En kort innføring i R og RStudio

Merk at dette bare er en kort innføring i bruk av R og RStudio, og er ikke kvalitetssikret på samme måte som en lærebok. Bruk gjerne *Applied Statistics using R* av Mehmetoglu og Mitter (2022) eller *Lær deg R* av Silje Synnøve Hermansen (2019) som støtte i arbeidet med å lære deg R.

Trondheim 9. februar 2026
Arild Blekesaune

Introduksjon til R

Introduksjon

Dette heftet er en kort innføring i bruk av R og RStudio. Det er ikke kvalitetssikret på samme måte som en lærebok. Bruk gjerne "Applied Statistics using R" av Mehmetoglu og Mitter (2022) eller "Lær deg R" av Silje Synnøve Hermansen (2019) som støtte i arbeidet med å lære R.

Forkunnskaper som forutsettes

Vi forutsetter grunnleggende ferdigheter i bruk av datamaskin. Du bør blant annet:

- kunne laste ned og installere programmer på egen datamaskin
- kunne lage mapper og mappestrukturer lokalt, og ha oversikt over filer
- kunne laste ned en fil direkte til en bestemt mappe uten å åpne den

Installere R og RStudio

R og RStudio er to programmer som er integrert i hverandre, og du starter alltid programmet R ved å åpne RStudio. R er navnet på selve programmeringsspråket og er programmet som gjør selve utregningene, men R er ikke veldig brukervennlig alene. RStudio er et "integrated development environment" (IDE) til R. Det integrerer R med en konsoll, et grafikk-vindu og en del andre nyttige ting. Det gjør det lettere å bruke R.

Du må installere R før du kan installere RStudio, og du finner lenker til begge installasjonsprogrammene på https://posit.co/download/rstudio-desktop/. Der kan du trykke på «DOWNLOAD AND INSTALL R» og da kommer du inn på den offisielle hjemmesiden til R (https://cran.rstudio.com/). Her velger du den R-versjonen som passer for din datamaskin (Windovs, macOS eller Linux). De som velger på Windows trykker «install R for the first time», og de som har Mac må selv finne ut om de skal installere R for den gamle Intel-prosessoren eller for den nye Apple silicon. De som bruker Linux vet sannsynligvis selv hva de skal velge.

Etter å ha installert R, må Windows-brukerne gå tilbake til startsiden (https://posit.co/download/rstudio-desktop/), og trykke på «DOWNLOAD RSTUDIO FOR WINDOWS». Macbrukerne må rulle litt nedover på siden til de finner lenken for macOS.

Viktig: du må installere R før du installerer RStudio, og RStudio vil gi feilmelding hvis den ikke finner R. Hvis du har en eldre datamaskin, og du får feilmelding ved installasjon av RStudio, kan du vurdere å installere en eldre versjon av RStudio fra den samme web-siden.

Spesielt om Windows: Rtools

Hvis du bruker Windows og planlegger å kompilere R-pakker (f.eks. C/C++ eller Fortran) eller utvikle egne R-pakker, installer Rtools fra: https://cran.r-project.org/bin/windows/Rtools/.

Spesielt om Mac

R installeres normalt på macOS uten problemer. Noen kan få beskjed om også å installere XQuartz (se: https://cran.r-project.org/bin/macosx/tools/).

Spesielt om Linux

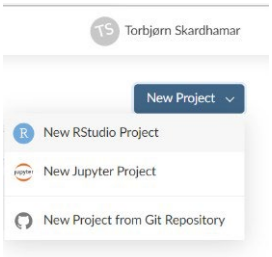
På Linux kan siste versjon av R og RStudio vanligvis installeres fra distribusjonens pakkebrønn (repository).

Spesielt om Chromebook

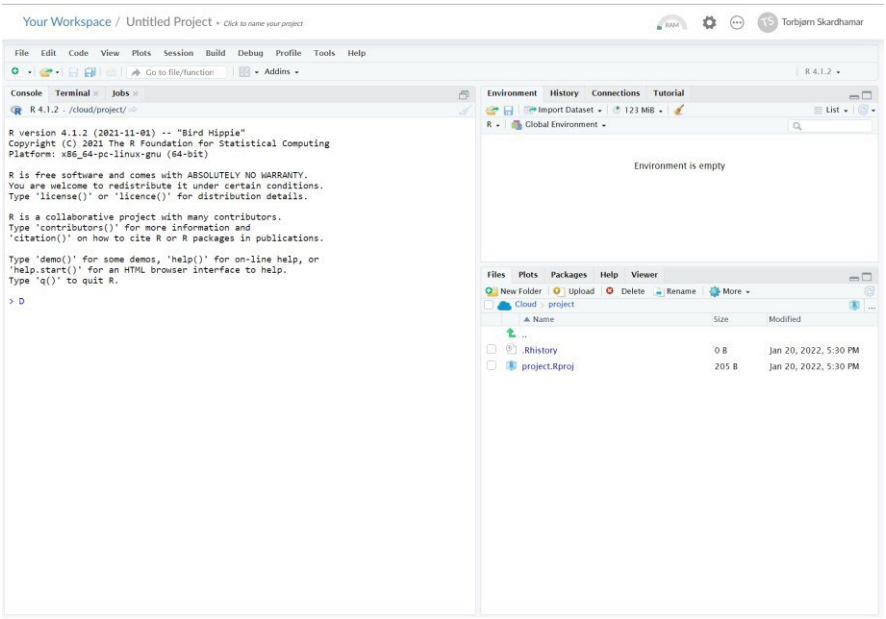
Chromebook bruker et annet operativsystem. R fungerer ikke nødvendigvis rett ut av boksen, men mange Chromebook kan kjøre Linux. Da kan du installere Linux-versjonene av R og RStudio.

Hvis lokal installasjon ikke er mulig

Dersom du har en Chromebook, en eldre maskin, lite lagringsplass, mangler administratorrettigheter eller av andre grunner ikke kan installere R og RStudio, kan du bruke R i nettleseren via Posit Cloud: https://posit.cloud/. Opprett en gratis konto («Free»). Vær oppmerksom på at du ikke har direkte tilgang til lokalt filsystem; last derfor opp nødvendige datafiler til skyen. [Skjermbilder utelatt]



Det skal nå se slik ut:



Kom i gang med R

Vinduene i RStudio

Når du åpner RStudio første gang vil du se de tre vinduene i figuren på forrige side. Hvis du trykker File, New File og R Script i menyen øverst så vil du få fram et fjerde vindu.

Øverst til venstre ser du da skript-vinduet. Det er her du skriver kommandoer. Dette er en tekstfil som du kan lagre i din egen mappa. Du kan ha flere skript åpent samtidig, og de vil da vises som faner ved siden av hverandre. Alle skriptene skal ha havn med filhale .R når du lagrer dem.

Nederst vil venstre ser du vinduet Console, der du ser resultatene fra kommandoer du kjører. Å «kjøre» en kommando kalles også å eksekvere, men jeg skal forsøke å holde meg til det norske begrepet.

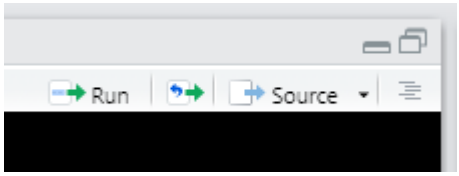
Øverst til høyre ser du vinduet Environment som viser hva som ligger i arbeidsminnet akkurat nå. Dette kan være datasett, men også andre ting. Her er det også en fane for «history» som viser nylige kommandoer du har kjørt.

Nederst til høyre vises et vindu der du kan se alt fra filene på arbeidsområdet ditt, tabeller og grafiske figurer, og hvilke pakker (tilleggsprogrammer) som du har lastet ned.

Nedtrekksmenyer?

RStudio har kun noen få nedtrekksmenyer, og du kan derfor ikke bruke pek-og-klikk når du jobber med RStudio. Alle kommandoene må skrives inn i et skript, så du er nødt til å lære deg en del grunnleggende programmeringskommandoer.

Men det er likevel noen få ting du kan klikkes på. I skriptvinduet øverst til venstre er det tre knapper for kjøring av kommandoer. Den første kjører det som er markert i skript-vinduet. Den neste kjører foregående kommandoer, og den siste «Source» kjører hele skriptet fra toppen av.



Den enkleste måten å kjøre en kommando på er å plassere markøren på linjen og trykke Ctrl+Enter (Cmd+Enter på Mac).

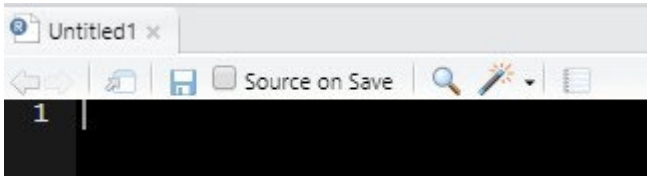
Skript

Du skal *alltid* jobbe med skript. Det er mulig å kjøre en kommando direkte fra konsollen, men ikke gjør det. Skriv alltid kommandoer i skriptet og kjør det derfra.

Alle skript bør lagres i en egen mappe med et navn som angir hva skriptet brukes til. For eksempel: hvis du har løst oppgave om tabellanalyse, så bør skriptet lagres med et navn som f.eks. tabellanalyse.R

Ta med innlesing av data og all koden som trengs for å reprodusere resultatene (omkodinger, analyser osv.) i hvert skript.

Ta vare på alle skript du skriver i løpet av kurset! Du lagrer et skript ved å trykke på save-ikonet eller via menyene fil➔ save / save as



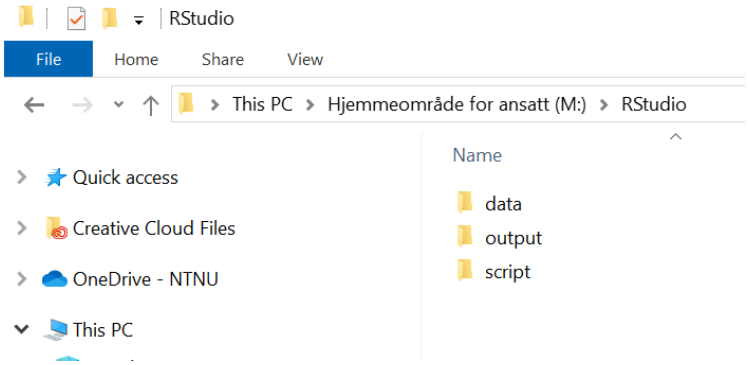
Det er mange fordeler med å bruke skript. Den viktigste er at du alltid har oversikt over hva du har gjort. Hvis alt går skeis og du ikke vet helt hva som er galt, kan du lukke datasettet og bare kjøre gjennom hele skriptet frem til dit du er usikker.

