

# Arbeidskrav2

Håvard Crantz Lorentzen, Hermann Moen, Margit Dahl Sørensen, Jacob Mollatt

28 9 2021

```
library(tidyverse)
library(readxl)
library(ggplot2)

lactate <- read_excel("data til laktatprofil.xlsx", na="NA")

lactate1 <- lactate %>%
  # Select columns needed for analysis
  select(FP, time, lac.75:lac.300) %>%
  # Only one participant and time-point
  filter(time == "pre", FP == 6) %>%
  # Pivot to long format data using the lactate columns
  pivot_longer(names_to = "watt",
               values_to = "lactate",
               names_prefix = "lac.",
               names_transform = list(watt = as.numeric),
               cols = lac.75:lac.300) %>%
  # Remove NA (missing) values to avoid warning/error messages.
  filter(!is.na(lactate))

# fit "straight line" model
m1 <- lm(lactate ~ watt, data = lactate1)

# fit second degree polynomial
m2 <- lm(lactate ~ poly(watt, 2, raw = TRUE), data = lactate1)

# fit third degree polynomial
m3 <- lm(lactate ~ poly(watt, 3, raw = TRUE), data = lactate1)

# fit forth degree polynomial
m4 <- lm(lactate ~ poly(watt, 4, raw = TRUE), data = lactate1)

# Store all residuals as new variables
lactate1$resid.m1 <- resid(m1)
lactate1$resid.m2 <- resid(m2)
lactate1$resid.m3 <- resid(m3)
lactate1$resid.m4 <- resid(m4)

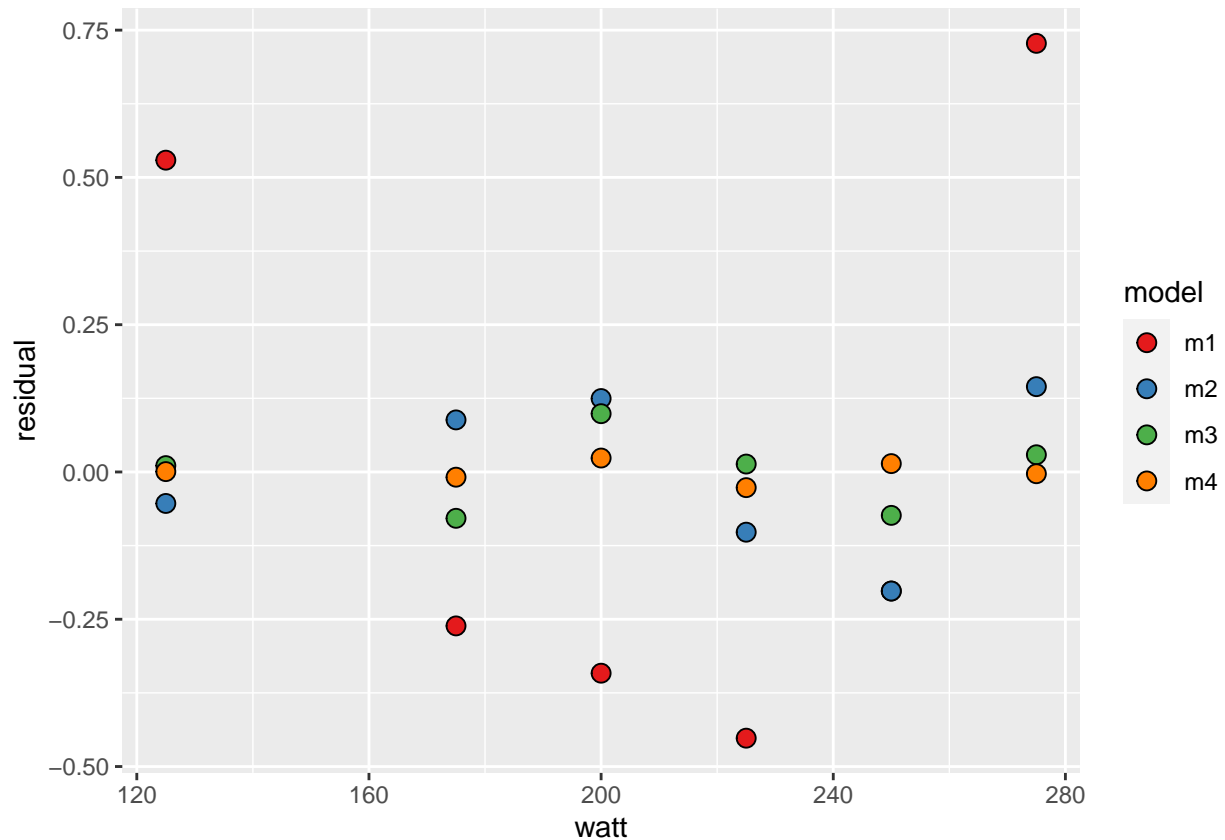
lactate1 %>%
  # gather all the data from the models
  pivot_longer(names_to = "model",
               values_to = "residual",
```

```

names_prefix = "resid.",
names_transform = list(residual = as.numeric),
cols = resid.m1:resid.m4) %>%
# Plot values with the observed watt on x axis and residual values at the y
ggplot(aes(watt, residual, fill = model)) + geom_point(shape = 21, size = 3) +

# To set the same colors/fills as above we use scale fill manual
scale_fill_manual(values = c("#e41a1c", "#377eb8", "#4daf4a", "#ff7f00"))

