Obligatoriske aflevering StatMet - Anders Tovler

Nikolaj Asgreen, William Baoxin Wang Lin, Jacob Christian Hede Dahlgaard

20 januar 2024

Contents

OPG 1

Hermed ses de 25 observationer fra uniform fordeling fra -1 til 1.

```
##  [1] -0.38446778 -0.48465500  0.10464487 -0.88723370 -0.06290143 -0.03245853
##  [7]  0.62480524 -0.25935893  0.09311719 -0.65947590  0.24999295  0.76433104
## [13] -0.43929232 -0.20302420  0.52510216  0.33804342 -0.59077568 -0.28495029
## [19] -0.28104977  0.38058106  0.07162231  0.42160769  0.07669740  0.49794445
## [25] -0.15979710
```

Vi beregner $\epsilon_{25}([-0.2, 0.2])$ ved

$$\epsilon_{25}([-0.2, 0.2]) = \frac{1}{25}\sum_{i=1}^{25} \delta_{x_i},$$

hvor $x_1, \ldots, x_{25}$ er de simulerede værdier fra ligefordelingen fra -1 til 1. Vi får ved brug af R.

```
length(subset(x,abs(x)<0.2))/25
```

```
## [1] 0.28
```

Her er den empiriske middelværdi samt varians beregnet ved brug af de indbyggede kommandoer i R.

```
## middelværdi     varians
## -0.02323803  0.18586547
```

OPG 2

Vi definerer $\hat{m}$ og $\hat{\tau}$ ved definitionen givet i opgaven.

```
med <- function(x){
  n<-length(x)
  sorted <- sort(x)
```

```
  if (n%%2==1) {res <- sorted[(n+1)/2]}
  else {res <- (sorted[n/2]+sorted[n/2+1])/2}
  return(res)
}

tau <- function(x,y){
  res <- sum((abs(x-y)<=1)*x)/sum(abs(x-y)<=1)
  return(res)
}
```

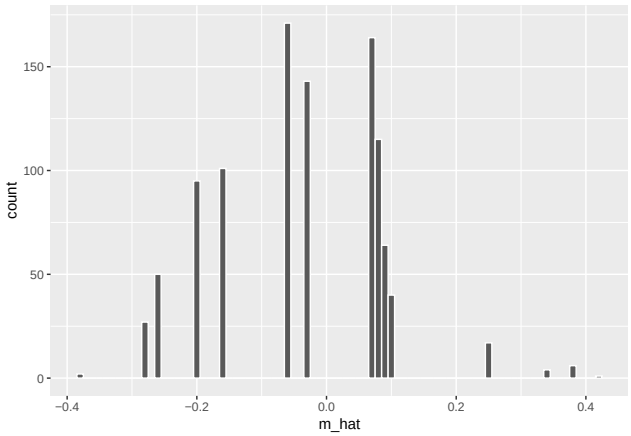
Bemærk, at vi vil bruge kommandoen "mean()" til at beregne $e\hat{t}a$ i de følgende opgaver. $\hat{m}$ er givet som.

```
##       m_hat     eta_hat     tau_hat
## -0.03245853 -0.02323803 -0.02323803
```

Vi vælger nu at bootstrap med 1000 samples, da vi føler at det er et repræsentativt/"standard" valg af antak replikationer.

```
B<-1000
data_x<-data.frame(x)
set.seed(100)
bootstrap<-data_x %>%
  rep_sample_n(size=25,replace=TRUE,reps=B) %>%
  group_by(replicate) %>%
  summarize(m_hat=med(x),
            eta_hat=mean(x),
            tau_hat=tau(x,m_hat))
```

$\hat{m}_{bootstrap}$'s sampling fordeling er visuelt givet nedenfor.



Vi bemærker at vores bootstrap af medianfunktionen er meget diskret fordelt. Dette skyldes at vi har et fast antal observationer (25) og dermed kan vores median kun være en af de 25 observationer, uanset hvor mange samples vi foretager. Hvorimod middelværdien $\hat{\eta}$ og den "centrerede" middelværdi $\hat{\tau}$ vil kunne antage flere værdier (som ses senere), selvom de også kun kan antage et tælleligt antal værdier.

Standard error konfidensintervallet, percentilintervallet og det simple bootstrap interval fås til følgende.

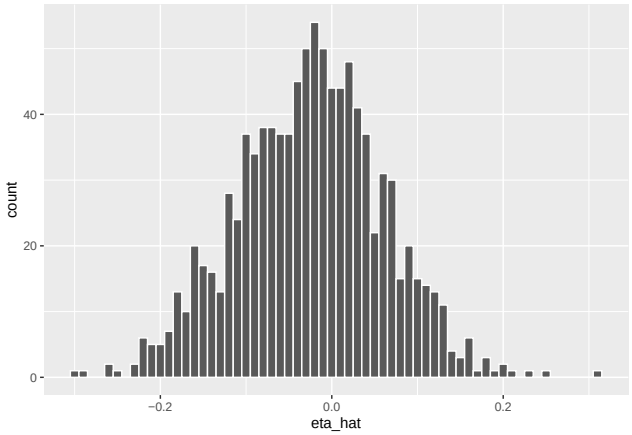
```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   -0.289    0.224
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
```

```
##      <dbl>    <dbl>
## 1   -0.281    0.250
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   -0.315    0.216
```

$\hat{\eta}_{bootstrap}$'s sampling fordeling er visuelt givet nedenfor.



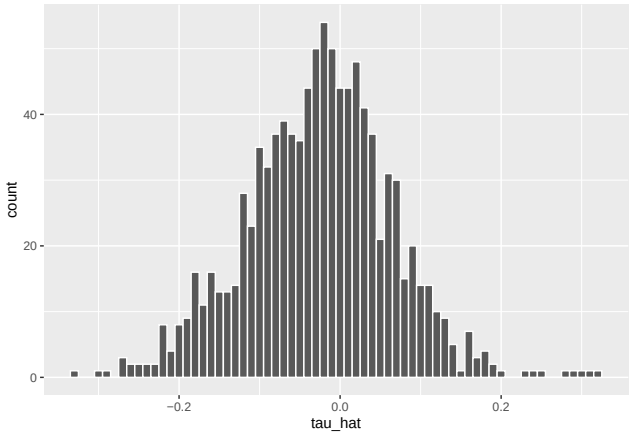
Standard error konfindensinterval, percentilinterval og simpel bootstrap konfidensinterval til følgende.

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   -0.190    0.144
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   -0.193    0.132
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   -0.178    0.146
```

$\hat{\tau}_{bootstrap}$'s sampling fordeling er visuelt givet nedenfor.



Standard error konfidensinterval, percentilinterval og simpel bootstrap konfidensinterval er givet nedenfor

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   -0.201    0.154
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   -0.210    0.153
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1   -0.200    0.164
```

OPG 3

Vi betagter $\mathbb{P}(|\hat{\eta}| > 0.2)$, hvor P er ligefordelingen på $[-1, 1]$, dvs. $X_i \sim \text{Unif}(-1, 1)$. Se

$$\text{E}(X_i) = \frac{-1+1}{2} = 0 \qquad \text{og} \qquad \text{Var}(X_i) = \frac{(1-(-1))^2}{12} = \frac{1}{3}.$$

Vi ser derfor

$$\text{E}(\hat{\eta}) = \text{E}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) = \frac{1}{n}\text{E}\left(\sum_{i=1}^{n} X_i\right) = \frac{1}{n}\sum_{i=1}^{n}\text{E}(X_i) = 0$$

$$\text{Var}(\hat{\eta}) = \text{Var}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) = \frac{1}{n^2}\text{Var}\left(\sum_{i=1}^{n} X_i\right) \overset{i.i.d.}{=} \frac{1}{n^2}\sum_{i=1}^{n}\text{Var}(X_i) = \frac{1}{n^2}\cdot n \cdot \frac{1}{3} = \frac{1}{3n}.$$

