Thema09_Log

Marcel Setz

Contents

Introduction

A CpG island or CpG site is a part of the DNA where the GC content is greater than 50%. In this dataset methylation values of certain CpG sites are displayed with also the age, gender and smoking status for 671 people.

Research Question

Is it possible to identify a person's gender, age or status of smoking given their CpG values?

Data exploration

Codebook & Reading the data

```r
myData <- read.csv("data/Smoker_Epigenetic_df.csv")
myData <- myData %>% drop_na()

columns = colnames(myData[1:4])
columns <- append(columns, "Columns 5-24")
names <- c("Sample Accessions numbers", "Smoking status", "Gender", "Age", "CG Island")
type <- c("chr", "chr", "chr", "int", "num")
unit <- c(NA, "current/never", "f/m", NA, NA)
descriptions = c("GSM identifier testsubject", "Wether the person is smoking or not", "Gender", "Age",
codebook <- data.frame(columns, names, type, descriptions)
write.csv(codebook, "Codebook.csv", row.names = FALSE)
knitr::kable(codebook)
```

columns	names	type	descriptions
GSM	Sample Accessions numbers	chr	GSM identifier testsubject
Smoking.Status	Smoking status	chr	Wether the person is smoking or not
Gender	Gender	chr	Gender
Age	Age	int	Age
Columns 5-24	CG Island	num	Methylation Rate of CG Island

Summary

```
subsetdata <- head(myData)
dataset <- subsetdata[1:7]
knitr::kable(dataset) %>%
  row_spec(0, bold = TRUE) %>%
  column_spec(1, bold = TRUE)
```

GSM	Smoking.Status	Gender	Age	cg00050873	cg00212031	cg00213748
GSM1051525	current	f	67	0.6075634	0.4228427	0.3724549
GSM1051526	current	f	49	0.3450542	0.5686615	0.5005995
GSM1051527	current	f	53	0.3213497	0.3609091	0.3527315
GSM1051528	current	f	62	0.2772675	0.3044371	0.4752352
GSM1051529	never	f	33	0.4135991	0.1312511	0.3675446
GSM1051530	current	f	59	0.6228599	0.5016849	0.2632270

```
dataset <- subsetdata[c(1, 8:13)]
knitr::kable(dataset) %>%
  row_spec(0, bold = TRUE) %>%
  column_spec(1, bold = TRUE)
```

GSM	cg00214611	cg00455876	cg01707559	cg02004872	cg02011394	cg02050847
GSM1051525	0.6215619	0.2907773	0.2671431	0.1791439	0.4802517	0.3276078
GSM1051526	0.4986067	0.3745909	0.1902743	0.1559775	0.4180809	0.3464627
GSM1051527	0.3738240	0.2306740	0.3147052	0.1057448	0.6151030	0.2375392
GSM1051528	0.4862581	0.2951815	0.2957931	0.1112862	0.3010196	0.3045353
GSM1051529	0.7611667	0.2357703	0.2505265	0.1691084	0.3929746	0.3062257
GSM1051530	0.4157459	0.4751891	0.2539041	0.2607587	0.5097921	0.4052457

```
dataset <- subsetdata[c(1, 14:19)]
knitr::kable(dataset) %>%
  row_spec(0, bold = TRUE) %>%
  column_spec(1, bold = TRUE)
```

GSM	cg02233190	cg02494853	cg02839557	cg02842889	cg03052502	cg03155755
GSM1051525	0.2411204	0.0670696	0.2469934	0.4692396	0.4002466	0.4150313
GSM1051526	0.1754907	0.0469389	0.2367423	0.3074666	0.3770313	0.3973715
GSM1051527	0.2464092	0.0382371	0.2446117	0.3577526	0.3050442	0.5212775
GSM1051528	0.1770279	0.0267163	0.0016414	0.4457390	0.2714746	0.4344920
GSM1051529	0.3017014	0.0370164	0.3343197	0.3950396	0.3265530	0.4300966
GSM1051530	0.3852716	0.0258346	0.3092102	0.3218573	0.5333670	0.5715522

```
dataset <- subsetdata[c(1, 20:24)]
knitr::kable(dataset) %>%
  row_spec(0, bold = TRUE) %>%
  column_spec(1, bold = TRUE)
```