```



```

#new data data frame
ndf <- data.frame(watt = seq(from = 125, to = 275, by = 0.1)) # high resolution, we can find the nearest

ndf$predictions <- predict(m4, newdata = ndf)

lactate_threshold4 <- ndf %>%
  filter(abs(predictions - 4) == min(abs(predictions - 4)))

lactate_threshold2 <- ndf %>%
  filter(abs(predictions - 2) == min(abs(predictions - 2)))

lactate <- read_excel("data til laktatprofil.xlsx", na="NA")

lactate2 <- lactate %>%
  select(FP, time, lac.75:lac.300) %>%
  pivot_longer(names_to = "watt",

```

```

      values_to = "lactate",
      names_prefix = "lac.",
      names_transform = list(watt = as.numeric),
      cols = lac.75:lac.300) %>%
filter(FP == 6,
      time == "pre",
      !is.na(lactate)) %>% # Remove NA values
print()

```

```

## # A tibble: 6 x 4
##   FP time   watt lactate
##   <dbl> <chr> <dbl>   <dbl>
## 1     6 pre    125     1.48
## 2     6 pre    175     1.67
## 3     6 pre    200     2.08
## 4     6 pre    225     2.46
## 5     6 pre    250     3.2
## 6     6 pre    275     4.62

```

```

# Fit the model
model <- lm(lactate ~ watt + I(watt^2) + I(watt^3), data = lactate2)

# Predict lactate values over all observed watt values
# calculate the smallest distance from the fixed lactate value

new_data <- data.frame(watt = seq(from = min(lactate2$watt), to = max(lactate2$watt), by = 0.1))

new_data$dist <- abs(predict(model, newdata = new_data) - 4)

# Find the smallest value of predicted - fixed lactate value
new_data %>%
  filter(dist == min(dist)) # Where the dist value equals the minimum dist value

```

```

##   watt      dist
## 1 265.1 0.002391221

```

```

lt <- function(data) {

  # Fit a 3 degree polynomial model
  m <- lm(lactate ~ watt + I(watt^2) + I(watt^3), data = data)

  # Store a data frame with exercise intensities
  new_data <- data.frame(watt = seq(from = min(data$watt), to = max(data$watt), by = 0.01))

  # Predict using the new data, predicting lactate values at each
  new_data$pred <- predict(m, newdata = new_data)

  # calculate deviation from the lactate value of interest
  new_data$watt.4mmol <- abs(new_data$pred - 4)
  new_data$watt.2mmol <- abs(new_data$pred - 2)

