EMD Project 1

Mateusz Kapiszewski, Jakub Sztyma

01 December, 2019

Summary

The analysis of the data allowed to connect the size of the herring caught with the attributes from the data set. Many attributes have a strong correlations with each other. Based on the data used in the study, it can be concluded that the real impact on the length of herring caught have, most of all:

1. sst: temperature at the water surface [°C];
2. cfin2: plankton availability [compaction Calanus finmarchicus species 2].

The remaining attributes largely correlate with the above selected or are not correlated with the length of herring. Adding them did not improve the accuracy of the linear regression.

Used libraries

```
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
library(DAAG)
```

```
## Loading required package: lattice
```

```
library(ggplot2)
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

Set seed to make experiment results recurrent

```
set.seed(5)
```

Read data from file

```
filename <- "sledzie.csv"
df <- read.csv(filename, na.strings=c("?"))
print("Is data.frame instance?")
```

```
## [1] "Is data.frame instance?"
```

```
print(is.data.frame(df))
```

```
## [1] TRUE
```

Parse data (replace Na with a column mean value)

```
df <- df[, names(df) != 'X'] # Remove column X
df <- data.frame(
   sapply( df,
       function(x)ifelse(is.na(x), mean(x, na.rm=TRUE), x)
   )
)
```

Data set summary

```
print("Size of cleared data:")
```

```
## [1] "Size of cleared data:"
```

```
print(nrow(df))
```

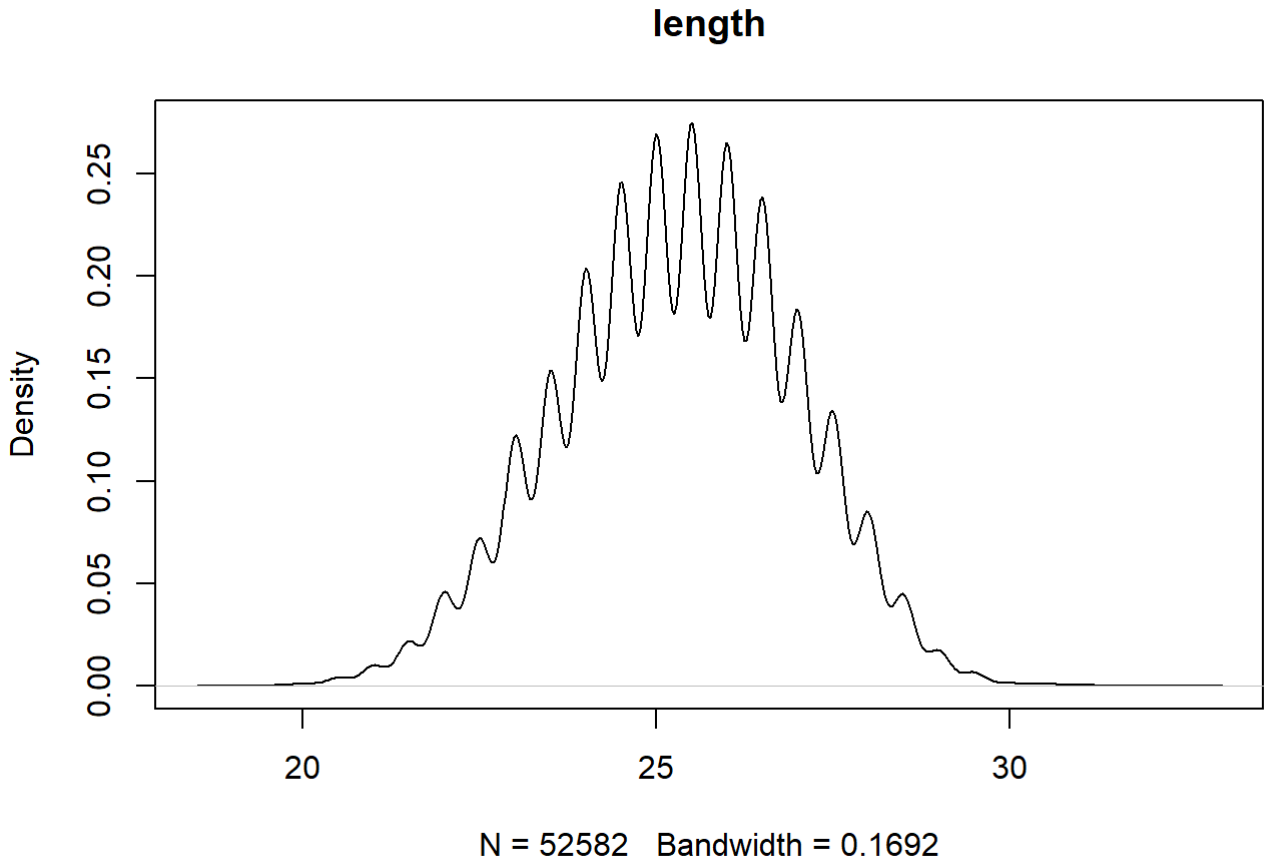
```
## [1] 52582
```

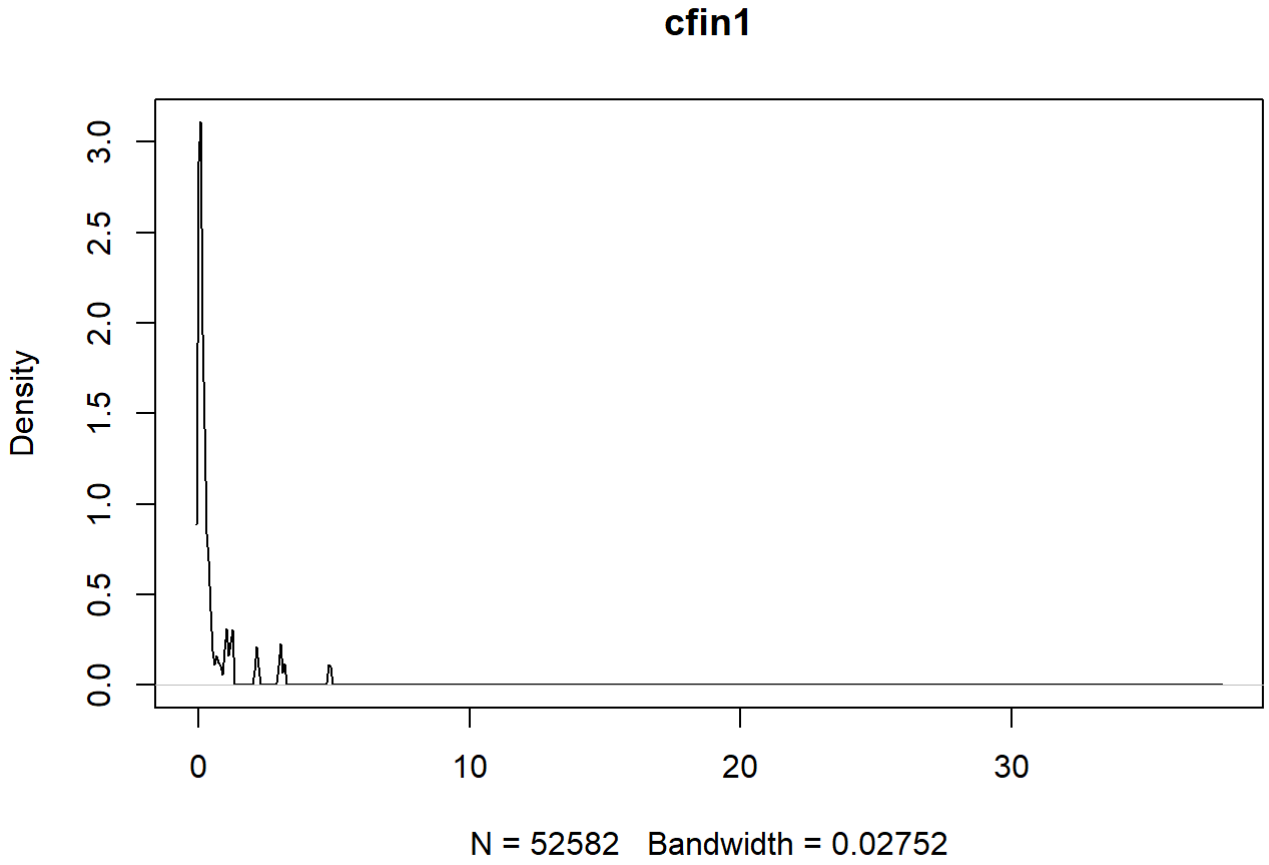
```
print(summary(df))
```

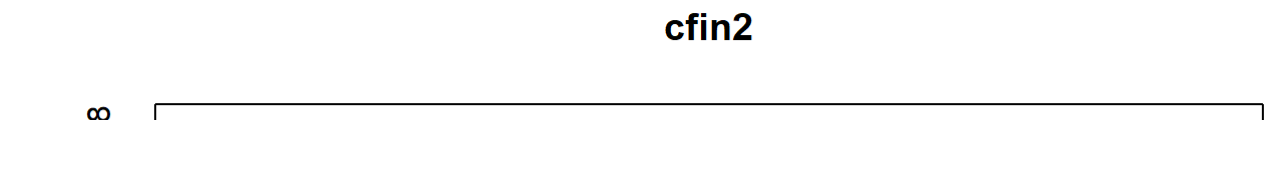
```
##      length         cfin1             cfin2             chel1
##  Min.   :19.0   Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.000
##  1st Qu.:24.0   1st Qu.: 0.0000   1st Qu.: 0.2778   1st Qu.: 2.469
##  Median :25.5   Median : 0.1333   Median : 0.7012   Median : 6.083
##  Mean   :25.3   Mean   : 0.4458   Mean   : 2.0248   Mean   :10.006
##  3rd Qu.:26.5   3rd Qu.: 0.3603   3rd Qu.: 1.9973   3rd Qu.:11.500
##  Max.   :32.5   Max.   :37.6667   Max.   :19.3958   Max.   :75.000
##      chel2            lcop1             lcop2             fbar
##  Min.   : 5.238   Min.   :  0.3074   Min.   : 7.849   Min.   :0.0680
##  1st Qu.:13.589   1st Qu.:  2.5479   1st Qu.:17.808   1st Qu.:0.2270
##  Median :21.435   Median :  7.1229   Median :25.338   Median :0.3320
##  Mean   :21.221   Mean   : 12.8108   Mean   :28.419   Mean   :0.3304
##  3rd Qu.:27.193   3rd Qu.: 21.2315   3rd Qu.:37.232   3rd Qu.:0.4560
##  Max.   :57.706   Max.   :115.5833   Max.   :68.736   Max.   :0.8490
##       recr              cumf             totaln             sst
##  Min.   : 140515   Min.   :0.06833   Min.   : 144137   Min.   :12.77
##  1st Qu.: 360061   1st Qu.:0.14809   1st Qu.: 306068   1st Qu.:13.63
##  Median : 421391   Median :0.23191   Median : 539558   Median :13.86
##  Mean   : 520367   Mean   :0.22981   Mean   : 514973   Mean   :13.87
##  3rd Qu.: 724151   3rd Qu.:0.29803   3rd Qu.: 730351   3rd Qu.:14.16
##  Max.   :1565890   Max.   :0.39801   Max.   :1015595   Max.   :14.73
##       sal              xmonth            nao
##  Min.   :35.40   Min.   : 1.000   Min.   :-4.89000
##  1st Qu.:35.51   1st Qu.: 5.000   1st Qu.:-1.89000
##  Median :35.51   Median : 8.000   Median : 0.20000
##  Mean   :35.51   Mean   : 7.258   Mean   :-0.09236
##  3rd Qu.:35.52   3rd Qu.: 9.000   3rd Qu.: 1.63000
##  Max.   :35.61   Max.   :12.000   Max.   : 5.08000
```

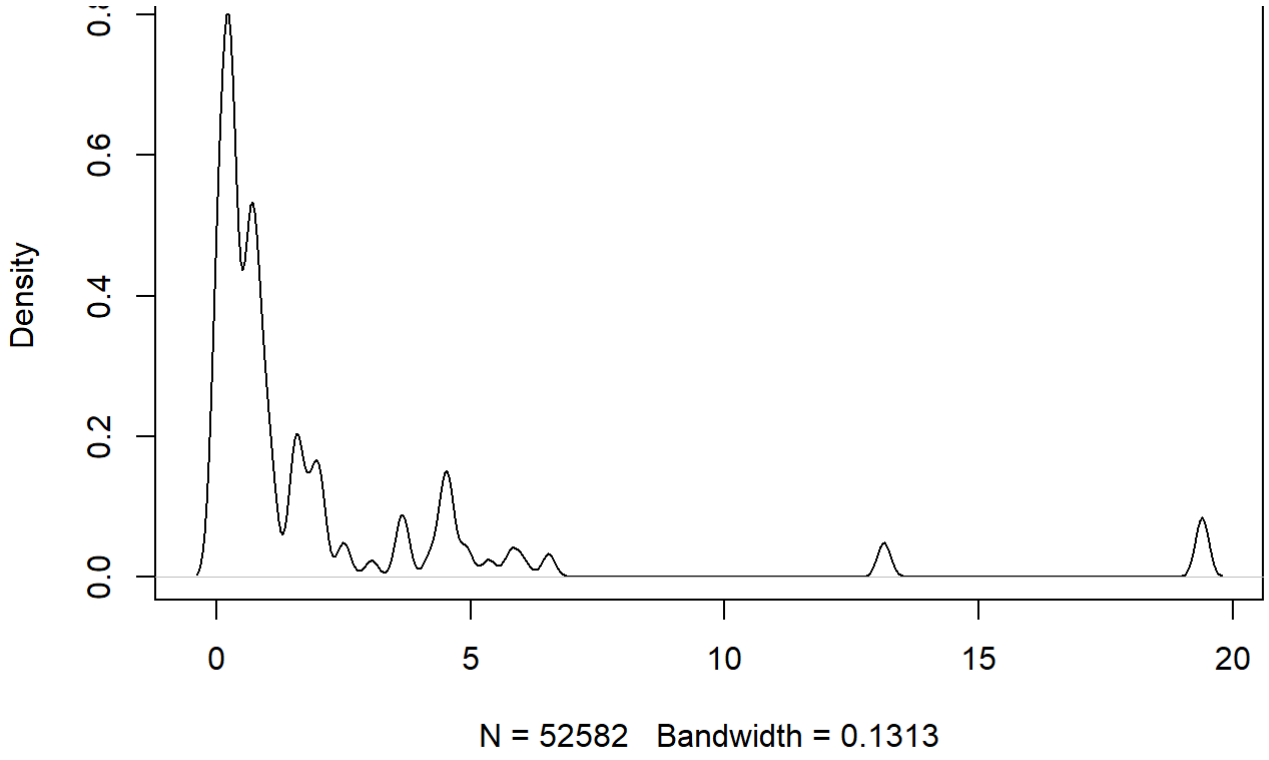
Show attribute distributions