For de oppgavene vi skal gjøre her vil jeg anbefale å bare lagre de originale versjonene av de datasettene vi bruker. Det går fort å kjøre gjennom et skript, så det holder å lese inn den originale datafila. Du trenger derfor aldri å lagre datasettet når du gjør endringer: skriptet dokumenterer alt arbeid fra de originale dataene.

Filstruktur og .Projects

Det første du må gjøre er å sørge for å ha orden i datasett, skript og annet *på din egen datamaskin*. Du bør for eksempel aldri lagre filer på skrivebordet, og særlig ikke i dette kurset eller når man jobber med større prosjekter og datasett.

For dette kurset skal du ha en mappestruktur med en hovedmappe for R og tilhørende undermapper. Det spiller ingen rolle hvor på datamaskinen du legger disse mappene, men du må vite hvor det er. Lag første en mappe som heter R, og innunder denne mappen lager du tre andre mapper med navnene *data*, *output* og *skript*. Du kan ha andre mapper i tillegg ved behov. Det kan se slik ut:



Installere pakker

Det er mange ulike brukere av R, og det er mange som bidrar med å utvikle lure funksjoner og systemer for ulik bruk av programmet. Vi kaller dette for *pakker*, og de kan gjerne sammenlignes med apper på em mobiltelefon. En pakke vil igjen inneholde flere *funksjoner*, som vil si en kommando med en parentes der vi skriver hva vi vil at kommandoen skal gjøre.

Den enkleste måten å installere pakker på er å gå inn på vinduet Packages nederst til høyre i RStudio, trykke på Install og skrive inn navnet på den pakken du ønsker å installere på datamaskinen.

Til dette kurset trenger vi flere pakker. Først og fremst pakken *tidyverse*.. Dette er den mest sentrale pakken vi bruker i dette kurset. I tillegg trenger vi senere i kurset pakkene *haven*, *sjlabelled*, *summarytools, gmodels, car*, *psych, jtools*, og *sjPlot*. Disse trengs bare til noen helt konkrete ting som forklares nærmere etter hvert som vi får bruk for dem. Det er likevel lurt å installere alle disse åtte pakkene i ditt eget R-program før du prøver ut eksemplene i denne manualen.

Du kan også installere en enkelt pakke slik:

```
install.packages("tidyverse")
```

Du trenger bare å laste ned en pakke en gang på din datamaskin, og hvis du vil se alle de pakkene du allerede har installert så trykker du på menyen **Packages** i vinduet nederst til høyre.

For å bruke funksjonene i en pakke må du laste den inn i arbeidsminnet i R. I dette kurset kommer vi hovedsakelig til å bruke pakken *tidyvers*, som er en statistikkpakke som er utviklet av det samme selskapet som har laget RStudio. Hver gang du åpner R på nytt må du derfor huske å laste inn pakkene *tidyverse* med følgende kommandoer:

```
library(tidyverse)
```

Når du trenger andre pakker, laster du disse inn i R på samme måte.

Last ned data

Dette heftet vil gi den en introduksjon og grunnleggende opplæring i bruk av programmet R som vil være nødvendig når du skal skrive en bacheloroppgave i kvantitativ metode. I tillegg til dette heftet kan det være svært nyttig å benytte seg av andre ressurser og lærebøker. R-skriptene til alle analysen i pensumboka *Enhet og mangfold* (Ringdal 2024) finner du på websiden https://enhetogmangfold5.akademisk.fagbokforlaget.no/. Boka *Applied Statistics using R* av Mehmetoglu og Mitter (2022) tar også opp mange av de samme temaene som vi går igjennom i denne innføringen. Den kan benyttes som et oppslagsverk, men samtidig også være nyttig for deg som vil sette deg ytterligere inn i R. Merk at boka bruker forskjellige datasett, mens vi her kun vil bruke datasettet European Social Survey (ESS), runde 11, fra 2024. Du kan laste ned et datasett med det norske utvalget fra ESS11 ved å gå inn på https://ess.sikt.no/en/data-builder/ og logg deg inn som NTNU-student via Feide. Marker så ruten som kombinerer runde 11 og Norway, marker alle variablene i datasettet. Trykk så på Download, velg at du vil laste ned dataene i Stata-format, og lagre fila i arbeidskatalogen din. Arbeidskatalogen min ligger på windows-katalogen C:\Users\arible\OneDrive - NTNU\R\kurs, men ettersom R bruker tegnet slash (\) til andre ting enn å angi nettkataloger så må vi endre de til backslash (/).

RStudio vil koble seg opp til katalogen min når jeg kjører denne kommandoen i skriptet mitt:

```
setwd("C:/Users/arable/OneDrive - NTNU/R/kurs")
```

Og hvis du bruker Mac, kan f.eks. du skrive:

```
setwd("/Users/ditt_brukernavn/Desktop/R/kurs")
```

Hvis du vil forsikre deg om at RStudio jobber i denne katalogen så kan du finne den aktive katalogen, og hvilke filer som ligger der, med disse kommandoene:

```
getwd()
list.files()
```

For å kunne se innholdet i den nedlastede ESS-fila må den konverteres til en R-fil. Ettersom vi valgte å laste ned ESS-dataene i Stata-format, så må vi laste ned pakken *haven.* Hvis du

foretrekker å ha datasettet i en annen katalog enn arbeidskatalogen din, så kan du bruke filbehandleren din til å markere fila, kopiere den, og lime den inn i R-skriptet ditt. Da vil RStudio lime inn filnavnet med en komplett adresse til den katalogen der datasettet ligger lagret. Ettersom jeg allerede har lagret datafila *ESS11-subset.dta* i arbeidskatalogen «C:\Users\arible\OneDrive - NTNU\R\kurs», er det ikke nødvendig å legge inn hvilken katalog den ligger i, og da kan jeg aktivere pakken haven og konvertere datafila til R-format med disse to kommandoen:

```
library(haven)
ess <- read_dta("ESS11-subset.dta")
```

Nå har datasettet blitt et nytt objekt som du kan bruke i R. I vinduet Environment vil du nå se at objektet *ess* er en datamatrise med 558 variabler og 1337 enheter. Hvis du klikker på fila vil du kunne bla deg rundt i datamaterisen i skript-vinduet, og da kan du se hvordan matamaterisen er bygd opp.

Du får enda bedre informasjon om variablene i ESS11 på websida til ESS: https://ess.sikt.no/en/datafile/242aaa39-3bbb-40f5-98bf-bfb1ce53d8ef?tab=0, og du kommer også på denne sida hvis du googler «ESS11 variables».

Hvis du ønsker å se hvilke variabler som er lagt under kategorien «Media and social trust», og trykker på denne, så vil du få informasjon om spørsmålsformuleringer og kodingen av svaralternativene for de seks variablene som er klassifisert under dette temaet.

Hvis du trykker på «politics», så vil du se at noen av variablene i denne kategorien kun er spurt i ett bestemt land, mens andre variabler er registrert for alle landene. Den første av disse variablene *polintr* måler hvor interessert respondentene er i politikk, og er kodet med verdiene 1 for «Very interested», 2 for «Quite interested», 3 for «Hardly interested» og 4 for «Not at all interested». For å se svarfordelingen på dette spørsmålet i Rstudio kan vi få fram en frekvenstabell med denne kommandoen:

```
table(ess$polintr)
```

Det betyr lag en enkel tabell med variabelen *polintr* fra datasettet *ess*, og svaret vi da får er:

```
  1   2   3   4
170 506 570  91
```

Det vil si at 170 av de spurte har blitt kodet med verdien 1 (Very interested), 506 med verdien 2 (Quite interested), 570 med verdien 3 (Hardly interested), og 91 med verdien 4 (Nota at all interested).

Hvis du i heller ønsker en mer oversiktlig tabell med navn på kategoriene, og med prosentfordelinger, så kan du installerer pakken *summarytools* og kjøre disse kommandoene:

```
library(summarytools)
```

```
freq(ess$polintr)
```

```
Frequencies
ess$polintr
Label: How interested in politics
Type: Numeric (labelled)
```

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
Very interested [1]	170	12.72	12.72	12.72	12.72
Quite interested [2]	506	37.85	50.56	37.85	50.56
Hardly interested [3]	570	42.63	93.19	42.63	93.19
Not at all interested [4]	91	6.81	100.00	6.81	100.00
<NA>	0			0.00	100.00
Total	1337	100.00	100.00	100.00	100.00

Her vises også kategorinavnene, prosenter og kumulative prosenter både for variabelen uten missingverdien <NA>, og for prosenter og kumulative prosenter for variabelen inkludert missingverdiene <NA>. I dette tilfellet er det ingen enheter som har manglende verdier, og da blir de to tabellene like.

Hvis du vil vise fordelingen for den kontinuerlige variabelen alder, så kan du også bruke funksjonen freq(ess$agea), men denne variabelen har så mange kategorier at vi får bedre oversikt hvis vi viser presenterer den med statistiske mål for kontinuerlige variabler.

```
descr(ess$agea)
```

```
Descriptive Statistics
ess$agea
Label: Age of respondent, calculated
N: 1337
```

	agea
Mean	48.46
Std.Dev	18.71
Min	15.00
Q1	33.00
Median	49.00
Q3	63.00
Max	90.00
MAD	22.24
IQR	30.00
CV	0.39
Skewness	0.04
SE.Skewness	0.07
Kurtosis	-0.96
N.Valid	1335.00
N	1337.00
Pct.Valid	99.85

Klargjøring av data

Tema:

- *Missing Values*
- *Omkoding av variabler på ulike målenivå*
- *Interpolering*
- *Sammensatte mål*

Missing eller manglende verdier

En av statistikkens grunnregler er at jo flere enheter som tas med i analysen, jo bedre er det. For at utvalgene vi studerer kan kunne fortelle oss noe om populasjonen vi studerer er det viktig at den enkelte variabel ikke har for mange ”missing values”. Det vil si observasjoner der vi mangler data, f.eks. personer som har unnlatt å svare på det spesifikke spørsmålet – disse vil nemlig ikke bidra med noe til analysen.

