Opg 7,8,9

2023-01-12

Opg 7:

Vi fokuserer på hunderacen Whippet.

Samplingfordelingen for $\hat{m}_1$ bestemmes ved ikke-parametrisk bootstrap::

```
set.seed(123)
m1_bootstrap <- whip %>%
      select(maxLA) %>%
      rep_sample_n(size = 50,
                   replace = TRUE,
                   reps = 10000) %>%
      summarise(stat = median(maxLA)) %>%
      mutate(est = "m1hat")
```

Samplingfordelingen for $\hat{m}_2$ bestemmes vha. parametrisk bootstrap, baseret på antagelsen om at $\log(Y_i)$ følger en normalfordeling med parametre μ og σ^2. Dvs. at vi først bestemmer $\hat{\mu}$ og $\hat{\sigma^2}$ for $\log(Y_i)$ ved: .

```
muhat <- mean(log(whip$maxLA))
sigmahat <- var(log(whip$maxLA))
```

Og dernæst resampler 10000 gange 17 realisationer fra en normalfordeling med parametre $\mu = \hat{\mu} = 10.54$ og $\sigma^2 = \hat{\sigma}^2 = 4.41$:

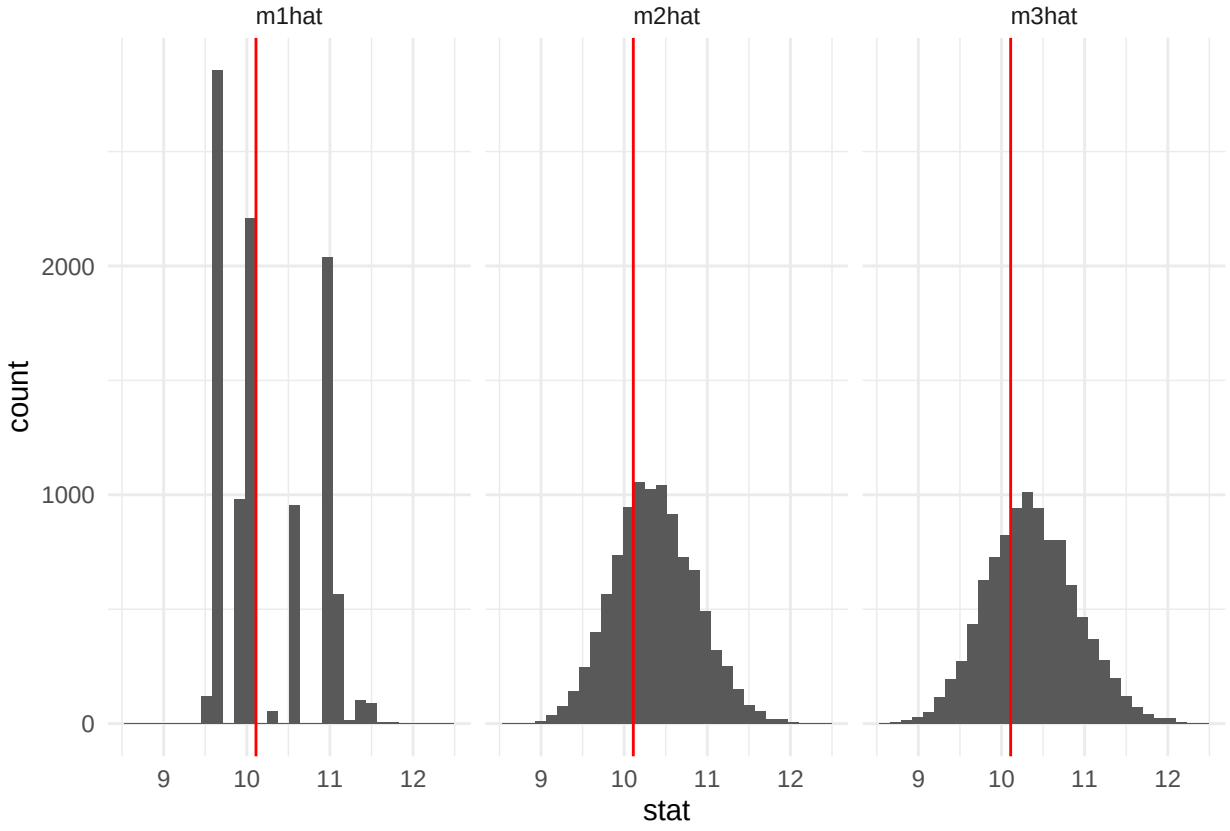
```
n <- nrow(whip)
B <- 10000
set.seed(123)
m2_bootstrap <- tibble(replicate = rep(1:B, each = n),
                logyi = rnorm(B*n, mean = muhat, sd = sqrt(sigmahat))) %>%
  group_by(replicate) %>%
  summarise(stat = exp(mean(logyi))) %>%
  mutate(est = "m2hat")
```

