STK1000: Løsningsforslag Uke 36

H2023

Oppgave 1.15 (R)

Datasettene kan bli lastet ned fra emnets semesterside og vi kan laste inn dette datasettet i R med

```
data = read.csv('../../ips10e_csv_data_sets/ips10e_ch1_csv_data_sets/ex01-015KPOT40.csv')
head(data)
```

```
  ID Potassium_mg Dose Source
1  1      3096.79   40 Potato
2  2      3607.14   40 Potato
3  3      3584.78   40 Potato
4  4      3256.73   40 Potato
5  5      3286.95   40 Potato
6  6      2788.97   40 Potato
```

Merk at vi har lastet ned CSV filene (ikke R filene). Snakk med gruppelærer hvis du har problemer med dette (det kan være litt vrient første gang). På Windows-maskiner kan det være man må bruke "\" i stedet for "/" i filstien i `read.csv`.

Variabelen `data` er nå en `data.frame` som inneholder flere kolonner med informasjon. Vi er kun interessert i `Potassium_mg` så vi henter ut denne

```
x = data$Potassium_mg
```

a) Vi kan lage et stemplot i R med følgende kommando som avrunder til nærmeste 10 (tallene er 10 ganger større enn i stemplottet). Prøv selv å forandre på `scale` variabelen for å se hvordan det påvirker stemplottet.

```
stem(x, scale = 1)
```

```
The decimal point is 2 digit(s) to the right of the |

26 | 69
28 | 5688
30 | 357702235
32 | 336689
34 | 9148
36 | 1
38 |
40 |
42 | 1
```

b) Vi ser at fordelingen er ganske symmetrisk, men det er litt få datapunkter til å kunne konkludere med dette.

c) Det ser ut til å kanskje være en outlier med verdi 4210. Denne observasjonen er ganske stor i forhold til resten.

d) Formen er ganske symmetrisk, midten er rundt 3010, og fordelingen er mellom 2660 og 4210.

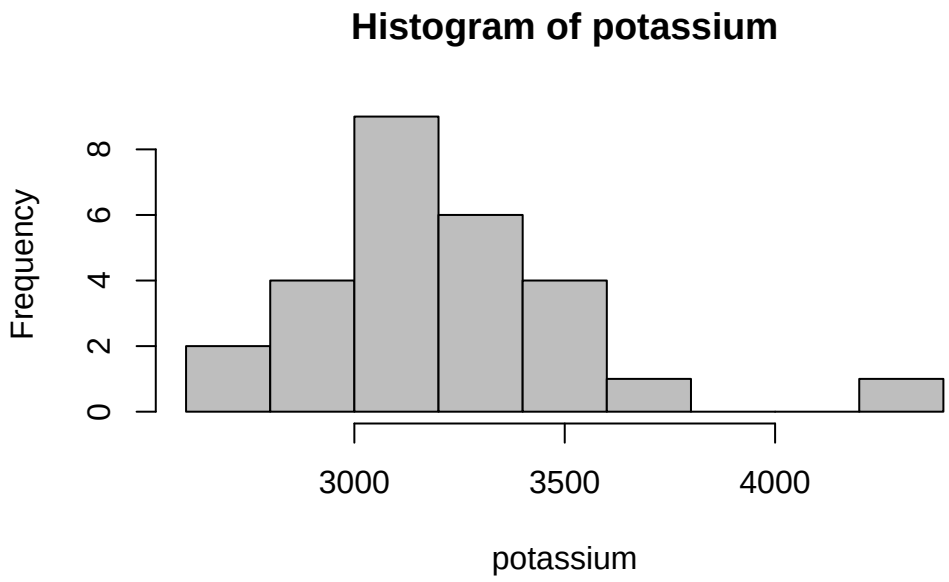
Oppgave 1.33(R)

a)

Dette er en fortsettelse på koden fra oppgave 1.15, så vi fortsetter fra der vi slapp (med dataene lastet inn i R).