Mange manglende verdier kan skape skjevheter i resultatene. Generelt er manglende informasjon minst skadelig hvis det opptrer helt tilfeldig. Hvis det derimot er snakk om systematisk frafall (eks. rike mennesker ønsker ikke svare på hvor mye de tjener) truer det utvalgets representativitet og kan gi skjeve resultater i analysen. Sjekk derfor om variablene dine har mange "missing” og vurder hva som kan være årsaken til dette.

I R blir alle missingverdiene kodet NA, som vil si at verdiene ikke er tilgjengelig (Not Available).

Kjør frekvensstatistikk som vi gikk gjennom sist for å se hvor mange missing det er for hver av disse variablene:

```
freq(ess$happy)
```

```
Frequencies
ess$happy
Label: How happy are you
Type: Numeric (labelled)

                          Freq   % Valid   % Valid Cum.   % Total   % Total Cum.
-------------------------- ------ --------- -------------- --------- --------------
      Extremely unhappy [0]      0     0.000          0.000     0.000          0.000
                    1 [1]      1     0.075          0.075     0.075          0.075
                    2 [2]      7     0.524          0.599     0.524          0.598
                    3 [3]     15     1.124          1.723     1.122          1.720
                    4 [4]     15     1.124          2.846     1.122          2.842
                    5 [5]     55     4.120          6.966     4.114          6.956
                    6 [6]     83     6.217         13.184     6.208         13.164
                    7 [7]    226    16.929         30.112    16.904         30.067
                    8 [8]    431    32.285         62.397    32.236         62.304
                    9 [9]    330    24.719         87.116    24.682         86.986
       Extremely happy [10]    172    12.884        100.000    12.865         99.850
                    <NA>      2                               0.150        100.000
                   Total   1337   100.000        100.000   100.000        100.000
```

Ser vi nærmere på variabelen happy vil vi se at det bare er to enheter som har blitt kodet som missing. Hvis en variabel har mange enheter som er kodet som missing må vi vurdere om bortfallet kan føre til at resultatene fra analysen vår blir skjeve i forhold til populasjonen.

I ESS11 er alle missing-verdiene allerede kodet som NA, og vi slipper derfor å kode bort missingverdiene når vi bruker disse variablene. Dersom vi ønsker å inkludere missing i analysen kan vi kode dem om slik at de blir numeriske. Her kan vi f.eks. velge å kode om alle med missing (NA) til medianverdien 8. For å gjøre dette kan vi bruke kommandoen *recode* som ligger i pakken «car», og når det er gjort kan du fjerne missinggruppen med følgende kommando:

```
library(car)
ess$happy <- recode(ess$happy, "NA=8")
freq(ess$happy)
```

Kommandoen *recode* finnes også i en del andre pakker, og noen ganger kan vi oppleve å få feilmelding hvis R ikke klarer å avgjøre hvilken recode-kommando vi mener. Når vi opplever dette problemet kan vi spesifisere navnet på pakken med denne kommandoen:

```
ess$happy <- car::recode(ess$happy, "NA=8")
```

Omkoding av variabler på ulike målenivå

For at resultatene dine skal være pålitelige er det helt avgjørende at variablene dine er kodet riktig. Noen av variablene kan brukes som de er, andre må man endre på. Er du i tvil så spør.

Dummykoding

Men "gndr" hadde jo kun to verdier, så hva med nominalvariabler som har flere enn to verdier? La oss se på variabelen "domicil". Denne måler hvor urbant/ruralt respondenten bor, og den har 5 kategorier/verdier som ser slik ut:

```
freq(ess$domicil)
```

```
Frequencies
ess$domicil
Label: Domicile, respondent's description
Type: Numeric (labelled)

                                        Freq   % Valid   % Valid Cum.   % Total   % Total Cum.
----------------------------------------- ------ --------- -------------- --------- --------------
                          A big city [1]    208    15.569         15.569    15.557         15.557
      Suburbs or outskirts of big city [2]    224    16.766         32.335    16.754         32.311
                  Town or small city [3]    407    30.464         62.799    30.441         62.752
                     Country village [4]    274    20.509         83.308    20.494         83.246
          Farm or home in countryside [5]    223    16.692        100.000    16.679         99.925
                                    <NA>      1                              0.075        100.000
                                   Total   1337   100.000        100.000   100.000        100.000
```

Denne variabelen kan også gjerne kodes om til en dummy med verdien 1 for de som bor i en bykommune, og verdien 0 for de som bor på landsbygda, og kalle den nye variabelen for bykommune. Det kan vi gjøre med disse kommandoene:

```
ess$bykommune <- recode(ess$domicil, "1:3=1; 4:5=0")
```

Hvis du prøver å få se frekvensfordelingen for den omkodede variabelen bykommune, så vil du se at merkelappene henger igjen i variabelen med tomme verdier. Dette unngår du hvis du endrer variabelen til en numerisk variabel ved å legge inn as.numeric i rekodingen

```
ess$bykommune <- recode(as.numerig(ess$domicil), "1:3=1; 4:5=0")
freq(ess$bykommune)
```

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
0	497	37.201	37.201	37.173	37.173
1	839	62.799	100.000	62.752	99.925
<NA>	1			0.075	100.000
Total	1337	100.000	100.000	100.000	100.000

La oss så se hvordan variabelen gndr (kjønn) ser ut:

```
freq(ess$gndr)
```

```
Frequencies
ess$gndr
Label: Gender
Type: Numeric (labelled)
```

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
Male [1]	673	50.34	50.34	50.34	50.34
Female [2]	664	49.66	100.00	49.66	100.00
<NA>	0			0.00	100.00
Total	1337	100.00	100.00	100.00	100.00

Her ser vi at menn er kodet med tallet 1, at kvinner er kodet med tallet 2, og at det er ingen i det norske utvalget som er kodet med missingkoden <NA>. Det er vanskelig å tolke kodene på slike nominalvariabler når vi ikke vet hvilke kjønn som er kodet 1 og 2. Derfor koder vi ofte todelte variabler på nominalnivå som ”mann” eller ”ikke mann” (eller ”kvinne” eller ”ikke kvinne”). Dette gjør vi ved å kode om verdiene til 1 og 0, der 1 betyr at man har egenskapen, og 0 betyr at man ikke har det. La oss kode om variabelen til å gjelde kvinne/ikke kvinne. Referansekategorien, det vil si den kategorien som andre kategorier måles opp mot, er alltid 0. Kvinne sammenlignes her med menn.

La oss kode om gndr til en dummyvariabel, og kalle den for kvinne:

```
ess$kvinne <- recode(as.numeric(ess$gndr), "1=0; 2=1")
```

Nå har det kommet en ny variabel med navnet kvinne helt bakerst i datasettet, og vi kan se fordelingen på denne variabelen med denne kommandoen:

```
freq(ess$kvinne)
```

```
             Freq   % Valid   % Valid Cum.   % Total   % Total Cum.
----------- ------ --------- -------------- --------- --------------
          0    673     50.34          50.34     50.34          50.34
          1    664     49.66         100.00     49.66         100.00
      <NA>      0                                0.00         100.00
      Total   1337    100.00         100.00    100.00         100.00
```

Det er også mulig å å trekke ut en enkelt verdi og kode denne med verdien 1 i en dummy. Du kan for eksempel lage en dummyvariabel for om man har stemt FrP eller ikke ved forrige valg med utgangspunkt i variabelen prtvtcno. Først må vi da se på hvordan variabelen den er kodet, og så kan vi lage en dummy som identifiserer de som stemte på FrP.

```
freq(ess$prtvtcno)
ess$frp <- recode(as.numeriv(ess$prtvtcno), "1:7=0; 8=1; 9:11=0")
freq(ess$frp)
```

Her kan vi se at det var 76 som stemte FrP mens 961 stemte enten et annet parti eller lot være å stemme.

Interpolering av variabel fra ordinalnivå til forholdstallsnivå

Interpolering handler om å omkode variabler til andre målenivåer. For eksempel kan vi interpolere hinctnta (inntekt) fra ordinalnivå til forholdstallsnivå. Dette vil gjøre tolkningen av variabelen enklere når vi senere skal lese resultatene av regresjonsanalysen. Variabelen er en kategori-variabel fordi hver verdi (1-10) står for en kategori- et inntektsdesil. Vi vil at variabelen skal kunne behandles kontinuerlig. Dette oppnår vi ved å interpolere variabelen. Da må vi finne ut av hva hvert desil egentlig betyr. Hjelpekortene (som intervjuerne bruker) kan fortelle oss dette. De norske hjelpekortene finner du på:

https://stessrelpubprodwe.blob.core.windows.net/data/round11/fieldwork/norway/ESS11_showcards_NO_nor.pdf

Spørsmål: F41

KORT 80

HUSHOLDNINGENS INNTEKT I NORSKE KRONER (NOK)

	Omtrentlig UKENTLIG	Omtrentlig MÅNEDLIG	Omtrentlig ÅRLIG	
J	Mindre enn 5 700	Mindre enn 24 700	Mindre enn 296 300	J
R	5 700 til 8 100	24 700 til 34 900	296 300 til 418 900	R
C	8 101 til 10 200	34 901 til 44 200	418 901 til 530 700	C
M	10 201 til 12 200	44 201 til 52 900	530 701 til 635 300	M
F	12 201 til 14 400	52 901 til 62 400	635 301 til 748 900	F
S	14 401 til 16 700	62 401 til 72 200	748 901 til 866 600	S
K	16 701 til 19 100	72 201 til 82 700	866 601 til 992 800	K
P	19 101 til 22 200	82 701 til 96 100	992 801 til 1 152 600	P
D	22 201 til 27 400	96 101 til 118 600	1 152 601 til 1 422 700	D
H	27 401 eller mer	118 601 eller mer	1 422 701 eller mer	H

Vi skal bruke årlig inntekt, og da regner vi ut midtpunktene i hvert desil. For eksempel vil midtpunktet i det første desilet være midt mellom 0-296 300, i det andre desilet vil midtpunktet være midt mellom 296 300-418 900. Vi finner ut at midtpunktet i desil 1 er (0+296 300)/2=148 150, og vi deler dette på 1000 og runder av desimalene. Da får vi at desil 1=148, desil 2=358, desil 3=475 osv (her runder vi oppover). Den øverste kategorien er åpen, og her velger vi den laveste verdien i intervallet.

```
freq(ess$hinctnta)
ess$inntekt <- recode(ess$hinctnta, "1=148;2=358;3=475;4=583;5=692;6=808;7=930;8=1073;9=1288;10=1423")
descr(ess$inntekt)
```

```
Descriptive Statistics
ess$inntekt
Label: Household's total net income, all sources
N: 1337

                     inntekt
----------------- ---------
             Mean    792.78
          Std.Dev    360.14
              Min    148.00
               Q1    475.00
           Median    808.00
               Q3   1073.00
              Max   1423.00
              MAD    392.89
              IQR    598.00
               CV      0.45
         Skewness      0.15
      SE.Skewness      0.07
         Kurtosis     -0.90
          N.Valid   1257.00
                N   1337.00
        Pct.Valid     94.02
```

Her ser vi at selv om frekvensene ikke har endret seg for hver kategori, så oppgir variabelen nå inntekt i antall 1000 kroner ,og er kontinuerlig med et gjennomsnitt på 792 780 kroner. Du finner igjen den nye inntektsvariabelen helt til slutt i datasettet ess.