```
for(name in names(df)){
  d <- density(df[, name])
  plot(d, main=name)
}
```

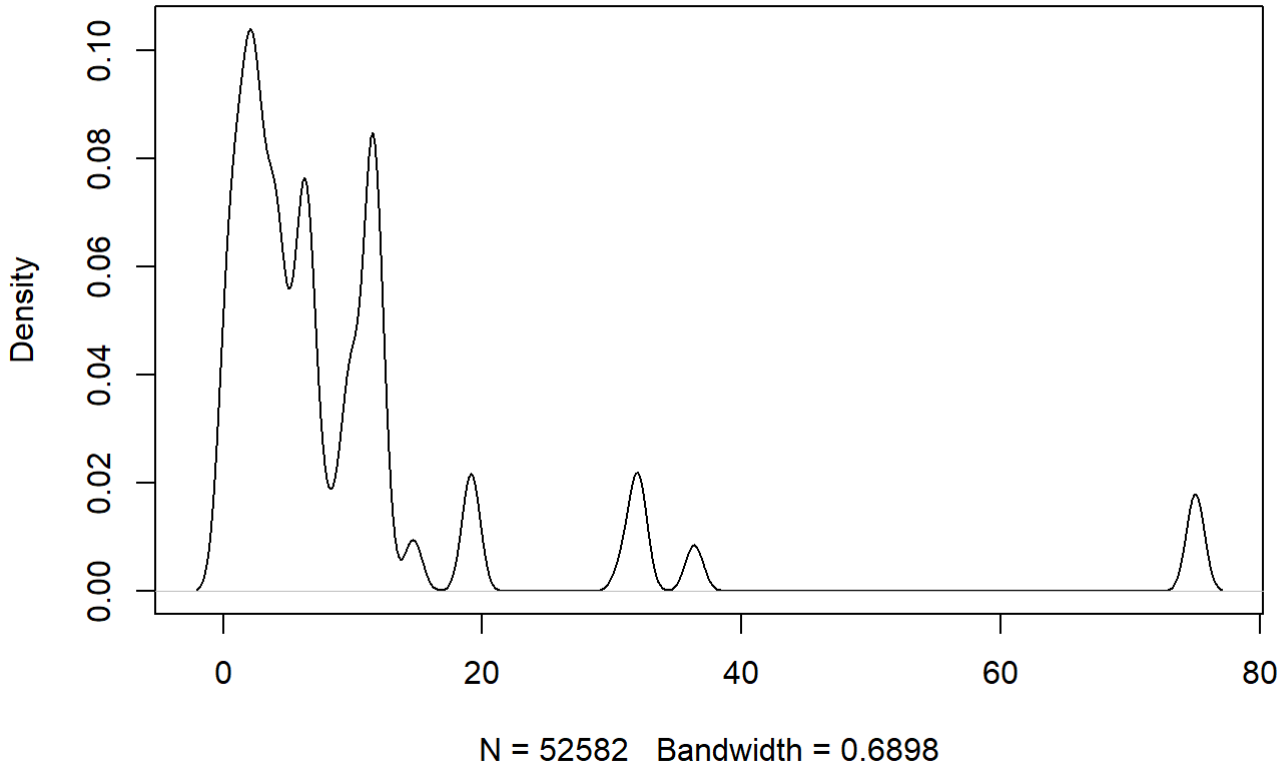

length
Density
0.00
0.05
0.10
0.15
0.20
0.25
20
25
30
N = 52582 Bandwidth = 0.1692

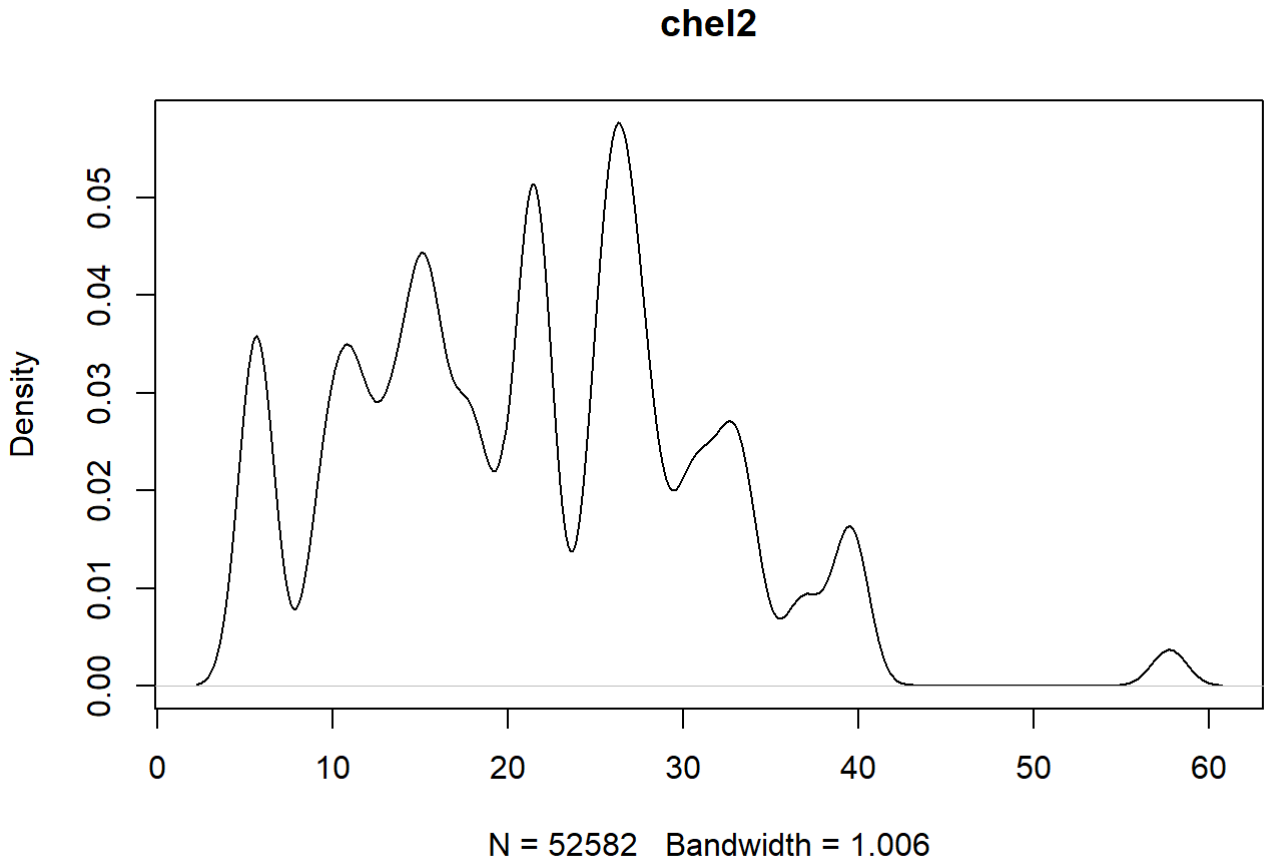

cfin1
Density
0.0
0.5
1.0
1.5
2.0
2.5
3.0
0
10
20
30
N = 52582 Bandwidth = 0.02752

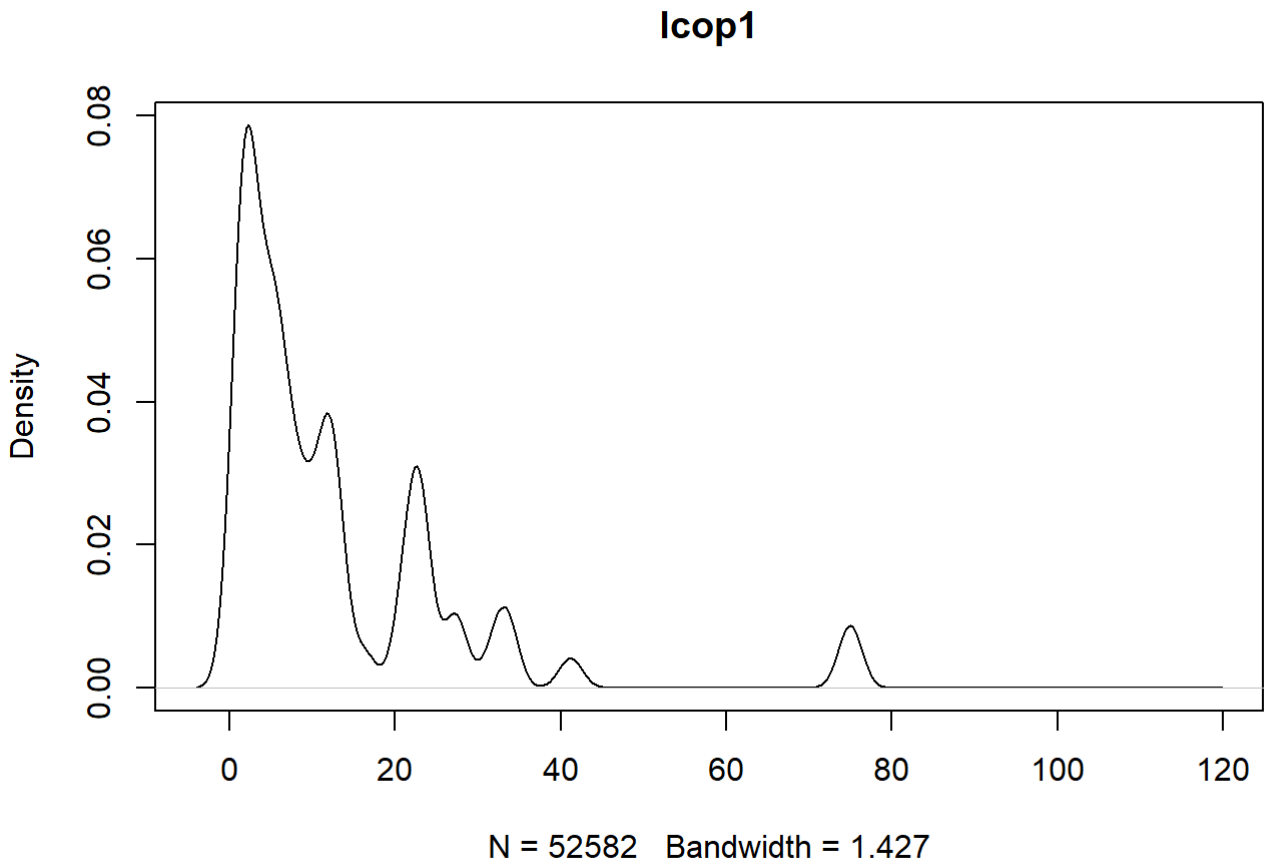

cfin2

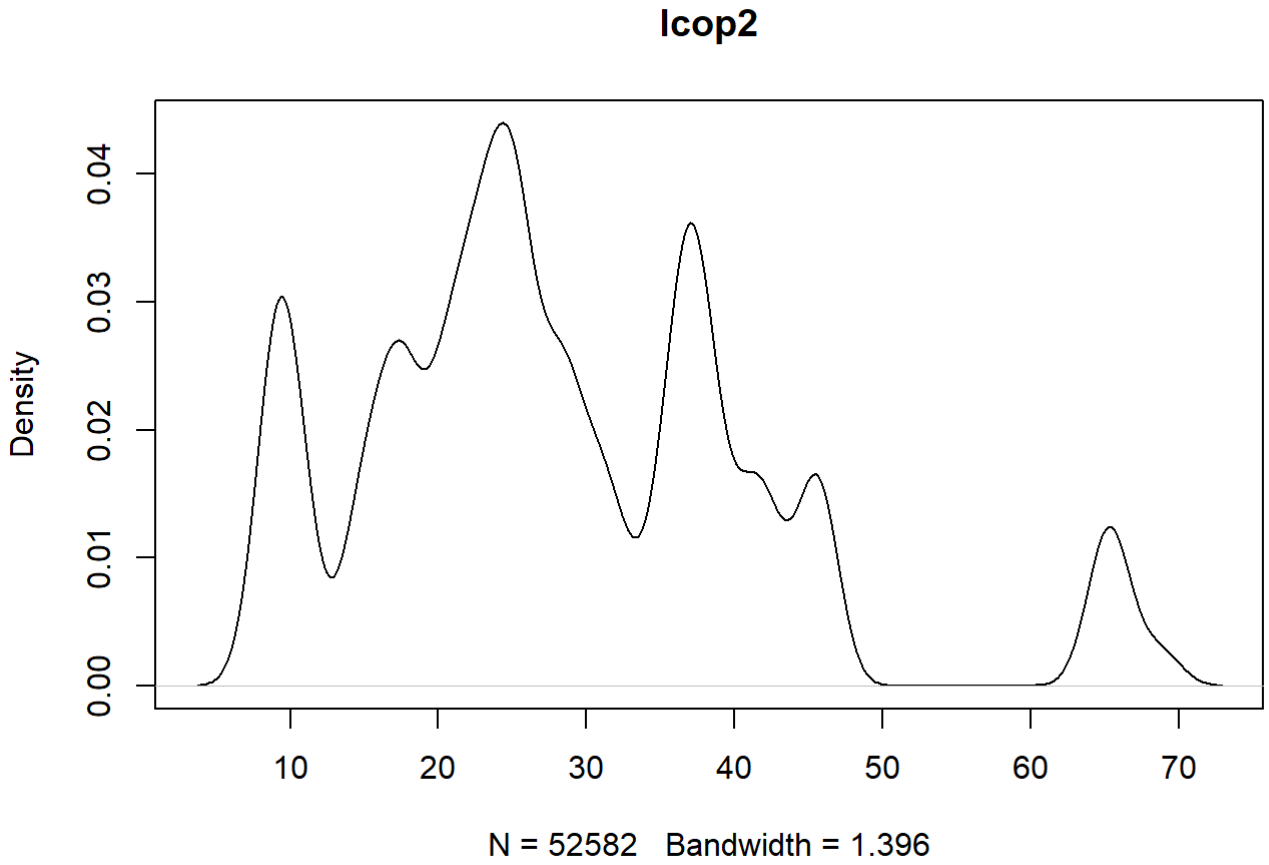

Density
0.0
0.2
0.4
0.6
0
5
10
15
20