GSM	cg03244189	cg03443143	cg03683899	cg03695421	cg03706273
GSM1051525	0.2214331	0.4758258	0.2077242	0.2091974	0.1299826
GSM1051526	0.2171221	0.5444690	0.1844462	0.1937732	0.0985327
GSM1051527	0.1850495	0.5370600	0.3931231	0.2680030	0.0402481
GSM1051528	0.1654187	0.5079167	0.2812089	0.2178572	0.1015163
GSM1051529	0.1811352	0.4054791	0.3107944	0.2800708	0.0778571
GSM1051530	0.2109749	0.3778239	0.4693609	0.3433317	0.0457791

```r
data_sum <- summary(myData)
data_sum
```

```
##      GSM            Smoking.Status        Gender               Age       
##  Length:621         Length:621         Length:621         Min.   :18.00  
##  Class :character   Class :character   Class :character   1st Qu.:46.00  
##  Mode  :character   Mode  :character   Mode  :character   Median :54.00  
##                                                           Mean   :52.59  
##                                                           3rd Qu.:61.00  
##                                                           Max.   :70.00  
##    cg00050873       cg00212031         cg00213748       cg00214611     
##  Min.   :0.1186   Min.   :0.006949   Min.   :0.0000   Min.   :0.01247  
##  1st Qu.:0.4131   1st Qu.:0.063172   1st Qu.:0.3635   1st Qu.:0.06946  
##  Median :0.5052   Median :0.365545   Median :0.4713   Median :0.41575  
##  Mean   :0.5600   Mean   :0.309601   Mean   :0.5191   Mean   :0.34106  
##  3rd Qu.:0.8144   3rd Qu.:0.459813   3rd Qu.:0.7278   3rd Qu.:0.49745  
##  Max.   :0.8989   Max.   :0.709992   Max.   :0.9236   Max.   :0.80606  
##    cg00455876        cg01707559        cg02004872         cg02011394    
##  Min.   :0.05917   Min.   :0.04333   Min.   :0.002616   Min.   :0.0000  
##  1st Qu.:0.29300   1st Qu.:0.11080   1st Qu.:0.042835   1st Qu.:0.4261  
##  Median :0.37968   Median :0.23873   Median :0.149332   Median :0.5157  
##  Mean   :0.44718   Mean   :0.21435   Mean   :0.155417   Mean   :0.6058  
##  3rd Qu.:0.66283   3rd Qu.:0.28061   3rd Qu.:0.242627   3rd Qu.:0.9412  
##  Max.   :0.85443   Max.   :0.46999   Max.   :0.473844   Max.   :0.9792  
##    cg02050847        cg02233190         cg02494853        cg02839557     
##  Min.   :0.05234   Min.   :0.008632   Min.   :0.01162   Min.   :0.00000  
##  1st Qu.:0.33963   1st Qu.:0.088375   1st Qu.:0.02865   1st Qu.:0.06384  
##  Median :0.42754   Median :0.259817   Median :0.03695   Median :0.35042  
##  Mean   :0.54369   Mean   :0.232498   Mean   :0.04077   Mean   :0.30088  
##  3rd Qu.:0.95558   3rd Qu.:0.337023   3rd Qu.:0.04677   3rd Qu.:0.45786  
##  Max.   :0.98320   Max.   :0.511730   Max.   :0.28947   Max.   :0.82739  
##    cg02842889        cg03052502       cg03155755       cg03244189     
##  Min.   :0.01346   Min.   :0.0000   Min.   :0.2020   Min.   :0.02972  
##  1st Qu.:0.05483   1st Qu.:0.4025   1st Qu.:0.4245   1st Qu.:0.11976  
##  Median :0.39757   Median :0.4940   Median :0.4962   Median :0.20397  
##  Mean   :0.32362   Mean   :0.5907   Mean   :0.5895   Mean   :0.19552  
##  3rd Qu.:0.47385   3rd Qu.:0.9631   3rd Qu.:0.8988   3rd Qu.:0.24921  
##  Max.   :0.85625   Max.   :0.9902   Max.   :0.9696   Max.   :0.54074  
##    cg03443143        cg03683899        cg03695421       cg03706273     
##  Min.   :0.06496   Min.   :0.00788   Min.   :0.0949   Min.   :0.01120  
##  1st Qu.:0.40963   1st Qu.:0.06159   1st Qu.:0.2566   1st Qu.:0.03413  
##  Median :0.48314   Median :0.34422   Median :0.3208   Median :0.04961  
##  Mean   :0.56841   Mean   :0.28442   Mean   :0.3978   Mean   :0.05769  
##  3rd Qu.:0.85436   3rd Qu.:0.41866   3rd Qu.:0.5965   3rd Qu.:0.06916  
##  Max.   :0.93589   Max.   :0.65925   Max.   :0.8433   Max.   :0.34380  
```

Visualization

Distribution plots

Below there are some histograms which visualizes the distribution of smoking status, age and gender.

```
ggplot(myData, aes(x=Smoking.Status)) +
  geom_histogram(stat="count", fill = c("pink", "lightblue"), col = "black") +
  ylim(0, 600) +
  labs(caption = "Figure 1: Number of people who are smoking") +
  theme(plot.caption = element_text(size=16)) +
  theme(plot.caption = element_text(size=16, face="italic")) +
  theme(axis.text = element_text(size = 20)) +
  theme(axis.title = element_text(size = 20, face="bold"))
```

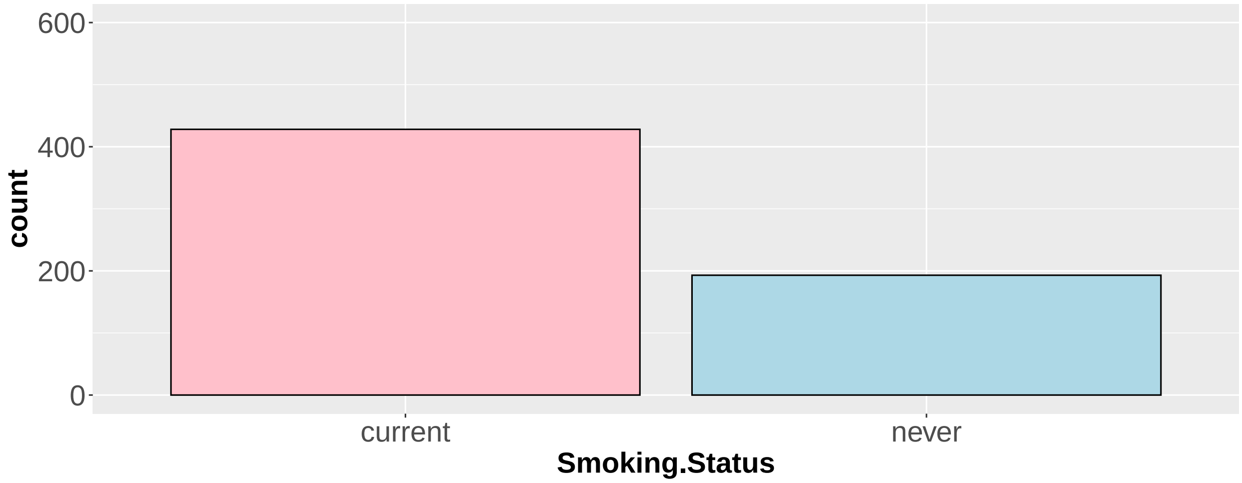


Figure 1: Number of people who are smoking

```
ggplot(myData, aes(x=Gender)) +
  geom_histogram(stat="count", fill = c("pink", "lightblue"), col = "black") +
  ylim(0, 600) +
  labs(caption = "Figure 2: Gender distribution") +
  theme(plot.caption = element_text(size=16)) +
  theme(plot.caption = element_text(size=16, face="italic")) +
  theme(axis.text = element_text(size = 20)) +
  theme(axis.title = element_text(size = 20, face="bold"))
```