Indekser

Refleksive indekser er basert på effektindikatorer. Svarene på spørsmålene (indikatorene) er skapt eller påvirket av en latent variabel (det vi ønsker å måle). Eksempler er politisk tillit eller holdning til innvandrere.

Refleksive indekser har et måleteoretisk grunnlag (i motsetning til formative indekser som ikke har et tilsvarende måleteoretisk grunnlag), og vi må dermed gjennomføre en del tester. For øvrig er det først og fremst sentralt at man ser på variablene (indikatorene) og gjør seg opp en teoretisk mening om de i det hele tatt kan slås sammen.

Vi lager en refleksiv indeks (skala) for politisk tillit med bruk av variablene trstprl, trstplt, trstprt som handler om tillit til ulike politiske institusjoner.

freq(ess$trstprl)

```
Frequencies
ess$trstprl
Label: Trust in country's parliament
Type: Numeric (labelled)
```

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
No trust at all [0]	32	2.40	2.40	2.39	2.39
1 [1]	14	1.05	3.46	1.05	3.44
2 [2]	38	2.85	6.31	2.84	6.28
3 [3]	62	4.66	10.97	4.64	10.92
4 [4]	71	5.33	16.30	5.31	16.23
5 [5]	154	11.57	27.87	11.52	27.75
6 [6]	148	11.12	38.99	11.07	38.82
7 [7]	242	18.18	57.18	18.10	56.92
8 [8]	337	25.32	82.49	25.21	82.12
9 [9]	150	11.27	93.76	11.22	93.34
Complete trust [10]	83	6.24	100.00	6.21	99.55
<NA>	6			0.45	100.00
Total	1337	100.00	100.00	100.00	100.00

freq(ess$trstplt)

```
Frequencies
ess$trstplt
Label: Trust in politicians
Type: Numeric (labelled)
```

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
No trust at all [0]	38	2.86	2.86	2.84	2.84
1 [1]	32	2.41	5.26	2.39	5.24
2 [2]	70	5.26	10.53	5.24	10.47
3 [3]	107	8.05	18.57	8.00	18.47
4 [4]	132	9.92	28.50	9.87	28.35
5 [5]	259	19.47	47.97	19.37	47.72
6 [6]	258	19.40	67.37	19.30	67.02
7 [7]	270	20.30	87.67	20.19	87.21
8 [8]	118	8.87	96.54	8.83	96.04
9 [9]	32	2.41	98.95	2.39	98.43
Complete trust [10]	14	1.05	100.00	1.05	99.48
<NA>	7			0.52	100.00
Total	1337	100.00	100.00	100.00	100.00

freq(ess$trstprt)

```
Frequencies
ess$trstprt
Label: Trust in political parties
Type: Numeric (labelled)
```

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
No trust at all [0]	26	1.96	1.96	1.94	1.94
1 [1]	22	1.66	3.61	1.65	3.59
2 [2]	63	4.74	8.35	4.71	8.30
3 [3]	101	7.60	15.95	7.55	15.86
4 [4]	134	10.08	26.03	10.02	25.88
5 [5]	281	21.14	47.18	21.02	46.90
6 [6]	250	18.81	65.99	18.70	65.59
7 [7]	285	21.44	87.43	21.32	86.91
8 [8]	129	9.71	97.14	9.65	96.56
9 [9]	28	2.11	99.25	2.09	98.65
Complete trust [10]	10	0.75	100.00	0.75	99.40
<NA>	8			0.60	100.00
Total	1337	100.00	100.00	100.00	100.00

Her ser vi at variablene er kodet på samme måte. Nå skal vi slå de sammen til en indeks.

Korrelasjonsmatrise med Pearsons r

Først lager vi et nytt objekt bestående av de variabelen som skal inngå i korrelasjonsanalysen. Deretter fjerne vi alle enhetene med missingverdien <NA> på en eller flere av de tre variablene, og lager et nytt objekt som består av en korrelasjonsmatrise med Pearsons r mellom alle variablene. Og så skriver vi ut denne korrelasjonsmatrisen.

```
matrise <- data.frame(ess$trstprl, ess$trstplt, ess$trstprt)
cor_matrix <- cor(matrise, use = "complete.obs")
print(cor_matrix)
```

```
            ess.trstprl ess.trstplt ess.trstprt
ess.trstprl   1.0000000   0.7693379   0.7466203
ess.trstplt   0.7693379   1.0000000   0.8443705
ess.trstprt   0.7466203   0.8443705   1.0000000
```

Man ønsker at korrelasjonen skal være på mer enn 0,4. Det ideelle er korrelasjoner mellom 0,4 og 0,8 (Ringdal 2025). Er det perfekt korrelasjon, kan den ene variabelen utelates, da den ikke tilfører noe ekstra informasjon. Hvis den er under 0,4 burde man fjerne variabelen. Korrelasjonsanalysen viser positive korrelasjoner over 0,4 mellom alle de tre variablene, og vi har da et godt grunnlag for å slå dem sammen til en indeks.

Cronbachs alfa

En annen måte for å vurdere grunnlaget for en refleksiv indeks er å beregne reliabilitetstesten Cronbachs alfa. Til det bruker vi funksjonen alpha som også ligger i pakken psych.

```
alfa <- alpha(matrise)
print(alfa)
```

```
Reliability analysis
Call: alpha(x = matrise)

  raw_alpha std.alpha G6(smc) average_r S/N    ase mean sd median_r
      0.91      0.92    0.89      0.79  11 0.0041  5.8  2     0.77

    95% confidence boundaries
          lower alpha upper
Feldt     0.90  0.91  0.92
Duhachek  0.91  0.91  0.92

 Reliability if an item is dropped:
             raw_alpha std.alpha G6(smc) average_r  S/N alpha se var.r med.r
ess.trstprl       0.91      0.92    0.84      0.84 10.9   0.0048    NA  0.84
ess.trstplt       0.85      0.85    0.75      0.75  5.9   0.0081    NA  0.75
ess.trstprt       0.87      0.87    0.77      0.77  6.7   0.0071    NA  0.77

 Item statistics
                n raw.r std.r r.cor r.drop mean  sd
ess.trstprl 1331  0.91  0.91  0.82   0.79  6.6 2.3
ess.trstplt 1330  0.94  0.94  0.91   0.86  5.4 2.1
ess.trstprt 1329  0.93  0.93  0.89   0.84  5.5 2.0
```

Cronbachs alfa varierer fra 0 til 1. Generelt ønsker vi at det skal være over 0,70 før vi konstruerer en skala. Her ser vi at vi har en høy verdi på Cronbachs alfa på 0,91 hvis vi tar med alle tre variablene. Hvis du vurderer om alfa-verdien blir høyere om du tar bort en av variablene, så bør du se på endringene i «raw_alpha» hvis vi fjerne en av variablene fra indeksen. Her ser vi at vi får samme verdi (0.91) hvis vi fjerne trstprl, og at alfa får lavere verdi hvis vi utelater en av de andre variablene. Så her anbefaler R oss å brukede tre variablene vi har valgt.

Faktoranalyse

En tredje metode for å vurdere grunnlaget for en refleksiv indeks er å gjennomføre en faktoranalyse. Her kan vi bruke funksjonen fa på den samme matrisen som vi nettopp definerte, og fa-funksjonen ligger i pakken stats.

```
library(psych)
faktor <- fa(matrise)
print(faktor)
```

```
Factor Analysis using method =  minres
Call: fa(r = matrise)
Standardized loadings (pattern matrix) based upon correlation matrix
             MR1   h2   u2 com
ess.trstprl 0.83 0.68 0.32   1
```

```
ess.trstplt 0.93 0.87 0.13   1
ess.trstprt 0.90 0.82 0.18   1

                 MR1
SS loadings     2.37
Proportion Var 0.79

Mean item complexity =  1
Test of the hypothesis that 1 factor is sufficient.

df null model =  3  with the objective function =  2.23 with Chi Square =  2975.75
df of  the model are 0  and the objective function was  0

The root mean square of the residuals (RMSR) is  0
The df corrected root mean square of the residuals is  NA

The harmonic n.obs is  1327 with the empirical chi square  0  with prob <  NA
The total n.obs was  1337  with Likelihood Chi Square =  0  with prob <  NA

Tucker Lewis Index of factoring reliability =  -Inf
Fit based upon off diagonal values = 1
Measures of factor score adequacy
                                                MR1
Correlation of (regression) scores with factors   0.96
Multiple R square of scores with factors          0.93
Minimum correlation of possible factor scores     0.86
```

I denne tabellen ser vi at en faktor vil fange fange opp 2,37 av den totale variansen i de tre variablene som inngår i faktoranalyse. Når vi sier at faktor 1 forklarer 79 prosent av variansen, betyr det at én underliggende dimensjon kan forklare nesten alt som måles i de tre variablene. De måler stort sett det samme, og at vi har et godt grunnlag for å lage en endimensjonal refleksiv indeks.

Alle de tre testene viser at vi har et godt statistisk grunnlag for å slå sammen de tre variablene i en reflektiv indeks som vi kan kalle politisktillit.

```
ess$politisktillit <- (ess$trstprl+ess$trstplt+ess$trstprt)
freq(ess$politisktillit)
```

Denne indeksen går fra 0 til 30, og hvis vi heller vil at den skap vise gjennomsnittet for de tre variablene så dividerer vi summen på 3.

