STK1000: Løsningsforslag Uke 35

2019

Oppgave 1.4

Individene (cases) i dette datasettet er leiligheter (apartments) og vi har 5 variabler:

- Husleie: Kvantitativ
- Inkludert kabel-TV : Kategorisk
- Lov med Kjæledyr : Kategorisk
- Antall soverom: Kvantitativ
- Avstand til campus: Kvantitativ

Oppgave 1.14

a) Her er statene individene (cases).

c) Her er både "antall studenter fra staten som går på college" og "antall studenter som går på college i hjemstaten" kvantitative.

Oppgave 1.17

Vi får et stemplottet

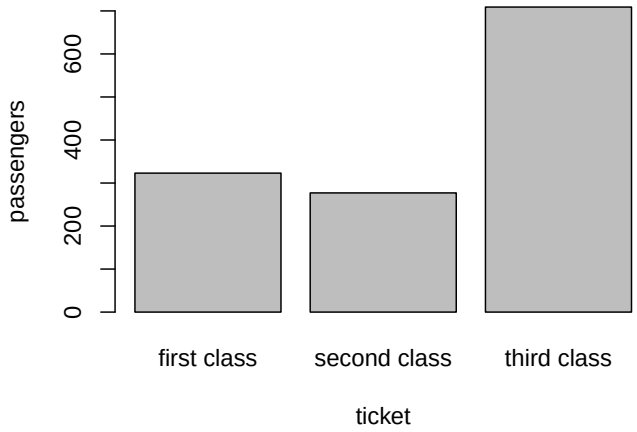
5 | 79
6 | 158
7 | 00335558
8 | 000223557
9 | 00122448

Vi ser at fordelingen ikke er symmetrisk, men i stedet venstreforskjøvet.

Oppgave 1.27 (R og for hånd)

a) I figuren under vise vi et stolpediagram som beskriver passasjerene.

```
passengers = c(323, 277, 709)
tickets = c('first class', 'second class', 'third class')
barplot(passengers, names.arg = tickets, ylab = 'passengers', xlab = 'ticket')
```



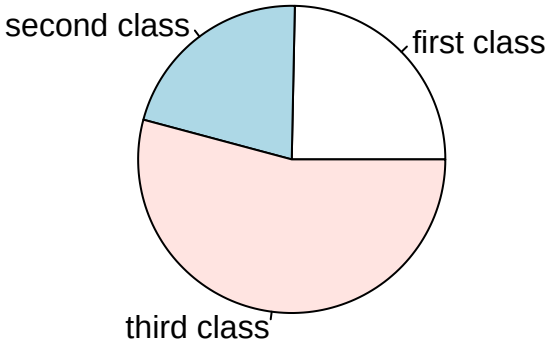
b) Det er klart flest passasjerer i tredje klasse og litt flere passasjerer i første enn i andre klasse.

c) Hvis vi i stedet lager et stolpediagram med prosentandel vil figuren se visuelt lik ut, men tallene på y-aksen vil forandre seg.

Oppgave 1.28(R)

a) Vi lager et sektordiagram av Titanic datasettet.

```
pie(passengers, labels = tickets)
```



b) Fordelen med sektordiagrammet er at vi umiddelbart ser at over halvparten av passasjerene er i tredje klasse. Med andre ord, det er lettere å se de relative størrelsene.

Oppgave 1.30(R)

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

```
data = read.csv('CSV/Chapter 1/EX01-030KPOT40.csv')
```

Merk at vi har lastet ned CSV filene (ikke R filene). Snakk med øvingsassistent 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.