```

```

# Create a results data frame
results <- data.frame(watt.4mmol = new_data[which.min(new_data$watt.4mmol),1], watt.2mmol = new_data[

# Return the data frame
return(results)

}

lactate3 <- lactate %>%
  select(FP, time, lac.75:lac.300) %>%
  pivot_longer(names_to = "watt",
               values_to = "lactate",
               names_prefix = "lac.",
               names_transform = list(watt = as.numeric),
               cols = lac.75:lac.300) %>%
  filter(!is.na(lactate)) %>% # Remove NA values
  group_by(time, FP) %>%
  mutate(n = n()) %>%
  filter(n >= 4) %>%
  # Use group_modify to apply the function to all participants per time-point (and group)
  group_modify(~ lt(.)) %>%
  print()

```

```

## # A tibble: 12 x 4
## # Groups:   time, FP [12]
##   time    FP watt.4mmol watt.2mmol
##   <chr> <dbl>      <dbl>      <dbl>
## 1 post     1      172.        75
## 2 post     2      297.       239.
## 3 post     3      135.       96.8
## 4 post     4      169.       75.7
## 5 post     5      205.       125
## 6 post     6      279.       197.
## 7 post     7      130.        75
## 8 pre      1      126.       78.3
## 9 pre      2      275.       215.
## 10 pre     4      157.       119.
## 11 pre     5      241.       144.
## 12 pre     6      265.       201.

```

```

lactate4 <- lactate3 %>%
  group_by(FP) %>%
  mutate(n = n()) %>%
  filter(n == 2) %>%
  pivot_wider(names_from = time, values_from = c(watt.4mmol, watt.2mmol)) %>%
  print()

```

```

## # A tibble: 5 x 6
## # Groups:   FP [5]
##   FP      n watt.4mmol_post watt.4mmol_pre watt.2mmol_post watt.2mmol_pre
##   <dbl> <int>          <dbl>          <dbl>          <dbl>          <dbl>