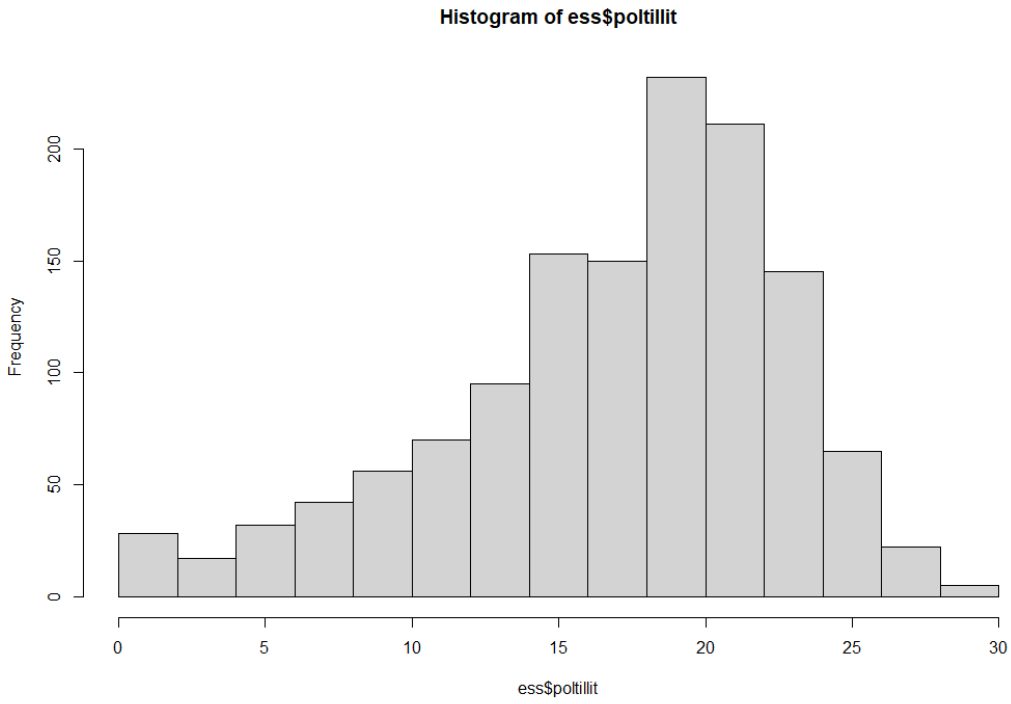
```
ess$politisktillit <- (ess$trstprl+ess$trstplt+ess$trstprt)/3
descr(ess$politisktillit)
```

```
Descriptive Statistics
ess$politisktillit
N: 1337
```

	politisktillit
Mean	5.81
Std.Dev	1.94
Min	0.00
Q1	4.67
Median	6.33
Q3	7.33
Max	10.00
MAD	1.48
IQR	2.67
CV	0.33
Skewness	-0.78
SE.Skewness	0.07
Kurtosis	0.33
N.Valid	1323.00
Pct.Valid	98.95

Vi har også vise fordelingen med et histogram, og det ser slik ut:

```
hist(ess$politisktillit)
```



Sammenheng mellom variabler

Tema:

- *Krysstabeller med kjikvadrattest*
- *Kjikvadrattest*
- *Introduksjon til lineær regresjon*

Krysstabeller og og kjikvadrater

I pakken *summarytools* finner du funksjonen ctable(). Denne funksjonen gir oss mye mer oversiktige krysstabeller enn det som er mulig med basiskommandoene i R, og den kan gi oss en kjikvadrattest for krysstabellen.

```
ctable(ess$polintr, ess$gndr, prop = "c", useNA = "no")
```

```
Cross-Tabulation, Column Proportions
polintr * gndr
Data Frame: ess

--------- ------ -------------- -------------- ---------------
            gndr              1              2           Total
  polintr
        1          105 ( 15.6%)    65 (  9.8%)    170 ( 12.7%)
        2          280 ( 41.6%)   226 ( 34.0%)    506 ( 37.8%)
        3          259 ( 38.5%)   311 ( 46.8%)    570 ( 42.6%)
        4           29 (  4.3%)    62 (  9.3%)     91 (  6.8%)
    Total          673 (100.0%)   664 (100.0%)   1337 (100.0%)
--------- ------ -------------- -------------- ---------------
```

I denne tabellen får vi oppgitt både antallet enheter i hver kombinasjon av de to variablene og hver verdi på den uavhengige variabelen blir prosentuert. Hvi du i tillegg ønsker å fa utført en kjikvadrattest for sammenhengen, så legger du bare inn det i kommandoen.

```
ctable(ess$polintr, ess$gndr, prop = "c", useNA = "no", chisq = TRUE)
```

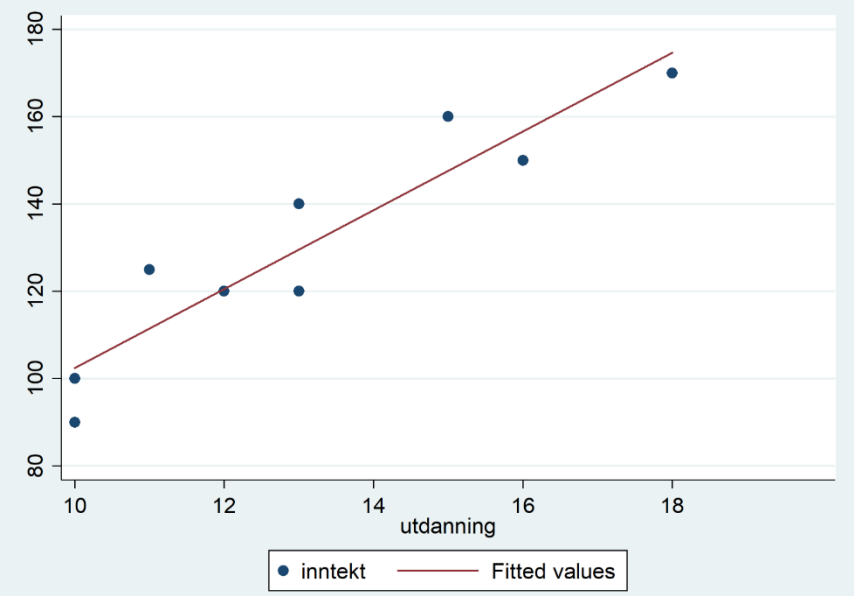
```
---------------------------
 Chi.squared   df   p.value
------------- ---- ---------
   31.8264     3       0
---------------------------
```

Lineær regresjonsanalyse

Hensikten med en regresjonsanalyse er å se hvorvidt og i hvilken grad vi har en statistisk signifikant sammenheng mellom en avhengig og en eller flere uavhengig(e) variabler. En regresjonsanalyse med bare to variabler, en avhengig og en uavhengig, kalles en *bivariat regresjonsanalyse*. Denne kan utvides og inkludere flere uavhengige variabler, og kalles da *multippel regresjonsanalyse*.

Bivariat regresjon

Vi begynner med en bivariat analyse. Se for deg et koordinatsystem med to akser, der vi plotter inn hver enkel enhet vi studerer som et punkt. *OLS (ordinary least squares) regresjonsanalyse* er en teknikk som tilpasser en rett linje til observasjonene, slik at avstandene mellom punktene i diagrammet og linjen blir minst mulig.



(Konstruert eksempel på sammenhengen mellom inntekt og utdanning)

Her skal vi kjøre en regresjonsanalyse for å se om variabelen utdanning (som måler utdanning i antall år) har en statistisk påvirkning på tillit bekymringer for klimaendringer (wrclmch). Siden vi forutsetter at det er utdanning som påvirker klimabekymring, og ikke omvendt, blir klimabekymring avhengig variabel. Først tar vi en nærmere titt både på den avhengige og den uavhengige variabelen.

freq(ess$wrclmch)

```
Frequencies
ess$wrclmch
Label: How worried about climate change
Type: Numeric (labelled)
```

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
Not at all worried [1]	57	4.266	4.266	4.263	4.263
Not very worried [2]	180	13.473	17.740	13.463	17.726
Somewhat worried [3]	639	47.829	65.569	47.794	65.520
Very worried [4]	413	30.913	96.482	30.890	96.410
Extremely worried [5]	47	3.518	100.000	3.515	99.925
<NA>	1			0.075	100.000
Total	1337	100.000	100.000	100.000	100.000

freq(ess$eduyrs)

```
Frequencies
ess$eduyrs
Label: Years of full-time education completed
Type: Numeric
```

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
0	2	0.150	0.150	0.150	0.150
2	2	0.150	0.301	0.150	0.299
3	2	0.150	0.451	0.150	0.449
4	4	0.301	0.751	0.299	0.748
5	4	0.301	1.052	0.299	1.047
6	2	0.150	1.202	0.150	1.197
7	11	0.826	2.029	0.823	2.019
8	18	1.352	3.381	1.346	3.366
9	36	2.705	6.086	2.693	6.058
10	70	5.259	11.345	5.236	11.294
11	76	5.710	17.055	5.684	16.978
12	137	10.293	27.348	10.247	27.225
13	126	9.467	36.814	9.424	36.649
14	115	8.640	45.455	8.601	45.251
15	132	9.917	55.372	9.873	55.123
16	141	10.594	65.965	10.546	65.669
17	120	9.016	74.981	8.975	74.645
18	134	10.068	85.049	10.022	84.667
19	75	5.635	90.684	5.610	90.277
20	61	4.583	95.267	4.562	94.839
21	24	1.803	97.070	1.795	96.634
22	18	1.352	98.422	1.346	97.981
23	9	0.676	99.098	0.673	98.654
24	7	0.526	99.624	0.524	99.177
25	2	0.150	99.775	0.150	99.327
26	2	0.150	99.925	0.150	99.476
27	1	0.075	100.000	0.075	99.551
<NA>	6			0.449	100.000
Total	1337	100.000	100.000	100.000	100.000

I kommandoen regress skal avhengig variabel først, etterfulgt av en eller flere uavhengige variabler.