N = 52582 Bandwidth = 0.1313

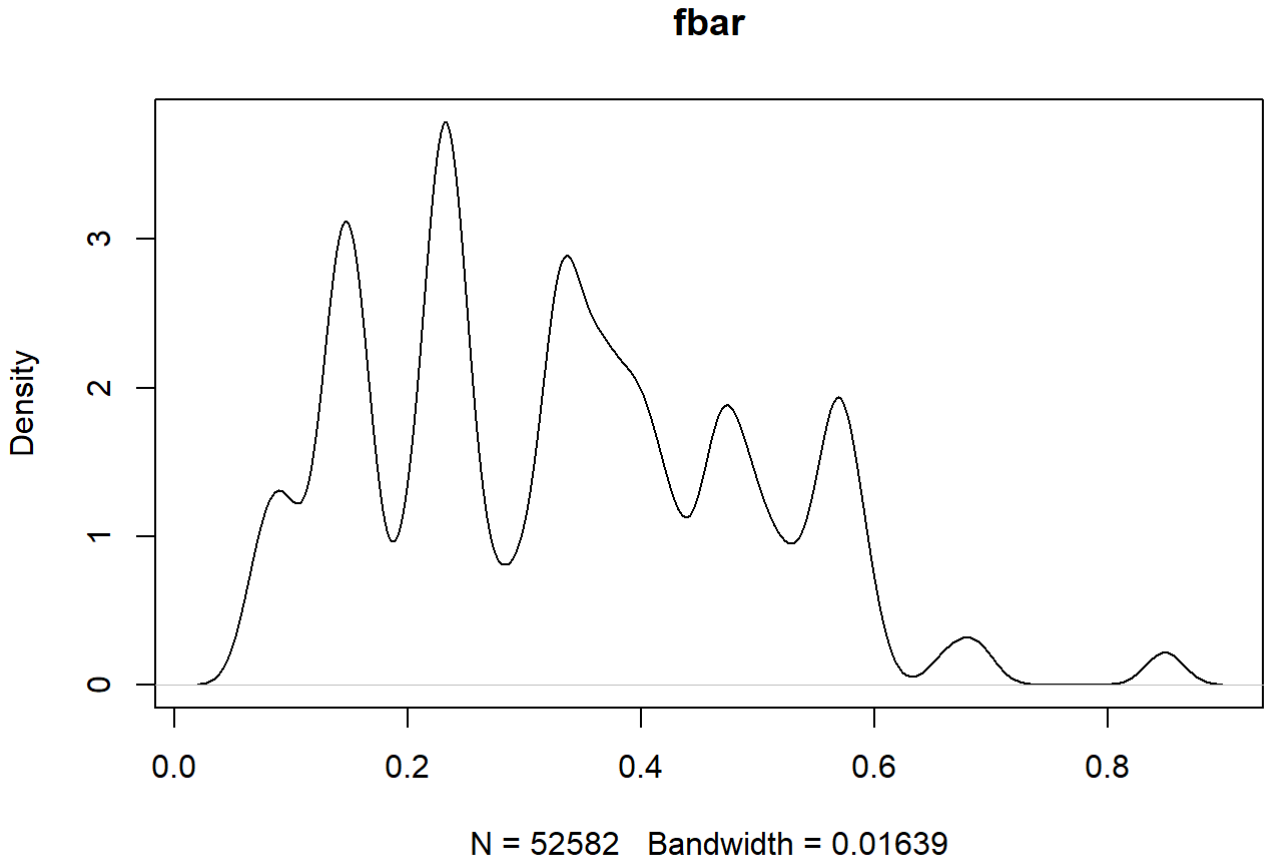
chel1

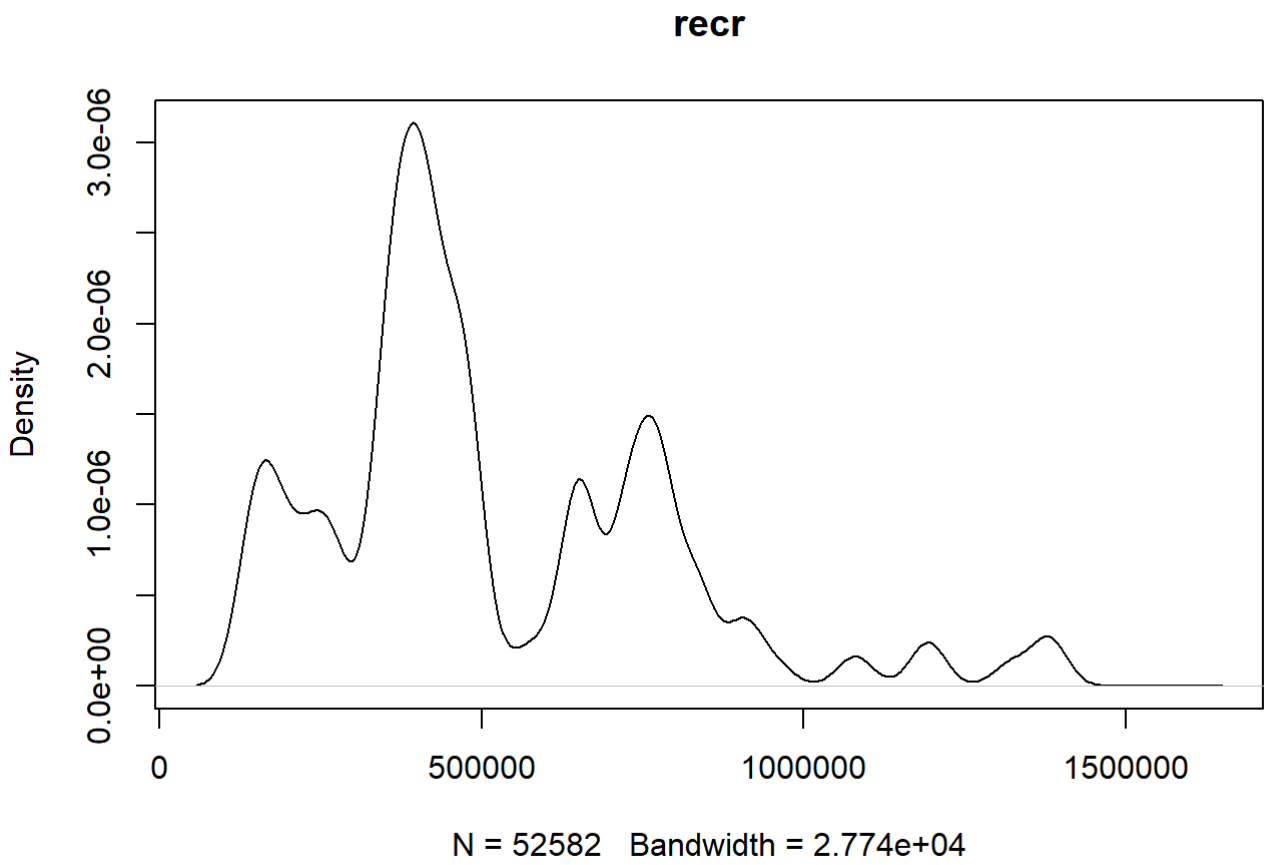

Density
0.00
0.02
0.04
0.06
0.08
0.10
0
20
40
60
80
N = 52582 Bandwidth = 0.6898

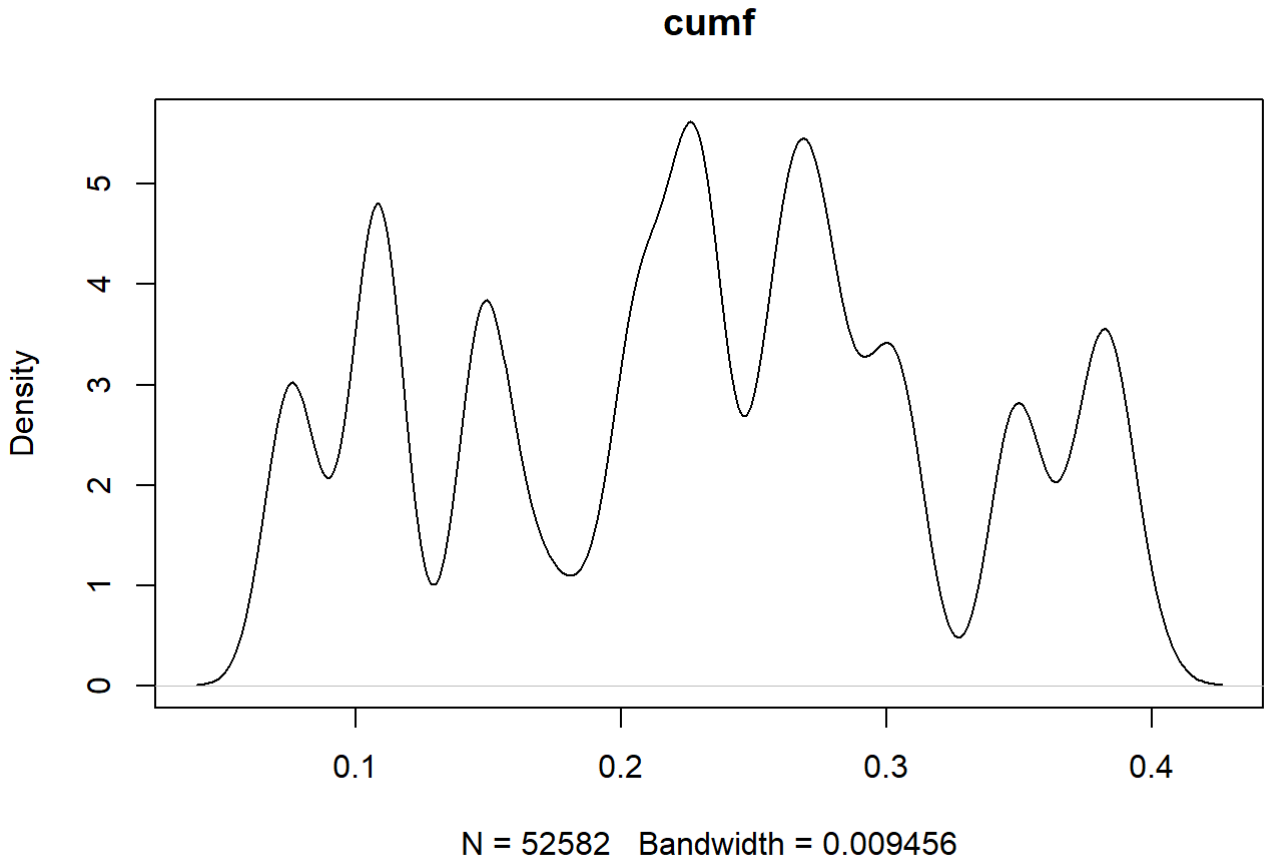

chel2
Density
0.00
0.01
0.02
0.03
0.04
0.05
0
10
20
30
40
50
60
N = 52582 Bandwidth = 1.006

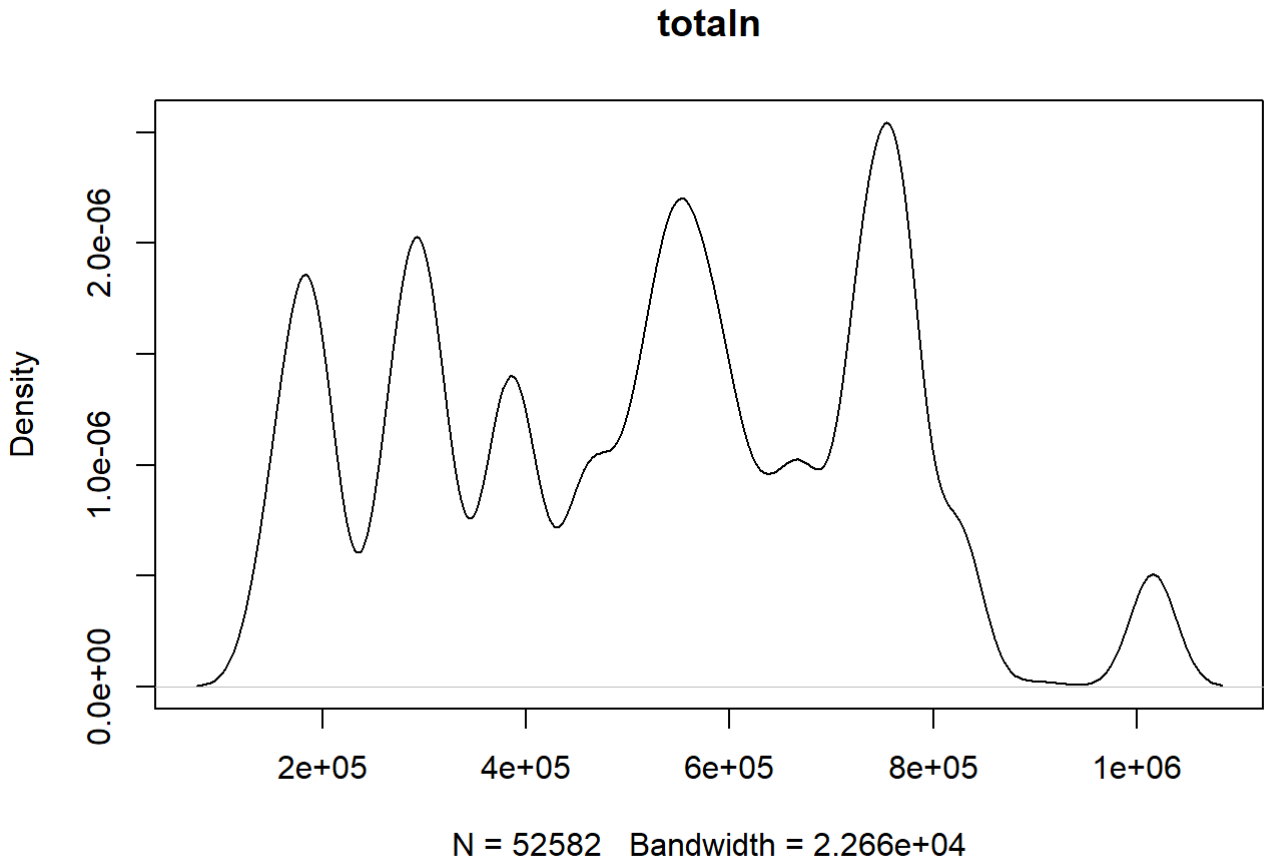

lcop1
Density
0.00
0.02
0.04
0.06
0.08
0
20
40
60
80
100
120
N = 52582 Bandwidth = 1.427

