Obligatorisk innlevering 2 – STK1000 – Geologi versjon

Oppgave 1

a)

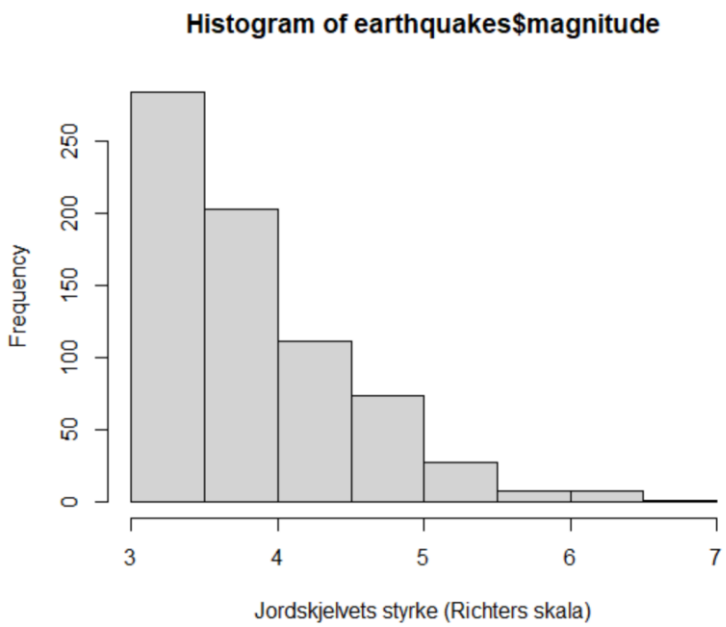
R-kode:

```
#Oppgave 1
#a)
#Laster inn dataene
data <- "https://www.uio.no/studier/emner/matnat/math/STK1000/data/obligdata/oblig2/earthquakes.txt"
earthquakes <- read.table(data, header=TRUE)
#Lager et histogram over dataene
hist(earthquakes$magnitude, xlab='Jordskjelvets styrke (Richters skala)')
#Sjekker om styrken til jordskjelvet er tilnærmet normalfordelt ved qqnorm()
qqnorm(earthquakes$magnitude)
qqline(earthquakes$magnitude)
```

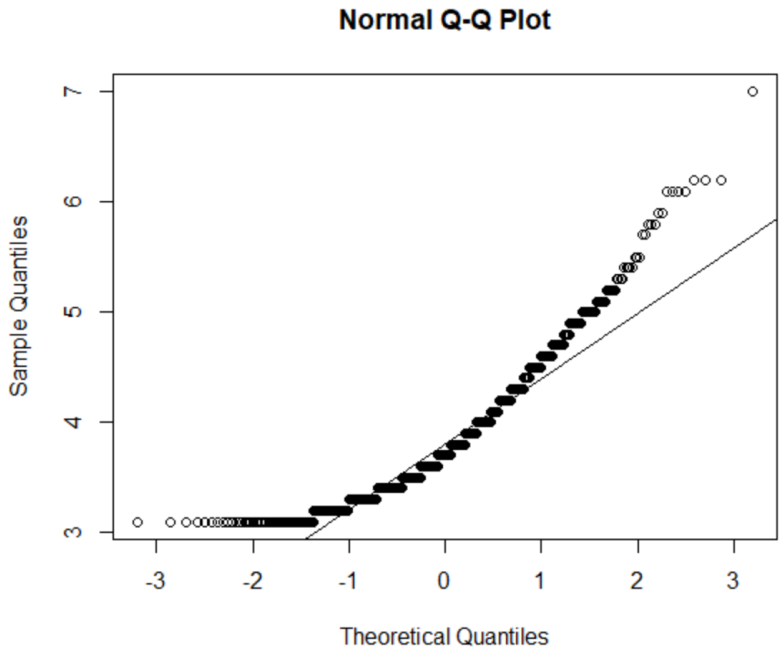
Output fra R/kjøreeksempel:

```
> #Oppgave 1
> #a)
> #Laster inn dataene
> data <- "https://www.uio.no/studier/emner/matnat/math/STK1000/data/obligdata/oblig2/earthquakes.txt"
> earthquakes <- read.table(data, header=TRUE)
> #Lager et histogram over dataene
> hist(earthquakes$magnitude, xlab='Jordskjelvets styrke (Richters skala)')
> #Sjekker om styrken til jordskjelvet er tilnærmet normalfordelt ved qqnorm()
> qqnorm(earthquakes$magnitude)
> qqline(earthquakes$magnitude)
```

Plottet fra R er inkludert i Figur 1 og Figur 2.



Figur 1: Histogram over magnitude



Figur 2: QQ Plott over magnitude

Histogrammet i Figur 1 viser at magnitude ikke virker å være normalfordelt, den er helt klart venstre-skjev. For at et histogram skal være normalfordelt, må det ha en symmetrisk unimodal topp. Benyttet også qqline og qqplot for å være helt sikker på mine antagelser.