$\hat{m}_3$ bestemmes også ved parametrisk boostrap, dvs. vi resampler fra den samme fordeling som for $\hat{m}_2$:

```
set.seed(123)
m3_bootstrap <- tibble(replicate = rep(1:B, each = n),
                       logyi = rnorm(B*n, mean = muhat, sd = sqrt(sigmahat))) %>%
  group_by(replicate) %>%
  summarise(stat = exp(mean(quantile(x = logyi, probs = c(0.25, 0.75))))) %>%
  mutate(est = "m3hat")
```

De tre samplingfordelinger visualiseres og sammenlignes med den observerede, empiriske median i datasættet:

```
d7 <- bind_rows(m1_bootstrap, m2_bootstrap, m3_bootstrap)
d7 %>% ggplot(aes(x = stat)) +
  geom_histogram(bins = 30) +
  facet_wrap(~factor(est)) +
  geom_vline(xintercept = median(whip$maxLA), col = "red") +
  theme_minimal()
```



En god estimator skal helst være både precise, accurate og unbiased. For at være "precise", skal alle estimaterne være samlet tæt om det samme punkt. En måde at vurdere dette på er den observerede varians af samplingfordelingen for hver estimator:

```
d7 %>% group_by(est) %>%
  summarise(var(stat))
```

```
## # A tibble: 3 x 2
##   est   `var(stat)`
## * <chr>       <dbl>
## 1 m1hat       0.322
## 2 m2hat       0.246
## 3 m3hat       0.296
```

$\hat{m}_2$ er altså den mest precise, mens $\hat{m}_1$ er mindst precise, hvilket også stemmer overens med samplingfordelingernes udseende i histogrammerne ovenfor. Visuelt ser især $\hat{m}_1$ ud til at være meget unprecise. Accuracy og bias er i sagens natur svære at vurdere her, da vi ikke kender den sande median for fordelingen af venstre forkammer-volumen i Whippet-hunde. Hvis vi for øvelsens skyld antog at den observerede median

i vores (meget lille) datasæt var den sande median, kan vi se, at både $\hat{m}_2$ og $\hat{m}_3$ rammer en lille smule ved siden af, dvs. de er en anelse inaccurate og biased. I sidste ende kommer valget af estimator primært an på, hvilke antagelser vi tør gøre om Whippet-hundes hjerter: Hvis vi er overbeviste om, at antagelsen $\log(Y_i)$ er normalfordelt er en god antagelse, så er $\hat{m}_2$ formentlig den bedste estimator at bruge, da den er den mest precise, og teorien fortæller os, at medianen er lig gennemsnittet for en normalfordelt variabel.

Opg 8:

Vi bemærker først at wgt > 0 og maxLA > 0 for alle hunde i datasættet (se tabel i opgave 4), og at vi derfor godt kan bruge logaritmen på de to variable.