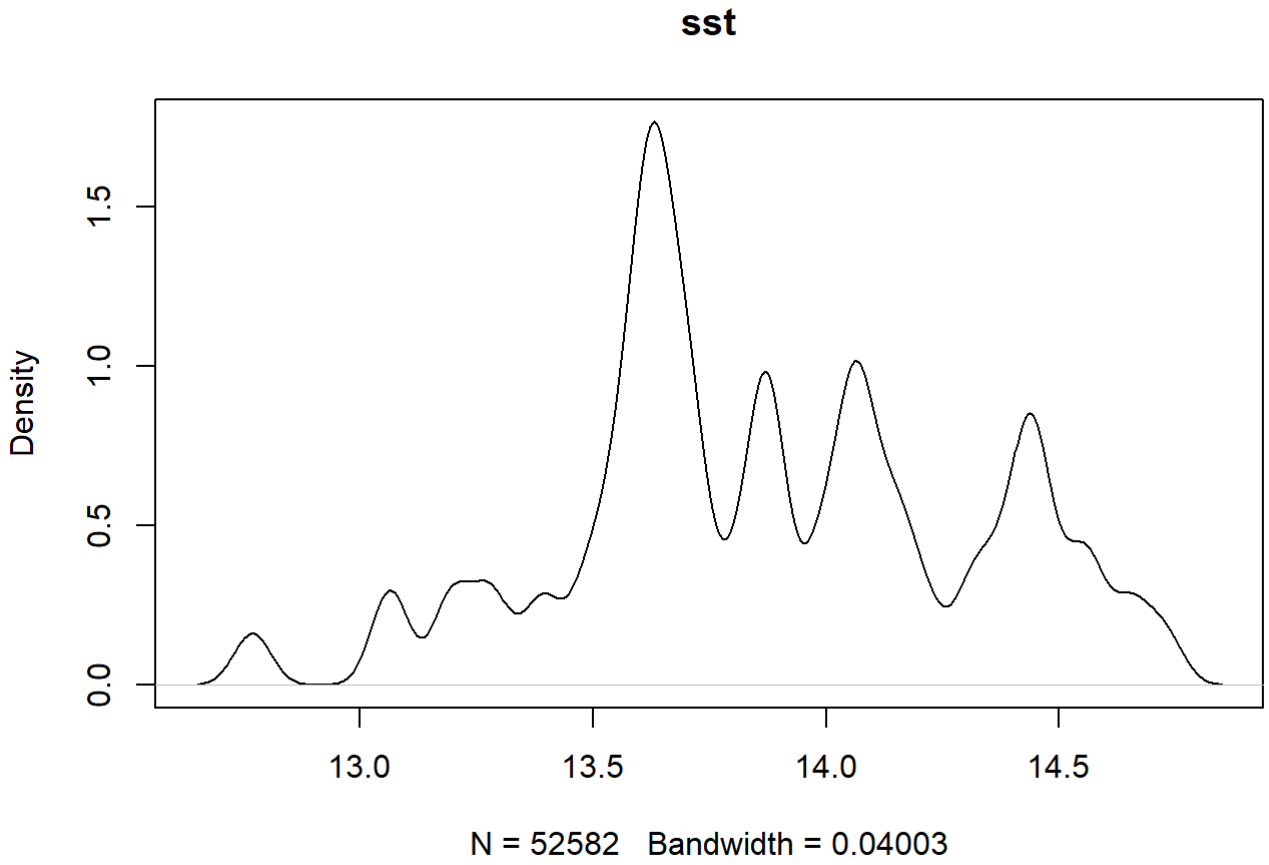

lcop2
Density
0.00
0.01
0.02
0.03
0.04
10
20
30
40
50
60
70
N = 52582 Bandwidth = 1.396

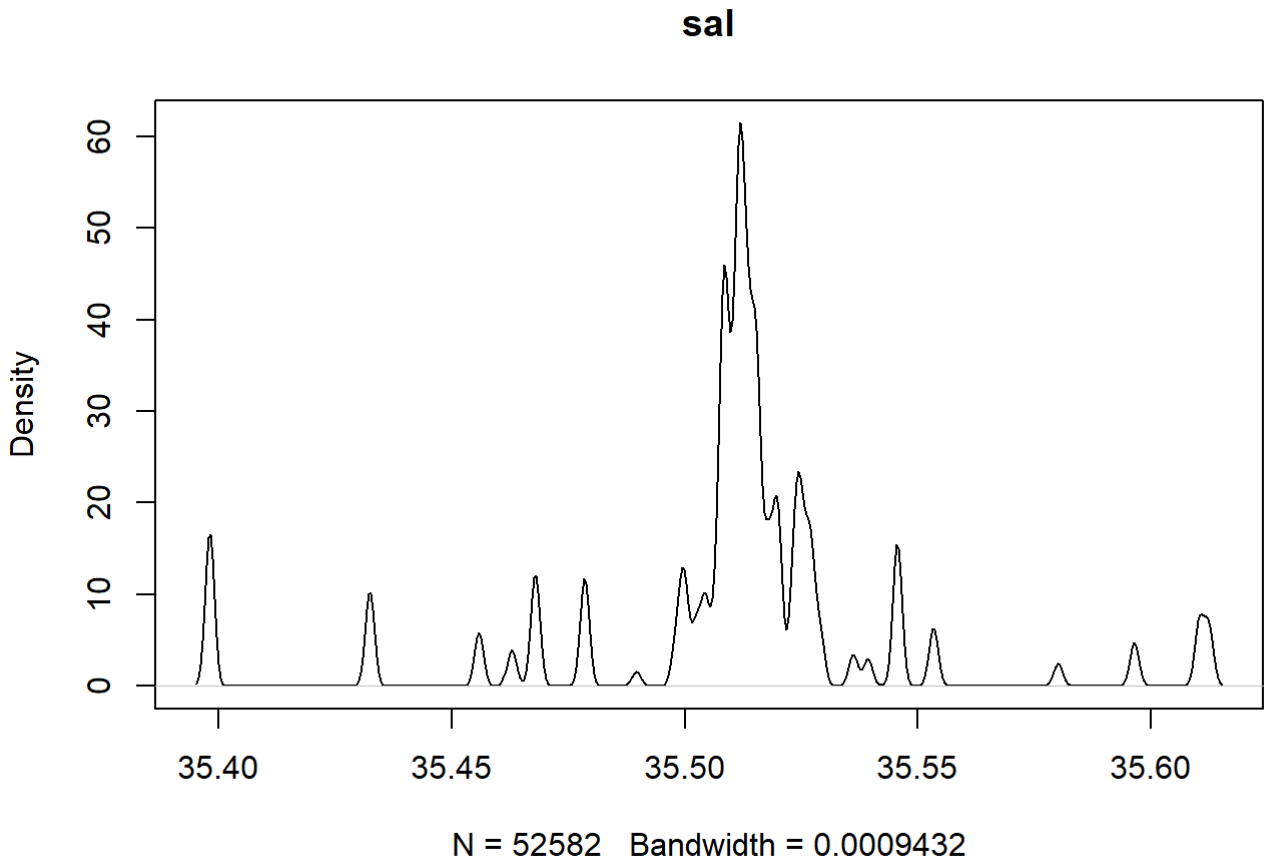

fbar
Density
0
1
2
3
0.0
0.2
0.4
0.6
0.8
N = 52582 Bandwidth = 0.01639

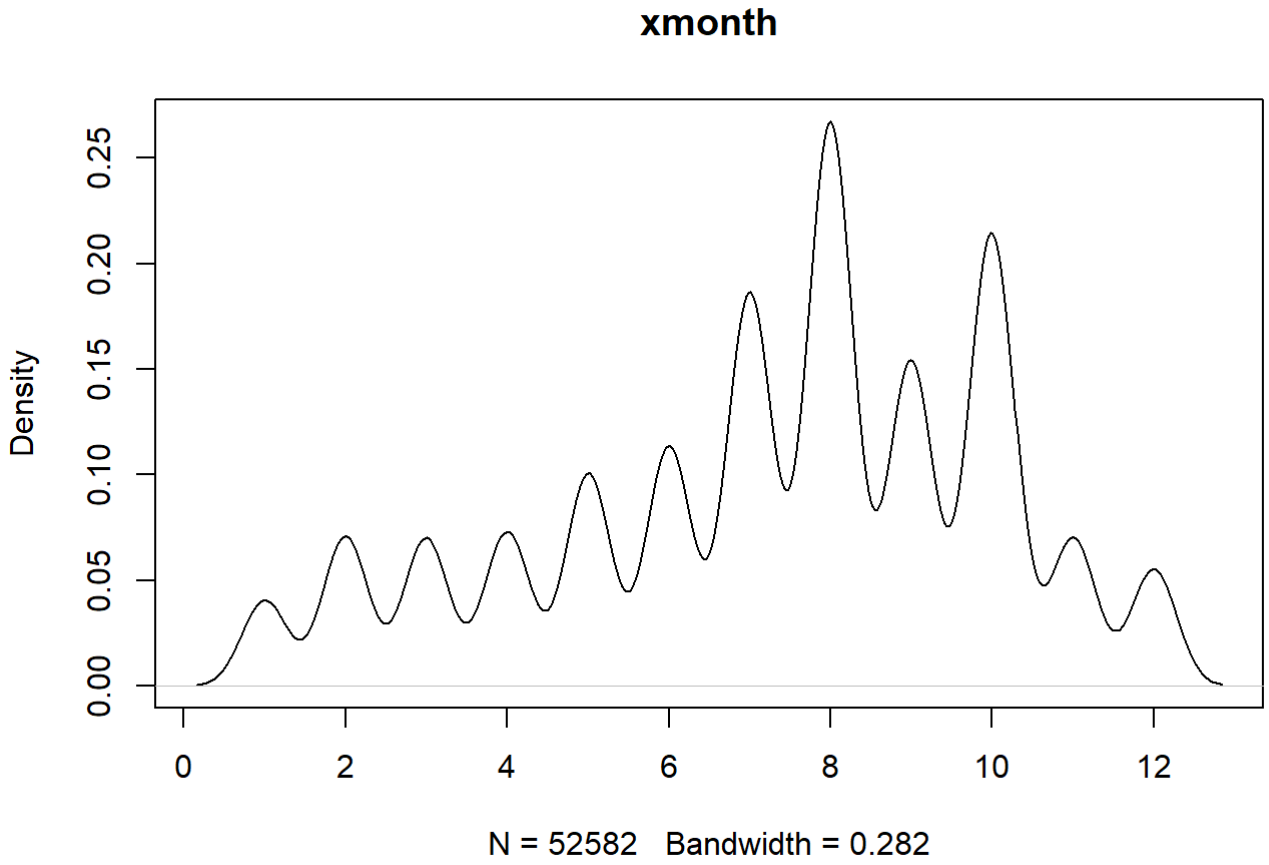

recr
Density
0.0e+00
1.0e-06
2.0e-06
3.0e-06
0
500000
1000000
1500000
N = 52582 Bandwidth = 2.774e+04

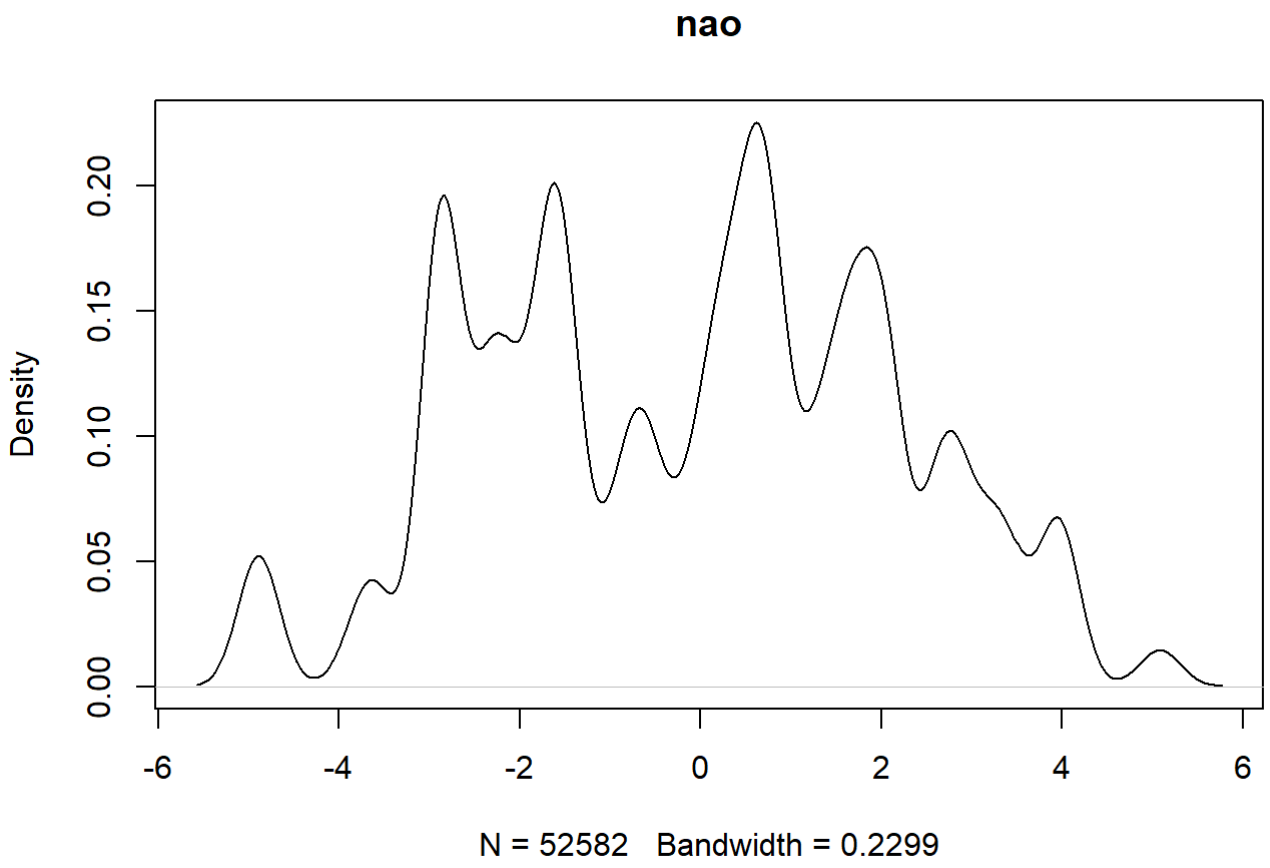

cumf
Density
0
1
2
3
4
5
0.1
0.2
0.3
0.4
