Variansanalyse

Plan for forelesningen

- Regresjon med dummyvariabler
- Sammenligning av gruppegjennomsnitt
- Toveis variansanalyse

PSY2014, 29. mars 2022

Målenivåer

Ting vi måler kan deles inn i fire ulike kategorier:

- **Nominal**: Kategorier som ikke kan rangeres etter størrelse. F.eks. farge, kjønn, land.
- **Ordinal**: Kategorier som kan rangeres etter størrelse, men avstanden mellom dem er ikke meningsfull. F.eks. "litt fornøyd", "ganske fornøyd", "veldig fornøyd".
- **Intervall**: Avstanden mellom ulike verdier er meningsfull, men nullverdien betyr ikke at det ikke finnes noe av det vi måler. F.eks. temperatur i Celsius.
- **Forholdstall/ratio**: Mengdeskala som også har et naturlig nullpunkt, f.eks. vekt i kg.

De tre siste er **kvantitative** variabler. Nominale variabler er **kvalitative**.

Tre barn ble utsatt for tre ulike intervensjoner, som skulle øke prestasjonen på en lesetest.

```
lesedata ← data.frame(
  Gruppe = rep(c("a", "b", "c"), each = 3),
  Prestasjon = c(3, 4, 5, 3, 9, 6, 2, 1, 3)
)
head(lesedata, 5)
```

```
##   Gruppe Prestasjon
## 1      a          3
## 2      a          4
## 3      a          5
## 4      b          3
## 5      b          9
```

- Hvilket målenivå har **Gruppe** og **Prestasjon**?
- Nominalt og intervall.

Tabulering av dataene

- Vi kan oppsummere noen tall for hver gruppe

```
library(tidyverse)
lesedata %>%
  group_by(Gruppe) %>%
  summarise(
    Antall = n(),
    Gjennomsnitt = mean(Prestasjon),
    Standardavvik = sd(Prestasjon),
    .groups = "drop"
  )
```

```
## # A tibble: 3 × 4
##   Gruppe Antall Gjennomsnitt Standardavvik
##   <chr>   <int>        <dbl>         <dbl>
## 1 a           3            4             1
## 2 b           3            6             3
## 3 c           3            2             1
```

Dummyvariabler i R

Som dere også har sett tidligere, en dummyvariabel er en variabel som kun tar verdiene 0 og 1. For å representere tre ulike grupper trenger vi to dummyvariabler.

R lager dummyvariabler automatisk for oss.

```
lm(Prestasjon ~ Gruppe, data = lesedata)
```

```
##
## Call:
## lm(formula = Prestasjon ~ Gruppe, data = lesedata)
##
## Coefficients:
## (Intercept)      Gruppeb      Gruppec
##           4            2           -2
```

Gruppe A er referansenivå, mens `Gruppeb` og `Gruppec` representerer forskjellen mellom disse gruppene og gruppe A.

```
(lesemodell ← lm(Prestasjon ~ Gruppe, data = lesedata))
```

```
##
## Call:
## lm(formula = Prestasjon ~ Gruppe, data = lesedata)
##
## Coefficients:
## (Intercept)      Gruppeb      Gruppec
##           4            2           -2
```

- La z_1 og z_2 være dummyvariablene. Disse tilsvarer `Gruppeb` og `Gruppec` i R-utskriften.

Gruppe	z_1 (**`Gruppeb`**)	z_2 (**`Gruppec`**)
a	0	0
b	1	0
c	0	1

En ting å være obs på!

- R skjønner ikke uten videre at tall betyr grupper!

```
head(lesedata2, 5)
```

```
##   Gruppe Prestasjon
## 1      1          3
## 2      1          4
## 3      1          5
## 4      2          3
## 5      2          9
```

En ting å være obs på!

Hvordan tolker vi her koeffisienten for **Gruppe**?

```
(lm(Prestasjon ~ Gruppe, data = lesedata2))
```

```
##
## Call:
## lm(formula = Prestasjon ~ Gruppe, data = lesedata2)
##
## Coefficients:
## (Intercept)       Gruppe
##           6           -1
```

R tror `Gruppe` er kvantitativ:

```
class(lesedata2$Gruppe)
```

```
## [1] "numeric"
```

En ting å være obs på!