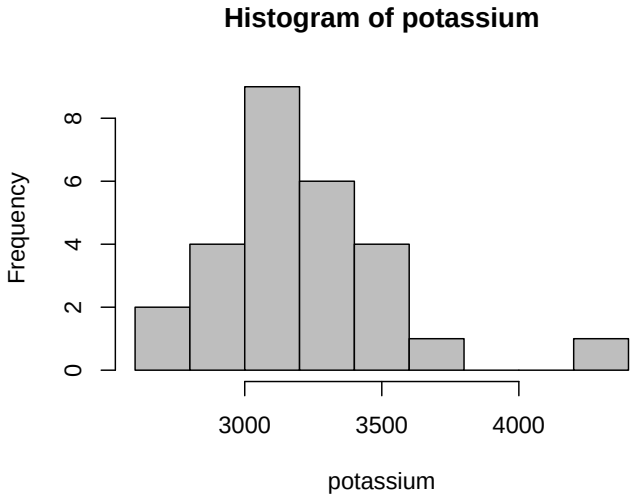
Oppgave 1.61(R)

a)

Dette er en fortsettelse på koden fra oppgave 1.30, så vi fortsetter fra der.

Vi lager først et histogram

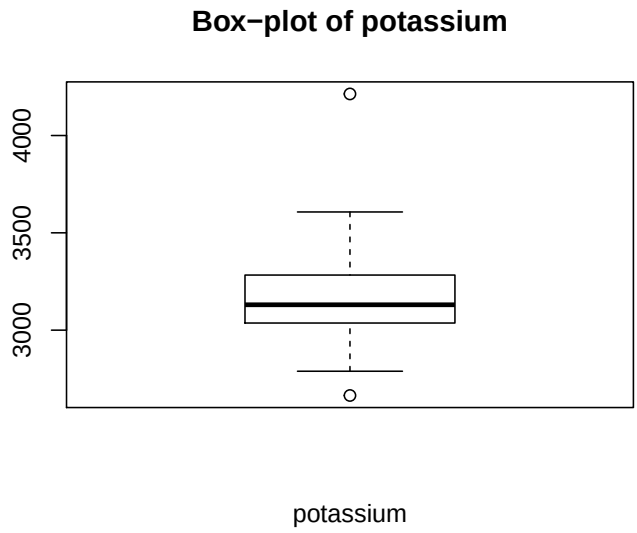
```
hist(x, col = "gray", xlab = "potassium", main = "Histogram of potassium")
```



b)

Vi lager et boksplott av de samme dataene

```r
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 1.72(R)

a)

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

```r
beer = read.csv('CSV/Chapter 1/EX01-072BEER.csv')
alcohol = beer$Alcohol
c(min(alcohol), median(alcohol), mean(alcohol), sd(alcohol), max(alcohol))
```

```
## [1]  0.400000  4.900000  5.171557  1.337357 11.500000
```

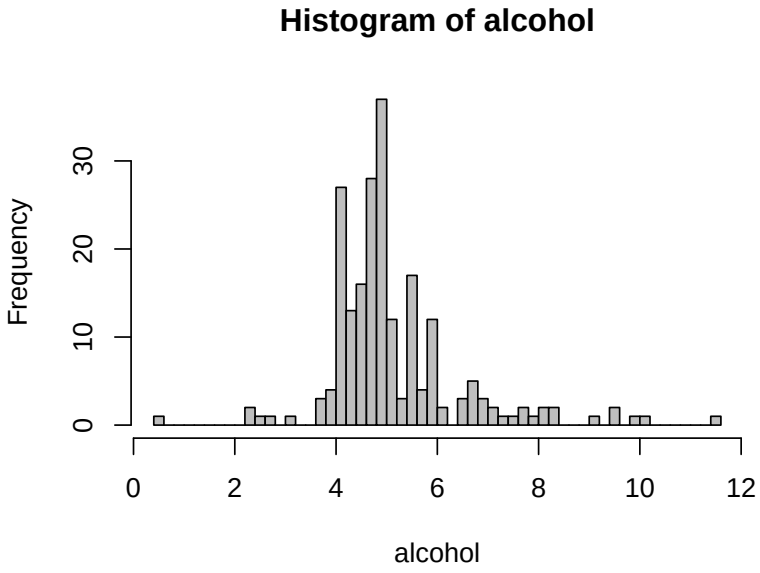
Alternativt kan vi bruke `summary` funksjonen som gir kvartiler i stedet for standardavvik.

