

Oppgave 3

Mohamad Ali Al-Mousawy

2/22/2022

```
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5    v purrr  0.3.4
## v tibble  3.1.6    v dplyr  1.0.8
## v tidyr   1.2.0    v stringr 1.4.0
## v readr   2.1.2    v forcats 0.5.1
```

```
## Warning: package 'tidyr' was built under R version 4.0.5
```

```
## Warning: package 'readr' was built under R version 4.0.5
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
```

```
library(rvest)
```

```
##
## Attaching package: 'rvest'
```

```
## The following object is masked from 'package:readr':
##
##   guess_encoding
```

```
library(janitor)
```

```
##
## Attaching package: 'janitor'
```

```
## The following objects are masked from 'package:stats':
##
##   chisq.test, fisher.test
```

Oppgave 1

Ved å bruke `read_html` funksjonen får jeg lastet ned dataen der jeg videre henter ut de forskjellige tabelene.

```
url1 <- read_html("https://www.motor.no/aktuelt/motors-store-vintertest-av-rekkevidde-pa-elbiler/217132")

ggg <- url1 %>%
  html_nodes(xpath="//div") %>%
  html_table(trim = TRUE)
```

Jeg ser videre fra dataen jeg har lastet inn at og velger ut listen jeg trenger Ved funksjonen dobbel brackets `[[]]`, velger jeg liste nr 35. Der jeg ser at dataframe table. Og rydder unna resten av informasjonen jeg har fra nettsiden. Sletter også bilene som ikke har informasjon og har verdien X i seg.

```
ggg[[35]]
```

```
## # A tibble: 68 x 12
##   X1      X2    X3    X4    X5    X6    X7    X8    X9    X10   X11   X12
##   <chr>   <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr>
## 1 Modell (te~ WLTP~ STOPP Avvik <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
## 2 Tesla Mode~ 614 ~ 521 ~ -15,~ <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
## 3 Mercedes-B~ 645 ~ 513 ~ -20,~ <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
## 4 BMW iX xDr~ 591 ~ 503 ~ -14,~ <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
## 5 Tesla Mode~ 507 ~ 451 ~ -11,~ <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
## 6 Volkswagen~ 539 ~ 435 ~ -19,~ <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
## 7 Kia EV6 2WD 528 ~ 429 ~ -18,~ <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
## 8 NIO ES8 LR~ 488 ~ 425 ~ -12,~ <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
## 9 Kia EV6 4WD 484 ~ 423 ~ -12,~ <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
## 10 Volkswagen~ 485 ~ 414 ~ -14,~ <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>
## # ... with 58 more rows
```

```
Table <- ggg[[35]] %>%
  slice(1:34) %>%
  select(1:4)

Table <- Table %>% row_to_names(row_number = 1)

Table <- Table %>%
  slice(1:18, 20:25, 27:33)
```

Separerer bort km/h fra wltp radene for å få orden på tabellen, og verdiene hver for seg.

```
Table <- Table %>%
  separate(`WLTP-tall`, sep = "/", into=c("wltp","kWh"))

Table$STOPP <- gsub("km","", as.character(Table$STOPP))

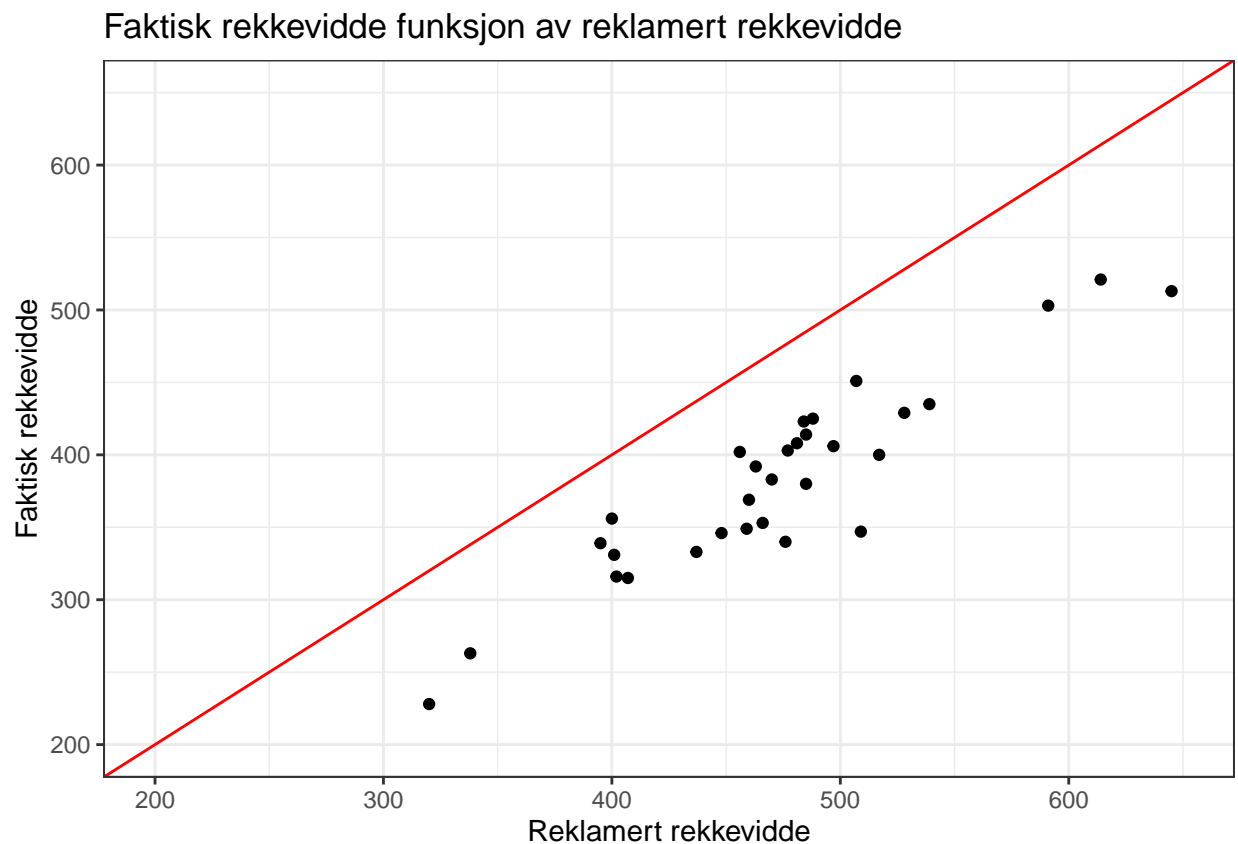
Table$wltp <- gsub("km","", as.character(Table$wltp))
```

