong relationships can be seen, for example between eggs and white meat and between eggs and red meat. Milk and eggs and nuts and starch are two other examples of variables that has a positive relationship. Examples of negative relationships can be seen between starch and cereals and between nuts and white meat.

2.

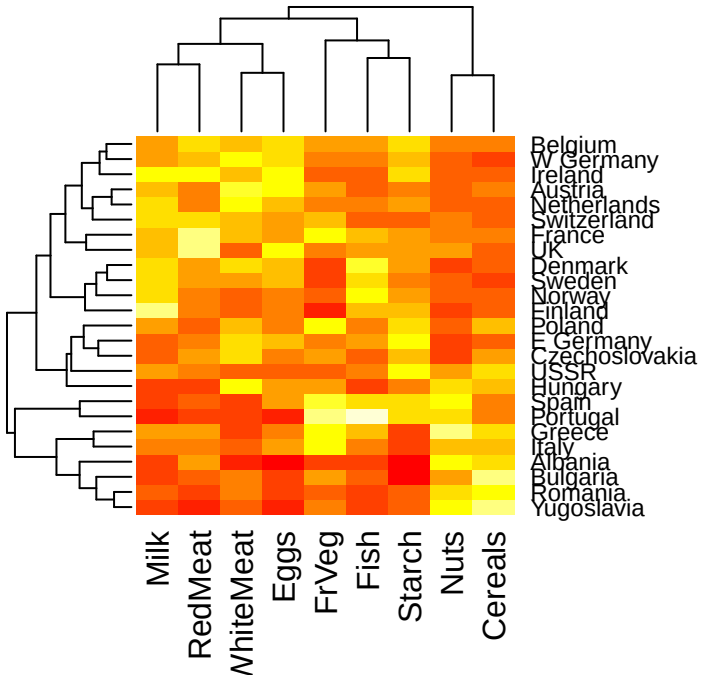
A heatmap for the data with no permutations over the rows or the columns is plotted.



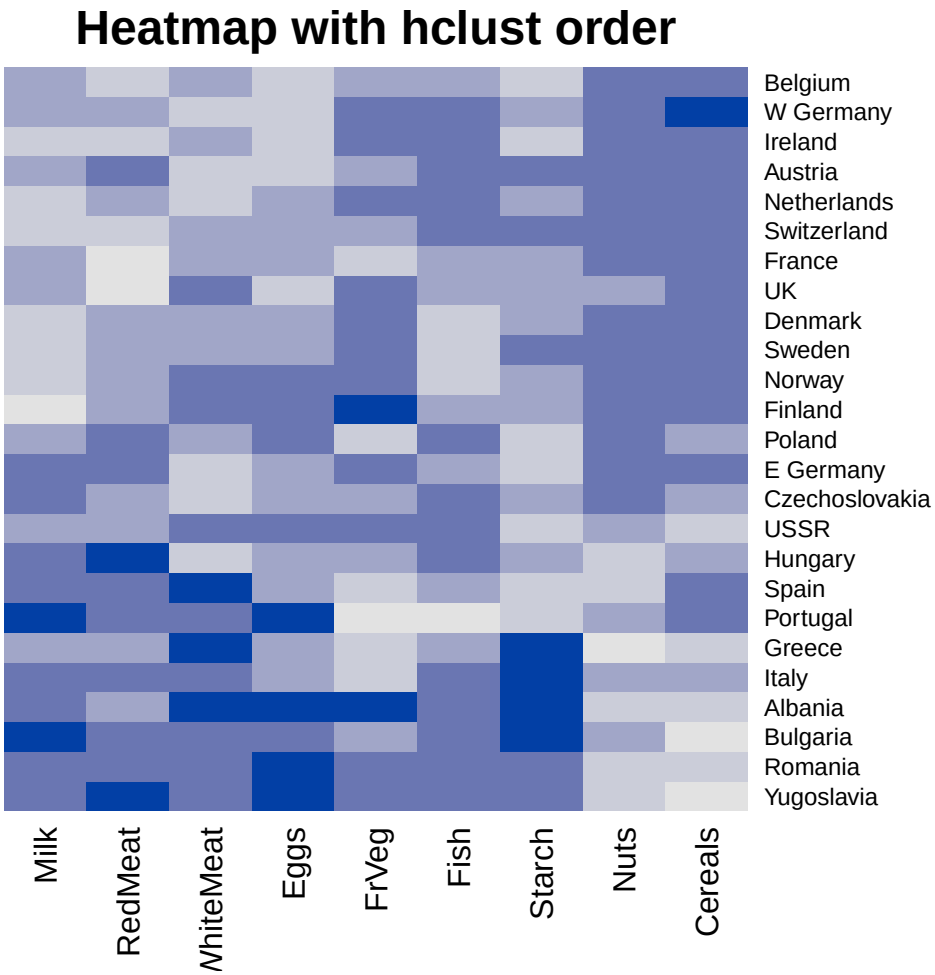
It is hard to find any clear clusters or outliers in the heatmap. Perhaps a correaltion can be noted for red meat and white meat where low values for the former coincides with low values for the latter. The same pattern seem to be present also for high values. Eggs and milk seem to have a fairly similar correlation.