Figur 2 viser også at magnitude ikke kan konkluderes å være normalfordelt, ettersom verdiene ikke ligger lineært og nær linjen. Plottet viser også at verdiene er sterkt forskjøvet.

b)

R-kode:

```
#b)
#Regner ut gjennomsnittlig størrelse og standardavvik av alle jordskjelvene
mean(earthquakes$magnitude)
sd(earthquakes$magnitude)
```

Output fra R/kjøreeksempel:

```
> #b)
> #Regner ut gjennomsnittlig størrelse og standardavvik av alle jordskjelvene
> mean(earthquakes$magnitude)
[1] 3.874334
> sd(earthquakes$magnitude)
[1] 0.6623718
```

Gjennomsnittet for alle jordskjelvene i datasettet er 3,874334.

Standardavviket for alle jordskjelvene i datasettet er 0,6623718.

c)

R-kode:

```
#c)
#Trekker et tilfeldig utvalg av 50 jordskjelv fra datasettet
magnitude = earthquakes$magnitude
sample_x = sample(magnitude, 50)
mean(sample_x)
```

Output fra R/kjøreeksempel:

```
> #c)
> #Trekker et tilfeldig utvalg av 50 jordskjelv fra datasettet
> magnitude = earthquakes$magnitude
> sample_x = sample(magnitude, 50)
> mean(sample_x)
[1] 3.782
```

Gjennomsnittet for et tilfeldig utvalg av 50 jordskjelv fra datasettet er 3,782. Sammenligner vi med resultatene i b), ser vi at gjennomsnittet for alle jordskjelvene er større enn det tilfeldige utvalget på 50 jordskjelvet. Dette kan bety at det tilfeldige utvalget består av jordskjelv med en styrke på rundt det samme som for hele gjennomsnittet av datasettet.

d)

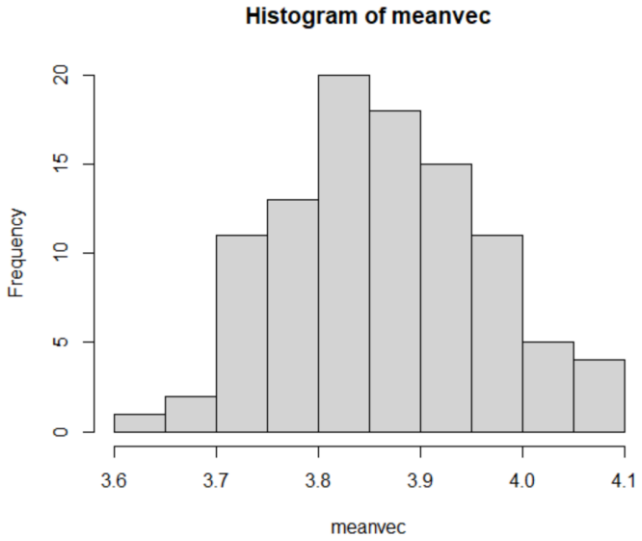
R-kode:

```
#d)
#Trekker et tilfeldig utvalg av 100 jordskjelv fra datasettet
meanvec <- rep(0, 100)
for(i in 1:100) {
  sample.now <- sample(magnitude, 50)
  meanvec[i] <- mean(sample.now)
}
#Plotter et histogram av gjennomsnittet til meanvec
hist(meanvec)
```

Output fra R/kjøreeksempel:

```
> #d)
> #Trekker et tilfeldig utvalg av 100 jordskjelv fra datasettet
> meanvec <- rep(0, 100)
> for(i in 1:100) {
+   sample.now <- sample(magnitude, 50)
+   meanvec[i] <- mean(sample.now)
+ }
> #Plotter et histogram av gjennomsnittet til meanvec
> hist(meanvec)
```

Plottet fra R er inkludert i Figur 3.



Figur 3: Histogram plott over meanvec

Fra histogrammet ser vi at fordelingen er tilnærmet symmetrisk og unimodal, og er nærmere normalfordelt. Ser vi på gjennomsnittet for fordelingen i oppgave b) ser vi at histogrammet i Figur 3 kan tyde på at det er et gjennomsnitt for 100 tilfeldige jordskjelv rundt denne verdien. Men den har tendenser til å være venstre-forskjøvet.

e)

R-kode:

```
#e)
#Teoretiske verdier, basert på verdiene fra oppgave b)
mean(earthquakes$magnitude)
sd(earthquakes$magnitude)/sqrt(713)

#Empiriske verdier, basert på de hundre simulerte gjennomsnittene i vektoren meanvec
mean(meanvec)
sd(meanvec)/sqrt(100)
```