Chebychev's ulighed giver nu

$$\mathbb{P}(|\hat{\eta}| > 0.2) = \mathbb{P}(|\hat{\eta} - \text{E}(\hat{\eta})| > 0.2) \leq \frac{\text{Var}(\hat{\eta})}{0.2^2} = \frac{25}{3n}.$$

Vi betagter nu $\mathbb{P}(|\hat{\eta}| > 0.2)$, hvor P er normalfordelingen med middelværdi 0 og varians 1, dvs. $X_i \sim \mathcal{N}(0, 1)$. Se $\text{E}(X_i) = 0$ og $\text{Var}(X_i) = 1$, og vi ser derfor

$$\text{E}(\hat{\eta}) = \text{E}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) = \frac{1}{n}\text{E}\left(\sum_{i=1}^{n} X_i\right) = \frac{1}{n}\sum_{i=1}^{n}\text{E}(X_i) = 0$$

$$\text{Var}(\hat{\eta}) = \text{Var}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) = \frac{1}{n^2}\text{Var}\left(\sum_{i=1}^{n} X_i\right) \overset{i.i.d.}{=} \frac{1}{n^2}\sum_{i=1}^{n}\text{Var}(X_i) = \frac{1}{n^2}\cdot n = \frac{1}{n}.$$

Chebychev's ulighed giver nu

$$\mathbb{P}(|\hat{\eta}| > 0.2) = \mathbb{P}(|\hat{\eta} - \text{E}(\hat{\eta})| > 0.2) \leq \frac{\text{Var}(\hat{\eta})}{0.2^2} = \frac{25}{n}.$$

Vi betagter nu $\mathbb{P}(|\hat{\eta}| > 0.2)$, hvor P er laplacefordelingen med tæthed $f(x) = \frac{1}{2}e^{-|x|}$, dvs. $X_i \sim \text{laplace}(0, 1)$. Se

$$\text{E}(X_i) = 0 \text{ og } \text{Var}(X_i) = 2 \cdot 1^2 = 2.$$

Vi ser derfor

$$\text{E}(\hat{\eta}) = \text{E}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) = \frac{1}{n}\text{E}\left(\sum_{i=1}^{n} X_i\right) = \frac{1}{n}\sum_{i=1}^{n}\text{E}(X_i) = 0$$

$$\mathrm{Var}(\hat{\eta}) = \mathrm{Var}\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) = \frac{1}{n^2}\mathrm{Var}\left(\sum_{i=1}^{n} X_i\right) \overset{i.i.d.}{=} \frac{1}{n^2}\sum_{i=1}^{n}\mathrm{Var}(X_i) = \frac{1}{n^2}\cdot 2n = \frac{2}{n}.$$

Chebychev's ulighed giver nu

$$\mathbb{P}(|\hat{\eta}| > 0.2) = \mathbb{P}(|\hat{\eta} - \mathrm{E}(\hat{\eta})| > 0.2) \leq \frac{\mathrm{Var}(\hat{\eta})}{0.2^2} = \frac{50}{n}.$$

Vi ser, at for $n \to \infty$ vil sandsynligheden for, at $|\hat{\eta}| > 0.2$ gå mod 0 for alle fordelinger.

OPG 4

Et eksempel på et symmetrisk sandsynlighedsmål, der ikke har tæthed kan være

$$P_0(\{1\}) = P_0(\{-1\}) = \frac{1}{2}.$$

Det gælder trivielt, at dette sandsynlighedsmål er symmetrisk, pr. definitionen givet i opgaven. Et eksempel på et symmetrisk sandsynlighedsmål, der har tæthed mht. lebesguemålet, er standard normalfordelingen, $P_0 = \mathcal{N}(0,1)$. Den opfylder $F(x) = 1 - F(-x)$, hvilket er tilstrækkeligt til at slutte, at sandsynlighedsmålet er symmetrisk omkring 0 pr. opgave 1.6.

Vi ønsker nu at vise, at 0 er en median for ethvert sandsynlighedsmål omkring 0. Da P_0 er symmetrisk, gælder

$$\begin{aligned} P_0((-\infty,0)) &= P_0((0,\infty)) \\ &= 1 - P_0((-\infty,0]) \\ &= 1 - (P_0((-\infty,0)) + P_0(\{0\})) \\ &= 1 - P_0((-\infty,0)) - P_0(\{0\}). \end{aligned}$$

Vi får nu

$$2P_0((-\infty,0)) = 1 - P_0(\{0\}) \implies P_0((-\infty,0)) = \frac{1}{2} - \frac{P_0(\{0\})}{2}$$

Vi ser nu, at $P_0((-\infty,0)) = F(0-) \leq \frac{1}{2}$, da $0 \leq P_0(\{0\})$. Vi ser også

$$P_0((-\infty,0)) = \frac{1}{2} - \frac{P_0(\{0\})}{2} \implies P_0((-\infty,0]) = \frac{1}{2} + \frac{P_0(\{0\})}{2},$$

og da $0 \leq P_0(\{0\})$, må $P_0((-\infty,0]) = F(0) \leq \frac{1}{2}$. Dermed er 0 pr. definition en 0.5-kvantil.

Det ses for det symmetriske sandsynlighedsmål med $P_0(\{1\}) = P_0(\{-1\}) = \frac{1}{2}$, at den har flere 0.5-kvantiler, og dermed er medianen ikke entydig. For eksempel er $F(0.5-) \leq \frac{1}{2} \leq F(0.5)$ og $F(-0.2-) \leq \frac{1}{2} \leq F(-0.2)$ også medianer.

OPG 5

Til simulationsstudiet vælger vi at kigge på 3 forskellige fordelinger, med en sand location parameter $\mu = 50$, for at se om der er forskel på hvilken estimator, der er bedst. I simulationerne bruger vi desuden seed 100 til at generere vores data.

De første 3 funktioner defineret nedenfor simulerer hhv. uniform-, normal- og laplacefordelinger 1000 gange, som for et givet input n (antal udtræk fra fordelingen), beregner den tilhørende $\hat{m}$, $\hat{\eta}$ og $\hat{\tau}$, og tilføjer dem i en tabel. Dette bruger vi til at tegne fordelingen af estimatorerne.