Vi må passe på at gruppevariabelen er en faktor:

```
lesedata2$Gruppe ← factor(lesedata2$Gruppe)
```

Konvertering til `factor` ga oss dummyvariabler igjen

```
(lm(Prestasjon ~ Gruppe, data = lesedata2))
```

```
##
## Call:
## lm(formula = Prestasjon ~ Gruppe, data = lesedata2)
##
## Coefficients:
## (Intercept)      Gruppe2      Gruppe3
##           4            2           -2
```

Regresjonsmodellen med to grupper er

$$y = \alpha + \beta_1 z_1 + \beta_2 z_2 + \epsilon$$

where α er intercept/skjæringspunkt, β_1 og β_2 er effekten av de to dummy-variablene, og ϵ er residualet (støyledd).

- Gjennomsnitt i gruppe a er $\mu_a = \alpha + \beta_1 0 + \beta_2 0 + 0 = \alpha$.
- Gjennomsnitt i gruppe b er $\mu_b = \alpha + \beta_1 1 + \beta_2 0 + 0 = \alpha + \beta_1$.
- Gjennomsnitt i gruppe c er $\mu_c = \alpha + \beta_1 0 + \beta_2 1 + 0 = \alpha + \beta_2$.

Dette gir oss en tolkning av regresjonskoeffisientene som forskjellen mellom grupper:

$$\beta_1 = \mu_b - \mu_a$$

og

$$\beta_2 = \mu_c - \mu_a$$

```
coef(lesemodell)
```

```
## (Intercept)     Gruppeb     Gruppec
##           4           2          -2
```

Estimatene er altså: $\hat{\alpha} = 4$, $\hat{\beta}_1 = 2$ og $\hat{\beta}_2 = -2$.

- Forskjellen mellom gruppe b og gruppe a er estimert til 2.
- Forskjellen mellom gruppe c og gruppe a er estimert til -2.

Er forskjellen signifikant?

Nullhypotesen er at det ikke er forskjell mellom gruppene,

$$H_0 : \mu_a = \mu_b = \mu_c$$

H_0 er det samme som $\beta_1 = \beta_2 = 0$. Alternativhypotesen er at det er forskjell mellom minst to av gruppene.

Vi kan bruke `anova`-funksjonen i R til dette:

```
anova(lesemodell)
```

```
## Analysis of Variance Table
##
## Response: Prestasjon
##           Df Sum Sq Mean Sq F value Pr(>F)
## Gruppe     2     24 12.0000  3.2727 0.1094
## Residuals  6     22  3.6667
```

Er forskjellen signifikant?

Alternativt kan vi bruke `aov`-funksjonen til å gjøre variansanalysen direkte

```
summary(aov(Prestasjon ~ Gruppe, data = lesedata))
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## Gruppe       2     24  12.000   3.273  0.109
## Residuals    6     22   3.667
```

Testen vi har gjort sjekker om det er forskjell mellom noen av gruppene, men ikke mellom hvilke. Vi kan sjekke dette også.

La $\bar{y}_g$ være gjennomsnittet i gruppe g, n_g være antall observasjoner i gruppe g, og s^2 være Residual Mean Square. Da er et konfidensintervall for forskjellen $\mu_{g1} - \mu_{g2}$ mellom to grupper $g1$ og $g2$ gitt ved

$$(\bar{y}_{g1} - \bar{y}_{g2}) \pm ts\sqrt{1/n_{g1} + 1/n_{g2}}$$

Kan vi finne et konfidensintervall for forskjellen mellom gruppe c og gruppe b, $\mu_c - \mu_b$?

```
lesedata %>%
  group_by(Gruppe) %>%
  summarise(Gjennomsnitt = mean(Prestasjon), Antall = n())
```

```
## # A tibble: 3 × 3
##   Gruppe Gjennomsnitt Antall
##   <chr>         <dbl>  <int>
## 1 a                 4      3
## 2 b                 6      3
## 3 c                 2      3
```

Vi har $\bar{y}_b = 6$ $\bar{y}_c = 2$, $n_b = 3$ og $n_c = 3$.

```
summary(aov(Prestasjon ~ Gruppe, data = lesedata))
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## Gruppe       2     24  12.000   3.273  0.109
## Residuals    6     22   3.667
```

Videre har vi $s^2 = 3.667$.

Ved å plugge inn i formelen får vi derfor konfidensintervallet for $\mu_c - \mu_b$

$$(2 - 6) \pm t\sqrt{3.667}\sqrt{1/3 + 1/3} = -4 \pm 1.56t.$$

t-en kommer fra t-fordelingen med antall frihetsgrader gitt ved differansen mellom antall observasjoner og antall grupper, $df = 9 - 3$. Vi velger 95 % signifikansnivå, og får derfor $\alpha = 0.025$, siden vi skal ha 2,5 % i hver hale av fordelingen.