Output fra R/kjøreeksempel:

```
> #e)
> #Teoretiske verdier, basert på verdiene fra oppgave b)
> mean(earthquakes$magnitude)
[1] 3.874334
> sd(earthquakes$magnitude)/sqrt(713)
[1] 0.02480602
>
> #Empiriske verdier, basert på de hundre simulerte gjennomsnittene i vektoren meanvec
> mean(meanvec)
[1] 3.87777
> sd(meanvec)/sqrt(100)
[1] 0.006789666
```

Her har vi funnet μ_x og σ_x for både teoretisk og empirisk verdi. Fra utregningen kan vi se at gjennomsnittet er veldig lik mellom de teoretiske verdiene og de empiriske verdiene. Ser vi på standardavviket har vi en ganske stor forskjell i verdier, noe som er et stort sprik til å være standardavvik.

f)

R-kode:

```
#f)
#For 10 jordskjelv
meanvec2 <- rep(0, 100)
for(i in 1:100) {
  sample.now <- sample(magnitude, 10)
  meanvec2[i] <- mean(sample.now)
}
mean(meanvec2)
sd(meanvec2)
#For 100 jordskjelv
meanvec3 <- rep(0, 100)
for(i in 1:100) {
  sample.now <- sample(magnitude, 100)
  meanvec3[i] <- mean(sample.now)
}
mean(meanvec3)
sd(meanvec3)
```

Output fra R/kjøreeksempel:

```
> #f)
> #For 10 jordskjelv
> meanvec2 <- rep(0, 100)
> for(i in 1:100) {
+   sample.now <- sample(magnitude, 10)
+   meanvec2[i] <- mean(sample.now)
+ }
> mean(meanvec2)
[1] 3.8735
> sd(meanvec2)
[1] 0.1991668
> #For 100 jordskjelv
> meanvec3 <- rep(0, 100)
> for(i in 1:100) {
+   sample.now <- sample(magnitude, 100)
+   meanvec3[i] <- mean(sample.now)
+ }
> mean(meanvec3)
[1] 3.87699
> sd(meanvec3)
[1] 0.06271331
```

Det gir for 10 jordskjelv:

$\mu_x = \mu = 3.8735$

$\sigma_x = 0.1991668$

Og for 100 jordskjelv:

$\mu_x = \mu = 3.87699$

$\sigma_x = 0.06271331$

g)

R-kode:

```
#g)
#Regner først ut for n = 10
pnorm(4, mean(meanvec2), sd(meanvec2), lower.tail = FALSE)
#Regner ut for n = 50
#For 50 jordskjelv
meanvec4 <- rep(0, 100)
for(i in 1:100) {
  sample.now <- sample(magnitude, 50)
  meanvec4[i] <- mean(sample.now)
}
pnorm(4, mean(meanvec4), sd(meanvec4), lower.tail = FALSE)
#Regner ut for n = 100
pnorm(4, mean(meanvec3), sd(meanvec3), lower.tail = FALSE)
```

Output fra R/kjøreeksempel:

```
> #g)
> #Regner først ut for n = 10
> pnorm(4, mean(meanvec2), sd(meanvec2), lower.tail = FALSE)
[1] 0.2626666
> #Regner ut for n = 50
> #For 50 jordskjelv
> meanvec4 <- rep(0, 100)
> for(i in 1:100) {
+   sample.now <- sample(magnitude, 50)
+   meanvec4[i] <- mean(sample.now)
+ }
> pnorm(4, mean(meanvec4), sd(meanvec4), lower.tail = FALSE)
[1] 0.07834663
> #Regner ut for n = 100
> pnorm(4, mean(meanvec3), sd(meanvec3), lower.tail = FALSE)
[1] 0.02491237
```

Sannsynligheten for å trekke et tilfeldig utvalg med gjennomsnitt større enn 4.0 er:

For n = 10 = 0.2626666 = 26.2%

For n = 50 = 0.07834663 = 7.8%

For n = 100 = 0.02491237 = 2.4%

h)

Bias av en observator er tendensen til å se hva vi forventer å se, eller det vi ønsker eller vil se. Dette betyr at et eksperiment utføres med fordommer, og for eksempel man har noe man tror kommer til å se i en studie.

Varians av en observator er at flere observatorer ser flere typer forskjellige data utifra samme populasjon. Det kan være å ta prøver som gir utfall som er varierende, for eksempel et stoff som gir variasjon i resultatet av flere observatorer av samme stoff.

