Eksamensoppgave INS300 Data Science – Høsten 2018

Vi har valgt at bruke filen «restaurant-and-market-health- inspections.csv» som følger med eksamensoppgaven

Including the dataset

```
  dataset <- read.csv("csv/restaurant-and-market-health-inspections.csv")
```

Including libraries

```
library(ggplot2) # usage: plotting tool
library(magrittr) # usage: pipeline
library(zipcode) # usage: cleaning zipcodes / giving lat and lng coordinates
data(zipcode) # usage: dataset for matching zipcodes
library(lubridate) # usage: gives better date functions
require(dplyr) # usage: data manipulation
library(leaflet) # usage: creates a map with nodes
library(leaflet.extras) # usage: to get heatmap
library(htmlwidgets) # usage: save leaflet to html

# to install the development version from Github, run this command
devtools::install_github("rstudio/leaflet")
```

Data preperation

```
# cleaning zipcodes
dataset$facility_zip <- clean.zipcodes(dataset$facility_zip)
# format the date from POSIX to more usable date format
dataset$activity_date <- dataset$activity_date %>% as.Date()
# uses lubridate to create new date columns
dataset$activity_year <- dataset$activity_date %>% year()
dataset$activity_month <- dataset$activity_date %>% month()
dataset$activity_day <- dataset$activity_date %>% day()

# getting longitude and latitude
dataset$lng <- zipcode$longitude[match(dataset$facility_zip, zipcode$zip)]
dataset$lat <- zipcode$latitude[match(dataset$facility_zip, zipcode$zip)]

# clean away empty values in program_name
dataset <- dataset[!is.na(dataset$program_name),]

# clean away the empty value in grade
dataset <- dataset[-c(49376),]

# clean away empty values in lat and longitude values
dataset <- dataset[!is.na(dataset$lat),]
dataset <- dataset[!is.na(dataset$lng),]
```

Tabell mean score per år.

```r
# Average per year
dataset %>%
  group_by(activity_year) %>%
  summarize(average=mean(score)) %>%
  as.data.frame()
```

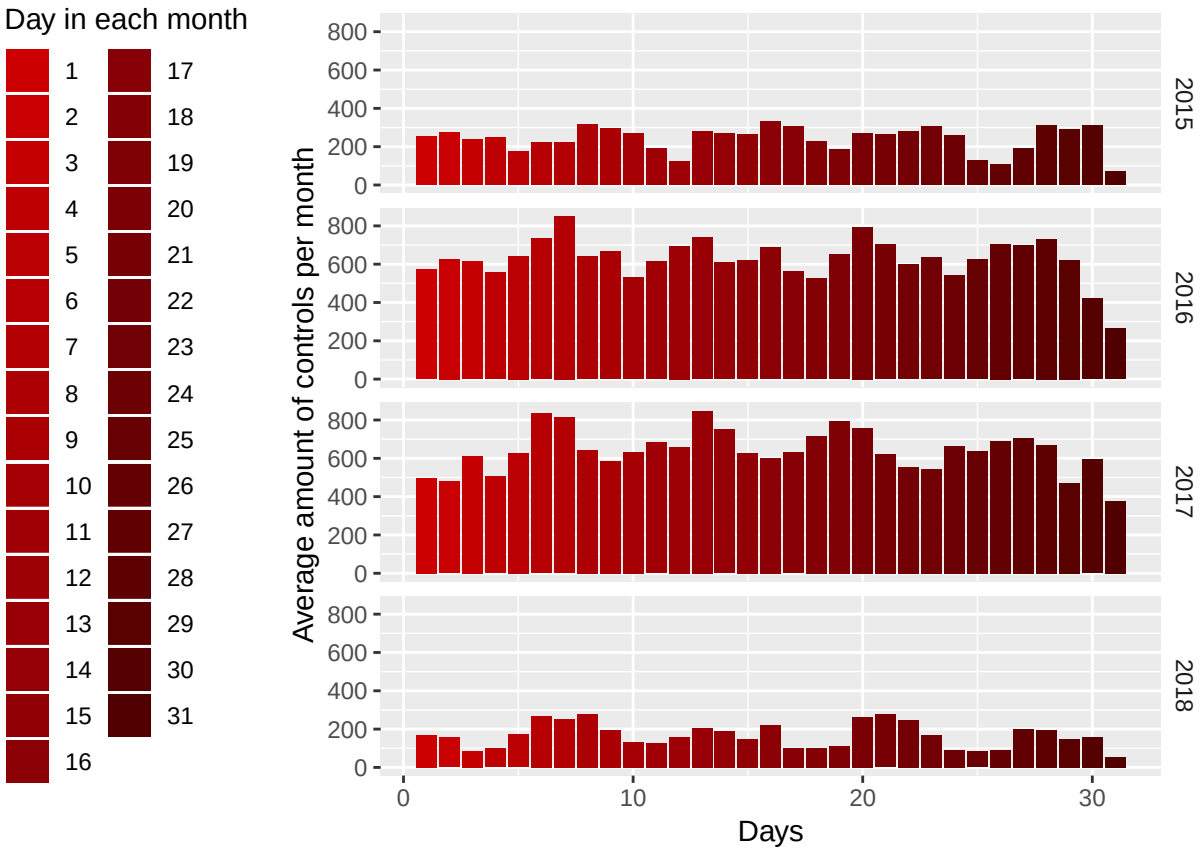
```
##   activity_year  average
## 1          2015 92.89445
## 2          2016 93.29196
## 3          2017 93.39620
## 4          2018 93.78696
```