N = 52582 Bandwidth = 0.009456


totaln
Density
0.0e+00
1.0e-06
2.0e-06
2e+05
4e+05
6e+05
8e+05
1e+06
N = 52582 Bandwidth = 2.266e+04


sst
Density
0.0
0.5
1.0
1.5
13.0
13.5
14.0
14.5
N = 52582 Bandwidth = 0.04003

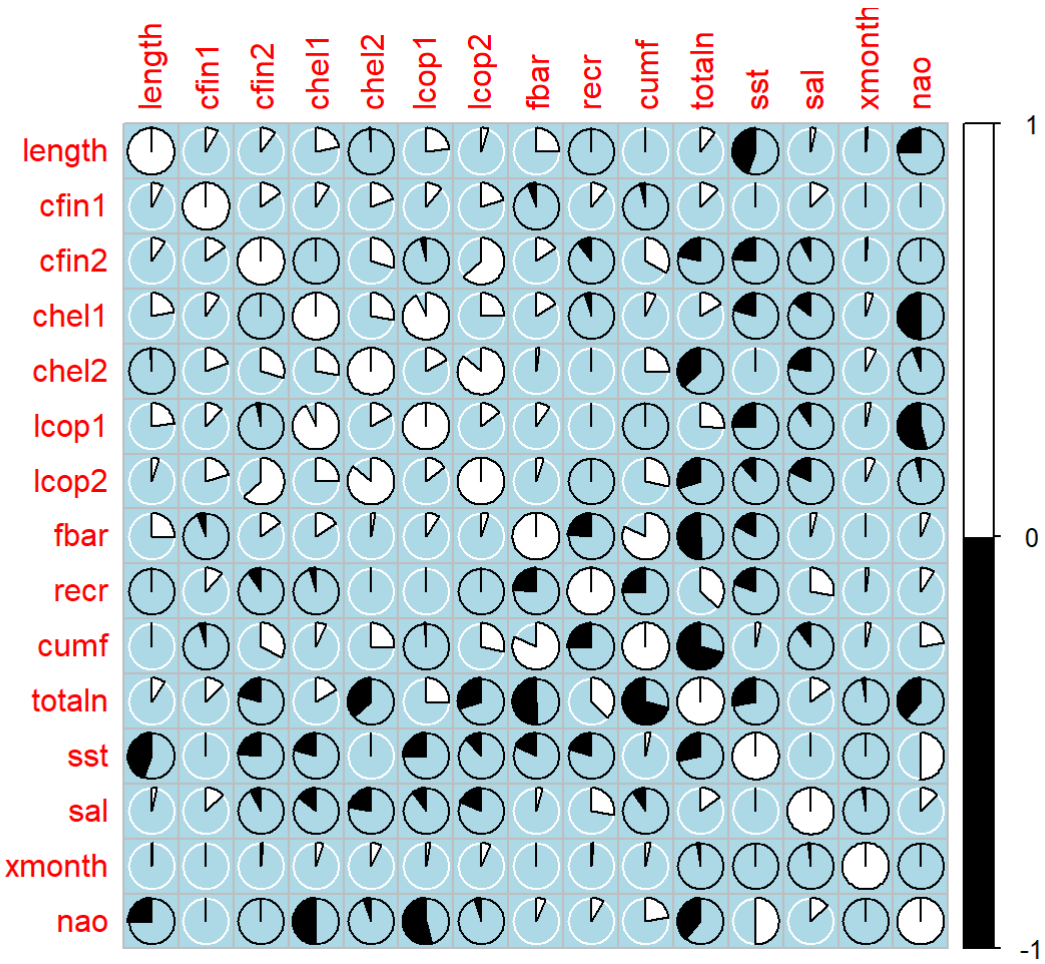

sal
Density
0
10
20
30
40
50
60
35.40
35.45
35.50
35.55
35.60
N = 52582 Bandwidth = 0.0009432


xmonth
Density
0.00
0.05
0.10
0.15
0.20
0.25
0
2
4
6
8
10
12
N = 52582 Bandwidth = 0.282



Show correlation matrix