Oppgave 2

a)

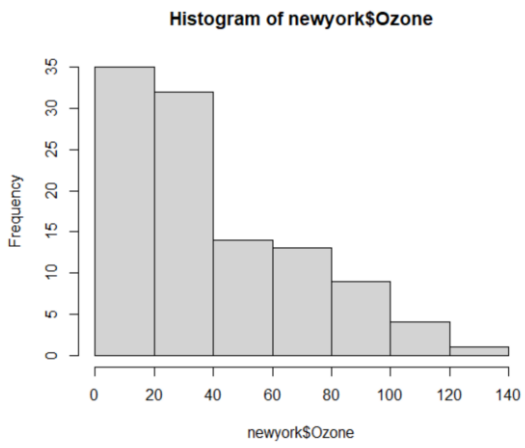
R-kode:

```
#Oppgave 2
#a)
data <- "https://www.uio.no/studier/emner/matnat/math/STK1000/data/obligdata/oblig2/ozone.txt"
newyork <- read.table(data,header=TRUE)
newyork$Ozone
hist(newyork$Ozone)
#Sjekker etter eventuelle uteliggere med tommelfingerregelen
summary(newyork$Ozone)
#Q1 - 1.5 X IQR =
18 - 1.5*(41.5)
#q3 - 1.5 X IQR =
59.50 + 1.5*(41.5)
#Sjekker om ozone er normalfordelt
qqnorm(newyork$Ozone)
qqline(newyork$Ozone)
```

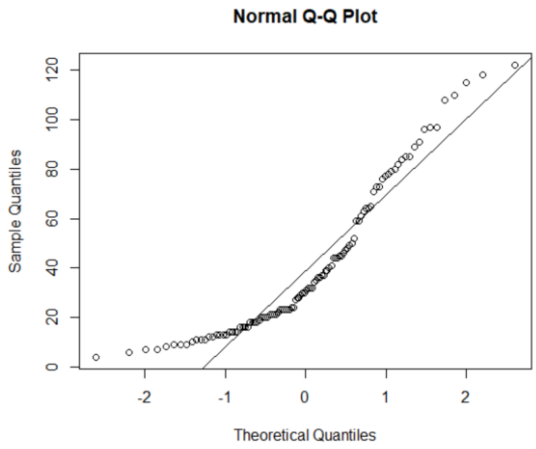
Output fra R/kjøreeksempel:

```
> #Oppgave 2
> #a)
> data <- "https://www.uio.no/studier/emner/matnat/math/STK1000/data/obligdata/oblig2/ozone.txt"
> newyork <- read.table(data,header=TRUE)
> newyork$Ozone
  [1]  41  36  12  18  23  19   8  16  11  14  18  14  34   6  30  11  11   4  32  23  45 115  37  29  71
 [26]  39  23  21  37  20  12  13  49  32  64  40  77  97  97  85  10  27   7  48  35  61  79  63  16  80
 [51] 108  20  52  82  50  64  59  39   9  16 122  89 110  44  28  65  22  59  23  31  44  21   9  45  73
 [76]  76 118  84  85  96  78  73  91  47  32  20  23  21  24  44  21  28   9  13  46  18  13  24  16  13
[101]  23  36   7  14  30  14  18  20
> hist(newyork$Ozone)
> #Sjekker etter eventuelle uteliggere med tommelfingerregelen
> summary(newyork$Ozone)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   4.00   18.00   30.50   40.45   59.50  122.00
> #Q1 - 1.5 X IQR =
> 18 - 1.5*(41.5)
[1] -44.25
> #q3 - 1.5 X IQR =
> 59.50 + 1.5*(41.5)
[1] 121.75
> #Sjekker om ozone er normalfordelt
> qqnorm(newyork$Ozone)
> qqline(newyork$Ozone)
```

Plottet fra R er inkludert i Figur 4 og 5.



Figur 4: Histogram over Ozone



Figur 5: QQ plott over Ozone

Ozonvariabelen er en ikke-symmetrisk uten unimodal topp og er en venstreforskjøvet fordeling, som vi ser i Figur 4. Dette kan vi også se i Figur 5, ettersom det er flere verdier som ligger under den rette linjen. Verdiene det er flest av har et gjennomsnittlig ozon-nivå på mellom 0 og 40. Ser vi på ozonvariabelens datasett, forekommer disse verdiene rundt de første månedene. Dette indikerer at det var lavere ozon-nivå rundt mai måneden og mot sommeren. Logisk sett er dette som regel vanlig, der sommer månedene er som regel varmest på grunn av lavere ozon-nivå, slik at det forekommer mer solstråling. Verdiene stiger mot september, og det er igjen logisk ettersom det blir kaldere temperaturer nettopp fordi mindre solstråling slipper til.

Fra tommelfingerregelen kan vi identifisere at verdier over 121,75 er uteliggere. Fra Histogrammet kan vi se at det finnes en uteligger, som vi også ser i QQ plottet (Figur 5) helt øverst i høyre hjørne. Vi kan også se fra datasettet at denne verdien ligger på 122 og er dermed en uteligger.