Vi lager først et histogram

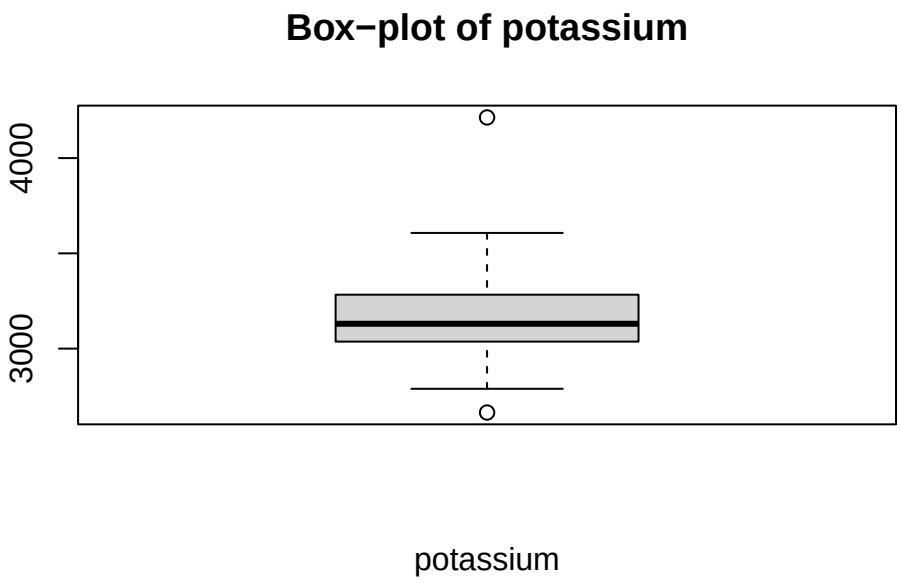
```
hist(x, col = "gray", xlab = "potassium", main = "Histogram of potassium")
```



b)

Vi lager et boksplott av de samme dataene

```
boxplot(x, xlab = "potassium", main = "Box-plot of potassium")
```



c)

Fordelen med stemplottet er at vi faktisk ser tallverdiene i plottet, ellers pleier vi å foretrekke et histogram (spesielt for større datasett). Et boksplott gir noe av den samme informasjonen som et histogram, men ikke like mye detaljer.

Oppgave 1.44(R)

For alkoholprosentene beregner vi minimum, median, gjennomsnitt, standardavvik og maksimum. Dette gir oss et innblikk i fordelingen til dataene.

```
options(width=100)
beer = read.csv('../../ips10e_csv_data_sets/ips10e_ch1_csv_data_sets/ex01-044BEER.csv')
head(beer)
```

```
     Type                          BEER        Brewery Calories Carbo.g. Alcohol.pct.
1 Domestic         American Amber Lager Straub Brewery      136     10.5          4.1
2 Domestic               American Lager Straub Brewery      132     10.5          4.1
3 Domestic               American Light Straub Brewery       96      7.6          3.2
4 Domestic                 Anchor Steam         Anchor      153     16.0          4.9
5 Domestic Anheuser Busch Natural Light Anheuser Busch       95      3.2          4.2
6 Domestic   Anheuser Busch Natural Ice Anheuser Busch      157      8.9          5.9
```

```
alcohol = beer$Alcohol
c(min(alcohol), median(alcohol), mean(alcohol), sd(alcohol), max(alcohol))
```

```
[1]  0.400000  4.900000  5.246875  1.404213 11.500000
```

Alternativt kan vi bruke `summary` funksjonen som gir kvartiler i stedet for standardavvik.

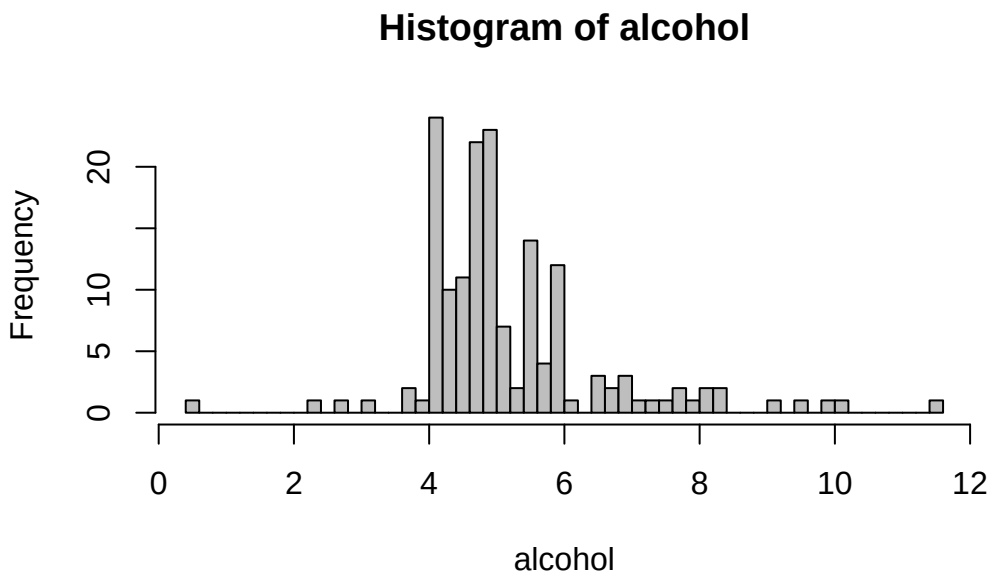
```r
summary(alcohol)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  0.400   4.400   4.900   5.247   5.625  11.500 
```

Vi kan gjerne bruke verdien for medianen, samt de to kvartilene for å beskrive senter og spredning i fordelingen.