Kommandoene er:

modell1 <- lm(trstprl ~ eduyrs, data=ess)

```
summary(modell1)
```

```
Call:
lm(formula = wrclmch ~ eduyrs, data = ess)

Residuals:
     Min       1Q   Median       3Q      Max
-2.37722 -0.29232 -0.08006  0.70768  2.13219

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 2.528203   0.096476   26.20  < 2e-16 ***
eduyrs      0.042451   0.006307    6.73 2.51e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.842 on 1328 degrees of freedom
  (7 observations deleted due to missingness)
Multiple R-squared:  0.03298,  Adjusted R-squared:  0.03226
F-statistic:  45.3 on 1 and 1328 DF,  p-value: 2.512e-11
```

Her har vi fått en tabell med en stor mengde tall, det viktigste å se på er R^2, B-koeffisienten og p-verdien.

Vi skal nå gå gjennom tabellen og hva du skal se etter.

B-koeffisienten (Coef.) forteller oss hvor mye et trinns økning i den uavhengige variabelen tilsvarer på den avhengige variabelen. Se også på fortegnet, for det kan være at en økning på den uavhengige variabelen gir lavere verdier på den avhengige, her kan kodingen være lite intuitiv, så vær sikker på at du vet hva høye og lave verdier tilsvarer for hver variabel du bruker. Når du etter hvert legger til flere variabler i regresjonsmodellen så vil også b-koeffisienten som oftest endre seg.

Signifikansverdien/P-verdien (P>|t|) er svært viktig i regresjonsanalyse. Tidligere har vi brukt kjikvadratet for å teste om korrelasjonene mellom to variabler var sterke nok til å kunne generalisere funnet til populasjonen for øvrig. Når kjikvadratet ble stort nok nådde det en «kritisk verdi», eller en slags kritisk størrelse, hvor vi ikke lenger kunne beholde nullhypotesen om ingen sammenheng i populasjonen, og vi måtte konkludere med at det vi ser i dataene våre også gjelder for populasjonen. I regresjonsanalyse bruker vi t-testen istedenfor kjikvadrattesten til å forkaste nullhypotesene våre om at de målte effektene bare er tilfeldige. Vi trenger ikke å se på t-verdien, som også står i koeffisienttabellen vår ettersom vi har signifikansverdien ved siden av. Når signifikansverdien er lavere enn 0,05 har t-verdien, akkurat som kjikvadratet, blitt stort nok til at vi må forkaste nullhypotesen og generalisere til populasjonen. I modell 1 er signifikansverdien for **Utdanning** oppgitt som $2{,}51*10^{-11}$ som vil si det samme som 0,0000000000251, og det er en langt lavere verdie enn 0,05. Det vi si at forholdet mellom utdanning og klimabekymring er så klar at vi kan forkaste nullhypotesen som antar at sammenhengen er tilfeldig. Dersom p-verdien blir større enn 0,05, f.eks. 0,10, må vi beholde nullhypotesen og konkludere med at sammenhengen i utvalget kun er tilfeldig gitt et signifikansnivå på 0,05.

R Square (R^2) forteller oss hvor mye av variansen på den avhengige variabelen som skyldes den eller de uavhengige variabelen/variablene i modellen, og varierer fra 0,0 til 1,0. Her er det 0,03298, som vil si at utdanningsvariabelen vår forklarer 3,3 % av variansen i folks klimabekymring. I samfunnsvitenskapelig statistikk er det ikke vanlig å få høye tall på R Square, spesielt hvis det er snakk om et holdningsmål som avhengig variabel. Det har blitt rettet kritikk mot R^2 fordi R Square uansett øker når man legger til flere uavhengige variabler. Har man flere uavhengige variabler kan man da heller se på **Adjusted R Square** (tilpasset R^2) som tar hensyn til antallet variabler. Forskjellene er stort sett ikke veldig store.

Standardfeil (Std. Err.) viser oss spredningen rundt b-koeffisienten (akkurat som standardavviket viser spredningen rundt gjennomsnittet). Den viser oss hvor mye vi forventer å bomme i populasjonen. Dersom dette tallet er høyt sett i forhold til b-koeffisienten er det stor sannsynlighet for at variabelen ikke er signifikant. *Det betyr at det er altfor stor variasjon rundt b-koeffisienten til at vi kan være sikre på et mønster.*

Funksjonen summary viser ikke hvor mange av dataenhetene (i ESS er dette personer) som er tatt med i regresjonsanalysen, men det oppgis at 7 observasjoner er fjernet på grunn av manglende data (missing). Antallet respondenter (N) i denne regresjonsmodellen blir derfor 1337 -7 = 1330.

t-verdien er signifikansmålet (testobservatoren) for variabler i regresjonsanalyser, og brukes til å beregne p-verdien. Den kan du også regne ut selv ved å dividere B-koeffisienten på standardfeilen. R oppgir i tillegg stjerner (*) som viser på hvilket nivå (0,5, 0,1, eller 0,01) variabelens effekt er statistisk signifikant.

Hvis du laster ned pakken *jtools* kan du bruke funksjonen summ i stedet for summary, og da får du en mye er ryddig regresjonstabell

summ(modell1)

```
MODEL INFO:
Observations: 1330 (7 missing obs. deleted)
Dependent Variable: wrclmch
Type: OLS linear regression

MODEL FIT:
F(1,1328) = 45.30, p = 0.00
R² = 0.03
Adj. R² = 0.03

Standard errors: OLS
------------------------------------------------
                    Est.   S.E.   t val.      p
----------------- ------ ------ -------- ------
(Intercept)         2.53   0.10    26.21   0.00
eduyrs              0.04   0.01     6.73   0.00
------------------------------------------------
```

Et problem med funksjonen summ er at den bare oppgir to desimaler, men antallet viste desimaler kan uvides med delkommandoen digits.

summ(modell1, digits=3)

```
MODEL INFO:
Observations: 1330 (7 missing obs. deleted)
Dependent Variable: wrclmch
Type: OLS linear regression

MODEL FIT:
F(1,1328) = 45.296, p = 0.000
R² = 0.033
Adj. R² = 0.032

Standard errors: OLS
--------------------------------------------------
                      Est.    S.E.   t val.       p
------------------ ------- ------- -------- -------
(Intercept)          2.528   0.096   26.205   0.000
eduyrs               0.042   0.006    6.730   0.000
```

Multippel lineær regresjon

Når man tester for korrelasjoner er det viktig å vite at korrelasjon ikke er det samme som kausalitet. Når vi kjører statistiske analyser for korrelasjon er det derfor viktig at man ikke tror at man finner frem til årsak -> virkning (i denne sammenheng kan det være nyttig å huske historien om storken og antall fødte barn, eller påstått sammenheng mellom nedgangen i antall pirater og global oppvarming). Det kan finnes et mønster mellom de to, men det betyr ikke nødvendigvis at det er en kausal sammenheng. Sammenhengen kan være spuriøs (falsk). Det kan for eksempel hende at det egentlig er en annen variabel som påvirker resultatet du har fått. I en regresjonsanalyse har man mulighet til å kontrollere for andre variabler når vi skal måle en årsakssammenheng. Dette er grunnen til at du har med kontrollvariabler i semesteroppgaven, og vi går derfor videre på multippel lineær regresjonsanalyse.

Hvis vi fortsetter å undersøke sammenhengen mellom utdanning og klimabekymring, så kan det tenkes at det er flere konfunderende variabler som påvirker bekymringer for klimaet. For eksempel kan kjønnsforskjeller i utdanning påvirke resultatene våre. I tillegg kan det tenkes at de med høy utdanning har også høy inntekt, og det og ha høy inntekt også kan innvirke på klimabekymringen. Det kan også tenkes at det er utdanningsforskjeller mellom de som bor i ker og de som bor på landet, og at unge har større grad av klimabekymring enn eldre personer. Vi legger derfor til kjønn, inntekt alder og bykommune som kontrollvariabler i tillegg til utdanning.

```
modell2 <- lm(wrclmch ~ eduyrs+kvinne+inntekt+agea+by, data=ess)
summ(modell2, digits=3)
```

```
MODEL INFO:
Observations: 1251 (86 missing obs. deleted)
Dependent Variable: wrclmch
Type: OLS linear regression

MODEL FIT:
F(5,1245) = 26.598, p = 0.000
```

```
R² = 0.097
Adj. R² = 0.093

Standard errors: OLS
--------------------------------------------------
                      Est.    S.E.   t val.       p
---------------- -------- ------- -------- -------
(Intercept)          2.237   0.131   17.053   0.000
eduyrs               0.040   0.007    5.902   0.000
kvinne               0.327   0.046    7.059   0.000
inntekt             -0.000   0.000   -1.627   0.104
agea                 0.002   0.001    1.645   0.100
bykommune                0.234   0.049    4.801   0.000
--------------------------------------------------
```

Det første vi legger merke til nå vi sammenligner de to modellene er at koeffisienten for antall år utdanning har blitt svakere i den var i den bivariate analysen. Resultatet er fortsatt signifikant, men koeffisienten har litt lavere verdi etter at vi kontrollerte for de andre variablene. Se vi på p-verdien ser vi at variabelen kvinne også er signifikant, noe som betyr at kvinner har signifikant større klimabekymring enn menn. Videre ser vi at verken inntekt eller alder har signifikant effekt på klimabekymring.