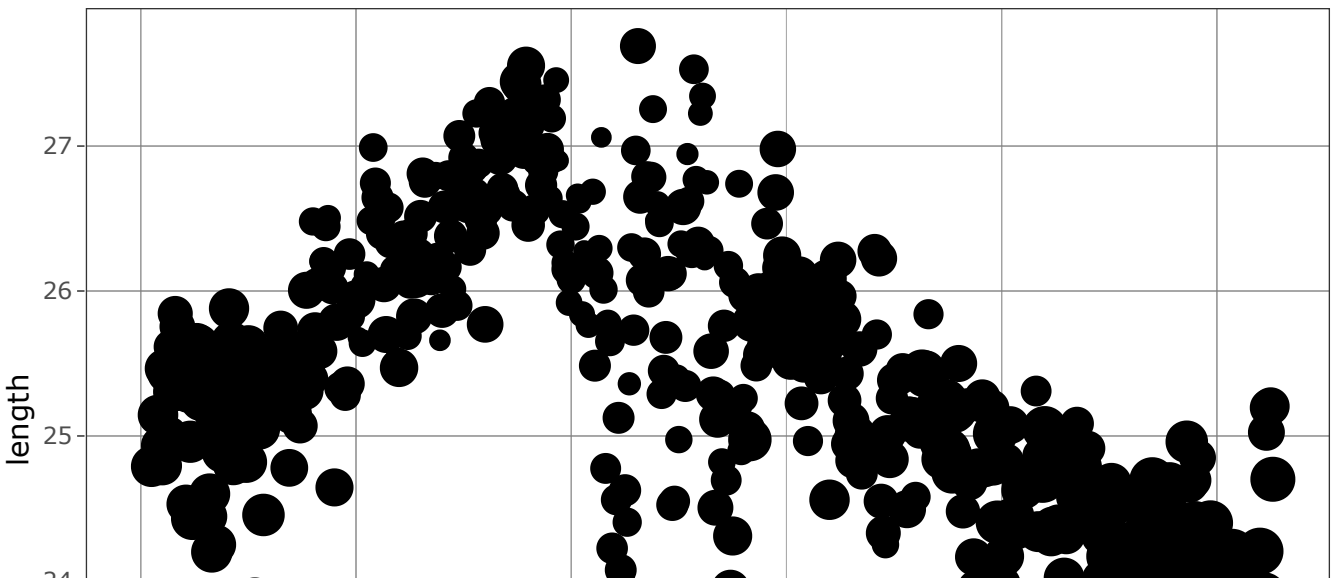
```
corr_matrix <- cor(df)
corrplot(corr_matrix, method="pie", col=c("black", "white"), bg="lightblue")
```

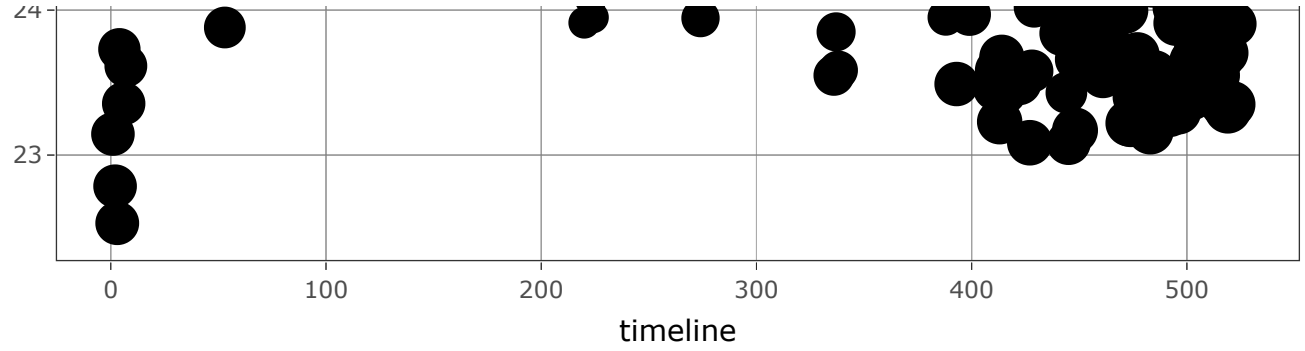


Herring length in time

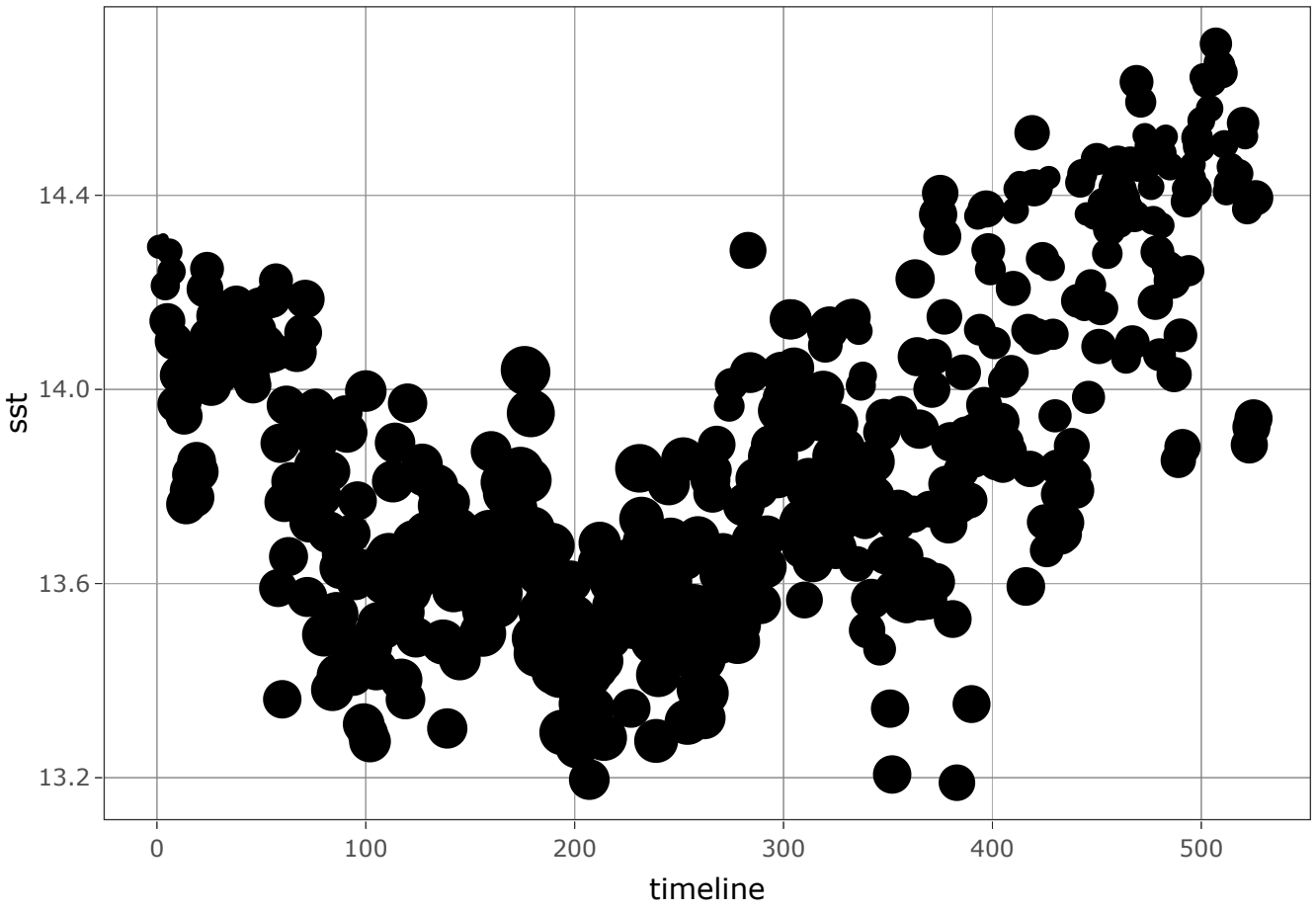
```
number_to_aggregate <- 100
aggregated <- aggregate(df
                        , list(rep(1:(nrow(df)%/%number_to_aggregate+1), each=number_to_aggre
gate, len=nrow(df)))
                        , mean)[-1]