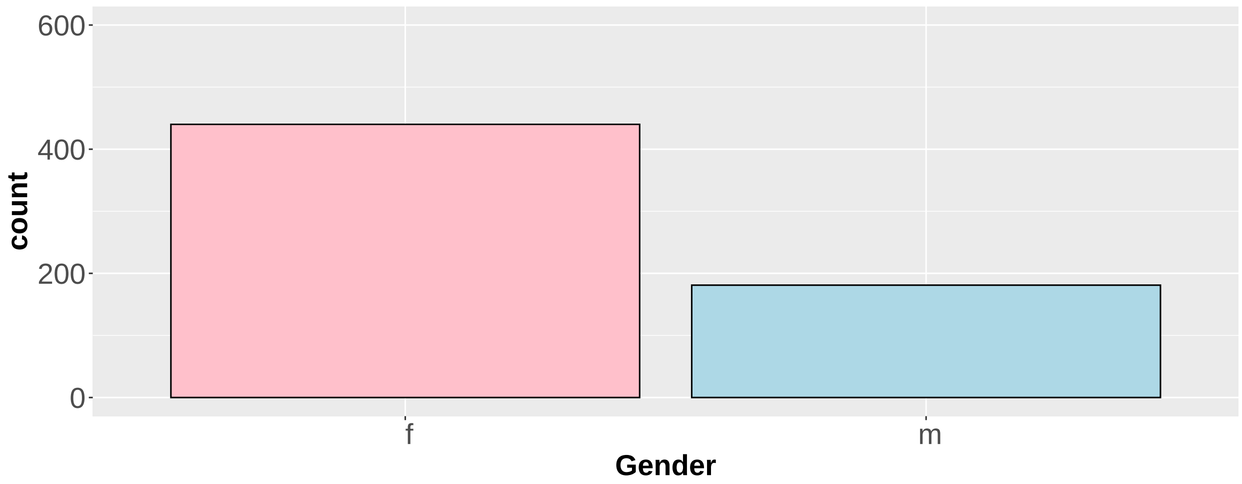


Figure 2: Gender distribution

```
ggplot(myData, aes(x=Age)) +
  geom_histogram(fill = "lightgrey", col = "black") +
  labs(caption = "Figure 3: Age distribution") +
  theme(plot.caption = element_text(size=16, face="italic")) +
  theme(axis.text = element_text(size = 20)) +
  theme(axis.title = element_text(size = 20, face="bold"))
```