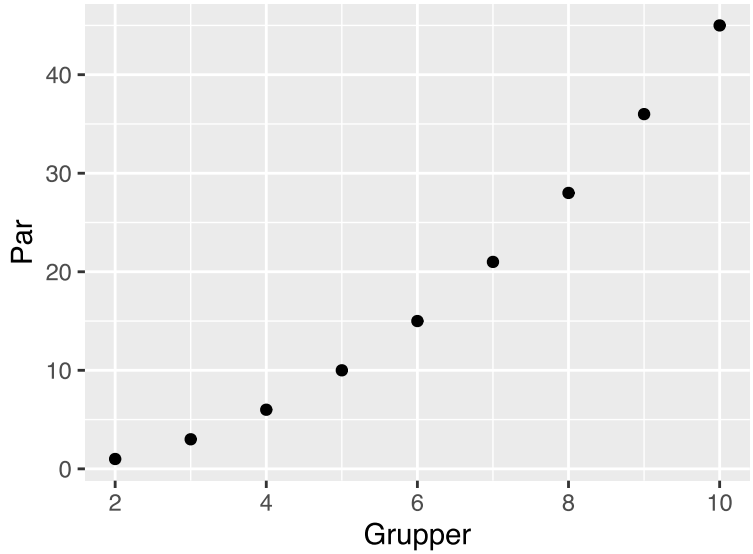
```
c(qt(p = .025, df = 6), qt(p = .975, df = 6))
```

```
## [1] -2.446912  2.446912
```

Intervallet er derfor

$$-4 \pm 2.45 \times 1.56 \text{ eller } (-7.8, -0.18)$$

Når antallet grupper vokser, vokser antallet parvise sammenligninger mye raskere.



Hvert 95 % konfidensintervall har 5 % sjanse for ikke å inneholde den sanne verdien. Hvis vi f.eks. sammenligner 7 grupper, har vi 21 konfidensintervall, og kan forvente at $21 \times 0.05 = 1.05$ av disse er feil.

Bonferroni-korreksjon

Bonferroni-korreksjon endrer bredden på konfidensintervallene, slik at vi kontrollerer den totale sannsynligheten for å gjøre feil.

Metoden er enkel å bruke. For eksemplet tidligere, dersom vi sammenligner alle tre par, $\mu_a - \mu_b$, $\mu_a - \mu_c$ og $\mu_b - \mu_c$, så deler vi vi på 3. Hvert enkelt konfidensintervall har derfor t-verdi basert på $0.025/3 = 0.00833$.

```
qt(p = .025 / 3, df = 6)
```

```
## [1] -3.287455
```

Intervallet vi regnet ut for $\mu_c - \mu_b$ blir da

$$-4 \pm 3.29 \times 1.56 \text{ eller } (-9.13, 1.13)$$

NB! Dette er bare nødvendig hvis vi faktisk ser på alle parvise differanser.

Tukeys metode for multippel korreksjon

Bonferronis metode er konservativ. Den garanterer at den totale feil er mindre eller lik signifikansnivået. Tukeys metode er litt mer liberal, ved at den sikrer at den forventede sannsynligheten for feil er lik signifikansnivået. Det finnes også en enkel funksjon for dette i R:

```
anova_modell ← aov(Prestasjon ~ Gruppe, data = lesedata)
TukeyHSD(anova_modell)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Prestasjon ~ Gruppe, data = lesedata)
##
## $Gruppe
##     diff       lwr       upr     p adj
## b-a    2 -2.797161 6.7971609 0.4554965
## c-a   -2 -6.797161 2.7971609 0.4554965
## c-b   -4 -8.797161 0.7971609 0.0947643
```