3.

Next, a heatmap given by a two-way hierarchical clustering is created.



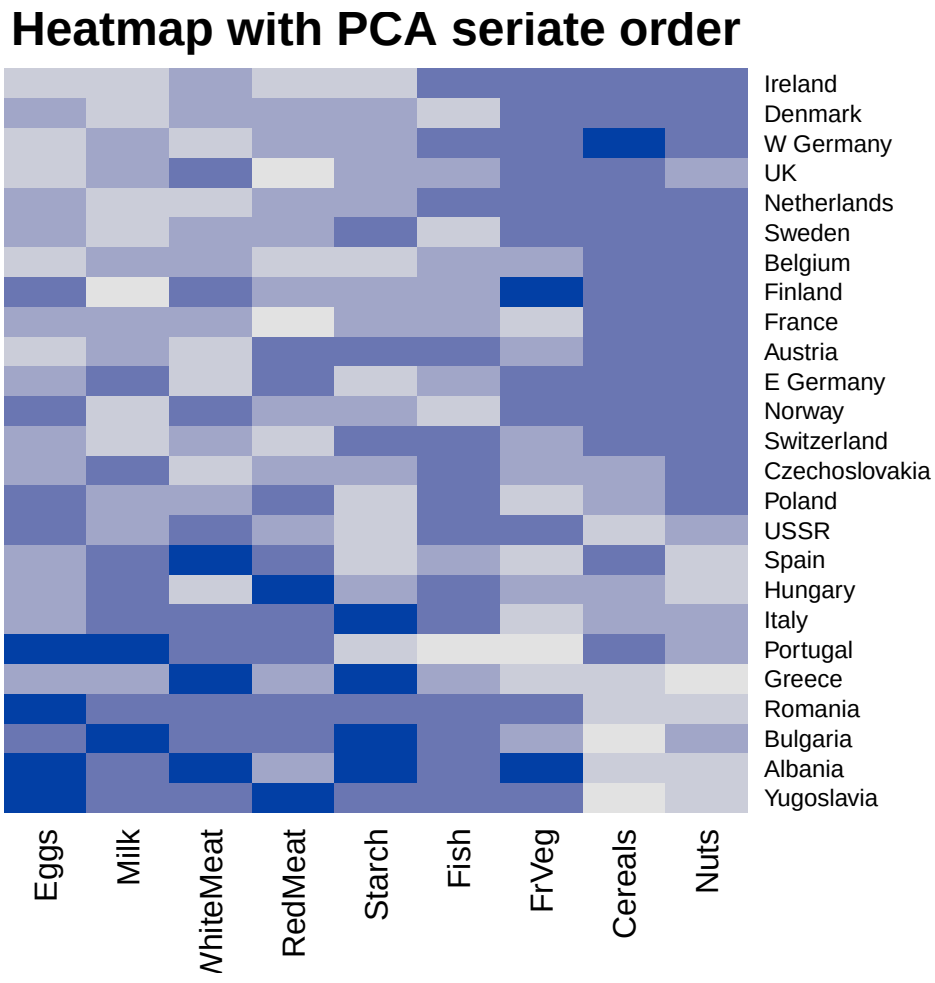
This plot is modified so that another color palette is used and the dendrograms are removed.



Two different clusters are thougt to be seen in the new heatmap, one at the bottom and one at the top. The first cluster, at the bottom, contains the countries with low values for milk, red meat, white meat and eggs and high values for nuts and cereals. The second cluster has the opposite pattern with high values for milk, red meat, white meat and eggs and low values for nuts and cereals. One possible outlier is the observation in the middle of the heatmap which has a high value for milk and a very low value for fruits and vegetables.
The first cluster contains the countries Yugoslavia, Romania, Bulgaria, Albania, Italy, Greece, Portugal and Spain. This is countries from eastern Europe and southern Europe. The second cluster contains Belgium, W Germany, Ireland, Austria, Netherlands, France, UK and Denmark. This is mostly countries from western Europe. It is then 8 countries left, which depending on how many clusters you want either could be splitted into two more clusters or assigned into the most fitting one of the two mentioned clusters. Sweden, Norway, Finland and Poland could either be a third cluster or added to the second. East Germany, Czechoslovakia, USSR and Hungary could be a fourth cluster or added to the first.
In general are countries in the same region clustered together. A reasonable result since it is fairly logical that people from the same region in general consumes the same type of products.