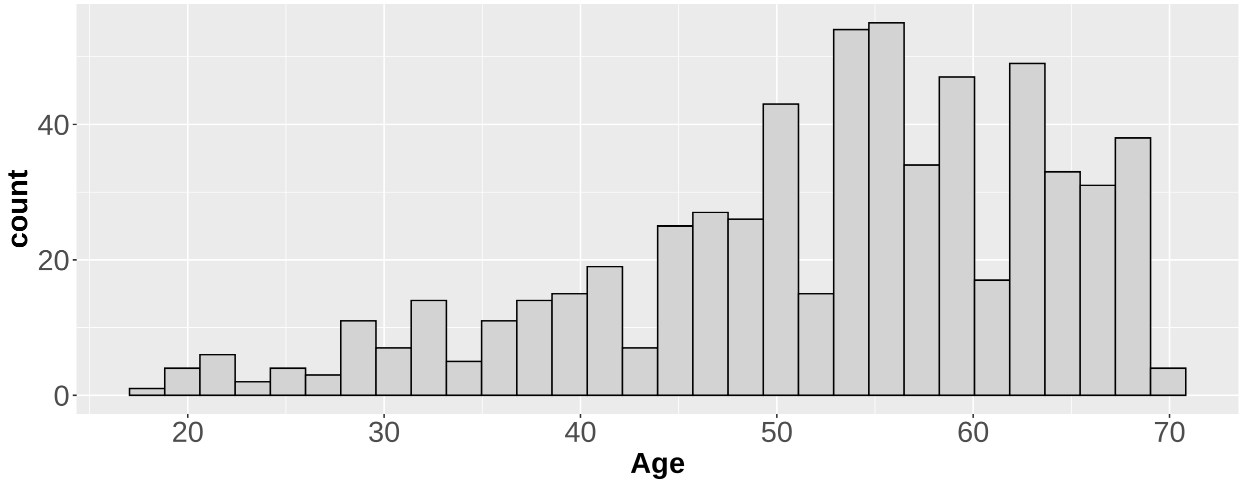


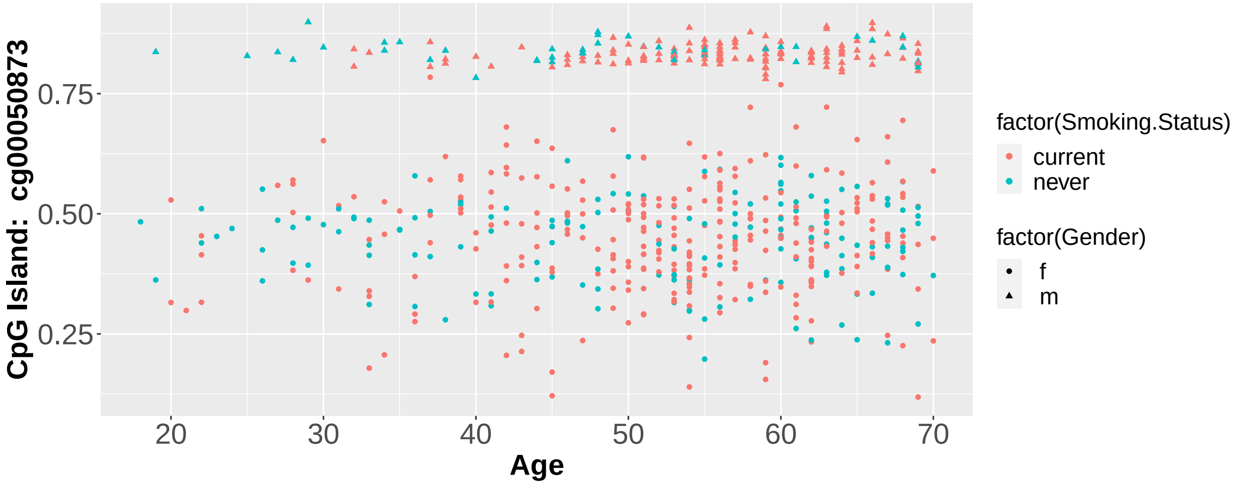
Figure 3: Age distribution

Plotting the data

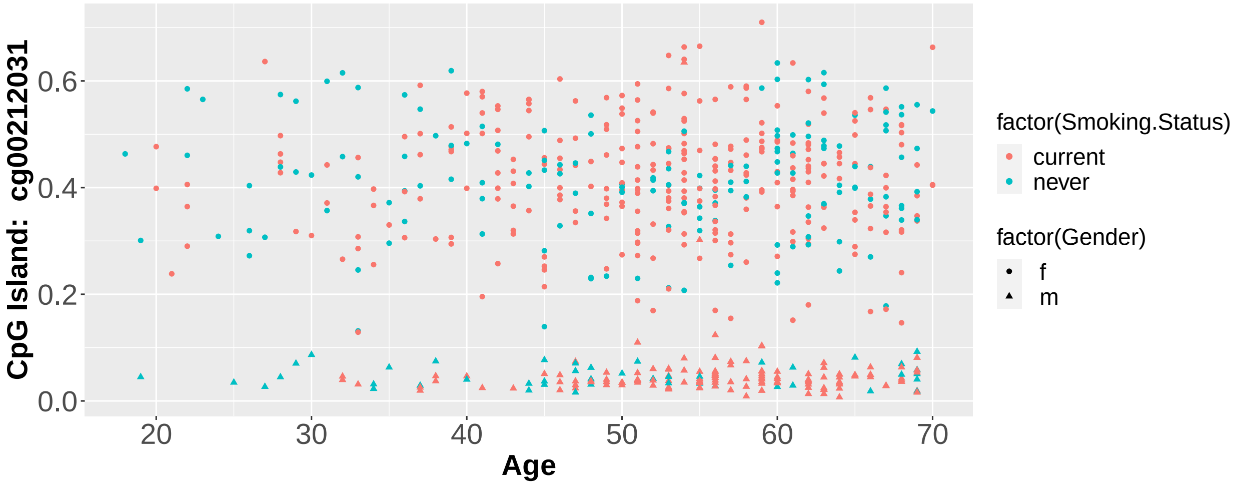
Scatterplots

These are all the CpG sites plotted against age, with smokin status as color groups and gender as shape groups. A few of the CpG islands have been removed, becaused they looked very similar. What stands out here is that you see two groups in almost every graph, one of men and one of women, so apparently the cg methylation rate is different between men and women.

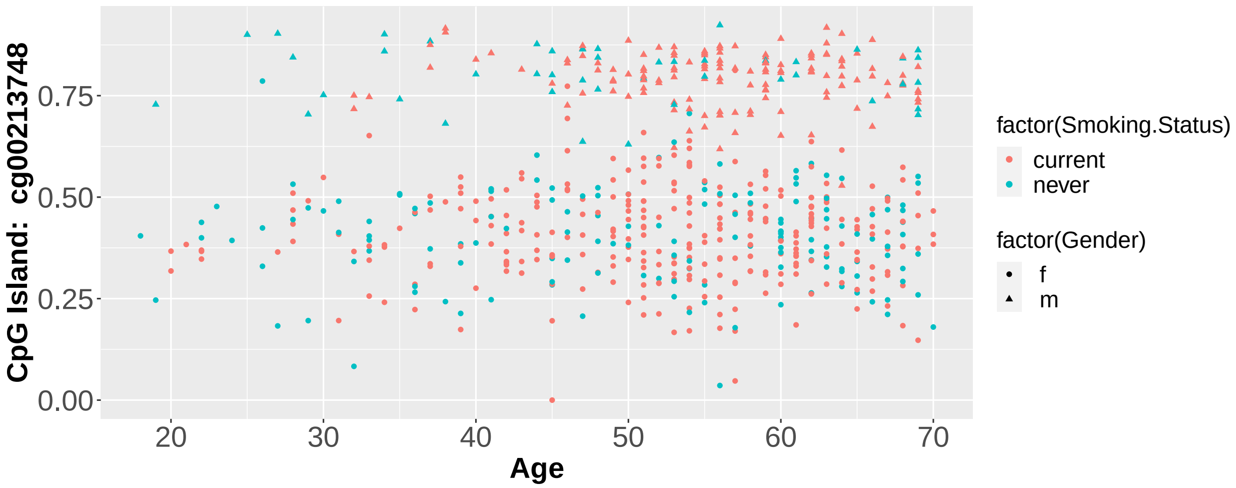
```r
plotfunction <- function(cg){
  ggplot(data = myData, mapping = aes_string(x = "Age", y = cg)) +
    geom_point(aes(shape = factor(Gender), color = factor(Smoking.Status))) +
    labs(caption = paste(cg, ": Scatterplot visualizing the methylation rate on this CpG island of diff
    theme(plot.caption = element_text(size=14, face="italic")) +
    theme(axis.text = element_text(size = 20)) +
    theme(axis.title = element_text(size = 20, face="bold")) +
    theme(legend.text = element_text(size = 16)) +
    theme(legend.title = element_text(size=16)) +
    ylab(paste("CpG Island: ", cg))
}
lapply(names(myData[c(5, 6, 7, 9, 10, 15, 20)]), plotfunction)
```