Vi betragter en additive noise model for log(wgt) på formen:

$$\log(\text{maxLA}) = \mu(\log(\text{weight}), \text{race}) + \epsilon_i$$

Hvor race indgår som faktor-variabel og log(weight) indgår som numerisk variabel, dvs. middelværdifunktionen μ er en lineær funktion af den kontinuerte variabel log(wgt) og faktorvariablen race givet ved:

Hvorfor helvede kan dette ikke knittes til pdf???

$$\mu(\log(\text{weight}), \text{race}) = \beta_0 + \beta_1 \cdot \log(\text{weight}) + \beta_2 \cdot 1(\text{race} = \text{Grand Danois}) +$$

$$\beta_3 \cdot 1(\text{race} = \text{Labrador}) + \beta_4 \cdot 1(\text{race} = \text{Petit Basset}) + beta_5 \cdot 1(\text{race} = \text{Whippet})$$

Dvs. dimensionen af det lineære underrum $L = L_{\log(\text{weight})} + L_{\text{race}}$ bliver (Jvf NRHAT lemma 3.2.14):

$$\dim(L) = \dim(L_{\log(\text{weight})}) + \dim(L_{\text{race}}) - \dim(L_{\log(\text{weight})} \cap L_{\text{race}}) =$$
$$\dim(L_{\log(\text{weight})}) + \dim(L_{\text{race}}) - \dim(L_1) = 2 + 5 - 1 = 6$$

Vores modelmatrix $\mathbf{X}$ ser dermed således ud (første seks rækker):

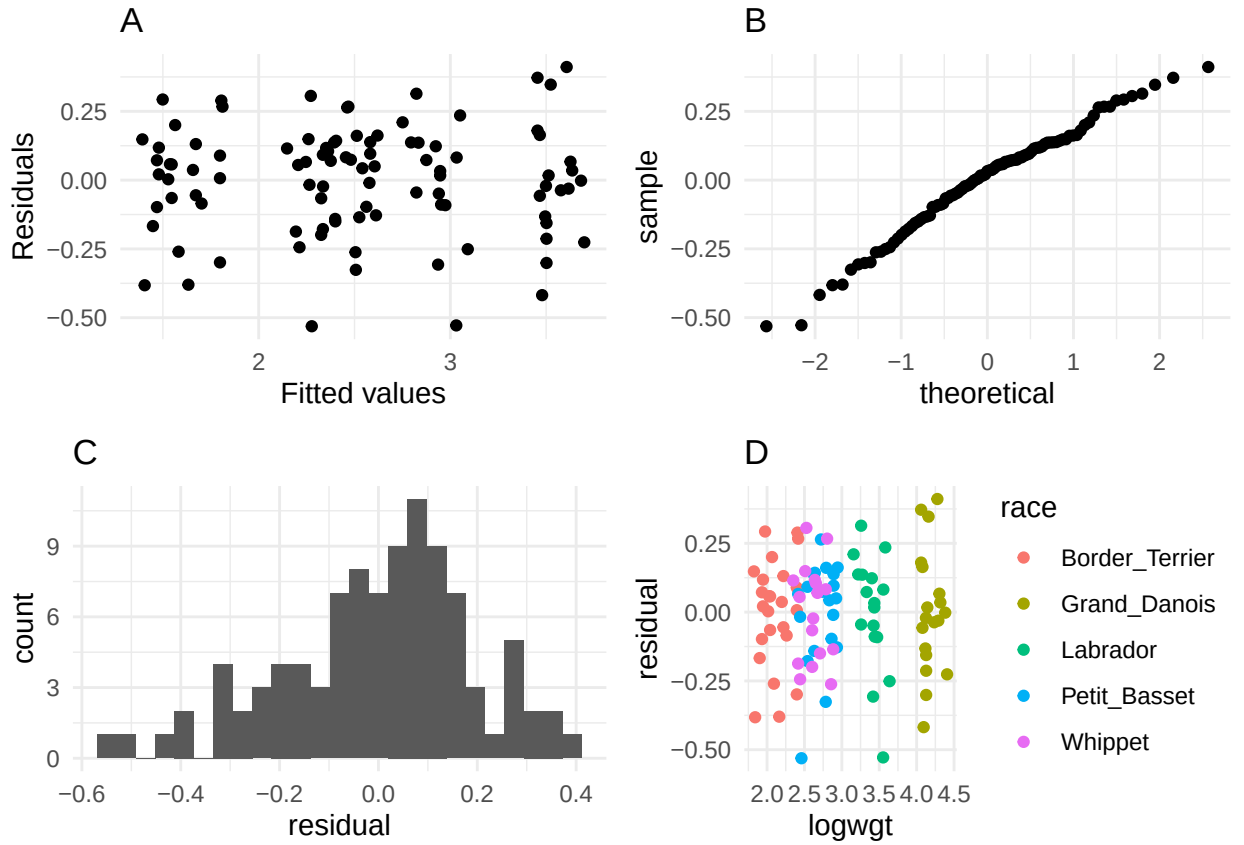
```
head(model.matrix(~ log(wgt) + race, data = hunde))
```

```
##   (Intercept) log(wgt) raceGrand_Danois raceLabrador racePetit_Basset
## 1           1 2.219203                0           0                0
## 2           1 1.931521                0           0                0
## 3           1 2.041220                0           0                0
## 4           1 2.397895                0           0                0
## 5           1 1.840550                0           0                0
## 6           1 1.902108                0           0                0
##   raceWhippet
## 1           0
## 2           0
## 3           0
## 4           0
## 5           0
## 6           0
```