```

## 1	1	2	172.	126.	75	78.3
## 2	2	2	297.	275	239.	215.
## 3	4	2	169.	157.	75.7	119.
## 4	5	2	205.	241.	125	144.
## 5	6	2	279.	265.	197.	201.

```
TE4 <- lactate4 %>%
  mutate(diff = watt.4mmol_post - watt.4mmol_pre) %>%
  ungroup() %>%
  # Change/difference score
  summarise(s = sd(diff, na.rm = TRUE), # Summarize to calculate sd, and...
            m = mean(c(watt.4mmol_post, watt.4mmol_pre)), # mean
            te = s / sqrt(2), 1, # the typical error.
            cv = 100 * (te / m), 1,
            L = qt(0.975, 4) * s) %>% # Calculate as a percentage of the mean
  print()
```

```
## # A tibble: 1 x 6
##       s      m    te   '1'    cv    L
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  29.7  219.  21.0     1  9.59  82.4
```

```
TE2 <- lactate4 %>%
  mutate(diff = watt.2mmol_post - watt.2mmol_pre) %>%
  ungroup() %>% # Change/difference score
  summarise(s = sd(diff), # Summarize to calculate sd, and...
            m = mean(c(watt.2mmol_post, watt.2mmol_pre)), # mean
            te = s / sqrt(2), 1, # the typical error.
            cv = 100 * (te / m), 1,
            L = qt(0.975, 4) * s) %>% # Calculate as a percentage of the mean
  print()
```

```
## # A tibble: 1 x 6
##       s      m    te   '1'    cv    L
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  24.5  147.  17.3     1  11.8  68.1
```

```
cv4 <- round(TE4$cv, 2)
cv2 <- round(TE2$cv, 2)
```

```
ladder <- data.frame(dist = c(1.0, 25.5, 63.0, 111.0, 165.5, 241.0, 332.5, 460.5, 540.5, 637.5),
                     mw = c(1000, 900, 800,
                           700, 600, 500,
                           400, 300, 250,
                           200))

# Create a new data frame of unknowns
unknown <- data.frame(dist = c(95.5, 303.8, 413.8, 95.5, 300.0, 409.5))
```

```
#Fit the model
cal <- lm(log(mw) ~ dist, data = ladder)
```

```
# Check model performance, R^2 should be ~ 1.
summary(cal)
```

```
##
## Call:
## lm(formula = log(mw) ~ dist, data = ladder)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.038846 -0.028023 -0.003263  0.018531  0.064677
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.846e+00  1.732e-02  395.28 < 2e-16 ***
## dist        -2.476e-03  5.167e-05  -47.92 3.98e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.035 on 8 degrees of freedom
## Multiple R-squared:  0.9965, Adjusted R-squared:  0.9961
## F-statistic: 2297 on 1 and 8 DF, p-value: 3.975e-11
```

```
preds <- exp(predict(cal, newdata = unknown))
```

```
preds
```

```
##      1      2      3      4      5      6
## 741.8119 442.9002 337.3038 741.8119 447.0870 340.9142
```

```
round(preds,2)
```

```
##      1      2      3      4      5      6
## 741.81 442.90 337.30 741.81 447.09 340.91
```

```
library(exscidata); library(tidyverse); library(dplyr)
data("hypertrophy")
```

```
Testo_DXA <- hypertrophy %>%
  select(PARTICIPANT, DXA_LBM_T1:DXA_LBM_T3, TESTOSTERONE_T1:TESTOSTERONE_T3) %>%
  rowwise() %>%
  mutate(testoavg. = mean(c(TESTOSTERONE_T1, TESTOSTERONE_T2, TESTOSTERONE_T3), na.rm=TRUE)) %>%

  ungroup() %>%
  mutate(DXA_change = DXA_LBM_T3 - DXA_LBM_T1) %>%
  group_by() %>%
  select(PARTICIPANT, testoavg., DXA_change) %>%
  filter(! is.nan(testoavg.))
```

```
# Fit the model
```

```
TDXA <- lm(DXA_change ~ testoavg., data = Testo_DXA)
```

```
summary(TDXA) %>%  
  print()
```

```
##  
## Call:  
## lm(formula = DXA_change ~ testoavg., data = Testo_DXA)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max   
## -3.2798 -0.8366 -0.1026  0.8446  3.2262   
##  
## Coefficients:  
##              Estimate Std. Error t value Pr(>|t|)      
## (Intercept)  3.975025   0.790774   5.027 3.88e-05 ***  
## testoavg.    -0.004339   0.001670  -2.599  0.0157 *    
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## Residual standard error: 1.471 on 24 degrees of freedom  
## Multiple R-squared:  0.2196, Adjusted R-squared:  0.1871   
## F-statistic: 6.755 on 1 and 24 DF,  p-value: 0.01574
```

```
#Reg
```

```
library(broom)
```