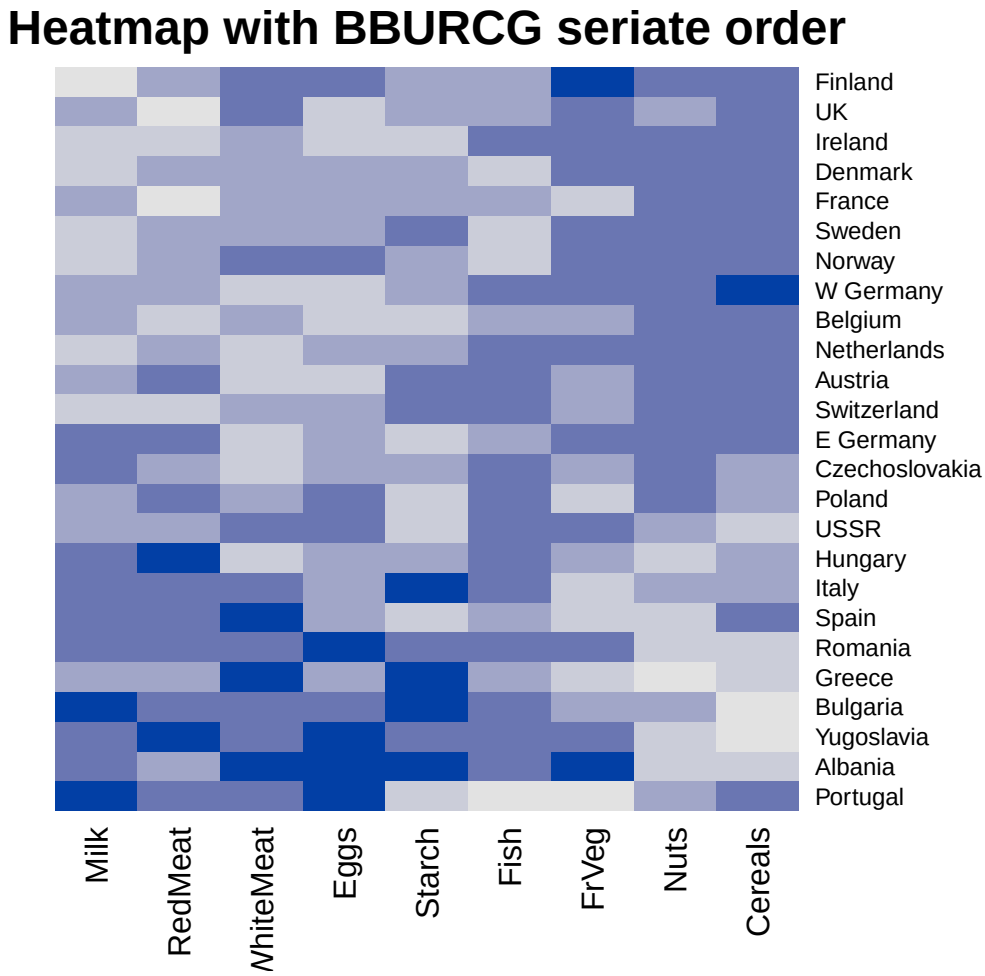
4.

The heatmap for the PCA seriations is plotted below.



Almost the same variables as before (red meat, white meat, eggs, milk, nuts, cereals) are the most important. The only difference is that the impact of the variable for fruits and vegetables is a bit clearer. Regarding the clusters can the same general pattern be seen. One cluster consists of countries from eastern and southern Europe and one of countries from western and northern Europe.

The corresponding heatmap for the BBURCG seriation.



Again, the order of the rows and columns has changed a little but the same overall pattern is present. Perhaps this is the heatmap that is easiest to analyze. The observations at the top has low values for milk, red meat, white meat and eggs and low values for fruits and vegetables, nuts and cereals. The opposite general pattern is seen for the countries at the bottom half of the heatmap. Just as before, the clusters consists of western/northern countries and eastern/southern countries.

5.

Common between the scatterplot matrix and the heatmaps is the information regarding the relationships between variables. The heatmaps clearly showed the relationship between white meat, red meat, eggs and milk. All of these relationships were apparent already during the analysis of the scatterplot matrix.

Assignment 2

1.

The wordcloud for the feedback from the pleased customers.

Large words in the wordcloud are the name of the brand, Casio, and affection words like “great” and “like”. Other, assumed, positives like “price”, “nice”, “good” and “looks” can be seen among the most frequently used words.

The wordcloud created from the feedback given by the unpleased customers.

Frequently used words by unpleased customers are Casio, “time” and Amazon which is the website where the clock was bought. Some other, relatively, frequent words are “back”, “great” and “like”.

2-3.

For the given set of connector words are four phrase nets for each text file created. Pictures of all phrase-nets are included in appendix 1. Starting with the unpleased customers the first phrase net showed a link between display and useless. By setting these words as keywords for a word tree it was given that people are complaining about the quality of the display. In the second phrase net there was an evident link between replace and battery. The word tree then added the information that some of the customers were unsatisfied with the life span of the battery and the cost of replacing it. The third phrase net had links from keeping to terrible and to lousy. These links comes from customers who has complained about that the the clock sometimes stops, or in some other way fails to keep time. In the final phrase net the link I found most interesting was the one between piece and junk. This link is a result of the unpleased customers being unhappy with the quality of the clock.