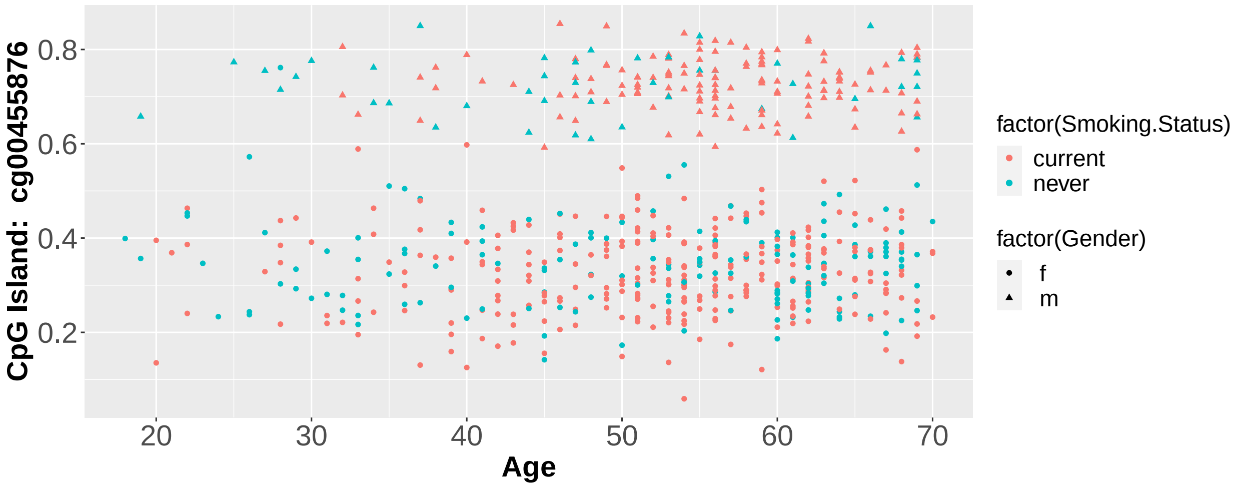
cg00050873 : Scatterplot visualizing the methylation rate on this CpG island of different ages and genders.



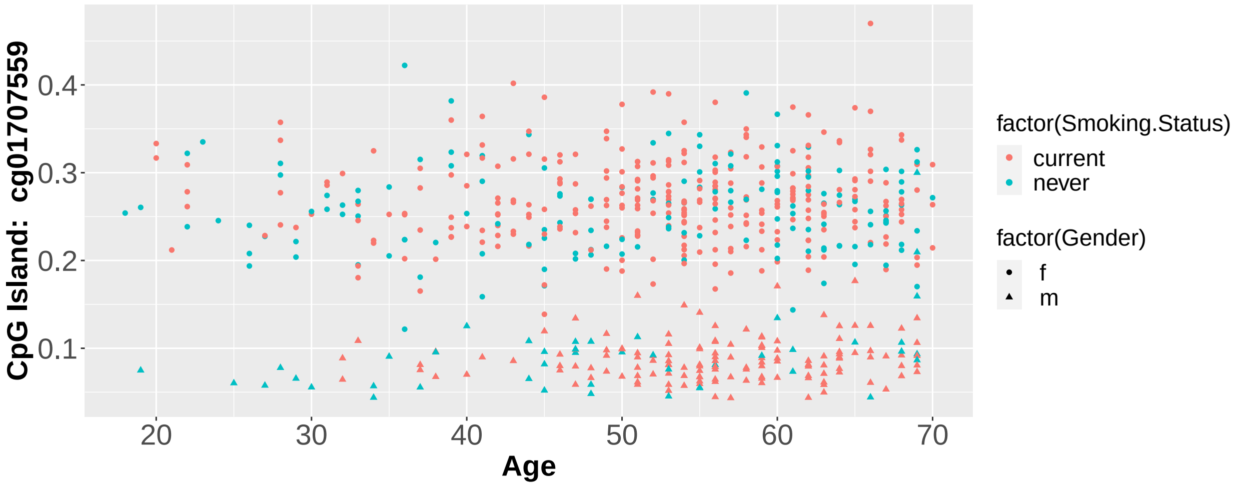
cg00212031 : Scatterplot visualizing the methylation rate on this CpG island of different ages and genders.