aggregated$timeline <- as.numeric(row.names(aggregated))
p <- ggplot(aggregated, aes(timeline, length, size=sst)) + geom_point() + theme_bw()
plotly::ggplotly(p)
```






```
p_sst <- ggplot(aggregated, aes(timeline, sst, size=length)) + geom_point() + theme_bw()
plotly::ggplotly(p_sst)
```

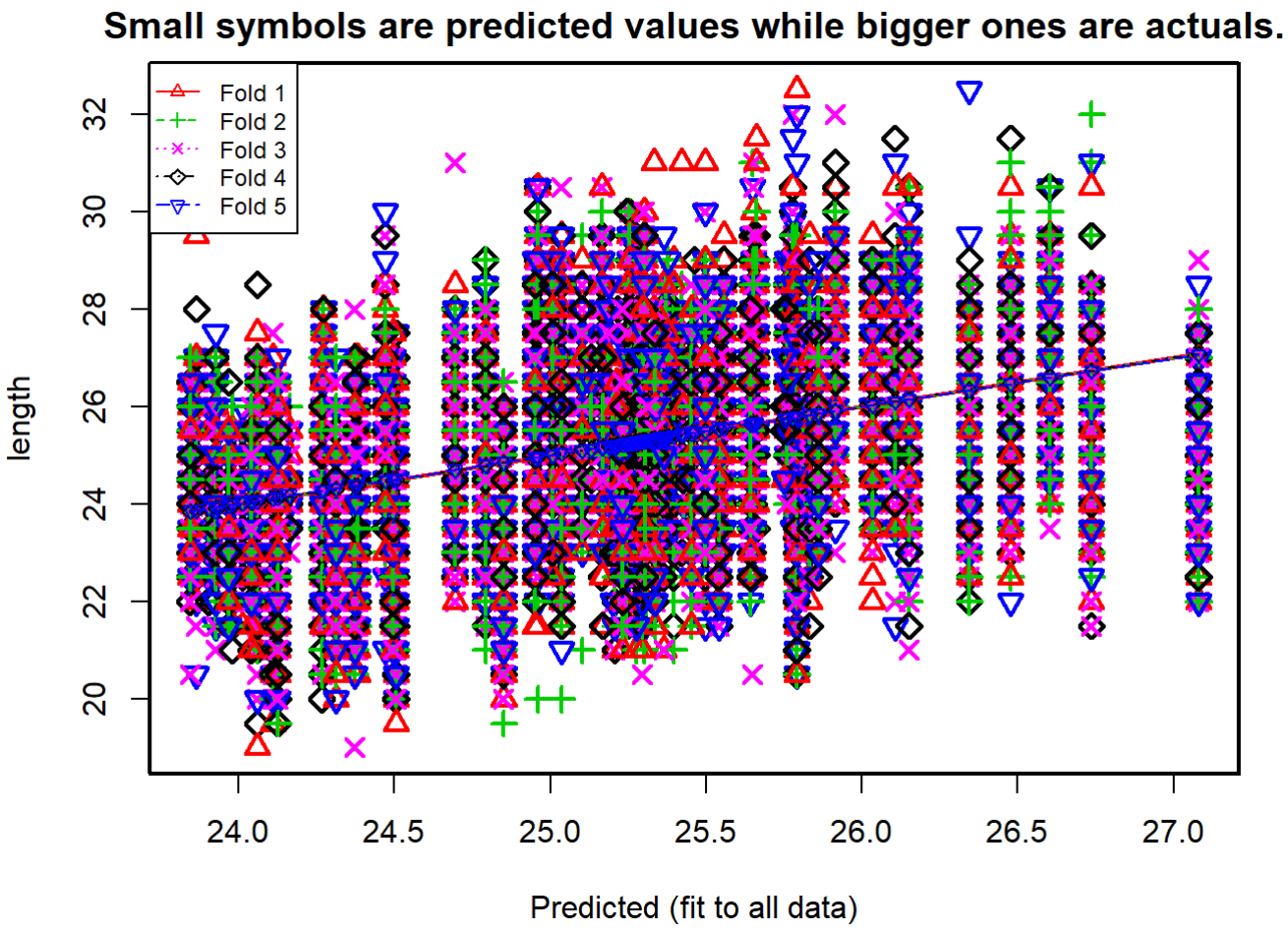


Regression model