The last four phrase nets in the appendix are created from the feedback given by the pleased customers. In the first are the links from watch to awesome and from it to comfortable interesting. A further look with the word tree gives that given the price are these customers happy with the clock in some different ways. A link from battery to change and to replace can be seen for the pleased customers as well. However, for these customers the context more often is an old clock with a bad battery or that it not is such a big problem to replace the battery. The third phrase net shows links from night to readable and to viewable. In this case most of the customers are happy with the readability at night and some are not. In the last phrase nets links from years to service and from lots to compliments were among the most interestig ones. Customes are pleased since the clock have worked for years.

Appendix 1 - Phrase nets

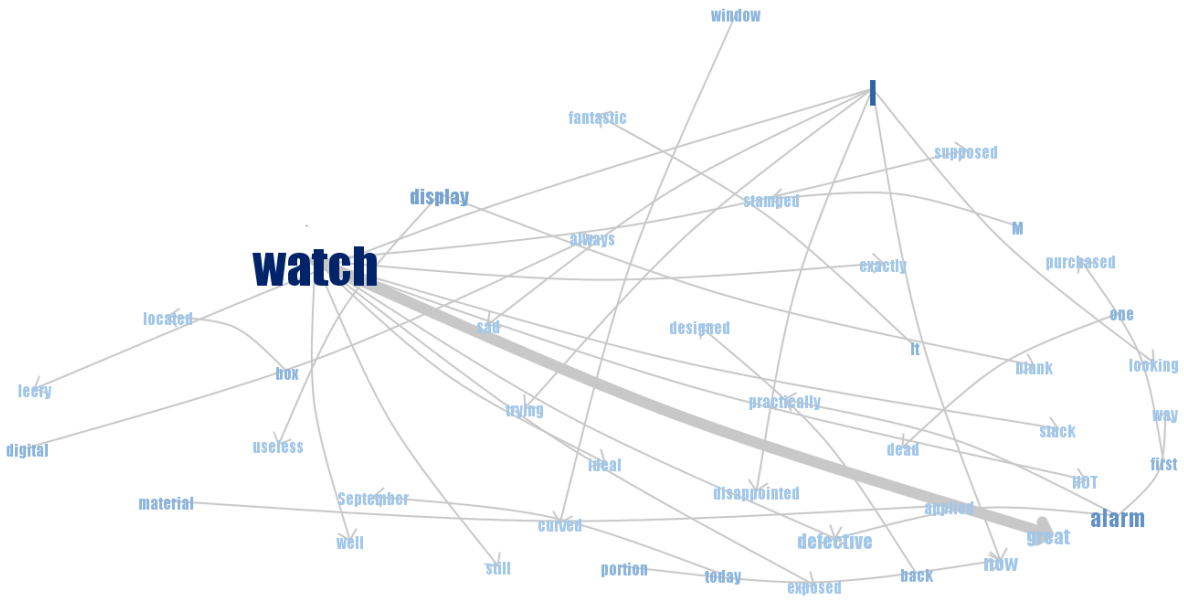


Figure 1: oneTwo1: Connecting words: am, is, are, was, were

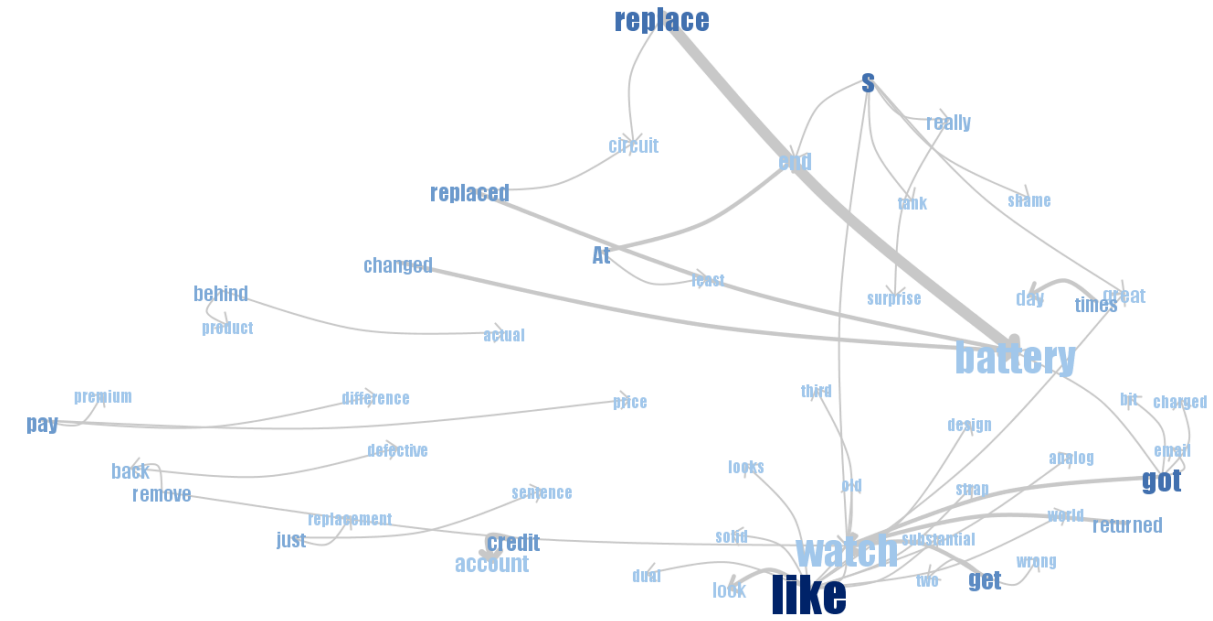


Figure 2: oneTwo2: Connecting words: a, the

Figure 3: oneTwo3: Connecting words: at

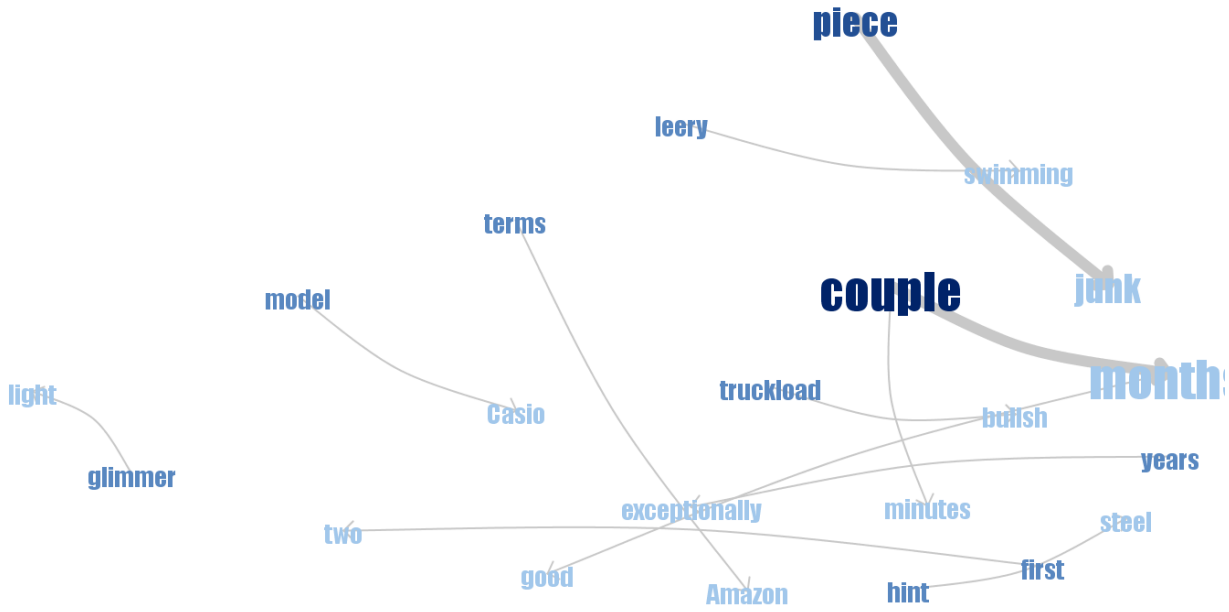


Figure 4: oneTwo4: Connecting words: of

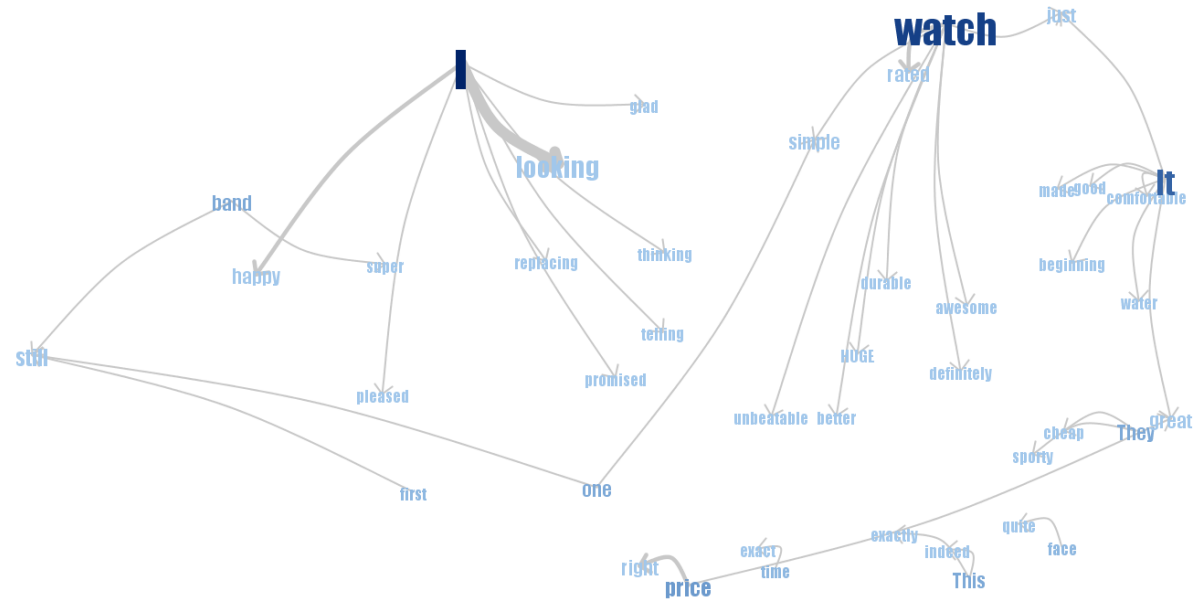