Fra QQ plottet (Figur 5) kan vi konkludere med at ozonvariabelen ikke er normalfordelt, ettersom verdiene ikke ligger inntil den lineære linjen i Figur 5.

b)

Antagelser som ligger i grunn for å bruke t-test:

- Vi ønsker å vite om populasjonene vi tester er relatert til hverandre og kan derfor sammenlignes til bruk i for eksempel behandling
- Vi sjekker om gjennomsnittet mellom populasjonene er like eller ikke like, slik at vi vet at egenskapene er like for testing

c)

R-kode:

```
#c)
#Kjører t-test på variabelen fra deloppgave a)
ozonelevel <- c(30, 30)
t.test(newyork$Ozone, ozonelevel)
```

Output fra R/kjøreeksempel:

```
> ozonelevel <- c(30, 30)
> t.test(newyork$Ozone, ozonelevel)

        Welch Two Sample t-test

data:  newyork$Ozone and ozonelevel
t = 3.6395, df = 107, p-value = 0.0004224
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  4.759688 16.147719
sample estimates:
mean of x mean of y
  40.4537   30.0000
```

Nullhypotese: det er ingen relasjon mellom forventet ozonnivå på 30ppb og forventet ozonnivå i perioden mai – september på syttitallet.

Alternativ hypotese: det er en relasjon mellom forventet ozonnivå på 30ppb og forventet ozonnivå i perioden mai – september på syttitallet.

Som vi ser har vi en lav p-verdi som tilsier at vi kan forkaste nullhypotesen. Vi kan dermed konkludere med at forventet ozonnivå var mer enn 30ppb på syttitallet i perioden mai – september i New York.

d)

R-kode:

```
#d)
#90% konfidensintervall
t.test(newyork$Ozone, conf.level = 0.9)
#95% konfidensintervall
t.test(newyork$Ozone, conf.level = 0.95)
#99% konfidensintervall
t.test(newyork$Ozone, conf.level = 0.99)
```

Output fra R/kjøreeksempel:

```
> #d)
> #90% konfidensintervall
> t.test(newyork$Ozone, conf.level = 0.9)

        One Sample t-test

data:  newyork$Ozone
t = 14.084, df = 107, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
90 percent confidence interval:
 35.68791 45.21949
sample estimates:
mean of x
  40.4537

> #95% konfidensintervall
> t.test(newyork$Ozone, conf.level = 0.95)

        One Sample t-test

data:  newyork$Ozone
t = 14.084, df = 107, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 34.75969 46.14772
sample estimates:
mean of x
  40.4537

> #99% konfidensintervall
> t.test(newyork$Ozone, conf.level = 0.99)

        One Sample t-test

data:  newyork$Ozone
t = 14.084, df = 107, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
99 percent confidence interval:
 32.9209 47.9865
sample estimates:
mean of x
  40.4537
```

Konfidensintervall	Min – verdi	Max – verdi
90%	35.68791	45.21949
95%	34.75969	46.14772
99%	32.9209	47.9865

Minimumsverdiene minker når intervallene går mot sikrere antydninger, og dermed vil maksverdiene øke og skape en større bredde i konfidensintervallet. 99% vil være veldig bred, og tilsier at det er flere verdier vi forventer å se i større konfidensintervall

e)

R-kode:

```
#e)
oz.juli.august <- newyork[newyork$Month %in% c(7,8),"Ozone"]
oz.mai.juni.sept <- newyork[newyork$Month %in% c(5,6,9),"Ozone"]
t.test(oz.juli.august, oz.mai.juni.sept)
```

Output fra R/kjøreeksempel:

```
> #e)
> oz.juli.august <- newyork[newyork$Month %in% c(7,8),"Ozone"]
> oz.mai.juni.sept <- newyork[newyork$Month %in% c(5,6,9),"Ozone"]
> t.test(oz.juli.august, oz.mai.juni.sept)

        Welch Two Sample t-test

data:  oz.juli.august and oz.mai.juni.sept
t = 4.9526, df = 80.631, p-value = 3.957e-06
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 16.06035 37.63270
sample estimates:
mean of x mean of y
 55.61702  28.77049
```

Nullhypotese: det er ingen relasjon mellom mai, juni, september og juli, august.

Alternativ hypotese: det er en relasjon mellom månedene i mengden ozon ettersom begge intervaller av måneder er rundt sommeren.

Testen fra output viser en veldig lav p-verdi på $3.9 * 10^{-6}$, som viser at nullhypotesen kan forkastes. Dermed er det en relasjon mellom månedene, som er veldig sannsynlig ettersom det er sommermåneder i begge intervaller av måneder.