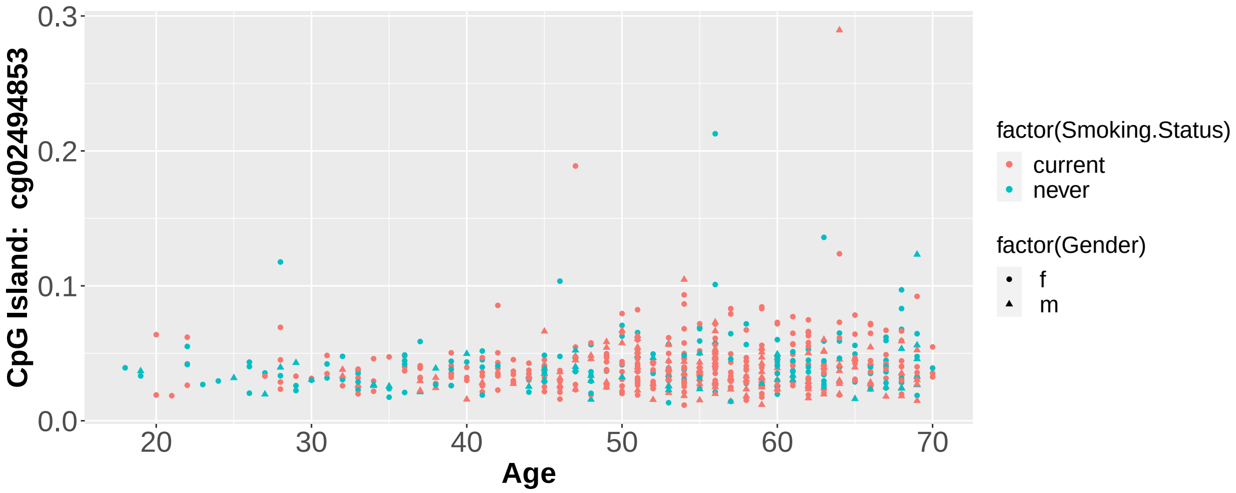
cg00213748 : Scatterplot visualizing the methylation rate on this CpG island of different ages and genders.



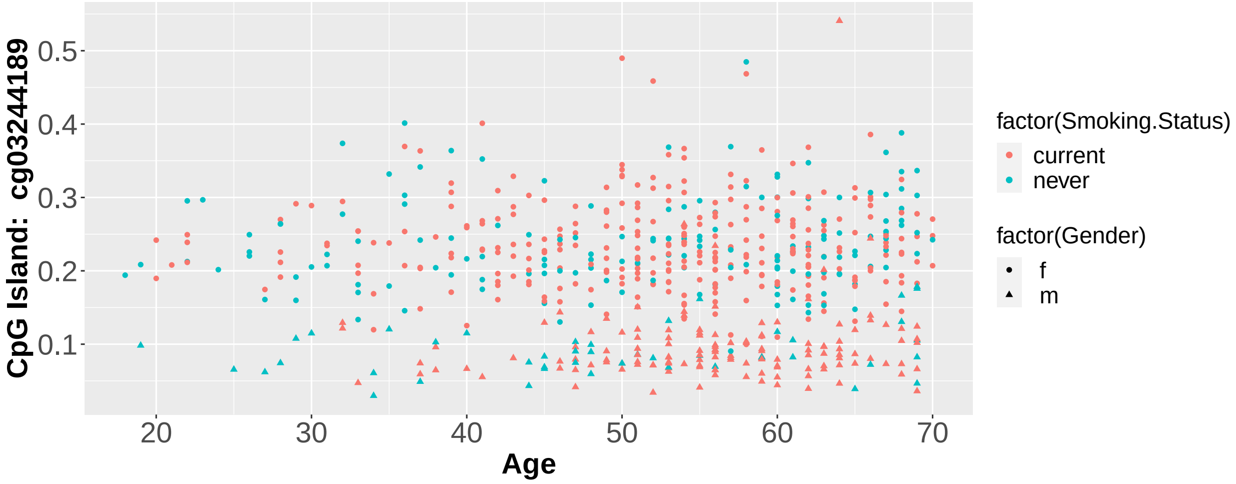
cg00455876 : Scatterplot visualizing the methylation rate on this CpG island of different ages and genders.



cg01707559 : Scatterplot visualizing the methylation rate on this CpG island of different ages and genders.



cg02494853 : Scatterplot visualizing the methylation rate on this CpG island of different ages and genders.



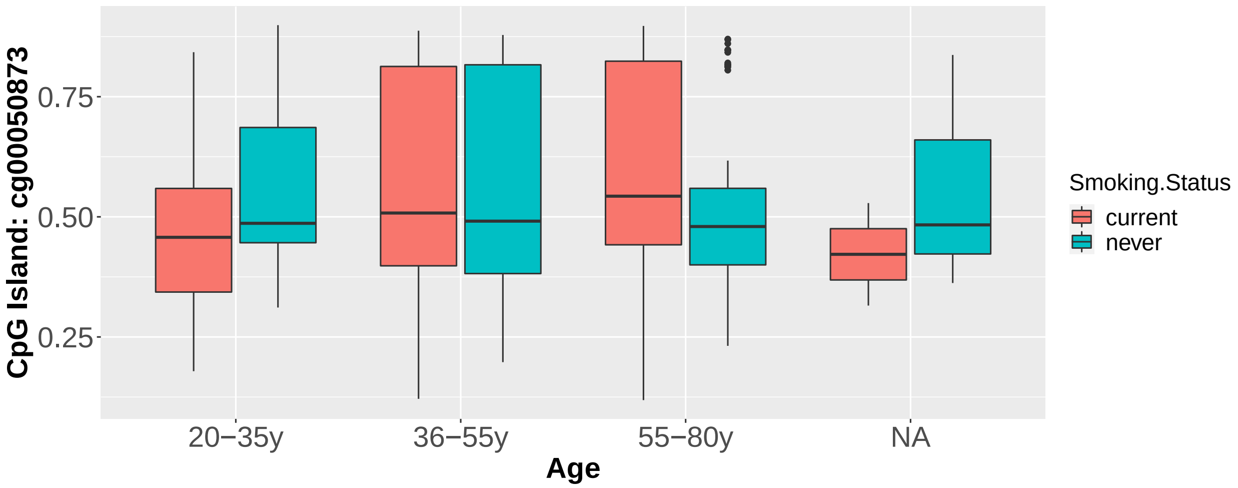
cg03244189 : Scatterplot visualizing the methylation rate on this CpG island of different ages and genders.