Figure 5: Five1: Connecting words: am, is, are, was, were

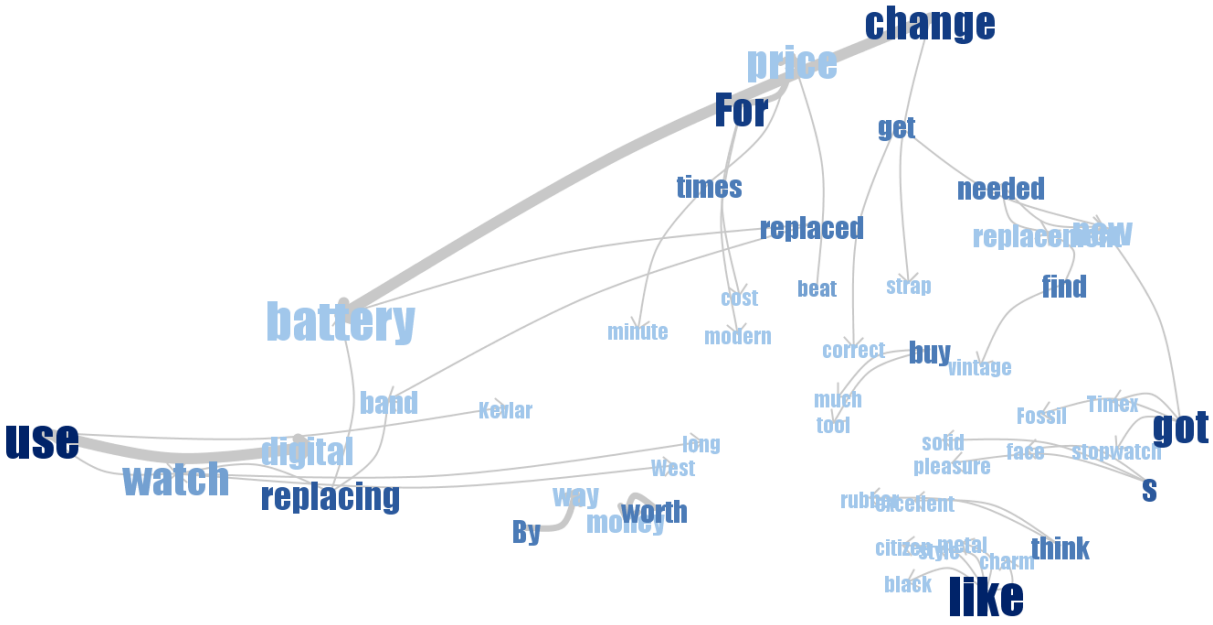


Figure 6: Five2: Connecting words: a, the

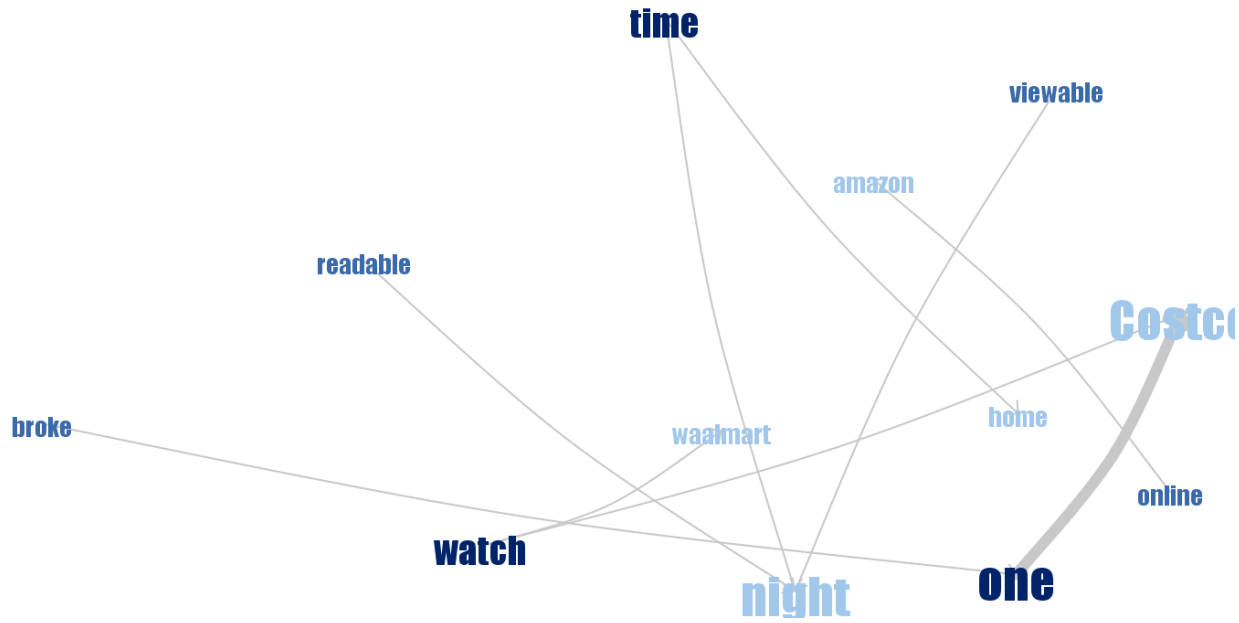


Figure 7: Five3: Connecting words: at

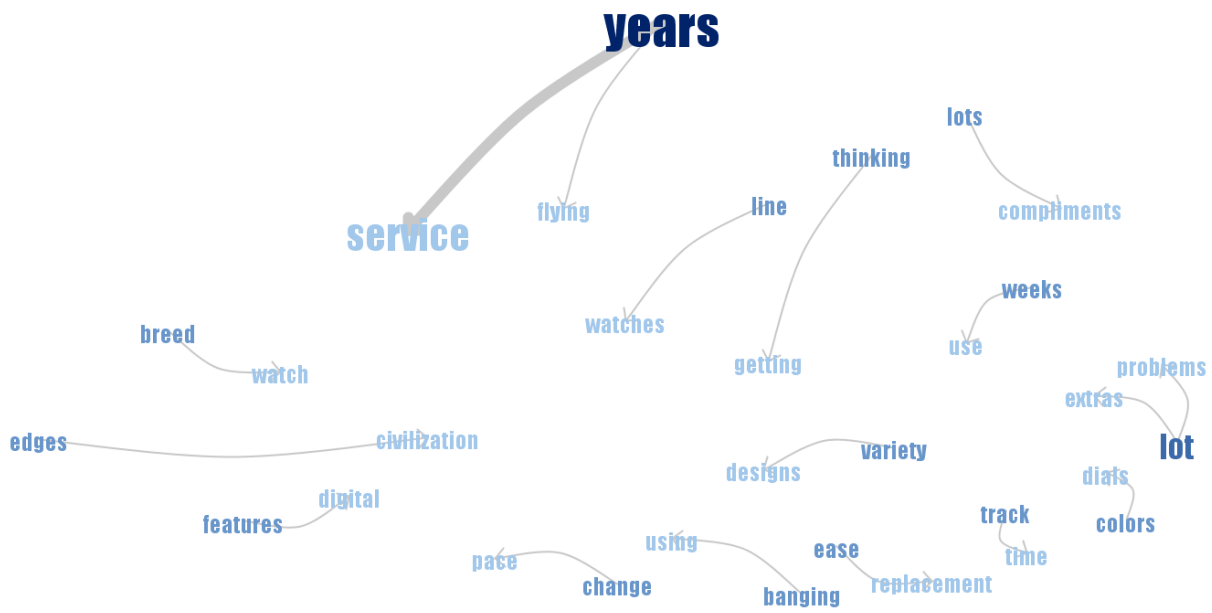


Figure 8: Five4: Connecting words: of

Appendix 2 - R-code

```
## ---- echo=FALSE, message=FALSE, warning=FALSE--------------------------
library(ggplot2)
library(GGally)
library(seriation)
library(colorspace)

# assignment 1
protein <- read.table("C:\\Users\\Gustav\\Documents\\Visualization\\Lab3\\protein.txt",
    sep = "\t", header = TRUE)

## ---- echo=FALSE, fig.height=4.5, fig.width=8---------------------------- 1
## code to add regression curves
regCurve <- function(data, mapping, method = "lm") {
    Plot <- ggplot(data = data, mapping = mapping) + geom_point() + geom_smooth(method = method,
        se = FALSE)
    Plot
}
ggpairs(protein, columns = 2:10, upper = list(continuous = regCurve), lower = list(continuous = "blank"
    title = ("Scatterplot matrix for Protein data set"), diag = list(continuous = "blankDiag"),
    showStrips = TRUE, axisLabels = "internal")

## ---- echo=FALSE, fig.height=3.5, fig.width=6---------------------------- 2
row.names(protein) <- protein[, 1]
scaleP <- as.matrix(scale(protein[, 2:10]))