Og den multiple lineære regressionsmodel kan fittes i R med lm:

```
hunde$logmaxLA <- log(hunde$maxLA)
hunde$logwgt <- log(hunde$wgt)
fit <- lm(logmaxLA ~ logwgt + race, data = hunde)
```

For at undersøge antagelsen om at fejlene/residualerne er normalfordelt visualiserer vi residualerne i nedenstående fire grafer. Figur A residualerne plottet mod de fittede værdier i modellen, og generelt ser der ikke ud til at være nogen sammenhæng mellem residualer og fittede værdier. Figur B er et QQ-plot, som viser at fraktilerne i den observerede fordeling af residualer plottet mod fraktilerne fra en normalfordeling generelt ligger på en lige linje. Figur C viser den observerede fordeling af residualer i et histogram, og selvom der er lidt flere observationer i den mest negative ende af fordelingen, end der er i den positive ende, så ser normalfordelingsantagelsen alligevel ikke helt forfærdelig ud, da observationerne generelt er centreret omkring 0 og har en nogenlunde symmetrisk fordeling. Figur D viser residualerne plottet mod log(wgt) og farvet efter hunderace. Også her ser der ikke ud til at være den store forskel i fordeling af residualer på baggrund af de forklarende variable.



Konklusionen bliver, at vi vælger at tro på antagelsen om, at residualerne er normalfordelt, og dermed kan vi tillade os at konstruere konfidensintervaller for parameterestimaterne ud fra t-fordelingen. Svarende til vores valgte parametrisering bliver parameterestimaterne med tilhørende 95%-konfidensintervaller:

```
##     beta                term estimate lower_ci upper_ci
## 1 beta_0           intercept    0.103   -0.448    0.655
## 2 beta_1              logwgt    0.707    0.448    0.966
## 3 beta_2 race: Grand_Danois    0.481   -0.069    1.031
## 4 beta_3     race: Labrador    0.416    0.059    0.773
## 5 beta_4 race: Petit_Basset    0.435    0.236    0.634
## 6 beta_5      race: Whippet    0.383    0.201    0.565
```