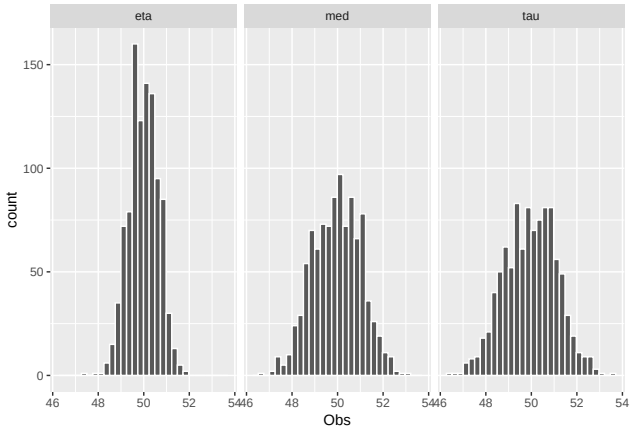
```
uniffunc <- function(n){
  n_pkt <- 1000
  set.seed(100)
  data1_summary <- tibble(med=numeric(n_pkt),eta=numeric(n_pkt),tau=numeric(n_pkt))
```

```
  raekke<-1
  for (i in seq(1,n_pkt,1)){
    data1 <- runif(n,45,55) %>%
      data.frame()
    colnames(data1)<-c("Obs")

    med1 <- med(data1$Obs)
    eta1 <- mean(data1$Obs)
    tau1 <- tau(data1$Obs,med1)

    data1_summary[raekke,1]<-med1
    data1_summary[raekke,2]<-eta1
    data1_summary[raekke,3]<-tau1
    raekke<-raekke+1
  }
  return(data1_summary)
}

normfunc <- function(n){
  n_pkt <- 1000
  set.seed(100)
  data1_summary <- tibble(med=numeric(n_pkt),eta=numeric(n_pkt),tau=numeric(n_pkt))
  raekke<-1
  data1_summary
  for (i in seq(1,n_pkt,1)){
    data1 <- rnorm(n,50,5) %>%
      data.frame()
    colnames(data1)<-c("Obs")

    med1 <- med(data1$Obs)
    eta1 <- mean(data1$Obs)
    tau1 <- tau(data1$Obs,med1)

    data1_summary[raekke,1]<-med1
    data1_summary[raekke,2]<-eta1
    data1_summary[raekke,3]<-tau1
    raekke<-raekke+1
  }
  return(data1_summary)
}

laplacefunc <- function(n){
  n_pkt <- 1000
  set.seed(100)
  data1_summary <- tibble(med=numeric(n_pkt),eta=numeric(n_pkt),tau=numeric(n_pkt))
  raekke<-1
  data1_summary
  for (i in seq(1,n_pkt,1)){
    data1 <- rlaplace(n,50,5) %>%
      data.frame()
    colnames(data1)<-c("Obs")

    med1 <- med(data1$Obs)
    eta1 <- mean(data1$Obs)
```

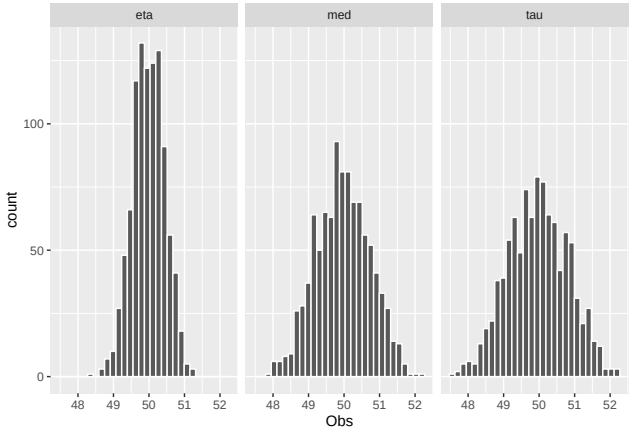
```
    tau1 <- tau(data1$Obs,med1)

    data1_summary[raekke,1]<-med1
    data1_summary[raekke,2]<-eta1
    data1_summary[raekke,3]<-tau1
    raekke<-raekke+1
  }
  return(data1_summary)
}

graf <- function(func){
  data_1_tidy <- func %>%
    data.frame() %>%
    pivot_longer(names_to = "Estimator",values_to = "Obs",cols = c(med,eta,tau))
  data_1_tidy

  ggplot(data_1_tidy,aes(x=Obs))+
    geom_histogram(color="white")+
    facet_wrap(~Estimator)
}
```

Vores plots bliver nu for en uniform fordeling (dvs. Unif(45,55), som er en scale transformation af Unif(-5,5) med $\mu = 50$) for $n = 20, 40, 200$:

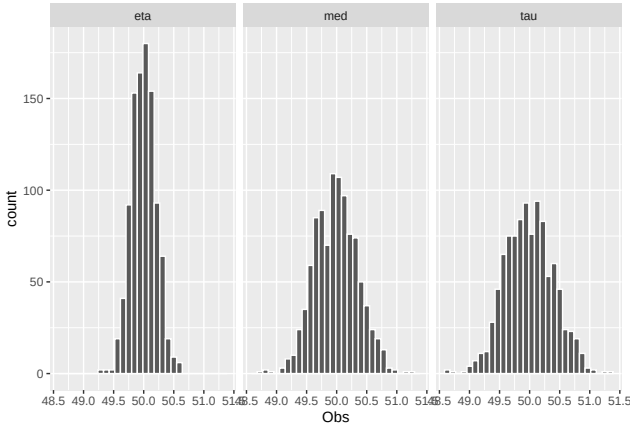
For $n = 20$.



For $n = 40$.

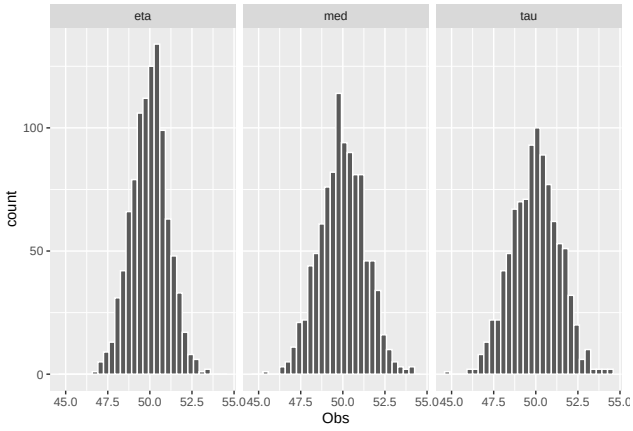


For $n = 200$.

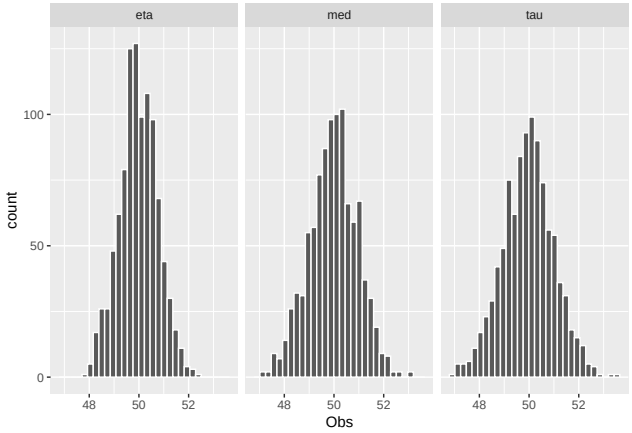


For en normalfordeling (dvs. Norm(50,5), som er en scale transformation af Norm(0,5) med $\mu = 50$) for $n = 20, 40, 200$:

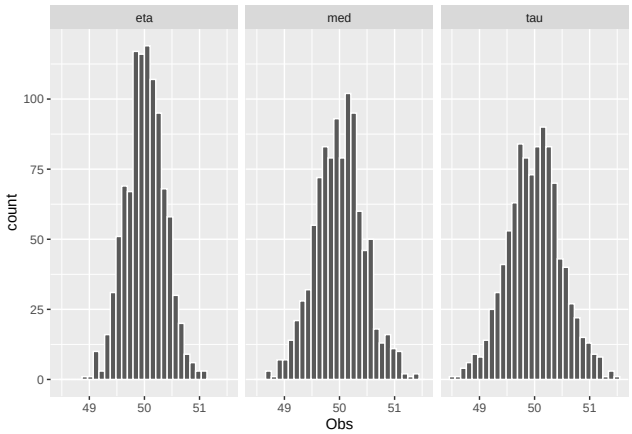
For $n = 20$.



For $n = 40$.

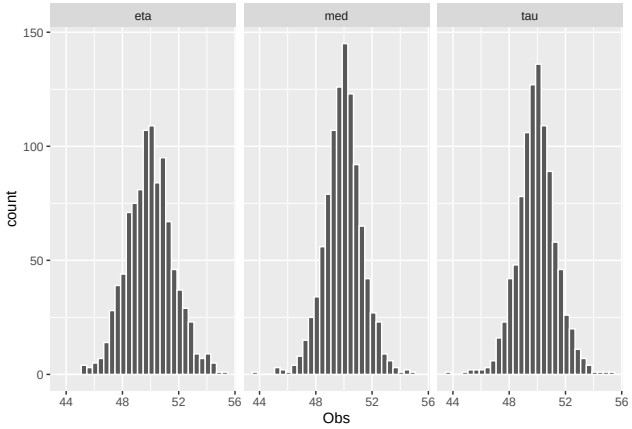


For $n = 200$.