heatmap(scaleP, Rowv = NA, Colv = NA, scale = "none")
```

```
## ---- echo=FALSE, fig.height=3.5, fig.width=6---------------------------- 3
pl <- heatmap(scaleP, scale = "none")


## ---- echo=FALSE, fig.height=5, fig.width=6------------------------------
reOrd <- scaleP[(pl$rowInd), pl$colInd]
heatmap(reOrd, scale = "none", Rowv = NA, Colv = NA, col = sequential_hcl(5),
    main = "Heatmap with hclust order")
# pimage(reOrd, col=sequential_hcl(5), key=TRUE, axes = 'x', main='Heatmap
# with hclust order') protein[rev(pl$rowInd), c(1,pl$colInd +1)]

## ---- echo=FALSE, fig.height=5, fig.width=6------------------------------ 4
## pca
seri_pca <- seriate(scaleP, method = "PCA")
seri_pca2 <- seriate(t(scaleP), method = "PCA")
ordPCARow <- get_order(seri_pca)
ordPCACol <- get_order(seri_pca2)
reOrdPCA <- scaleP[rev(ordPCARow), ordPCACol]

heatmap(reOrdPCA, Rowv = NA, Colv = NA, scale = "none", col = sequential_hcl(5),
    main = "Heatmap with PCA seriate order")

# protein[rev(ordPCA),]

## ---- echo=FALSE, fig.height=5, fig.width=6------------------------------
## anti-robinson
rowdist <- dist(scaleP)
coldist <- dist(t(scaleP))
order1 <- seriate(rowdist, "BBURCG")
order2 <- seriate(coldist, "BBURCG")
ord1 <- get_order(order1)
ord2 <- get_order(order2)
reordmatr <- scaleP[(ord1), ord2]

ordBB <- ser_permutation(ord1, ord2)