Inspeksjoner årligt

```r
# Gives a color scale for 31 days
colorscale <- c('#cc0000','#c80001','#c40001','#be0002','#ba0002','#b60003','#b20003','#ad0004','#aa000

# Removes inactive dates
dataset_no_inactive <- dataset[dataset$program_status == "ACTIVE",]

# inspections on specific days, each year
ggplot(dataset_no_inactive, aes(activity_day)) +
  geom_bar(aes(fill = factor(activity_day))) +
  scale_fill_manual(values = colorscale) +
  facet_grid(activity_year ~ .) +
  theme(legend.position = "left", strip.background = element_rect(colour = "#FFFFFF",fill = "#FFFFFF"))
  guides(fill = guide_legend("Day in each month")) +
  ylab("Average amount of controls per month") + xlab("Days")
```



Kod för maps

```
# Only shows results with:
# A
sorted <- dataset[dataset$score <= 100 & dataset$score >= 90,]
# B
sorted <- dataset[dataset$score <= 89 & dataset$score >= 80,]
# C
sorted <- dataset[dataset$score <= 79 & dataset$score >= 64,]

# sets the scale for the legend
mybins=seq(64,79, by=1)
# sets colors for everything in the maps
mypalette = colorBin(palette="PuBu", domain=sorted$score, na.color="transparent", bins=mybins)

# Prepar the text for the tooltip:
textnode=paste("Grade: ", sorted$grade, "<br/>", "Score: ", sorted$score, "<br/>", "Zipcode: ", sorted$

# Creates the map
savedMap <- leaflet(sorted) %>%
  addTiles() %>%
  setView(lat=34, lng=-118, zoom=10) %>%
  addProviderTiles(providers$CartoDB) %>%
  addCircleMarkers(~lng, ~lat,
    fillColor = ~mypalette(score), fillOpacity = 0.7, color="black", radius=8, stroke=TRUE,
    label = textnode,
```

```
    labelOptions = labelOptions(style = list("font-weight" = "normal", padding = "3px 8px"), textsize =
  addLegend(pal=mypalette, values=mybins, opacity=0.7, title = "Grade", position = "bottomright")

savedMap

# Saves to a html file for webview
saveWidget(savedMap, file = "c-map.html", selfcontained = TRUE)
```

länk till lösning: https://exam-data-science.firebaseapp.com