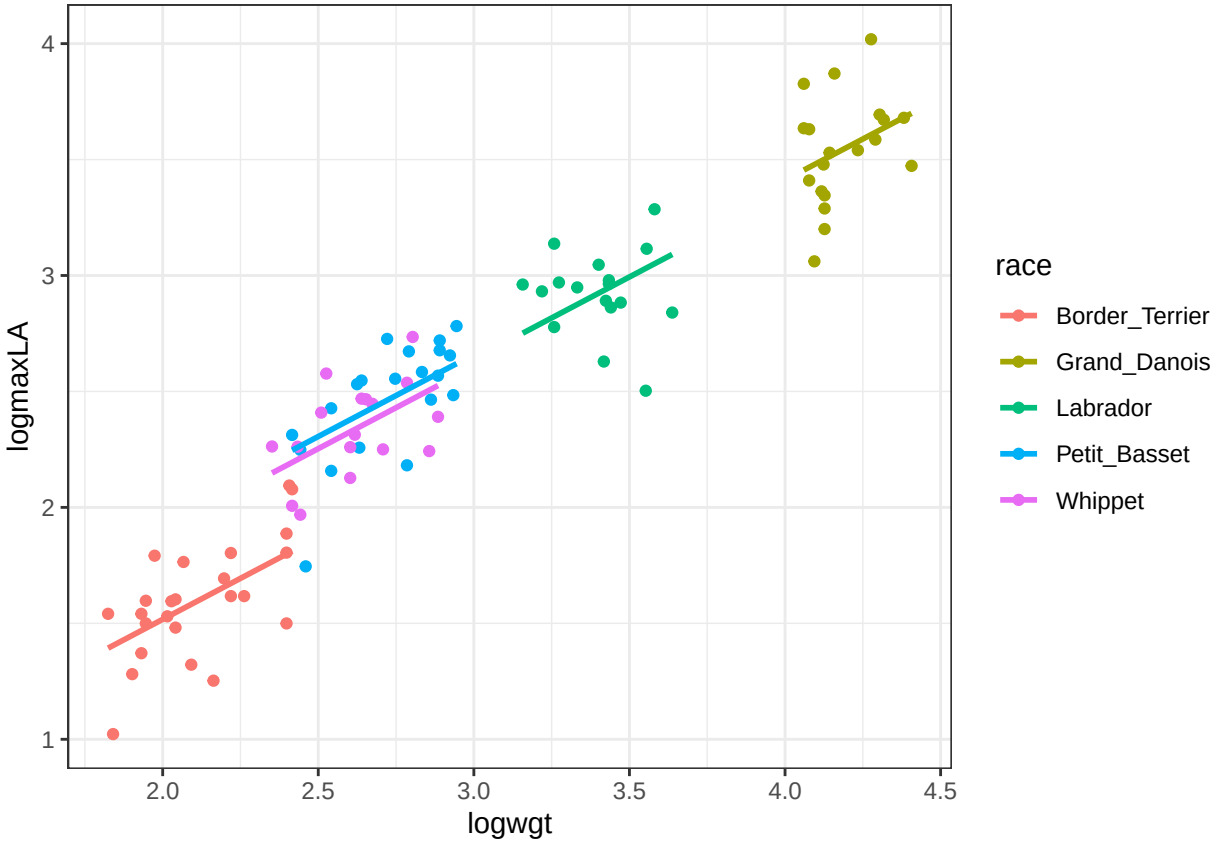
Modellen er en additiv model, hvor $\beta_0 = 0.10$ (interceptet) er estimatet af log(maxLA) for Border Terriers med log(wgt) = 0. β_1 er koefficienten for "hældningen" i den affine funktion der beskriver sammenhængen mellem log(wgt) og log(maxLA), dvs. at for hver stigning i log(wgt) på 1 stiger log(maxLA) med 0.71 uanset hunderace. β_{2-5} er forskellen i log(maxLA) for de øvrige hunderacer sammenlignet med en Border Terrier.

Hvis man f.eks. vil bruge modellen til at estimere venstre forkammervolumen i mL hos en Grand Danois med en vægt på 70 kg (dvs. log(wgt) = 4.25) skal man altså udregne:

$$\exp(\beta_0 + \beta_1 \cdot 4.25 + \beta_2) = \exp(0.10 + 0.71 \cdot 4.25 + 0.48) = 36.15 \text{ mL}$$

(Det angivne resultat er udregnet med flere decimaler i mellemregningerne end vist ovenfor)

Modellen kan også visualiseres således:



Vi bemærker desuden, at hvis vi ville teste om det var rimeligt at antage, at hældningen for de fem hunderacer var ens, kunne vi foretage et test for vekselvirkning, dvs. f.eks. ved at undersøge modellen med med parametriseringen $\mu(\log(\text{weight}), \text{race}) = \beta_0 + \beta_{\text{race},\log(\text{wgt})} \cdot \log(\text{wgt})$, dvs. ved at lade hver hunderace have sin egen hældning af linjen for sammenhængen med log(wgt), og man kunne sammenligne de to modeller, f.eks. med et F-test. Baseret på opgaveformuleringen betragter vi dog dette som uden for denne opgave.