heatmap(reordmatr, Rowv = NA, Colv = NA, scale = "none", col = sequential_hcl(5),
    main = "Heatmap with BBURCG seriate order")

# pimage(scaleP,ordBB,col=sequential_hcl(5), key=FALSE, axes = 'x',
# main='Heatmap with BBURCG seriate order') protein[rev(ord1),c(1,ord2+1)]

## ---- echo=FALSE, message=FALSE, warning=FALSE--------------------------
## Assignment 2
library(tm)
library(wordcloud)
library(RColorBrewer)
pleased <- read.table("C:\\Users\\Gustav\\Documents\\Visualization\\Lab3\\five.txt",
    header = F, sep = "\n")
unpleased <- read.table("C:\\Users\\Gustav\\Documents\\Visualization\\Lab3\\OneTwo.txt",
    header = F, sep = "\n")  #Read file

## ---- echo=FALSE--------------------------------------------------------
mycorpus <- Corpus(DataframeSource(pleased))  #Creating corpus (collection of text data)
```

```
mycorpus <- tm_map(mycorpus, removePunctuation)
mycorpus <- tm_map(mycorpus, function(x) removeWords(x, stopwords("english")))
# Remove some more words
mycorpus <- tm_map(mycorpus, function(x) removeWords(x, c("watch", "one", "watches")))
tdm <- TermDocumentMatrix(mycorpus)  #Creating term-document matrix
m <- as.matrix(tdm)
v <- sort(rowSums(m), decreasing = TRUE)  #Sum up the frequencies of each word
d <- data.frame(word = names(v), freq = v)  #Create one column=names,second=frequences