Oppgave 3

a)

R-kode:

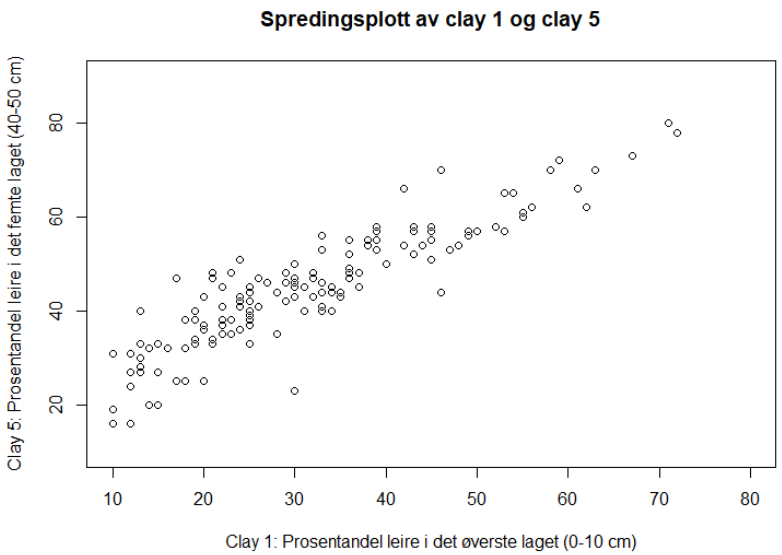
```
#Oppgave 3
data <- "https://www.uio.no/studier/emner/matnat/math/STK1000/data/obligdata/oblig2/cameroonclay.txt"
cameroon <- read.table(data,header=TRUE)

#a)
#La til passende navn på aksene og tittel
clay1 <- cameroon$clay1
clay5 <- cameroon$clay5
plot(clay1, clay5, xlim = c(10, 80), ylim = c(10, 90),
     xlab = "Clay 1: Prosentandel leire i det øverste laget (0-10 cm)",
     ylab = "Clay 5: Prosentandel leire i det femte laget (40-50 cm)",
     main = "Spredingsplott av clay 1 og clay 5")
#Figuren viser en lineær spredning mellom de to variabelene, legger derfor til en test av korrelasjon
cor(clay1, clay5)
```

Output fra R/kjøreeksempel:

```
> #Oppgave 3
> data <- "https://www.uio.no/studier/emner/matnat/math/STK1000/data/obligdata/oblig2/cameroonclay.txt"
> cameroon <- read.table(data,header=TRUE)
>
> #a)
> #La til passende navn på aksene og tittel
> clay1 <- cameroon$clay1
> clay5 <- cameroon$clay5
> plot(clay1, clay5, xlim = c(10, 80), ylim = c(10, 90),
+      xlab = "Clay 1: Prosentandel leire i det øverste laget (0-10 cm)",
+      ylab = "Clay 5: Prosentandel leire i det femte laget (40-50 cm)",
+      main = "Spredingsplott av clay 1 og clay 5")
> #Figuren viser en lineær spredning mellom de to variabelene, legger derfor til en test av korrelasjon
> cor(clay1, clay5)
[1] 0.8977721
```

Plottet fra R er inkludert i Figur 6.



Figur 6: Spredningsplott over clay 1 og clay 5

Som vi ser fra plottet i Figur 6 ser vi en lineær vekst i spredningen. Sjekket også korrelasjonen mellom variablene og dette impliserer en sterk korrelasjon. Dermed kan man med god grunn anta at det er et lineær forhold mellom variablene.

b)

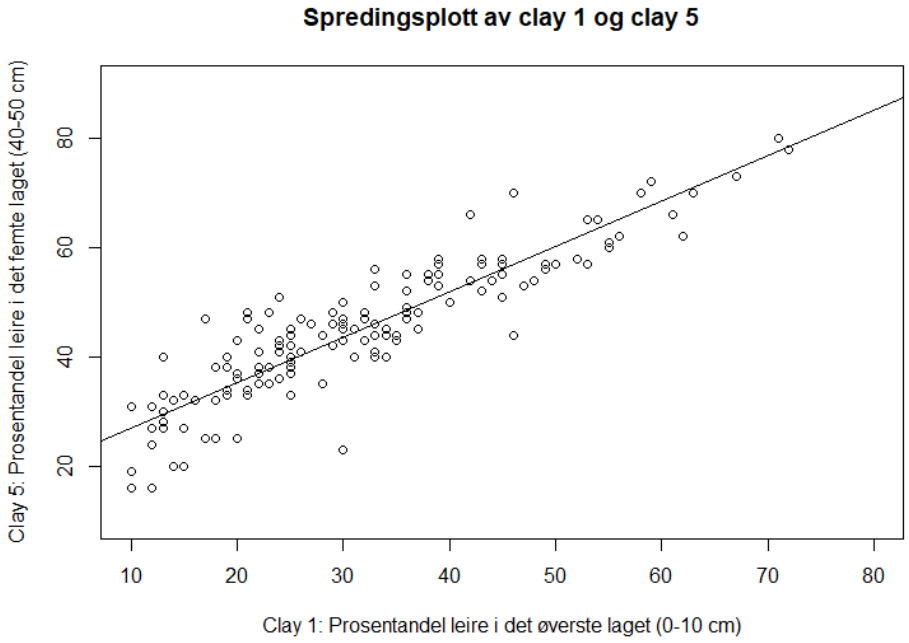
R-kode:

```
#b)
#Lager en regresjonsmodell med gitt R-kode
fit <- lm(clay5 ~ clay1)
abline(fit)
```

Output fra R/kjøreeksempel:

```
> #b)
> #Lager en regresjonsmodell med gitt R-kode
> fit <- lm(clay5 ~ clay1)
> abline(fit)
```

Plottet fra R er inkludert i Figur 7.