Det er likevel problematisk å sammenligne koeffisientene i de to modellene når de ikke er basert på de samme enhetene. Modell 1 er baser på 1330 personer, mens modell 2 er basert på 1251 personer. Det er 79 personer som er med i modell 1 men som ikke er med i modell 2, og denne forskjellen skyldes i hovedsak at det er mange som har missngverdi <NA> på inntektsvariabelen. For å forsikre oss om at de to modellen bykommunegger på data fra de samme personene lager vi et underutvalg bestående av de seks variablene som inngår i den største av disse modellene, og fjerner alle personene med missing <NA> på en eller flere av de seks variablene, og så estimerer vi både modell 1 og modell 2 med data fra dette reskede utvalget.

```
utvalg <- subset(ess, select = c(wrclmch, eduyrs, kvinne, inntekt, agea, bykommune))
utvalg=na.omit(utvalg)
modell1 <- lm(wrclmch ~ eduyrs, data=utvalg)
summ(modell1, digits=3)
MODEL INFO:
Observations: 1251
Dependent Variable: wrclmch
Type: OLS linear regression

MODEL FIT:
F(1,1249) = 50.700, p = 0.000
R² = 0.039
Adj. R² = 0.038

Standard errors: OLS
--------------------------------------------------
                      Est.    S.E.   t val.       p
---------------- ------- ------- -------- -------
(Intercept)          2.469   0.100   24.745   0.000
eduyrs               0.046   0.006    7.120   0.000
--------------------------------------------------
```

```
modell2 <- lm(wrclmch ~ eduyrs+kvinne+inntekt+agea+bykommune, data=utvalg)
```

summ(modell2, digits=3)

```
MODEL INFO:
Observations: 1251
Dependent Variable: wrclmch
Type: OLS linear regression

MODEL FIT:
F(5,1245) = 26.598, p = 0.000
R² = 0.097
Adj. R² = 0.093

Standard errors: OLS
-------------------------------------------------
                      Est.    S.E.   t val.       p
----------------- -------- ------- -------- -------
(Intercept)           2.237   0.131   17.053   0.000
eduyrs                0.040   0.007    5.902   0.000
kvinne                0.327   0.046    7.059   0.000
inntekt              -0.000   0.000   -1.627   0.104
agea                  0.002   0.001    1.645   0.100
bykommune             0.234   0.049    4.801   0.000
-------------------------------------------------
```

Da har vi to modeller med det samme antallet enheter, noe som gjør at de kan sammenlignes. Modell 1 har en forklart varians på (R^2 jurstert) på 0,038, og at den forklarte variansen øker til 0,093 i modell 2. Vdere ser vi at effekten av hvert år med utdanning går ned fra 0,046 til 0,040 når vi kontrollerer for effektene av kvinne, inntekt og bykommune. Vi ser visere at kvinnene har større grad av klimabekymring enn menn, og at kjønnsforskjellen på 0,327 trinn på klimavariabelen er statistisk signifikant på 5%-nivået. Variablene inntekt og bykommune har derimot ikke signifikante effekter på kvimabekymring.

Regresjonsmodeller med dummysett

I alle typer regresjonsanalyse bør vi skille klart mellom uavhengige kontinuerlige variabler som måler effekten som lineære sammenhenger, og uavhengige variabler som måler effekten av dummyer. I de siste modellene har vi vi kodet om bostedsvariabelen *domicil* til å måle eventuelle forskjeller mellom de som bor i bykommune og de som bor på landsbygda. Den opprinnelige bostedsvariabelen *domicil* hadde opprinnelig 5 kategorier, og vi kan få et enda mer nyansert bilde av bostedseffekten hvis vi lager et dummysett som måler all informasjonen som ligger i denne variabelen. Ettersom dummyvariabler alltid måler forskjellen effekt mellom en gruppe og en referansegruppe, slik vi har målt effekten av å bo i en bykommune i forhold til referansegruppen landsbygd, så kan vi også bruke den første av kategoriene i en kategorisk variabel som referanse, og se hvordan enhetene i den gruppen skille seg fra enhetene i hver av de andre gruppene. Hvis vi ber R om å transformere variabelen *domicil* fra å måle en enkel lineær effekt om til et dummysett kan vi bruke kommandoen *as.factor* for å gjøre denne om til et dummysett med storby som referansekategori.

```
ess$bosted <- as.factor(ess$domicil)
modell3 <- lm(wrclmch ~ bosted, data=ess)
summ(modell3)
```

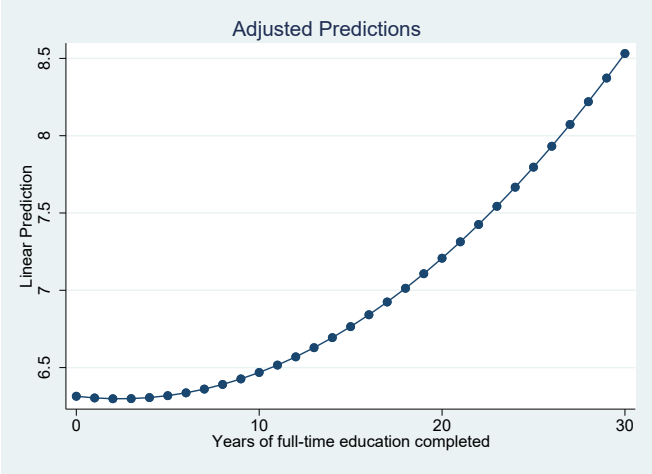
```
MODEL INFO:
Observations: 1335 (2 missing obs. deleted)
Dependent Variable: wrclmch
Type: OLS linear regression

MODEL FIT:
F(4,1330) = 9.86, p = 0.00
R² = 0.03
Adj. R² = 0.03

Standard errors: OLS
------------------------------------------------
                     Est.   S.E.   t val.      p
----------------- ------- ------ -------- ------
(Intercept)          3.39   0.06    58.12   0.00
bosted2             -0.19   0.08    -2.33   0.02
bosted3             -0.19   0.07    -2.65   0.01
bosted4             -0.33   0.08    -4.23   0.00
bosted5             -0.47   0.08    -5.85   0.00
------------------------------------------------
```

Her ser vi resultatene av dummymodellen med verdien Storby som referansegruppe. De som bor i utkanten av en storby (bosted 2) ligger i gjennomsnitt 0,19 trinn lavere på variabelen som måler klimabekymring enn de som bor på landsbygda,og at denne forskjellen er statistisk signifikant på 5%-nivået. De som bor i småbyer (bosted 3) ligger også i gjennomsnitt 0,19 trinn lavere på 5-trinnsskalaen for klimabekymring enn de som bor i storbyen, og også denne forskjellen er statistisk signifikant på 5%-nivået. De som bor i tettsteder på landsbygd (bosted 4) ligger i gjennomsnitt 0,33 trinn lavere på klimabekymring enn de som bor i storby, og de som bor i spredtbygde (bosted 5) ligge i gjennomsnitt 0,47 trinn lavere enn de som bor i storby på femtrinnsvariabelen som måler klimabekymring. Alle de fire dummyene er dermed signifikant lavere grad av klimabekymring enn de som bor storbyer.

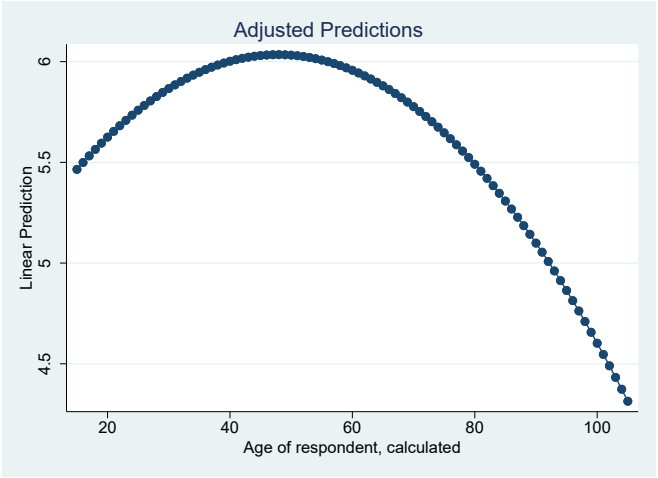
Regresjonsmodeller med andregradsledd / kurvelinearitet

I en regresjonsmodell måles alle sammenhenger i utgangspunktet som om sammenhengen mellom X og Y er lineær, men det er ikke alle sammenhenger som er lineære. Kvadratledd kan brukes til å avsløre såkalte kurvelineære mønstre i en regresjonsmodell. Hvis du vil teste ut om du har et kurvemønster i modellen så estimerer du både effekten av den uavhengige variabelen og den samme uavhengige variabelen opphøyd i andre i en og samme regresjonsmodell.

Først kan det være greit å få et grafisk inntrykk av hva kurvelinearitet handler om.



Figur 1.Sammenheng mellom utdanning og tillit til Stortinget



Figur 2. Sammenheng mellom alder og holdning til innvandrere

Vi kan se at sammenhengen mellom utdanning og tillit til Stortinget er tilnærmet lineær. Sammenhengen mellom alder og holdninger til innvandrere (ifm økonomi) er derimot tydelig kurvelineær. Denne sammenhengen skal vi se nærmere på nå.

Først må vi kunne å lage et andregradsledd. Andregradsledd lages ved å gange variabelen med seg selv. Andregradsleddet legges enklest inn med kommandoen **I(agea^2)**. Da skjønner R at den skal måle variabelen agea med to ulike variabelledd; et førstegradsledd og et andregradsledd.

La oss kjøre to stegvise regresjonsmodeller. Vi bruker **imbgeco** som avhengig variabel (som måler om respondenten mener innvandring er bra eller dårlig for økonomien på en skala fra 0 (dårlig for økonomien) til 10 (godt for økonomien). I den modell 1 legger vi kun inn alder (agea) og andregradsleddet for alder (I(agea^2)) som uavhengige variabler i modellen.

```
modell4 <- lm(imbgeco ~ agea+I(agea^2), data=ess)
summary(modell4, digits=5)
```

```
MODEL INFO:
Observations: 1321 (16 missing obs. deleted)
Dependent Variable: imbgeco
Type: OLS linear regression

MODEL FIT:
F(2,1318) = 8.86089, p = 0.00015
R² = 0.01327
Adj. R² = 0.01177

Standard errors: OLS
--------------------------------------------------------------
                        Est.      S.E.     t val.          p
----------------- ---------- --------- ---------- ----------
(Intercept)          5.16524   0.33946   15.21606   0.00000
agea                 0.05176   0.01490    3.47397   0.00053
I(agea^2)           -0.00058   0.00015   -3.86523   0.00012
--------------------------------------------------------------
```

Her ser vi at både variabelen agea og andregradsleddet I(agea^2) får signifikante effekter på 5%-nivået, og at koeffisienten for agea har positivt fortegn mens koeffisienten for I(agea^2) har negativt fortegn.