Tukeys metode

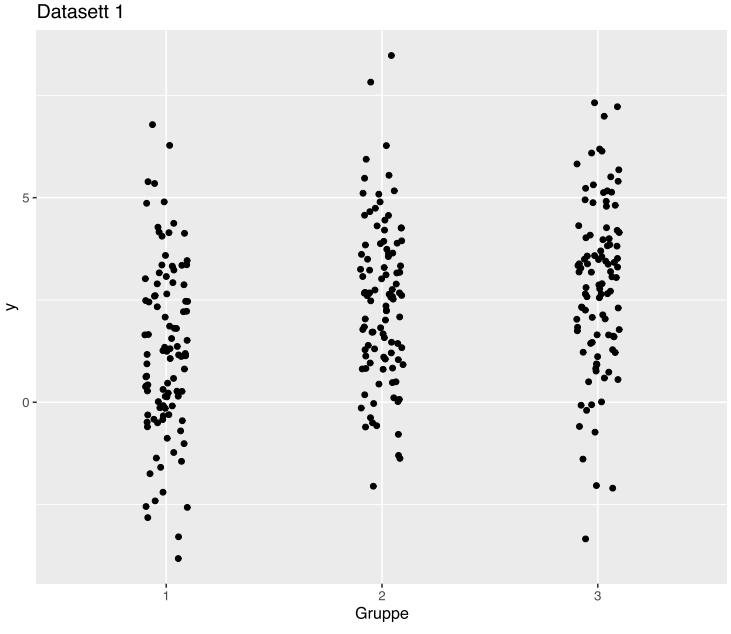
```
plot(TukeyHSD(anova_modell))
```

For g grupper kan vi teste nullhypotesen

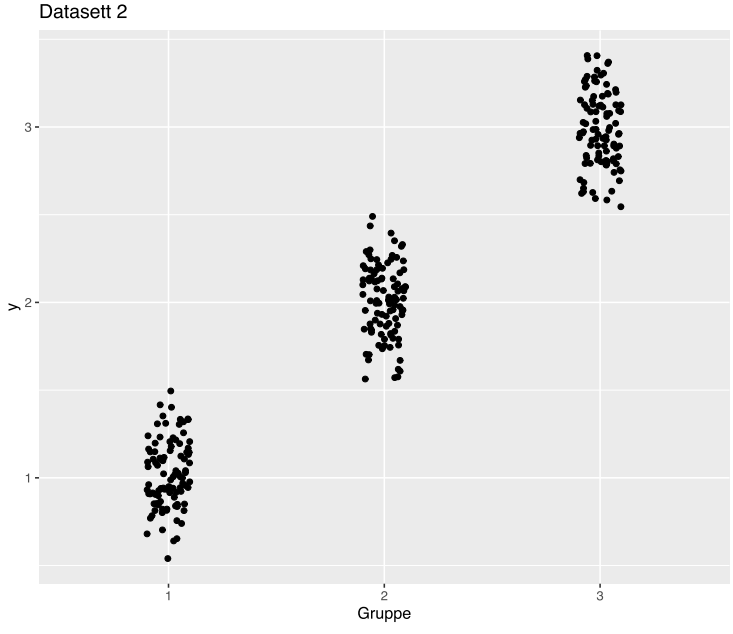
$$H_0 : \mu_1 = \mu_2 = \cdots = \mu_g$$

mot alternativhypotesen at det er forskjell mellom minst ett par, uten å bruke regresjonsanalyse. Dette kan gi et bedre innblikk i hva vi egentlig tester.

Eksempel



- Variasjon innad i gruppene er større enn variasjonen mellom gruppene.



- Variasjonen innad i gruppene er mindre enn variasjonen mellom gruppene.

- Variasjon innad i gruppene (**within-groups variance**) beregnes utifra hvor mye observasjonene i én gruppe varierer fra gjennomsnittet $\bar{y}_g$.

- Variasjonen mellom gruppene (**between-groups variance**) beregnes utifra hvor mye gjennomsnittene i hver gruppe varierer fra det total gjennomsnittet $\bar{y}$.