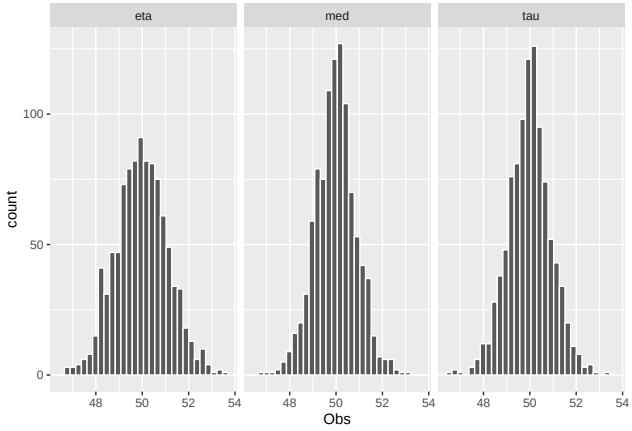


Og for en laplacefordeling (dvs. Laplace(50,5), som er en scale transformation af Laplace(0,5) med $\mu = 50$) for $n = 20, 40, 200$:

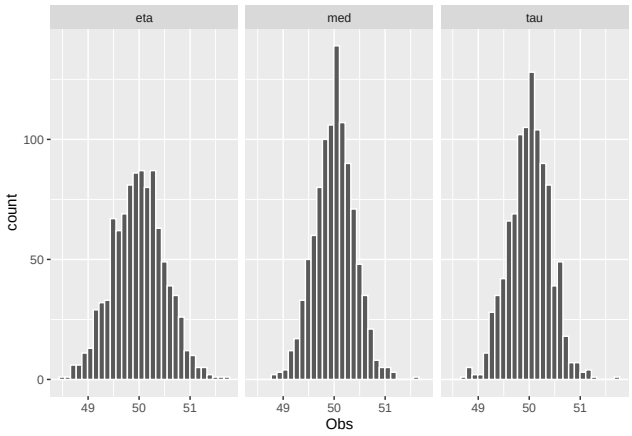
For $n = 20$.



For $n = 40$.



For $n = 200$.

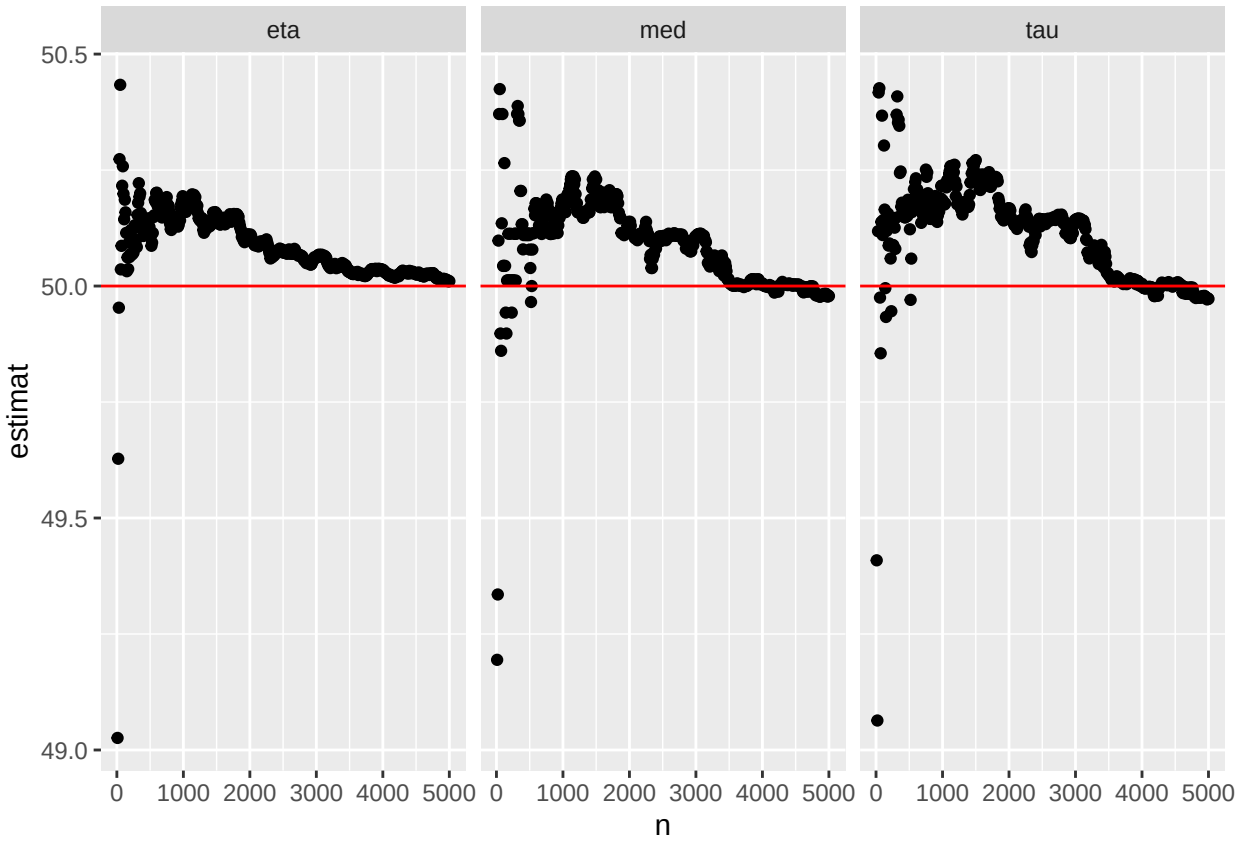


Det ses, at alle estimatorerne centrerer godt omkring 50 (da den sande location parameter i vores simulationsstudie for alle 3 fordelinger er $\mu = 50$). Alle fordelingerne ser nogenlunde normalfordelte ud, men for den uniforme fordeling og normalfordelingen ser det ud til, at $\hat{\eta}$ har en mindre spredning, mens for laplacefordelingen har $\hat{m}$ og $\hat{\tau}$ en mindre spredning. Det ligner, at den foretrukne estimator afhænger af, hvilken fordeling der undersøges for.

For at få et mere klart billede om disse observationer passer generelt, undersøger vi nærmere effekten af n. Vi vil i det følgende lave plots for $\hat{\eta}$, $\hat{m}$ og $\hat{\tau}$ med 500 forskellige n, ligeligt fordelt fra $n = 10$ til $n = 5000$, for hver af de tre fordelingstyper, og se hvordan estimatorerne forbedres når n øges. Vi får:

For en uniform fordeling (Unif(45,55)).