pal <- brewer.pal(4, "Dark2")
pal <- pal[-(1:2)]  #Create palette of colors
wordcloud(d$word, d$freq, scale = c(4, 0.3), min.freq = 2, max.words = 100,
    random.order = F, rot.per = 0.15, colors = pal, vfont = c("sans serif",
        "plain"))

## ---- echo=FALSE-----------------------------------------------------------
mycorpus <- Corpus(DataframeSource(unpleased))  #Creating corpus (collection of text data)
mycorpus <- tm_map(mycorpus, removePunctuation)
mycorpus <- tm_map(mycorpus, function(x) removeWords(x, stopwords("english")))
# Remove some more words
mycorpus <- tm_map(mycorpus, function(x) removeWords(x, c("watch", "one", "watches")))
tdm <- TermDocumentMatrix(mycorpus)  #Creating term-document matrix
m <- as.matrix(tdm)
v <- sort(rowSums(m), decreasing = TRUE)  #Sum up the frequencies of each word
d <- data.frame(word = names(v), freq = v)  #Create one column=names,second=frequences
pal <- brewer.pal(4, "Dark2")
pal <- pal[-(1:2)]  #Create palette of colors
wordcloud(d$word, d$freq, scale = c(4, 0.3), min.freq = 2, max.words = 100,
    random.order = F, rot.per = 0.15, colors = pal, vfont = c("sans serif",
        "plain"))

## ----code=readLines(knitr::purl('C:\\Users\\Gustav\\Documents\\Visualization\\Lab3\\Lab3_1.Rmd',docume
## = 1)), eval = FALSE, tidy=TRUE---- NA
```