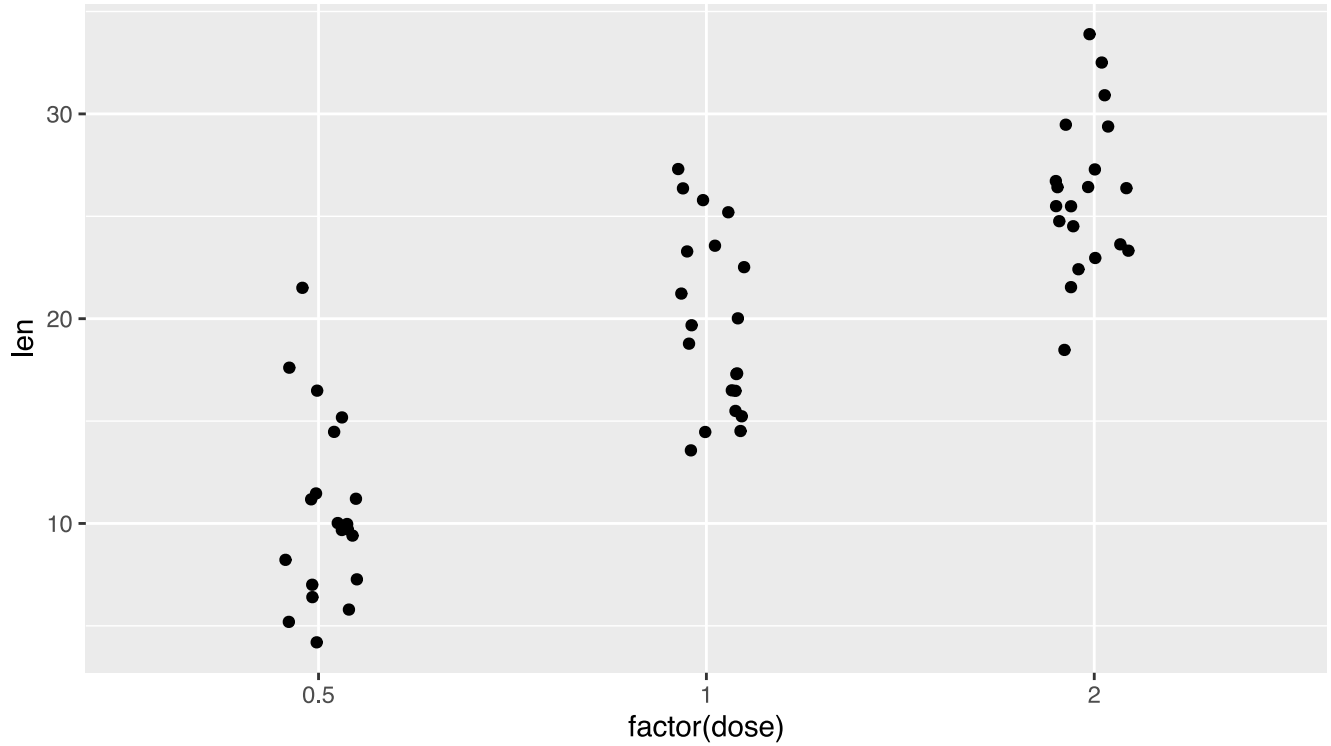
- Teststatistikken er

$$F = \frac{\text{Varians mellom}}{\text{Varians innad}}$$

Frihetsgradene er df_1 gitt ved antall grupper minus én, og df_2 gitt ved antall observasjoner minus antall grupper.

Tolkning av F-statistikken: Stor **Varians mellom** i forhold til **Varians innad** gjør oss sikrere på at det er reell forskjell mellom gruppene.

La oss ta et litt større eksempel. `ToothGrowth` er et datasett som følger med R, med 60 hamstere som har fått 0,5, 1 eller 2 milligram C-vitamin, og den avhengige variabelen er lengden på cellene som sørger for tannvekst.



```
ToothGrowth$dose ← factor(ToothGrowth$dose)
summary(aov(len ~ dose, data = ToothGrowth))
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## dose          2   2426    1213   67.42 9.53e-16 ***
## Residuals    57   1026      18
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Kolonnen `Mean Sq` viser varians, raden `Residuals` tilsvarer **within-group** og `group` tilsvarer **between-group**.

Teststatistikken blir derfor

$$F = \frac{1213}{18} = 67.389$$

Med 60 observasjoner og 3 grupper har vi $df_1 = 3 - 1 = 2$ og $df_2 = 60 - 3 = 57$. P-verdien blir