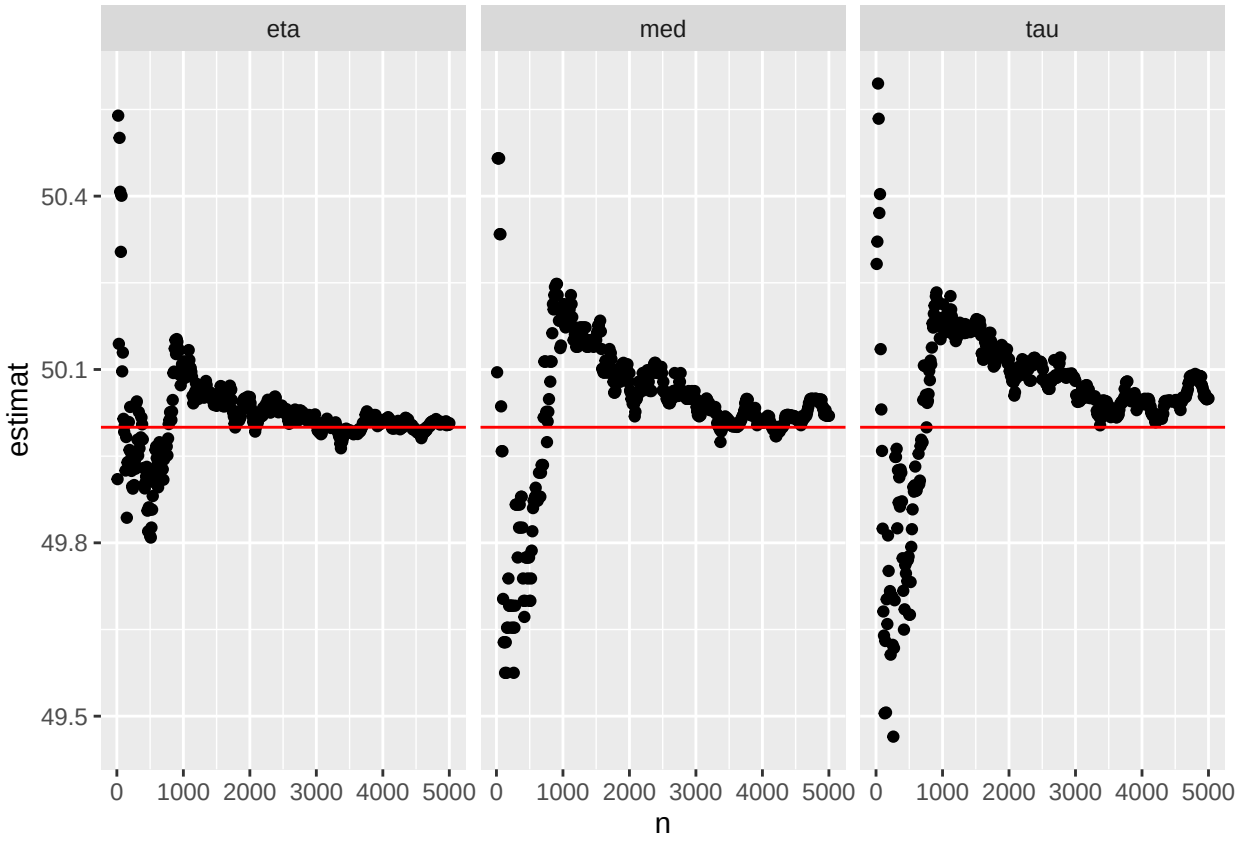
```
#Uniform test
n_pkt <- 500
data1_summary <- tibble(med=numeric(n_pkt),
                        eta=numeric(n_pkt),
                        tau=numeric(n_pkt),
                        n=seq(10,n_pkt*10,10))
raekke <- 1
for (i in seq(10,n_pkt*10,10)){
  set.seed(100)
  data1 <- runif(i,45,55)

  med1 <- med(data1)
  eta1 <- mean(data1)
  tau1 <- tau(data1,med1)

  data1_summary[raekke,1]<-med1
  data1_summary[raekke,2]<-eta1
  data1_summary[raekke,3]<-tau1
  raekke <- raekke+1
}

data1_df <- data1_summary %>%
  data.frame() %>%
  pivot_longer(names_to = "estimator",values_to = "estimat",cols=c(med,eta,tau))

ggplot(data1_df,aes(x=n,y=estimat))+
  geom_point()+
  geom_hline(yintercept = 50,color="red")+
  facet_wrap(~estimator)
```



Koden for plottet til normalfordelingen laves ligesom for den uniforme fordeling, men "runif(i,45,55)" byttes med "rnorm(i,50,5)".

```
#Normalfordeling test
n_pkt <- 500
data1_summary <- tibble(med=numeric(n_pkt),
                        eta=numeric(n_pkt),
                        tau=numeric(n_pkt),
                        n=seq(10,n_pkt*10,10))
raekke <- 1
for (i in seq(10,n_pkt*10,10)){
  set.seed(100)
  data1 <- rnorm(i,50,5)

  med1 <- med(data1)
  eta1 <- mean(data1)
  tau1 <- tau(data1,med1)

  data1_summary[raekke,1]<-med1
  data1_summary[raekke,2]<-eta1
  data1_summary[raekke,3]<-tau1
  raekke <- raekke+1
}

data1_df <- data1_summary %>%
  data.frame() %>%
  pivot_longer(names_to = "estimator",values_to = "estimat",cols=c(med,eta,tau))
```

```
ggplot(data1_df,aes(x=n,y=estimat))+
  geom_point()+
  geom_hline(yintercept = 50,color="red")+
  facet_wrap(~estimator)
```



koden for plottet til laplace fordelingen laves ligesom for den uniforme fordeling, men "runif(i,45,55)" byttes med "rlaplace(i,50,5)" (bemærk den hentes fra R-package "extraDistr").

```
#Laplace test
n_pkt <- 500
data1_summary <- tibble(med=numeric(n_pkt),
                        eta=numeric(n_pkt),
                        tau=numeric(n_pkt),
                        n=seq(10,n_pkt*10,10))

raekke <- 1
for (i in seq(10,n_pkt*10,10)){
  set.seed(100)
  data1 <- rlaplace(i,50,10)

  med1 <- med(data1)
  eta1 <- mean(data1)
  tau1 <- tau(data1,med1)

  data1_summary[raekke,1]<-med1
  data1_summary[raekke,2]<-eta1
  data1_summary[raekke,3]<-tau1
  raekke <- raekke+1
```

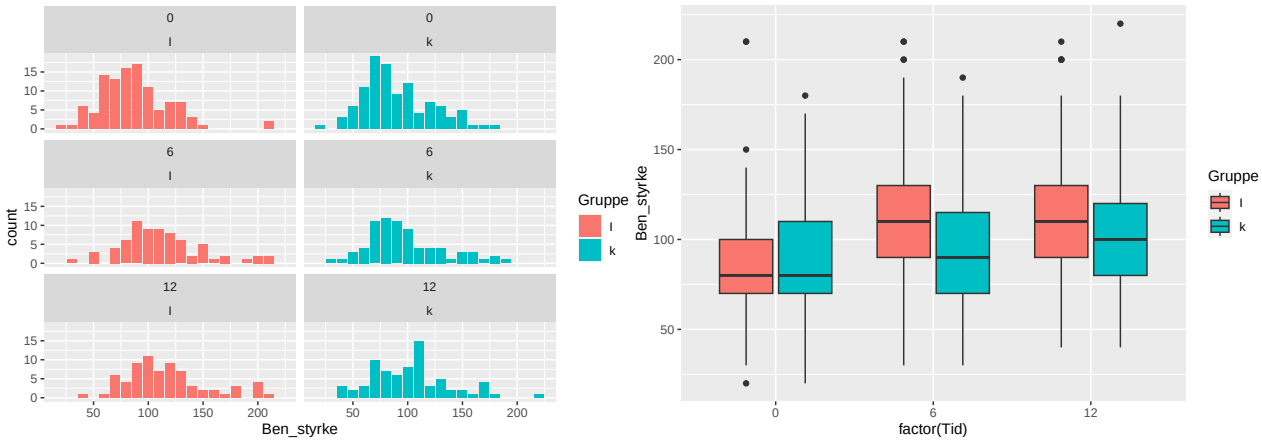
```
}

data1_df <- data1_summary %>%
  data.frame() %>%
  pivot_longer(names_to = "estimator",values_to = "estimat",cols=c(med,eta,tau))

ggplot(data1_df,aes(x=n,y=estimat))+
  geom_point()+
  geom_hline(yintercept = 50,color="red")+
  facet_wrap(~estimator)
```

Ifølge de store tals lov, skulle vi gerne se værdierne for hhv. $\hat{\eta}$, $\hat{m}$ og $\hat{\tau}$ konvergere mod den forventede værdi, 50, for større og større n, hvilket også er tilfældet. De er altså alle tre gode estimatorer ved store n. Vi ser at $\hat{\eta}$ konvergerer hurtigst for uniform- og normalfordeling, men for laplacefordelingen konvergerer $\hat{m}$ og $\hat{\tau}$ hurtigere. Dette stemmer helt overens med vores betragtninger fra histogrammerne ovenfor.