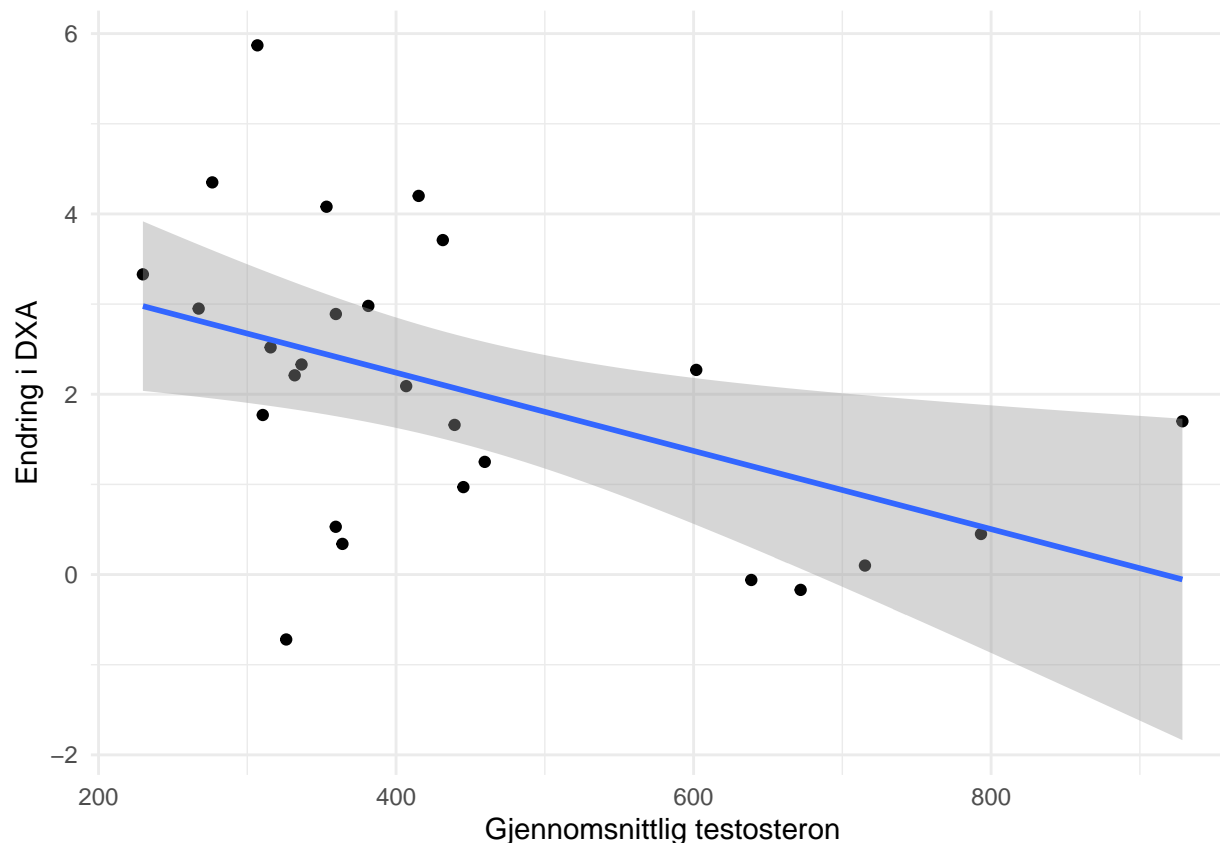
```
tidy(TDXA)
```

```
## # A tibble: 2 x 5  
##   term          estimate std.error statistic  p.value  
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>  
## 1 (Intercept)  3.98         0.791      5.03 0.0000388  
## 2 testoavg.    -0.00434     0.00167    -2.60 0.0157
```

```
#plotte figur
```

```
Testo_DXA %>%  
  ggplot(aes(testoavg., DXA_change)) + geom_point(shape = 19) + theme_minimal() + geom_smooth(method =
```

```
## 'geom_smooth()' using formula 'y ~ x'
```



## Resultater

Typefeil er ved 4mmol laktat er 9.59% og ved 2mmol laktat er den 11.8% Predikerte størrelser på DNA-fragmentene er 741.81, 442.9, 337.3, 741.81, 447.09, 340.91.

I datasettet hypertrophy finner vi flere forskjellige variabler tilknyttet muskelvekst. Blant annet finnes det en måling av deltakernes testosteronverdier ved tre tidspunkt fra intervensjonen, og mager kroppsmasse målt i DXA ved tre tidspunkter. Vi ville se på en eventuell korrelasjon mellom den gjennomsnittlige testosteronverdien og endringen i mager kroppsmasse fra test 1 til 3.

### ##Diskusjon

Testosteron er et potent anabolsk hormon, som stimulerer muskel-proteinsyntesen (Florini 1970; Mauras et al. 1998), og forbedrer intramuskulært opptak av aminosyrer (Vingren et al. 2010) som sammen resulterer i en forbedring i total protein balanse (Ferrando et al. 2002). Basert på dette ser vi for oss at testosteron nivået, vil ha en positiv innvirkning på endringen i mager kroppsmasse.