Rydder unna og gjør om wltp og stopp til numeric for å ha verdiene til tabbelen senere i oppgaven.

```
Table$wltp <- as.numeric(Table$wltp)
```

```
Table$STOPP <- as.numeric(Table$STOPP)
```

```
ggplot(Table, aes(wltp,STOPP)) +  
  geom_point() +  
  geom_abline(col = "red") +  
  scale_x_continuous(limits=c(200,650)) +  
  scale_y_continuous(limits=c(200,650)) +  
  labs(title = "Faktisk rekkevidde funksjon av reklamert rekkevidde",  
        x = "Reklamert rekkevidde", y = "Faktisk rekkevidde") +  
  theme_bw()
```



Oppgave 2

En linjær funksjon har likningen $Y = a + bX$, hvor verdien av Y er den avhengige variabelen. Der verdien av y når $x = 0$.

```
lm(Table$STOPP~Table$wltp, data = Table)
```

```
##
```

```
## Call:
```

```
## lm(formula = Table$STOPP ~ Table$wltp, data = Table)
```

```
##
## Coefficients:
## (Intercept)    Table$wltp
##      -26.6450      0.8671
```

Modellen sier oss at Faktisk rekkevidde er den avhengige variabelen (y), som blir forklart av en Reklamert rekkevidde. Der 1 enhets endring i variabelen x gir Beta endring i variabelen y. Ved å sette inn en (lm) funksjon som er lineær funksjon, får vi korrelasjonen mellom Faktisk rekkevidde og Reklamert rekkevidde. der den har en stigningstall på 0.8671.

Bruker lm funksjonen for å få den lineære modellen vi blir spurt om. Og ser at det er forventet 0.86km i WLTP rekkevidden. Som sier oss at for hver rekkevidde som er reklamert er det ca. 87% prosent av den kjørelengden og -13% i avvik. Ved å plusse sammen den lineære funksjonen og gange en valgt bil får vi da den faktiske rekkevidden i -10 grader. Feks $-26.6450 + 0.8671 * \text{Tesla 3 (614km)} = 505.7544$ er den faktiske rekkevidden i -10 grader. Ved å bruke Geom_smooth funksjonen får vil til å plote det. Jeg valgte å bruke "SE = FALSE", for å ikke vise koffdiens linje eller nivåene i plotten, som er på 95%.

```
ggplot(Table, aes(wltp, STOPP)) +
  geom_point(aes(color="Faktisk")) +
  labs(color = "Rekkevidder") +
  ggtitle("Faktisk rekkevidde som funksjon av Reklamert rekkevidde") +
  scale_x_continuous(name="Reklamert rekkevidde", limits=c(200,650)) +
  scale_y_continuous(name="Faktisk rekkevidde", limits=c(200,650)) +
  geom_abline(aes(intercept=0, slope=1, color="Reklamert"), show.legend = FALSE) +
  geom_smooth(method = lm, aes(color="lm_smooth"), se = FALSE) +
  scale_color_manual(values = c("Faktisk"="black", "Reklamert" = "firebrick", "lm_smooth"="turquoise3"),
    labels=c("Faktisk", "Reklamert", "lineær")) +
  theme_bw()
```

```
## 'geom_smooth()' using formula 'y ~ x'
```

Faktisk rekkevidde som funksjon av Reklamert rekkevidde