Vi kan altså konkludere at der ikke er en enkelt estimator der er den bedste for alle fordelinger, men at man kan have grund til at foretrække en estimator frem for de andre ved bestemte fordelinger. Eksempelvis kan man have grund til at foretrække $\hat{\eta}$ for uniform- og normalfordelinger, og man kan have grund til at foretrække de to andre estimatorer for laplacefordelinger. Vi bemærker dog, at for meget store n vil alle estimatorer være rimelig gode estimatorer for μ for vores valgte fordelinger.

OPG 6

Vi vil i det følgende fokusere på "Tid", "Gruppe" og "Ben_styrke", og vores plots vil derfor bruges til at få overblik over disse variable.



I forhold til histogrammet, lægger vi især mærke til, hvordan folks benstyrke efter 6 og 12 måneder generelt ligger højere end for folk i måned 0. Vi ser blandt andet, at antallet af folk, der har benstyrke over 150 kg, øges markant fra baseline til 12 måneder efter. Dette gælder især for "intervention" gruppen, men også for kontrol gruppen, dog i en mindre markant form.

I forhold til vores boxplot, skal der især lægges mærke til hvordan intervention gruppen "overhaler" kontrol gruppen, hvor deres masse rykkes højere. Bemærk blandt andet, at medianen for begge grupper er meget tæt på hinanden ved baseline, med ved 6. og 12. måned ligger medianen for interventionsgruppen højere end for kontrolgruppen.

I de følgende to tabeller vil vi stadigvæk fokusere på sammenhænge mellem benstyrke, gruppe og tid.

```
## # A tibble: 6 x 5
## # Groups:   Gruppe [2]
##   Gruppe   Tid  mean    sd median
##   <chr>  <int> <dbl> <dbl>  <dbl>
## 1 I          0  87.4  31.9     80
## 2 I          6 112.   37.4    110
## 3 I         12 116.   37.3    110
## 4 k          0  90    31.6     80
## 5 k          6  97.5  34.7     90
## 6 k         12 102.   35.7    100
```

```
## # A tibble: 4 x 7
##   term        estimate std_error statistic p_value lower_ci upper_ci
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
## 1 intercept     89.9       3.12     28.8     0        83.8    96.0
## 2 Tid            2.53      0.437     5.79    0         1.67    3.39
## 3 Gruppe: k      0.419     4.43      0.095   0.925    -8.28    9.12
## 4 Tid:Gruppek   -1.48      0.617    -2.40    0.017    -2.69   -0.268
```

I den første tabel har vi angivet nogle hyppigt brugte stikprøvestørrelser. Vi skal her ligesom ved boxplottet i første tabel, lægges mærke til hvordan middelværdien starter lavere hos interventionsgruppen, men allerede efter 6 måneder har de en markant større benstyrke end kontrol gruppen. Dog er der ca. samme stigning i middelværdi fra måned 6 til 12 for både interventionsgruppen og kontrolgruppen. Dermed kan man formode, at det har stor effekt på hvordan man hurtigt får sin benstyrke tilbage, hvorvidt man har været til "styrketræning" med fysisk træner eller om man selv har stået for det.

I vores anden tabel ser vi på en linæer model. Her ser vi, som vi også har observeret før, at benstyrken ved baseline for kontrolgruppen er større end intervention gruppen. Det ses også, som formodet, at vores interventionsgruppe har en hurtigere forøgelse af styrke i benene end kontrolgruppen. Bemærk, at vi i modellen betragter "Tid" som en numerisk variabel.

Ud fra disse plots og tabeller kan man derfor have den formodning, at gruppe kan have indflydelse på, hvordan genoptræningen af benene går.

Det skal bemærkes at der i udarbejdelsen af histogram og boxplot ikke er fjernet værdier for de "patienter/observationer", der ikke har fuldt "forsøget" hele vejen til mål. Altså vil der være flere observationer for til 0 ift tid 6 og 12.

OPG 7

Bemærk de observerede værdier for gruppe "I" og "k" til tid 0 er hhv. 87.41 og 90.00.

Vi vælger at lave ikke-parametrisk bootstrap og angive standard error konfidensintervallet.

For gruppe I er konfidensintervallerne ved baseline.

```
set.seed(100)
bootdataI <- pactdata0_small %>%
  filter(Gruppe=="I") %>%
  rep_sample_n(size=nrow(pactdata0_small %>% filter(Gruppe=="I")),
               replace=TRUE,reps=1000) %>%
  group_by(replicate) %>%
  summarize(mean=mean(Ben_styrke))

get_confidence_interval(bootdataI,level=0.95,
                        type="se",point_estimate = mean_I_obs)
```

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1     81.4     93.4
```

For gruppe k er konfidensintervallerne ved baseline vist nedenfor. I koden har vi blot ændret filteret fra "I" til "k", og i standard error konfidensintervallet bruges den observerede værdi for "k".

```
## # A tibble: 1 x 2
##   lower_ci upper_ci
##      <dbl>    <dbl>
## 1     83.9     96.1
```

Konklusionen er da, (som beskrevet i MDive s.280) at hvis vi gentager vores sampling mange gange, så vil 95% af vores konstruerede konfidensintervaller indeholde den sande middelværdi for benstyrken, hvis man er i henholdsvis kontrol- eller interventionsgruppen. I dette tilfælde har vi at konfidensintervallet tilhørende kontrolgruppen er "højere/forskudt mod større benstyrke" end konfidensintervallet for interventionsgruppen. Dette betyder dog ikke at den sande middelværdi for kontrolgruppen nødvendigvis er højere end for interventionsgruppen.

Det kan især være svært at sige noget da der også er overlap mellem de to konfidensintervaller, så iforhold til hinanden kan det være svært at konkludere noget.

OPG 8

Vi udfører en test for hypotesen om, at fordelingen af "Ben_styrke" er ens for interventionsgruppen og kontrolgruppen efter interventionsperioden. Lad Y_i betegne folks "Ben_styrke". Lad P_I være fordelingen af Y_i, der er i gruppe I, og lad P_k være fordelingen af Y_i, som er i gruppe k. Under nulhypotesen er H_0: $P_I = P_k$, og dermed er $\mu_{P_I} - \mu_{P_k} = 0$, hvor μ_{P_I} og μ_{P_k} er middelværdierne for hhv. P_I og P_k. Dermed vil

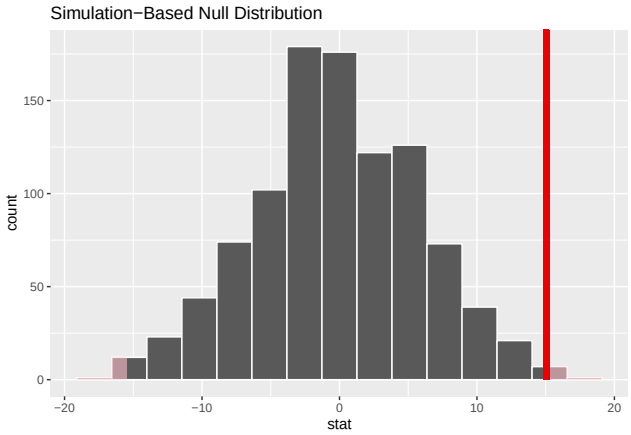
$$\hat{\mu}_{P_I} - \hat{\mu}_{P_k}$$

være en god stikprøvefunktion. Vi tester for "independence", da vi tester, om "Ben_styrke" er uafhængig fra "grp". Bemærk, at vi anvender permutationssampling, da under H_0 er Y_i-erne exchangable.