```
# sst has the biggest correlation with length
# cfin2 has big correlation with length and relatively small with sst
# nao has correlation with length but does not have big correlation with sst or cfin2
# There is no need in adding more attributes to regression as the rest of attributes are very much correlated
formula <- length ~ sst + nao
linearMod <- lm(formula, data=df)  # build linear regression model on full data
summary(linearMod)
```

```
##
## Call:
## lm(formula = formula, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.3740 -0.9978  0.0235  1.0022  6.7087
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 47.830831   0.241370  198.16   <2e-16 ***
## sst         -1.623791   0.017379  -93.43   <2e-16 ***
## nao         -0.033318   0.003325  -10.02   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.479 on 52579 degrees of freedom
## Multiple R-squared:  0.1996, Adjusted R-squared:  0.1996
## F-statistic:  6557 on 2 and 52579 DF,  p-value: < 2.2e-16
```

```
cvResults <- suppressWarnings(CVlm(df, form.lm=formula, m=5, dots=FALSE, seed=29, legend.pos="topleft",  printit=FALSE, main="Small symbols are predicted values while bigger ones are actuals."));  # performs the CV
```




```
attr(cvResults, 'ms')
```

```
## [1] 2.186939
```

Regression model interpretation

According to the previously observed properties of the data set, the prepared linear regression model says that the sst attribute is the most important for the result of linear regression.The estimated regression coefficient value of this attribute is ~ -1.63, while the value of the next most-important attribute is ~ -0.33. Adding the remaining attributes did not improve the accuracy of the linear regression.