Figur 7: Spredningsplott fra a) med regresjonslinje

Som vi ser fra plottet i Figur 7 er det en lineær vekst, som tilsier en sterk korrelasjon.

c)

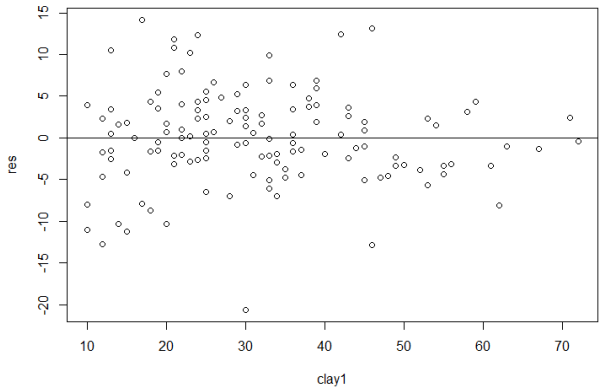
R-kode:

```
#c)
#Bruker gitt R-kode for å inspisere residualene
#Residualplott
res <- residuals(fit)
plot(clay1, res)
abline(h = 0)
#QQ plott
qqnorm(res)
qqline(res)
```

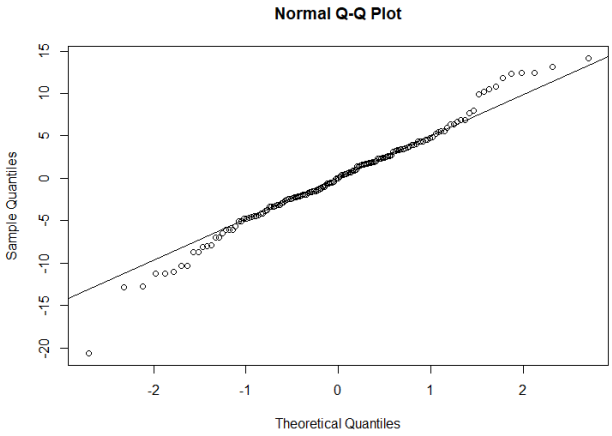
Output fra R/kjøreeksempel:

```
#c)
#Bruker gitt R-kode for å inspisere residualene
#Residualplott
res <- residuals(fit)
plot(clay1, res)
abline(h = 0)
#QQ plott
qqnorm(res)
qqline(res)
```

Plottet fra R er inkludert i Figur 8 og Figur 9.



Figur 8: Residualplott av clay 1 og clay 5



Figur 9: QQ plott over residualene

Antagelser for en lineær regresjonslinje er som følger:

- Residualene er uavhengige, det vil si at det er ønskelig at residualet er lik 0
    - Fra Figur 8 kan vi se at punktene virker å være plassert tilfeldig rundt linjen, og dermed kan vi si at residualet er uavhengig ettersom residualet sannsynligvis er i nærheten av 0
- Normalfordelt om regresjonslinja, med samme varians for alle verdier av x-variabelen
    - Hvis residualet er i nærheten av 0 kan vi også si at regresjonslinjen er normalfordelt med samme varians for alle verdier av x

d)

R-kode:

```
#d)
#Finner stigningstall og skjæringspunkt for modellen
summary(fit)
```

Output fra R/kjøreeksempel:

```
> #d)
> #Finner skjæringspunkt for modellen
> summary(fit)

Call:
lm(formula = clay5 ~ clay1)

Residuals:
     Min       1Q   Median       3Q      Max
-20.6258  -3.1907   0.0055   3.3875  14.1500

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 18.75856    1.15561   16.23   <2e-16 ***
clay1        0.82891    0.03377   24.54   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.687 on 145 degrees of freedom
Multiple R-squared:  0.806,     Adjusted R-squared:  0.8047
F-statistic: 602.4 on 1 and 145 DF,  p-value: < 2.2e-16
```

Det gir den lineære modellen:

y = 0.82891*x + 18.75856

Den forventede prosentandelen av leire i lag fem øker med 0.82891 prosent for hver gang leire i lag en øker med 1%.

e)

Vi ser at p-verdien er $2.2 * 10^{-16}$ som er en liten p-verdi. Dette indikerer at vi har bevis mot null-hypotesen, ettersom lave p-verdier indikerer dette. Dermed kan dette bety at vi kan avvise null-hypotesen, ettersom verdien er så liten og indikerer et forhold mellom populasjonene; og det motstrider null-hypotesen.