Vi lager også et histogram av alkoholprosentene som gir et grafisk innblikk i fordelingen.

```r
hist(alcohol, col = 'gray', breaks = 40)
```



Vi ser vi har en enkelt øl med veldig lav alkoholprosent. O'Doul's er en alkoholfri øl og skiller seg derfor fra resten. Den kan man finne med å se på datasettet eller man kan finne raden i R med følgende kommando

```r
idx_min = which.min(alcohol)
beer[idx_min,]
```

```
        Type     BEER        Brewery Calories Carbo.g. Alcohol.pct.
110 Domestic O'Doul's Anheuser Busch       70     13.3          0.4
```

Oppgave 1.45(R)

Koden i denne oppgaven baserer seg på forrige oppgave.

a)

Vi kan fjerne outlieren fra dataene med følgende kommando

```r
alcohol_no = alcohol[-idx_min]
```

Vi kan nå regne ut gjennomsnitt

```
c(mean(alcohol), mean(alcohol_no))
```

```
[1] 5.246875 5.277358
```

og median

```
c(median(alcohol), median(alcohol_no))
```

```
[1] 4.9 4.9
```

Vi ser at medianen ikke forandre seg når vi fjernere outlieren, mens gjennomsnitt verdien blir større.

b)

Vi gjentar dette for standardavvik og ser at det blir mindre uten outlieren.

```
c(sd(alcohol), sd(alcohol_no))
```

```
[1] 1.404213 1.354501
```

For kvartilene ser vi at det er mindre forskjeller (bortsett fra 0% som er minimumsverdien).

```
print(quantile(alcohol))
```

```
    0%    25%    50%    75%   100%
 0.400  4.400  4.900  5.625 11.500
```

```
print(quantile(alcohol_no))
```

```
   0%   25%   50%   75%  100%
 2.40  4.45  4.90  5.65 11.50
```

c)

Vi ser at det outlieren har liten påvirkning ettersom forskjellene i a) og b) er veldig små. Dette kommer av at verdien ikke er veldig forskjellig fra resten av datasette, i tillegg til at vi har ganske mange observasjoner.

Oppgave 1.56(R)

a)

Vi laster inn trediametrene og regner ut minimum, 25 % kvartil, median, 75 % kvartil og maksimum. Utskriften under inneholder også gjennomsnitt, men den kan vi se bort i fra.

```r
tree_data = read.csv('../../ips10e_csv_data_sets/ips10e_ch1_csv_data_sets/ex01-056PINES.csv')
head(tree_data)
```

```
  Diameter
1     10.5
2     13.3
3     26.0
4     18.3
5     52.2
6      9.2
```

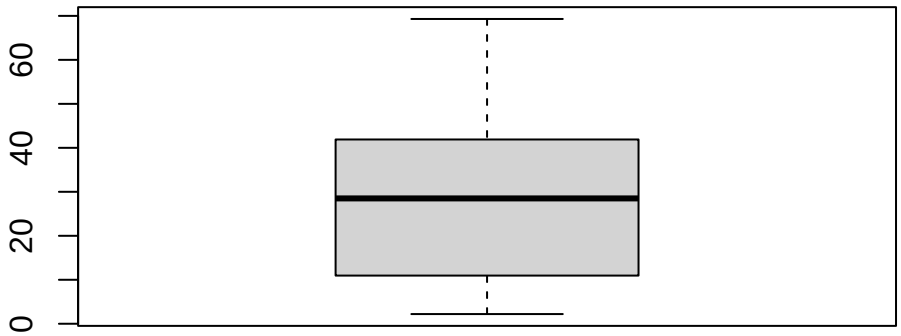
```r
trees = tree_data$Diameter
summary(trees)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   2.20   11.18   28.50   27.29   41.20   69.30
```

b)