```r
summary(alcohol)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.400   4.422   4.900   5.172   5.600  11.500
```

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

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



b)

Vi ser vi har en 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

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

```
##         BEER        Brewery Calories Carbohydrates Alcohol     Type
## 109 O'Doul's Anheuser Busch       70          13.3     0.4 Domestic
```

c)

Her er det ikke noe korrekt svar.

Oppgave 1.73(R)

Koden i denne oppgaven baserer seg på forrige oppgave.

a)

Vi kan fjerne outlieren fra dataene med følgende kommando

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

Vi kan nå regne ut gjennomsnitt

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

```
## [1] 5.171557 5.194171
```

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] 1.337357 1.299272
```

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

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

```
##      0%     25%     50%     75%    100%
##  0.4000  4.4225  4.9000  5.6000 11.5000
```

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

```
##     0%    25%    50%    75%   100%
##  2.300  4.465  4.900  5.600 11.500
```

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 1.75

En slik forskjell kan forekomme hvis vi har mange husstander med formue under 81200, og noen få med veldig mye større formue. Dette er et godt eksempel på farene med å kun oppgi et gjennomsnitt.

Oppgave 1.87

a) Hvis vi velger samme verdi for alle fire tallene får vi et standardavvik på 0.

b) Hvis vi velger 10, 10, 20, 20, får vi det størst mulige standard avvik.

c) I a) finnes det 11 mulige svar: (10, 10, 10, 10), (11, 11, 11, 11), ..., (20, 20, 20, 20). I b) finnes det bare en løsning, med mindre vi er interessert i forskjellige permutasjoner av løsningen: (10, 20, 10, 20), (20, 10, 10, 20), etc.

Oppgave 1.88(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.

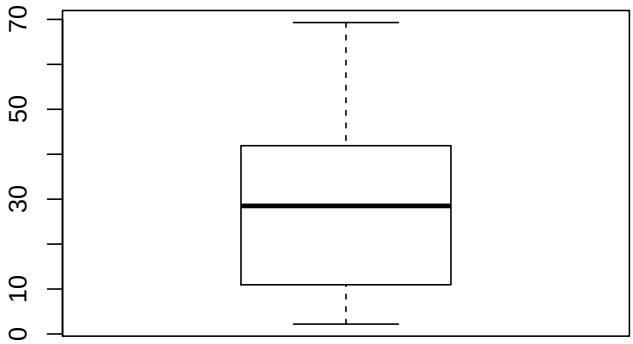
```
tree_data = read.csv('CSV/Chapter 1/EX01-088PINES.csv')
trees = tree_data$Diameter
summary(trees)
```

```
##    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

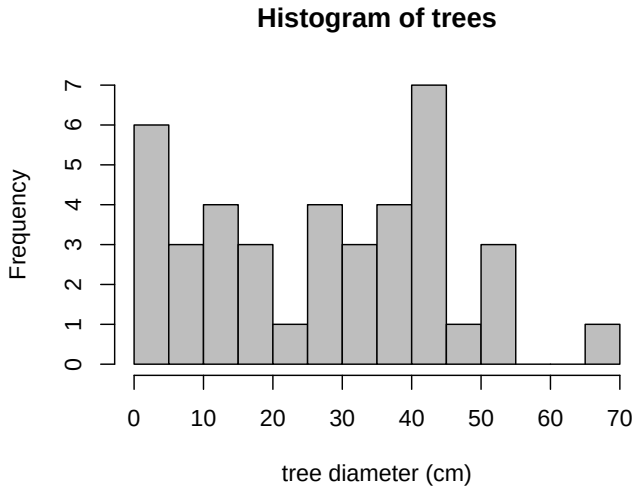
```
boxplot(trees)
```



c)

Vi kan lage et histogram med følgende kommando

```
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 1.92(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  11.120  10.640  16.070  27.030
```

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.