f)

R-kode:

```
#f)
#Finner 95% konfidensintervall for stignignstallet i modellen
b1 <- summary(fit)$coefficients[2, 1]
se.b1 <- summary(fit)$coefficients[2, 2]
df <- fit$df.residual
lower <- b1 + qt(0.025, df) * se.b1
upper <- b1 + qt(0.975, df) * se.b1
se.b1
lower
upper
```

Output fra R/kjøreeksempel:

```
> #f)
> #Finner 95% konfidensintervall for stignignstallet i modellen
> b1 <- summary(fit)$coefficients[2, 1]
> se.b1 <- summary(fit)$coefficients[2, 2]
> df <- fit$df.residual
> lower <- b1 + qt(0.025, df) * se.b1
> upper <- b1 + qt(0.975, df) * se.b1
> se.b1
[1] 0.03377248
> lower
[1] 0.7621583
> upper
[1] 0.8956582
```

95%-konfidensintervall for stigningstallet i modellen kan vi se i kjøreeksemplet i R ovenfor.

g)

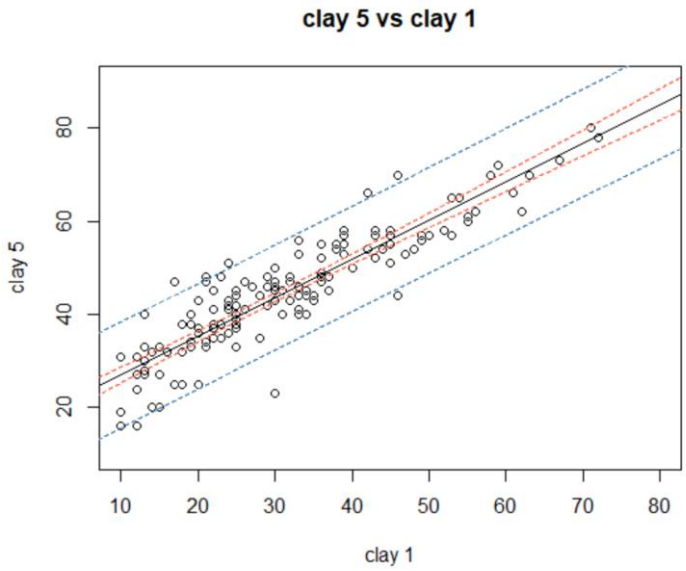
R-kode:

```
#g)
#Plotter modellens prediksjonsintervall og konfidensintervall for forventet respons
plot(clay1, clay5, xlim = c(10, 80), ylim = c(10, 90),
     xlab = "clay 1", ylab = "clay 5", main = "clay 5 vs clay 1")
abline(fit)
xval <- seq(0, 100, by = 0.01)
new <- data.frame(clay1 = xval)
pred.int <- predict(fit, newdata = new, interval = "prediction")
mean.int <- predict(fit, newdata = new, interval = "confidence")
matlines(xval, cbind(pred.int[, 2], pred.int[, 3]), lty = 2,
         col = "steelblue")
matlines(xval, cbind(mean.int[, 2], mean.int[, 3]), lty = 2,
         col = "tomato")
```

Output fra R/kjøreeksempel:

```
#g)
#Plotter modellens prediksjonsintervall og konfidensintervall for forventet respons
plot(clay1, clay5, xlim = c(10, 80), ylim = c(10, 90),
     xlab = "clay 1", ylab = "clay 5", main = "clay 5 vs clay 1")
abline(fit)
xval <- seq(0, 100, by = 0.01)
new <- data.frame(clay1 = xval)
pred.int <- predict(fit, newdata = new, interval = "prediction")
mean.int <- predict(fit, newdata = new, interval = "confidence")
matlines(xval, cbind(pred.int[, 2], pred.int[, 3]), lty = 2,
         col = "steelblue")
matlines(xval, cbind(mean.int[, 2], mean.int[, 3]), lty = 2,
         col = "tomato")
```

Plottet fra R er inkludert i Figur 10.



Figur 10: Prediksjonsintervall og konfidensintervall for forventet respons

Prediksjonsintervallet er i blått og konfidensintervallet er i rødt. Prediksjonsintervallet er bredere enn det andre fordi det er verdier vi predikerer å se, altså de verdiene vi tror vi kan se. Det er som regel flere verdier. Konfidensintervallet er smalere fordi det er verdier vi forventer å se eller de verdiene vi ønsker å se.

h)

Prediksjonsintervallet som vi ser i Figur 10, ser vi at for x = 60 ligger det femte laget på rundt 57 til 79. Konfidensintervallet som vi ser i Figur 10, ser vi at for x = 60 ligger det femte laget på rundt 60 til 65. Dermed antar jeg at innholdet i det femte laget ligger på rundt 65.