```
1 - pf(1213/18, df1 = 2, df2 = 57)
```

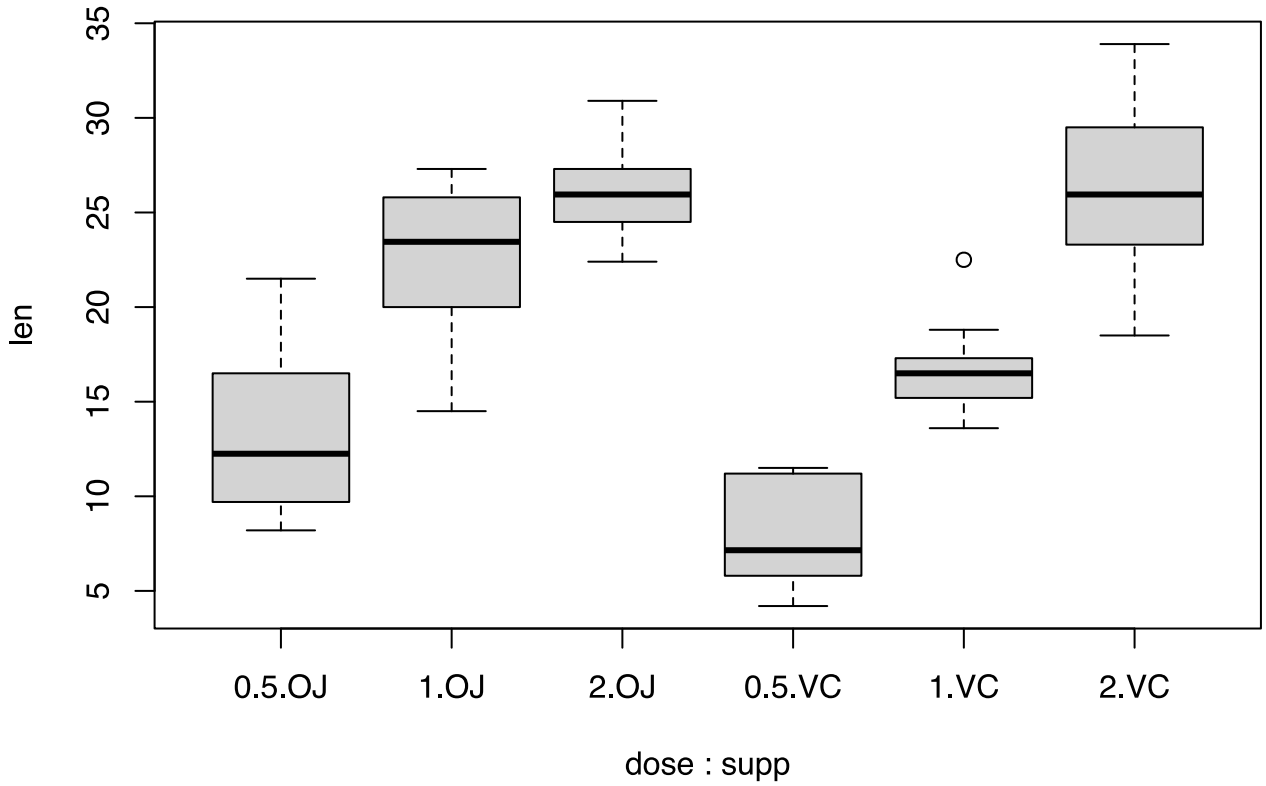
```
## [1] 8.881784e-16
```

Toveis variansanalyse

I en toveis variansanalyse har vi to uavhengige variabler. Vi kan også utføre treveis, fireveis, osv, men da blir det fort vanskelig å skjønne hva vi holder på med.

I `ToothGrowth`-datasettet har vi også en variabel `supp` som viser om C-vitaminen ble levert via appelsinjus eller askorbinsyre.

```
boxplot(len ~ dose + supp, data = ToothGrowth)
```



Vi har nå to nullhypoteser, én for hver faktor. For `dose` er nullhypotesen at det ikke er forskjell i lengde for ulike doser, og for `supp` er nullhypotesen at det ikke er forskjell i lengden mellom ulike måter å supplere C-vitamin på.

Vi får derfor to teststatistikker, én for hver faktor.

$$F = \frac{\text{Mean square for faktor}}{\text{Residual mean square}}$$

- Telleren er variasjonen mellom gruppene for faktoren.
- Nevneren er variasjonen innenfor grupper.

```
summary(aov(len ~ dose, data = ToothGrowth))
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## dose         2   2426    1213   67.42 9.53e-16 ***
## Residuals   57   1026      18
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(len ~ dose + supp, data = ToothGrowth))
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## dose         2 2426.4  1213.2   82.81  < 2e-16 ***
## supp         1  205.4   205.4   14.02 0.000429 ***
## Residuals   56  820.4    14.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(aov(len ~ dose + supp, data = ToothGrowth))
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## dose          2 2426.4  1213.2   82.81  < 2e-16 ***
## supp          1  205.4   205.4   14.02 0.000429 ***
## Residuals    56  820.4    14.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