```r
data <- pact_data %>%
  select(Tid,Gruppe,Ben_styrke) %>%
  filter(Tid==6) %>%
  mutate(Ben_styrke=as.double(Ben_styrke))

set.seed(100)
null_distribution <- data %>%
  specify(formula = Ben_styrke~Gruppe) %>%
  hypothesize(null="independence") %>%
  generate(reps=1000,type="permute")%>%
  calculate(stat="diff in means",order=c("I","k"))

obs_diff <- data %>%
  specify(formula = Ben_styrke~Gruppe) %>%
  calculate(stat="diff in means",order=c("I","k"))
obs_diff
```

```
## Response: Ben_styrke (numeric)
## Explanatory: Gruppe (factor)
## # A tibble: 1 x 1
##    stat
##   <dbl>
## 1  15.0
```

```r
visualize(null_distribution)+
  shade_p_value(obs_stat = obs_diff,direction = "both")
```




```r
null_distribution %>%
  get_p_value(obs_stat = obs_diff,direction="both")
```

```
## # A tibble: 1 x 1
##   p_value
##     <dbl>
## 1    0.01
```

Vi får en p-værdi på 0.01, og vi afviser H_0 hypotesen med signifikansniveau 0.95. Vi har her brugt "both" til beregning af p-værdien, da under vores alternative hypotese H_A: $P_I \neq P_k$ er $\mu_{P_I} - \mu_{P_k} \neq 0$.

OPG 9

Ved indlæsning af det nye data "statmet_base" har vi valgt at ændre dataen direkte fra filerne ved at finde de slåfejlene ved brug af "anti.join()" og derefter "joined" dem med "left.join()".

```
##   grp gender Bs0 Bs6
## 1   I      K 140 200
## 2   I      K 130 140
## 3   I      M 130 150
## 4   I      K 150 150
## 5   I      K  40  50
## 6   I      K  80  90
```

Vi bruger formelsyntaks "$y - y_0 \sim$ grp*gender" til vores parametrisering. Koefficienterne for vores lineære model vises nedenfor.

```
##  (Intercept)         grpk      genderM grpk:genderM
##     18.83333    -12.01515     19.50000    -16.31818
```

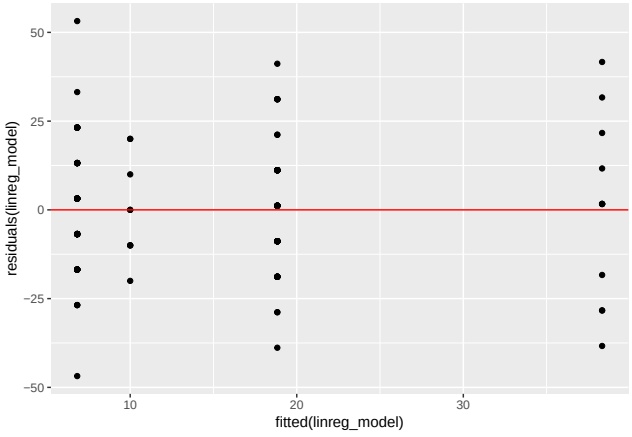
I denne konkrete parametrisering skal værdierne forstås som, at en kvinde i intervention gruppe har en forventet styrkeforøgelse på 18.833. Hvis hun nu er fra kontrolgruppen, estimeres hendes styrkeforøgelse på 18.833-12.015. Hvis vi kigger på en mand i interventionsgruppen, skal vi se på interceptet + genderM altså 18.833+19.50. For en mand fra kontrolgruppen, istedet for interventionsgruppen, ses en vekselvirkning, hvor Benstyrken er givet ved 18.833+19.50-16.318.

Hvis vi ser på den modelmatrix vi har brugt (vi har brugt "head()")

```
##   (Intercept) grpk genderM grpk:genderM
## 1           1    0       0            0
## 2           1    0       0            0
## 3           1    0       1            0
## 4           1    0       0            0
## 5           1    0       0            0
## 6           1    0       0            0
```

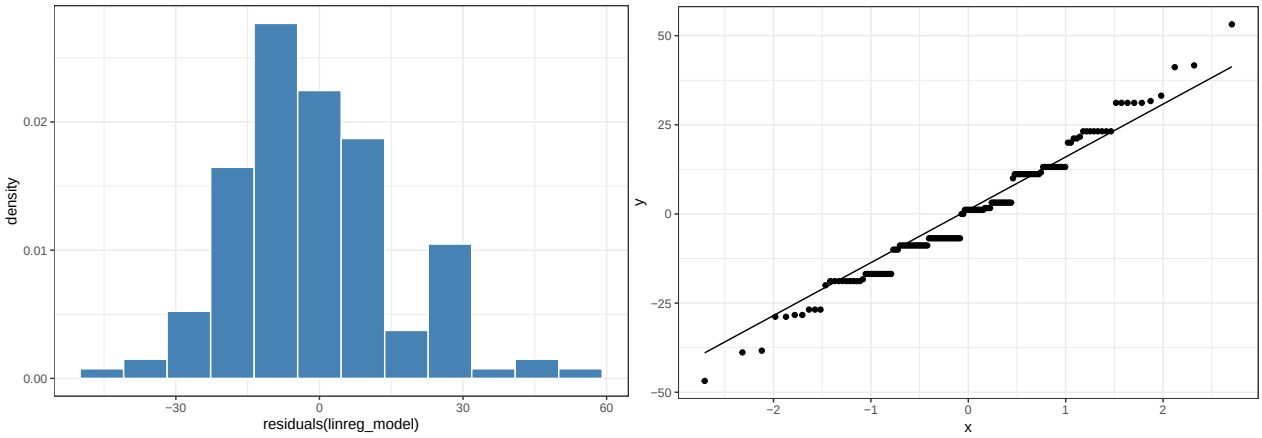
fungerer det sådan, at hvis vi ønsker at estimere den forventede benstyrke givet en kombination af gruppe og køn, skal vi blot addere de relevante parametre.

Vi undersøger nu, om vores lineære model er rimelig for vores datasæt. Til dette formål undersøger vi residualplottet.



Det ligner, at middelværdien for støjen er 0, datapunkterne fordeler sig symmetrisk omkring 0, og variansen er omtrent det samme for alle punkter. Derfor ser det ud til, at en lineær model er rimelig for datasættet.

Vi vil nu vurdere, om støjen for vores lineære models er normalfordelt .



Vi vurderer ud fra histogrammet, at residualerne er nogenlunde normalfordelt, og der ser ud til at være en god lineær sammenhæng i qq-plottet. Det er derfor rimeligt at tro, at $\epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$. Vi kan derfor beregne konfidensintervaller for koefficienterne for vores lineære model, β, ved

```
confint(linreg_model)
```

```
##                   2.5 %      97.5 %
## (Intercept)   14.386771 23.27989583
## grpk         -18.158961 -5.87134252
## genderM        8.608191 30.39180922
## grpk:genderM -32.701672  0.06530883
```

Hvis vi ikke havde troet på, at $\epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$, kan vi udføre resampling fra lineær regressionsmodel, og den vej igennem bestemme konfidensintervallerne. Vi viser nedenfor percentilintervallet, standard error intervallet og det simple bootstrap interval for parametrene ved resampling fra lineær regressisonsmodel.