Comparing two variables

With the age distribution in mind, let's try to plot 2 methylation rates with the age factored as groups.

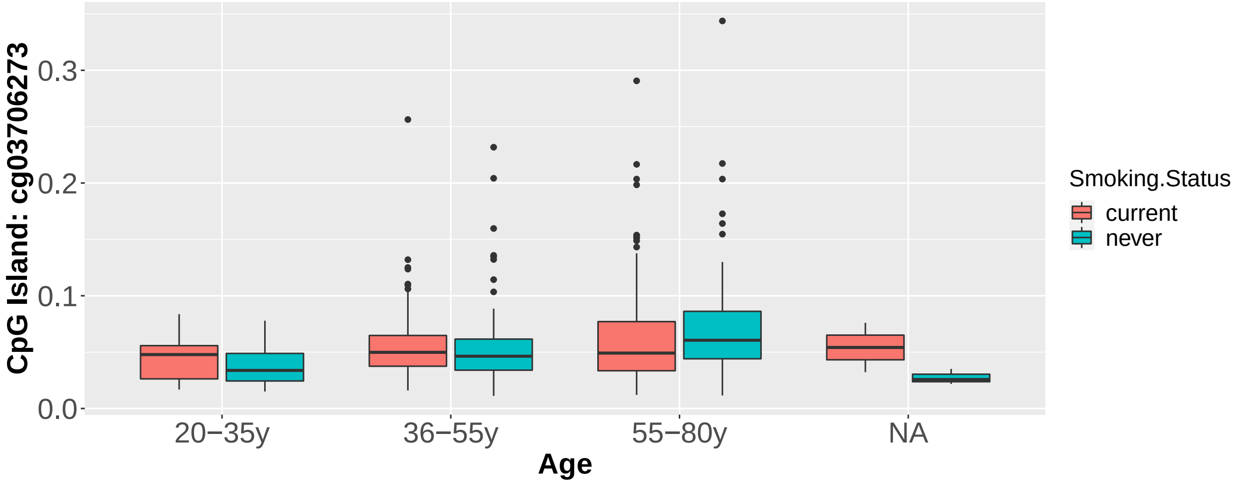
```
age.groups <- cut(myData$Age, breaks = c(20, 35, 55, 80), labels = c("20-35y", "36-55y", "55-80y"))

myData$ClassAge <- factor(age.groups)
ggplot(data = myData, mapping = aes(x = ClassAge, y = cg00050873, fill = Smoking.Status)) +
    geom_boxplot() +
    labs(caption = paste("cg00050873: Scatterplot visualizing the methylation rate on this CpG island o
    theme(plot.caption = element_text(size=14, face="italic")) +
    theme(axis.text = element_text(size = 20)) +
    theme(axis.title = element_text(size = 20, face="bold")) +
    theme(legend.text = element_text(size = 16)) +
    theme(legend.title = element_text(size=16)) +
    ylab(paste("CpG Island: cg00050873")) +
    xlab("Age")
```



cg00050873: Scatterplot visualizing the methylation rate on this CpG island of different ages and genders.

```
ggplot(data = myData, mapping = aes(x = ClassAge, y = cg03706273, fill = Smoking.Status)) +
    geom_boxplot() +
    labs(caption = paste("cg03706273: Scatterplot visualizing the methylation rate on this CpG island o
    theme(plot.caption = element_text(size=14, face="italic")) +
    theme(axis.text = element_text(size = 20)) +
    theme(axis.title = element_text(size = 20, face="bold")) +
    theme(legend.text = element_text(size = 16)) +
    theme(legend.title = element_text(size=16)) +
    ylab(paste("CpG Island: cg03706273")) +
    xlab("Age")
```



cg03706273: Scatterplot visualizing the methylation rate on this CpG island of different ages and genders.

Data mining

Quality metrics

In this week we define the best algorithm for our data with true positive, false positive, true negative and false negative.