Opgave 9

Vi betragter først underrummet svarende til vores model ovenfor, dvs. $L_{\log(\text{wgt})} + L_{\text{race}}$

Og det mindre underrum L_{race} svarende til en model, hvor vi kun bruger log(wgt) som forklarende variabel, og vi bemærker at:

$$L_{\text{race}} \subseteq L_{\text{race}} + L_{\log(\text{wgt})}$$

Vi bruger nu et F-test til at teste nulhypotesen $H_0 : \ \mu \in L_{\text{race}}$ under den større model givet ved $L_{\text{race}} + L_{\log(\text{wgt})}$. Dette gøres ved følgene R-kode:

```
fit0 <- lm(logmaxLA ~ logwgt, data = hunde)
a <- anova(fit0, fit)
```

Vi får en F-statistic på:

```
a$F[2]
```

```
## [1] 9.822703
```

Som giver $P < 1,2 \cdot 10^{-6}$, og vi forkaster dermed nulhypotesen og konkluderer, at modellen som indeholder både hunderace og log(vægt) er signifikant bedre til at beskrive fordelingen af log(maxLA), hvilket er ensbetydende med at datasættet *ikke* understøtter en hypotese om, at der ikke er forskel på venstre forkammervolumen for de forskellige racer, hvis man justerer for hundens kropsvægt.

Til sidst bemærker vi, at ovenstående F-test ligesom konfidensintervallerne i Opgave 8 kun kan bruges, hvis residualerne er normalfordelt. I Opgave 8 konkluderede vi, at dette var tilfældet, men vi observerede også en lille "hale" i den lave ende af fordelingen på histogrammet (Figur C). For at dobbelttjekke, at vi ikke forbryder os mod antagelsen om normalfordelte residualer tjekker vi lige om vi får det samme resultat hvis vi laver et F-test med bootstrapping (*efter samme fremgangsmåde som i opgave 6*).

Eftersom $q = \dim(L_{\text{race}}) = 2, p = \dim(L_{\text{race}} + L_{\log(\text{wgt})}) = 6$, vil F-teststørrelsen være F-fordelt med $(p-q, n-p) = (6-2, 97-6) = (4, 91)$ frihedsgrader.

```
X0 <- model.matrix(fit0)
n <- nrow(X0)

set.seed(2022)
B <- 10000
my_boot <- tibble(residuals = residuals(fit0)) %>%
    rep_sample_n(size = n, replace = TRUE, reps = B) %>%
    mutate(y = fitted(fit0) + residuals)

X <- model.matrix(fit)

F_test <- function(lm_null, lm_full) {
    p <- lm_full$rank
    q <- lm_null$rank

    lm_full$df.residual * sum((lm_full$fitted.values -
        lm_null$fitted.values)^2)/(sum(lm_full$residuals^2) *
        (p - q))
}

my_res <- my_boot %>%
    summarize(F = F_test(lm.fit(X0, y), lm.fit(X, y)))

F_obs <- F_test(lm.fit(X0, hunde$logmaxLA), lm.fit(X, hunde$logmaxLA))
p_value <- sum(my_res$F > F_obs) / B
data.frame(Ftest = F_obs, P = p_value)
```

```
##      Ftest P
## 1 9.822703 0
```

Vi får en observeret F-teststørrelse på 9.8 og $P = 0$, fordi der i de 10000 resamplinger ikke fandtes en eneste F-teststørrelse som var større end 9.8. Dette stemmer fint overens med resultaterne af den eksakte F-test ovenfor. Figuren nedenfor illustrerer placeringen af vores observerede F-teststørrelse (rød linje) ifht. de 10000 F-teststørrelser som er opnået ved resampling under nulhypotesen (dvs. bootstrapfordelingen). Desuden viser den overlejrede kurve den teoretiske F-fordeling med frihedsgrader (4,91), og vi kan se at den stemmer meget fint overens med de resamplede størrelser, hvilket igen indikerer at det var acceptabelt at antage at residualerne var normalfordelte.

```
ggplot(data = my_res) + geom_histogram(aes(x = F, y = ..density..),
                                       color = "white", fill = "steelblue",
                                       bins = 40) +
  labs(x = "F") +
  theme_bw() +
  stat_function(fun = df, args = list(df1 = 4, df2 = 91), geom = "area", fill = "pink",
    color = "blue", alpha = 0.25) +
  geom_vline(xintercept = F_obs, color = "red")
```