Vi lager et boksplott med følgende kommando

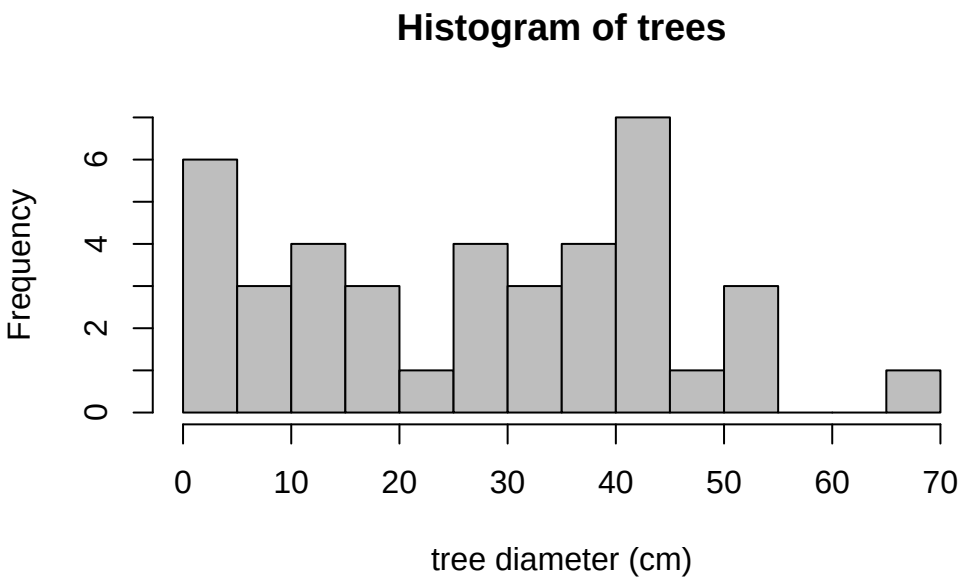
```r
boxplot(trees)
```



c)

Vi kan lage et histogram med følgende kommando

```r
hist(trees, breaks = 10, col = "gray", xlab = "tree diameter (cm)")
```



d)

Begge figurer viser en høyreforskyvning. Eller gir histogrammet mer detaljer enn boksplottet, noe som typisk er en fordel.

Oppgave 1.60(R)

En centimeter er 0.39 tommer. Vi kan derfor konvertere til tommer med å multipliseres med dette tallet.

```
trees_inch = trees * 0.39
summary(trees_inch)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  0.858   4.358  11.115  10.643  16.068  27.027 
```

Vi ser at disse verdiene har forandret seg med en faktor på 0.39.

Figurene fra 1.88 b) og c) er visuelt like bare med forskjellige tall på aksen som tilsvarer diameter. Man kan se noen små forskjeller i histogrammet på grunn av avrundinger, men dette er avhengig av ditt valg av `breaks` i `hist` kommandoen. Figurene kan bli plottet med samme kommandoer som tidligere, men vi tar de ikke med her.

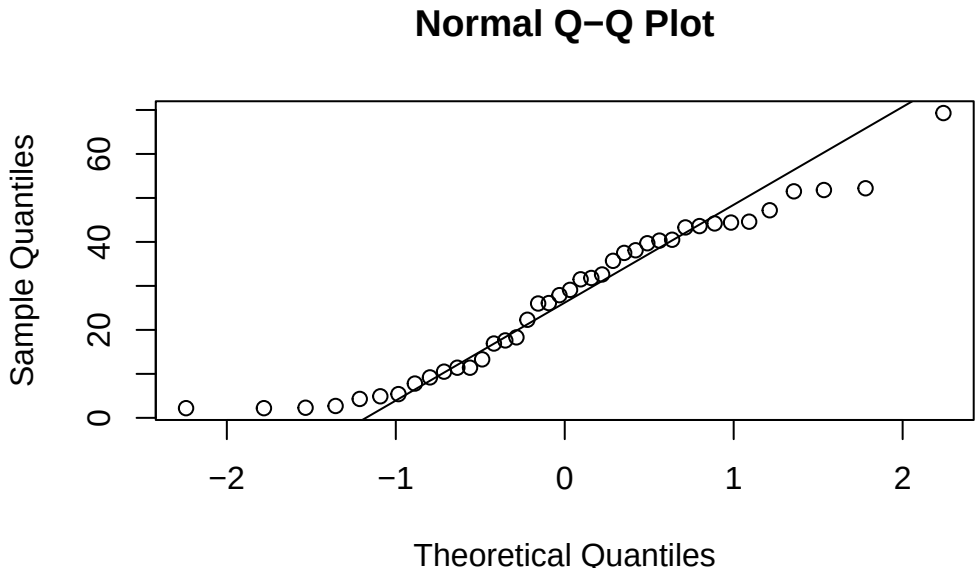
Oppgave 1.103(R)

Vi laster inn diameterdataene og lager et Q-Q plott med følgende kode

```
data = read.csv('../../ips10e_csv_data_sets/ips10e_ch1_csv_data_sets/ex01-103PINES.csv')
head(data)
```

```
  Diameter
1     10.5
2     13.3
3     26.0
4     18.3
5     52.2
6      9.2
```

```r
x = data$Diameter
qqnorm(x)
qqline(x)
```



Vi ser at punktene ikke følger linjen så bra, så fordelingen er nok ikke helt normal. Det er indikasjoner på høyreforskyvning, men det er ikke veldig tydelig.