For `dose` (jeg har tatt med én desimal mer enn det som vises):

$$F = \frac{1213.2}{14.7} = 82.53$$

$df_1 = 3 - 1 = 2, df_2 = 56$:

```
1 - pf(1213.2/14.7, df1 = 2, df2 = 56)
```

```
## [1] 0
```

```
summary(aov(len ~ dose + supp, data = ToothGrowth))
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## dose          2 2426.4  1213.2   82.81  < 2e-16 ***
## supp          1  205.4   205.4   14.02 0.000429 ***
## Residuals    56  820.4    14.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- For `supp`:

$$F = \frac{205.4}{14.7} = 13.97$$

$df_1 = 2 - 1 = 2, df_2 = 56$:

```
1 - pf(205.4/14.7, df1 = 1, df2 = 56)
```

```
## [1] 0.0004373507
```

Modellen vi her har estimert er nøyaktig den samme som

```
(mod2v ← lm(len ~ dose + supp, data = ToothGrowth))
```

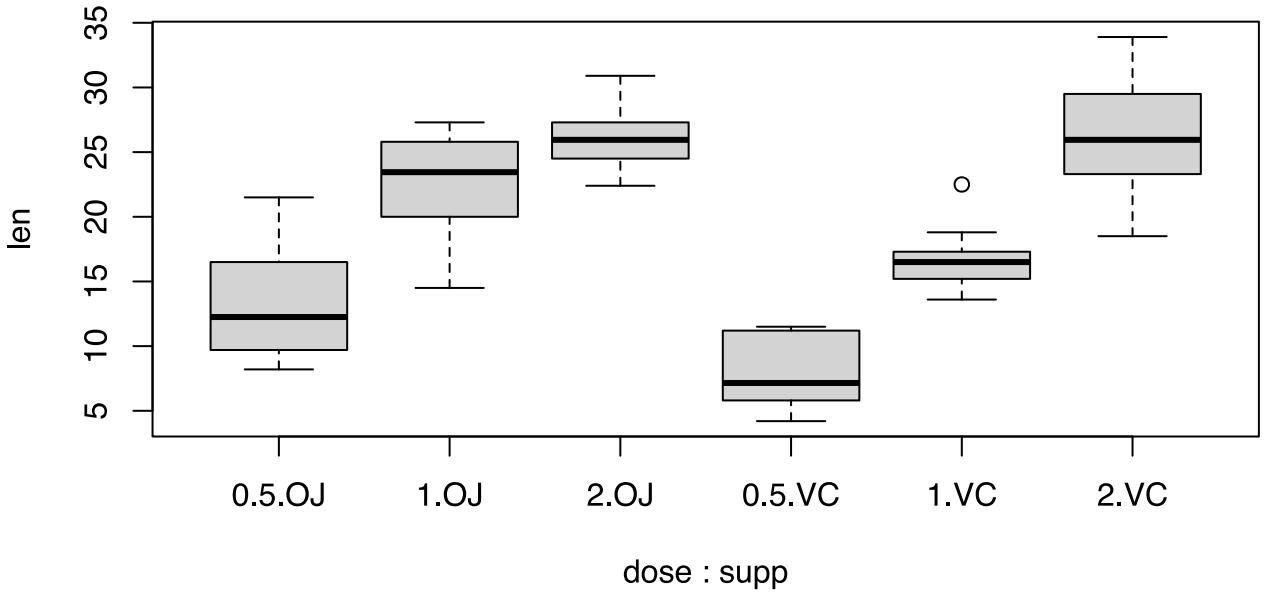
```
##
## Call:
## lm(formula = len ~ dose + supp, data = ToothGrowth)
##
## Coefficients:
## (Intercept)        dose1        dose2       suppVC
##       12.46         9.13        15.49        -3.70
```

```
anova(mod2v)
```

```
## Analysis of Variance Table
##
## Response: len
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## dose       2 2426.43 1213.22  82.811 < 2.2e-16 ***
## supp       1  205.35  205.35  14.017 0.0004293 ***
## Residuals 56  820.43   14.65
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Er effekten av dose avhengig av hvordan C-vitaminene leveres? Er effekten av leveringsmetoden avhengig av dosen?

```
boxplot(len ~ dose + supp, data = ToothGrowth)
```