Algorithms

The algorithms we are going to compare are: ZeroR, OneR, J48, NaiveBayes, IBK, Simple Logistics, SMO and Random Forest. Below you can see a table with the most important results of all algorithms.

```r
ML_dataLoc <- "data/Smoker_Epigenetic_dfExperiment.csv"
ML_data <- read.csv(ML_dataLoc)
ML_ZeroR <- ML_data[c(1:100),]
ML_OneR <- ML_data[c(101:200),]
ML_J48 <- ML_data[c(201:300),]
ML_IBK <- ML_data[c(301:400),]
ML_NaiveBayes <- ML_data[c(401:500),]
ML_SimpleLog <- ML_data[c(501:600),]
ML_SMO <- ML_data[c(601:700),]
ML_RandomForest <- ML_data[c(701:800),]
algos <- list(ML_ZeroR, ML_OneR, ML_J48, ML_IBK, ML_NaiveBayes, ML_SimpleLog, ML_SMO, ML_RandomForest)

avgs_pc <- list()
avgs_pi <- list()
avgs_tp <- list()
avgs_fp <- list()
avgs_tn <- list()
avgs_fn <- list()
avgs_pr <- list()
avgs_rc <- list()
avgs_roc <- list()

for (algo in algos) {
  percent_correct <- algo[3]
  avg <- lapply(percent_correct, mean)
  avgs_pc[[length(avgs_pc) + 1]] <- avg
}
for (algo in algos) {
  percent_incorrect <- algo[4]
  avg <- lapply(percent_incorrect, mean)
  avgs_pi[[length(avgs_pi) + 1]] <- avg
}
for (algo in algos) {
  TP <- algo[5]
  avg <- lapply(TP, sum)
  avgs_tp[[length(avgs_tp) + 1]] <- avg
}
for (algo in algos) {
  FP <- algo[6]
  avg <- lapply(FP, sum)
  avgs_fp[[length(avgs_fp) + 1]] <- avg
}
for (algo in algos) {
  TN <- algo[7]
  avg <- lapply(TN, sum)
  avgs_tn[[length(avgs_tn) + 1]] <- avg
}
for (algo in algos) {
  FN <- algo[8]
  avg <- lapply(FN, sum)
  avgs_fn[[length(avgs_fn) + 1]] <- avg
}
for (algo in algos) {
  precision <- algo[9]
  avg <- lapply(precision, mean)
  avgs_pr[[length(avgs_pr) + 1]] <- avg
}
for (algo in algos) {
  recall <- algo[10]
  avg <- lapply(recall, mean)
  avgs_rc[[length(avgs_rc) + 1]] <- avg
}
for (algo in algos) {
  roc_area <- algo[11]
  avg <- lapply(roc_area, mean)
  avgs_roc[[length(avgs_roc) + 1]] <- avg
}
avgs_df <- data.frame(row.names = c("ZeroR", "OneR", "J48", "IBK", "NaiveBayes", "SimpleLogistics", "SM

vec_pc <- unlist(avgs_pc)
vec_pi <- unlist(avgs_pi)
vec_tp <- unlist(avgs_tp)
vec_fp <- unlist(avgs_fp)
vec_tn <- unlist(avgs_tn)
vec_fn <- unlist(avgs_fn)
vec_pr <- unlist(avgs_pr)
vec_rc <- unlist(avgs_rc)
vec_roc <- unlist(avgs_roc)

avgs_df$percent_correct_avgs <- vec_pc
avgs_df$percent_incorrect_avgs <- vec_pi
avgs_df$TP_sum <- vec_tp
avgs_df$FP_sum <- vec_fp
avgs_df$TN_sum <- vec_tn
avgs_df$FN_sum <- vec_fn
avgs_df$precision_avgs <- vec_pr
avgs_df$recall_avgs <- vec_rc
avgs_df$ROC_Area_avgs <- vec_roc

knitr::kable(avgs_df)
```

	percent_correct_avgs	percent_incorrect_avgs	TP_sum	FP_sum	TN_sum	FN_sum	precis
ZeroR	68.92217	31.07783	4280	1930	0	0	0
OneR	65.64926	34.35074	3874	1727	203	406	0
J48	68.92217	31.07783	4280	1930	0	0	0
IBK	67.72888	32.27112	4156	1880	50	124	0
NaiveBayes	47.43958	52.56042	1584	568	1362	2696	0
SimpleLogistics	68.92217	31.07783	4280	1930	0	0	0
SMO	68.92217	31.07783	4280	1930	0	0	0
RandomForest	68.48797	31.51203	4249	1926	4	31	0

```
write.csv(avgs_df, file = "data/Smoker_Epigenetic_df_Summary.csv")
```

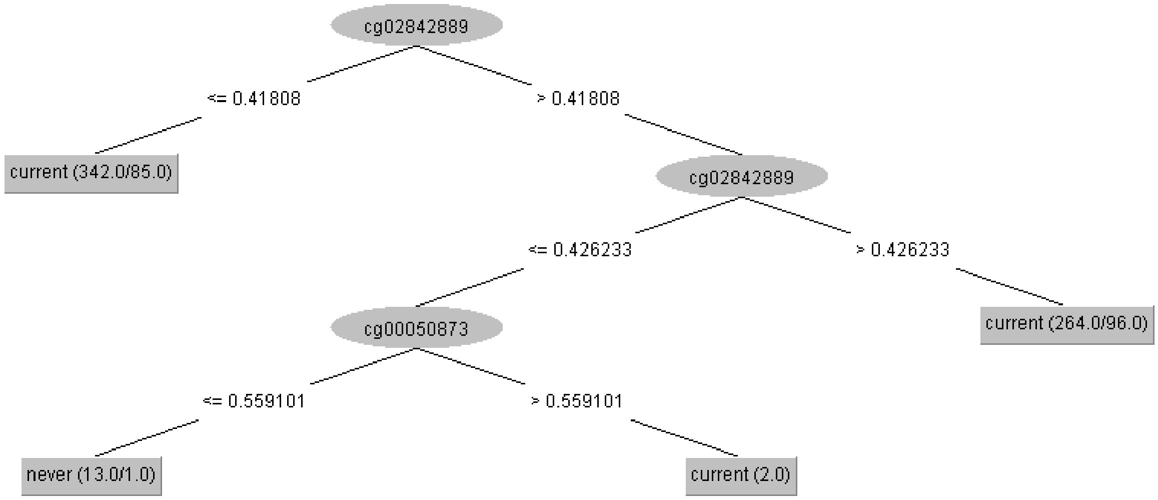


Figure 1: The generated tree of the J48 algorithm. Apparently these CpG islands say the most about whether a person smokes or not.

Discussion

So far we don't have a concrete answer to the research question, because the algorithms aren't extremely overwhelming. So for now it isn't really possible to determine whether someone is smoking or not, because the data is too much spreaded out and not really divided into groups.

It's quite obvious that every algorithm except for NaiveBayes are very close to eachother, the only significant lower scores according to weka are NaiveBayes and OneR, the best (not really significant) algorithm are surprisingly ZeroR, J48, SimpleLogistics and SMO.