For hver økning i testosteron vil endringen i mager kroppsmasse endres med  $-0,00434 \pm 0,00167$ std.. Dette betyr i praksis at høyere testosteron ga mindre endring i mager kroppsmasse  $p=0,0157$ . P-verdien forteller oss hvor overraskende utfallet er hvis null hypotesen er sann. Altså vil det være et veldig overraskende funn hvis null hypotesen var sann, og vi kan kalle funnet statistisk signifikant  $p \leq 0,05$ , når signifikansgrensen er satt til 0,05 Std. er en forkortelse for standard avvik, og viser oss det gjennomsnittlige avviket fra gjennomsnittet i målingen.

Vi så for oss en sammenheng mellom gjennomsnittlig testosteron verdi, og endringen i mager kroppsmasse. Vi tenkte korrelasjonen skulle være positiv, altså mere testosteron, ville resultere i større økning i mager masse. Dette er basert på litteratur nevnt over. Det viste seg at for hver verdi med testosteron ville den magre kroppsmassen endres med  $-0,00434$ ,  $p=0,0157$ .



Vi ønsker å vite om dette resultatet er noe vi kan stole på, og dermed kunne late som at funnet her er en sannhet. Vi må derfor kunne si noe om hvor reliabelt det er, og hvor sannsynlig det er at forskjellen i dataen skyldes sammenhengen mellom variablene og ikke støy. Standard feil er et estimat på reliabiliteten til dataene. Reliabiliteten kan forklares som hvor stor verdien til den tilfeldige variasjonen i et mål vil være fra gang til gang, når målet er gjort under like forhold (Hopkins 2000). Standardfeilen beregnes ved hjelp av «within subject standard deviation». Dette datasettet ga oss en standardfeil i målingene på 0,00167.

Nå vet vi hvor mye som kan forventes skyldes støy, og vi ønsker nå å vite om funnet vårt om korrelasjonen er noe vi kan late som er sant. Til dette vil vi bruke p-verdi. P-verdien forteller oss hvor overraskende utfallet er hvis null hypotesen er sann. Altså vil det være et veldig overraskende funn hvis null hypotesen var sann, og vi kan kalle funnet statistisk signifikant  $p \leq 0,05$ , når signifikans grensen er satt til 0,05. Det betyr at i denne situasjonen kan vi late som at funnet er sant.

Altså vet vi nå at det er et veldig overraskende funn om null hypotesen er sann, som i praksis gjør at vi kan late som at funnet er sant, da den som nevnt var under grenseverdien for statistisk signifikans. Vi kan også se på om null hypotesen skal avises eller følges ved hjelp av t-verdi. T-verdien er enkelt sagt forholdet mellom verdien og standardfeilen. Jo høyere verdien er, jo sikrere kan vi være på at null hypotesen kan avvises. Dette gjøres i hypotesetesting i students t-test. Et akseptabelt resultat i t-verdi er 2, eller -2. I dette tilfellet her var t-verdien 2,60, som gir oss enda en grunn til å late som at funnet vil være sant i alle tilfeller.

Ferrando, A. A., M. Sheffield-Moore, C. W. Yeckel, C. Gilkison, J. Jiang, A. Achacosa, S. A. Lieberman, K. Tipton, R. R. Wolfe, and R. J. Urban. 2002. "Testosterone administration to older men improves muscle function: molecular and physiological mechanisms." *Am J Physiol Endocrinol Metab* 282 (3): E601–607.

Florini, J. R. 1970. "Effects of testosterone on qualitative pattern of protein synthesis in skeletal muscle." *Biochemistry* 9 (4): 909–12.

Hopkins, W. G. 2000. "Measures of reliability in sports medicine and science." *Sports Med* 30 (1): 1–15.

Mauras, N., V. Hayes, S. Welch, A. Rini, K. Helgeson, M. Dokler, J. D. Veldhuis, and R. J. Urban. 1998. "Testosterone deficiency in young men: marked alterations in whole body protein kinetics, strength, and adiposity." *J Clin Endocrinol Metab* 83 (6): 1886–92.

Vingren, J. L., W. J. Kraemer, N. A. Ratamess, J. M. Anderson, J. S. Volek, and C. M. Maresh. 2010. "Testosterone physiology in resistance exercise and training: the up-stream regulatory elements." *Sports Med* 40 (12): 1037–53.