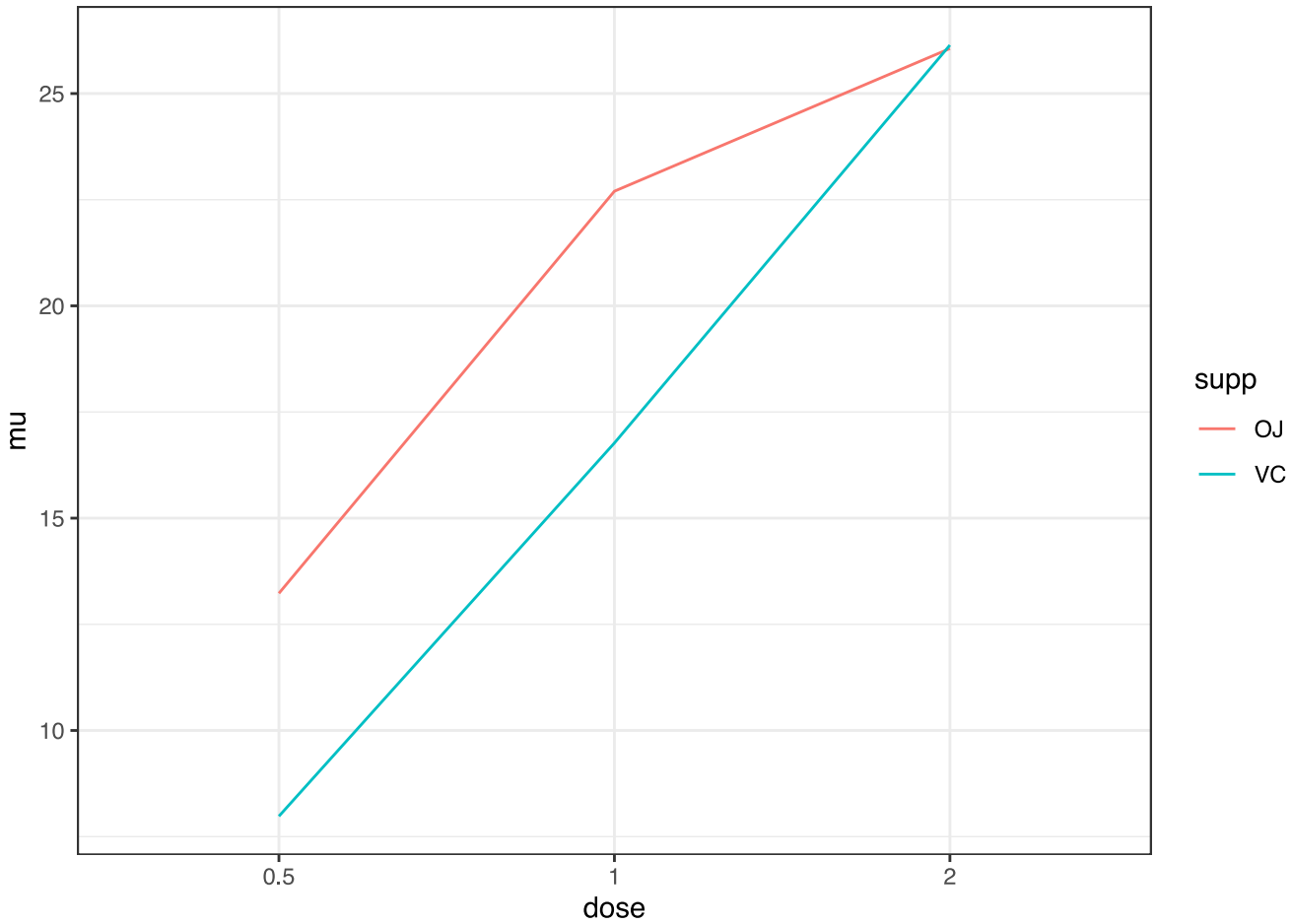


Ja, kanskje det er interaksjoner her?

```
intrxmod ← lm(len ~ dose * supp, data = ToothGrowth)
anova(intrxmod)
```

```
## Analysis of Variance Table
##
## Response: len
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## dose       2 2426.43 1213.22  92.000 < 2.2e-16 ***
## supp       1  205.35  205.35  15.572 0.0002312 ***
## dose:supp  2  108.32   54.16   4.107 0.0218603 *
## Residuals 54  712.11   13.19
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Hva betyr interaksjonene?



Noen eksempler på data hvor interaksjoner er viktige?

Slides created via the R package **xaringan**.

The chakra comes from remark.js, **knitr**, and R Markdown.