```
set.seed(100)
B <- 1000
n <- length(residuals(linreg_model))
X <- model.matrix(linreg_model)

my_boot <- tibble(residuals = residuals(linreg_model)) %>%
  rep_sample_n(size = n, replace = TRUE, reps = B) %>%
  mutate(y = fitted(linreg_model) + residuals) %>%
  summarize(estimate = lm.fit(X, y) %>% coef()) %>%
  mutate(term = names(coef(linreg_model)))

my_res <- my_boot %>%
  group_by(term) %>%
  summarize(q_low = quantile(estimate, 0.025)
            , q_upp = quantile(estimate, 0.975)
            , se = sd(estimate))

my_res <- mutate(my_res,
                 beta_hat = c(coef(linreg_model)[1],
                              coef(linreg_model)[3],
                              coef(linreg_model)[2],
                              coef(linreg_model)[4]),
                 q_low_se = beta_hat - 1.96 * se,
                 q_upp_se = beta_hat + 1.96 * se,
                 simp_low = 2*beta_hat - q_upp,
```

```
                simp_upp = 2*beta_hat - q_low) %>%
  select(-se,-beta_hat)
my_res
```

```
## # A tibble: 4 x 7
##   term          q_low  q_upp q_low_se q_upp_se simp_low simp_upp
##   <chr>         <dbl>  <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)   14.4  23.2      14.5   23.2       14.5     23.2
## 2 genderM        8.52 30.1       8.68  30.3        8.94    30.5
## 3 grpk         -17.6  -6.34    -17.7   -6.31     -17.7     -6.39
## 4 grpk:genderM -31.6  -0.252   -32.6   -0.0794   -32.4     -1.08
```

Det ses, at konfidensintervallerne her ligner dem fra før, hvilket bekræfter vores tidligere formodning om, at støjen er normalfordelt.

OPG 10

Vores nul-hypotese er, at interventionens effekt er ens for de to biologiske køn. Under H_0 skal ændringen i benstyrken for folk i interventionsgruppen minus ændringen i benstyrken for folk i kontrolgruppen være ens for de to køn. Vi kan tænke effekten af interventionen som

$$\text{effekt af intervention} = y_I - y_{0,I} - (y_k - y_{0,k}),$$

for begge køn, hvor $y_I - y_{0,I}$ er ændringen i benstyrken for folk i interventionsgruppen, og $y_k - y_{0,k}$ er ændringen i benstyrken for folk i kontrolgruppen. Vi benytter formelsyntaks "~grp+gender" til at parametrisere denne model.

```
##
## Call:
## lm(formula = Bs6 - Bs0 ~ grp + gender, data = new_data)
##
## Coefficients:
## (Intercept)         grpk      genderM
##       20.04       -14.31        12.29
```

Se, at en kvinde og en mand i interventionsgruppen vil have en ændring i benstyrke på hhv. 20.04 kg og $20.04 + 12.29 = 32.33$ kg. For begge køn ville deres ændring i benstyrke have været 14.31 kg lavere, havde de været i kontrolgruppen, hvorfor effekten af interventionen er ens for begge køn, nemlig på 14.31 kg.

Den alternative hypotese er, at effekten af interventionen er forskellig for de to køn. Dette opnås ved at tillade en vekselvirkning i denne model. Vi bruger derfor formelsyntaks "~grp*gender" og får følgende model.

```
##
## Call:
## lm(formula = Bs6 - Bs0 ~ grp * gender, data = new_data)
##
## Coefficients:
##  (Intercept)          grpk       genderM  grpk:genderM
##        18.83        -12.02         19.50        -16.32
```

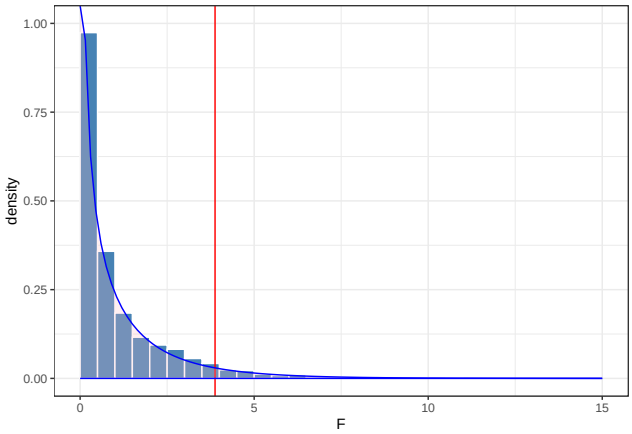
Vi får derfor følgende.

```
X0 <- model.matrix(lm_0)
n <- nrow(X0)
Bs_obs <- new_data %>%
  select(Bs0,Bs6) %>%
  na.omit()
X <- model.matrix(lm_A)
```

```r
set.seed(100)
B <- 1000
my_boot <- tibble(residuals = residuals(lm_0)) %>%
  rep_sample_n(size = n, replace = TRUE, reps = B) %>%
  mutate(y = fitted(lm_0) + residuals)

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

#observerede værdier og p-værdi
F_obs <- F_test(lm.fit(X0, Bs_obs$Bs6-Bs_obs$Bs0 ),lm.fit(X, Bs_obs$Bs6-Bs_obs$Bs0))

p_value <- sum(my_res$F > F_obs) / B

c(Observerede_F=F_obs,p_værdi=p_value)
```

```
## Observerede_F       p_værdi
##      3.876223      0.054000
```

```r
#plot
ggplot(data = my_res) +
  geom_histogram(aes(x = F, y = ..density..),
                 color = "white",
                 fill = "steelblue",
                 boundary = 0,
                 binwidth = 0.5) +
  labs(x = "F") +
  theme_bw() +
  geom_vline(xintercept = F_obs, color = "red") +
  stat_function(fun = df,
                args = list(df1 = 4 - 3 , df2 = 114 - 4),
                geom = "area",
                fill = "pink",
                color = "blue",
                alpha = 0.25,
                xlim = c(0., 15)) +
  lims(y = c(0, 1))
```



Da vi får en p-værdi på 0.054, kan vi ikke afvise nulhypotesen om at middelværdivektoren kan antages at ligge i $L_{grp} + L_{gender}$ med signifikansniveau 95%. Med andre ord, vi kan ikke afvise, at effekten af interventionen er ens for de to biologiske køn.

Lille reflektion over data og setup'et:

Der er to bagvedlæggende effekter (den personlige træners effekt og mænds VS kvinders "egentræningslyst") vi ikke bliver kloge på, på trods af vores test. Det skyldes setup'et af testen.

Måske ville det være mere gavnligt hvis vi faktisk vidste hvor mange gange en mand eller kvinde i kontrolgruppen har trænet. Vi ved faktisk kun at de har fået af vide, at det er meget vigtigt at træne. Måske lægger der også noget i hvorvidt at mænd eller kvinder er bedre til at træne på egenhånd. Altså kunne det have været spændende, hvis vi have haft lidt mere info fra vores kontrolgruppe. Lige nu kan vi nemlig ikke sige om det skyldes den "personlige træners effekt" eller om det skyldes at mænd eller kvinder er dårligere til at træne af sig selv. Dette er selvfølgelig kun spændende ift en videre vurdering af hvorvidt det er vigtigt med en personlig træner eller om det handler om motivationen. Blot for bedre at kunne besvare hvad der egentlig har en effekt på benstyrken og hvad der ikke har den store effekt.

Reflektionspapir

Vi har alle i gruppen bidraget ligelidt. Visse har haft en større finger i enkelte opgaver, men det handler i større grad om at fikse små problemer i de opgaver, og dermed har det haft ligeså stor indflydelse på det færdige produkt som at løse andre opgaver.

Vi vil derfor sige at alle skal bedømmes ligelidt da den er lavet i sammenarbejde.