Kurvelinearitet kan vise seg i fire forskjellige former:

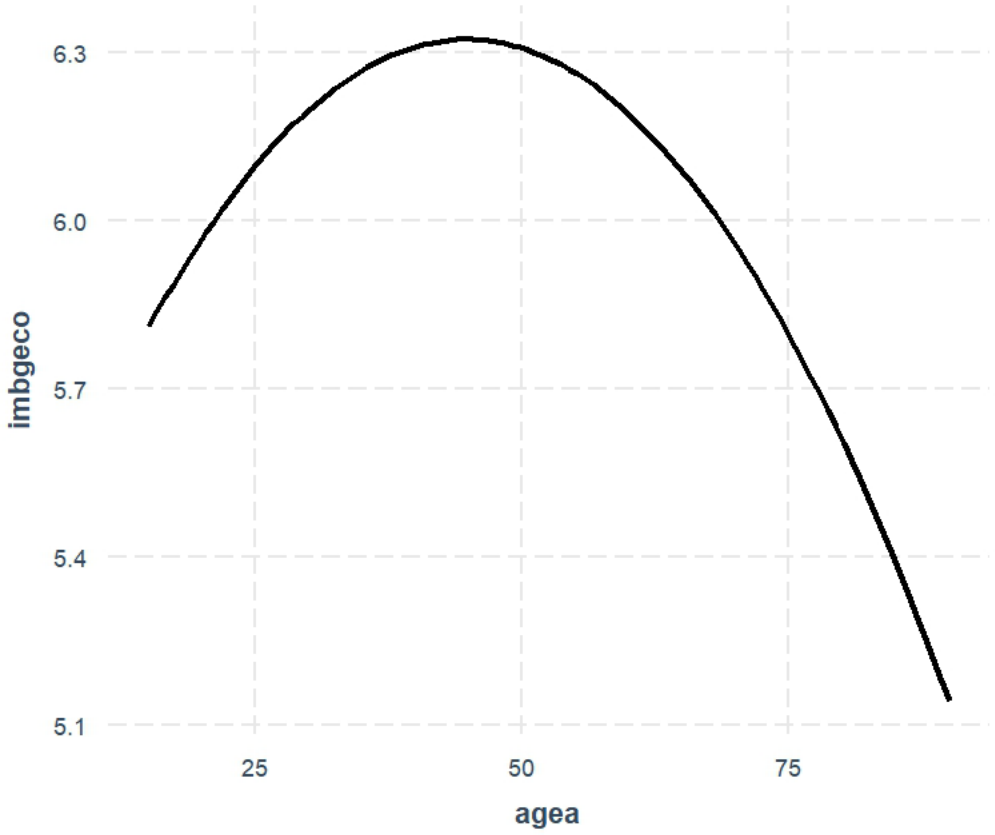
- Positiv første- og andregrads koeffisient: Stigende linje som bøyer ytterligere oppover
- Negativ første- og andregrads koeffisient: Synkende linje som stuper enda brattere
- Positiv første, negativ andregrads koeffisient: ∩-formet linje som først stiger, for så å flate ut og begynner å synke
- Negativ første, positiv andregrads koeffisient: U-formet linje som først synker, for så å flate ut for så å begynner å stige

Merk: Formelen for å regne ut vendepunktet er: $-b_{agea}/(2*b_{I(agea^2)})$. Dersom vi regner ut vendepunktet på kurven i vårt eksempel (-0,05176/(2*-0,00058)=44,62), finner vi at jo eldre man blir, jo mer positivt mener man innvandring er for økonomien, før sammenhengen snur og blir negativ rundt 45 år, og det kurvelineære mønsteret er statistisk signifikant på 5%-nivået.

Hvis du i tillegg ønsker å se å se den kurvelineære formen på sammenhengen mellom alder og syn på innvandring, så kan du bruke funksjonen *effect_plot* i pakken *jtools*. Vi kan lage plottet slik:

```
effect_plot(modell4, pred = agea)
```

Her vil du få fram et spredningsdiagram som viser den kurvelineære alderseffekten for alle aldersgruppene i datasettet:



Hvis du i tillegg ønsker i vise konfidensintervallet for hver alderskategori. Så kan du bruke denne kommandoen:

```
effect_plot(modell4, pred = agea, interval = TRUE)
```

Regresjonsmodell med samspillseffekter

Noen variabler får en ekstra effekt av at andre variabler er med i spillet. Vi har å gjøre med et samspill når effekten av en uavhengig variabel på den avhengige er forskjellig for ulike verdier på en annen uavhengig variabel.

Vi kan teste om effekten av aldersvariabelen **agea** på holdninger til homofile er ulik for de som bor i storby og de som bor andre steder. Variabelen **freehms** er her omkodet til **homoaksept,** og skalaen er endre fra 1-5 til 5-1 slik at de som er sterkt uenig i at homofile bør få leve slik de selv vil har blitt kodet om til verdien 0 og de som er sterkt enig i at homofile bør få leve slik de selv ønsker er kodet med verdien 4. Denne omkodingen gjør vi med følgende kommando:

```
ess$homoaksept <- recode(ess$freehms, "1=5; 2=4; 3=3; 4=2; 5=1")
```

Videre koder vi om den uavhengige variabelen **domicil** til den nye variabelen **storby** der de som bor i storby (1) og i omegnen til en storby (2) er kodet 1, mens de som bor i småbyer og på landsbygda er kodet 0.

```
ess$storby <- recode(ess$domicil, "1:2=1; 3:5=0")
```

Så kan vi estimere to regresjonsmodeller med homoaksept som avhengig variabel, der modell 5 måler effekten av alder og storby, mens modell 6 i tillegg måler et eventuelt samspill mellom alder og storby.

```
modell5 <- lm(homoaksept ~ agea+storby, data=ess)

summ (modell5)
```

```
MODEL INFO:
Observations: 1329 (8 missing obs. deleted)
Dependent Variable: homoaksept
Type: OLS linear regression

MODEL FIT:
F(3,1325) = 17.15, p = 0.00
R² = 0.04
Adj. R² = 0.04

Standard errors: OLS
------------------------------------------------
                     Est.   S.E.   t val.      p
----------------- ------- ------ -------- ------
(Intercept)          4.80   0.06    76.37   0.00
agea                -0.01   0.00    -4.98   0.00
storby              -0.06   0.11    -0.57   0.57
agea:storby          0.01   0.00     2.42   0.02
------------------------------------------------
```

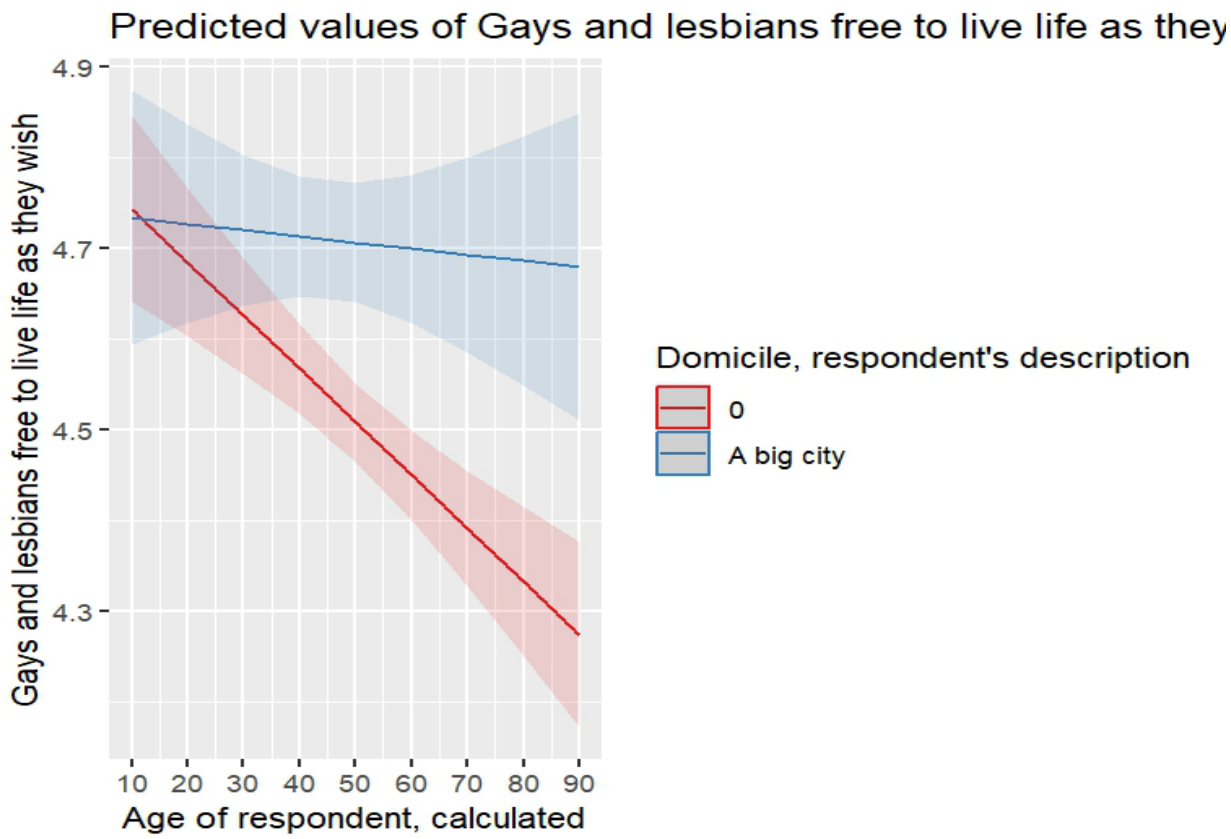
```
modell6 <- lm(homoaksept ~ agea+storby+agea:storby, data=ess)
summary(modell6)
```

```
MODEL INFO:
Observations: 1329 (8 missing obs. deleted)
Dependent Variable: homoaksept
Type: OLS linear regression

MODEL FIT:
F(3,1325) = 17.145, p = 0.000
R² = 0.037
Adj. R² = 0.035

Standard errors: OLS
--------------------------------------------------
                       Est.    S.E.   t val.      p
----------------- -------- ------- -------- -------
(Intercept)           4.801   0.063   76.375   0.000
agea                 -0.006   0.001   -4.984   0.000
storby               -0.061   0.108   -0.570   0.569
agea:storby           0.005   0.002    2.417   0.016
--------------------------------------------------
```

Den første av disse to modellene (modell5) måler effekten av å bo i storby og alder, mens den andre modellen (modell6) viser i tillegg om det er et samspill mellom storby og alder i forhold til holdning til homofile. I den siste modellen (modell6) ser vi at det ikke lenger er en signifikant forskjell i holdningen til homofile mellom de som bor i storby og de som bor andre steder. Men

nå får vi en positiv og statistisk signifikant samspillseffekt mellom alder og bosted. For å kunne tolke slike samspillet er det ofte lurt å bruke grafiske hjelpemidler. Funksjonen effect_plot, som vi brukte til å vise den kurvelineære grafen, egner seg ikke når vi ønsker å se effekten av to uavhengige variabler. For å lage figurer med med et amspill mellom to uavhengige variabler bruker vi heller funksjonen *plot_model* i pakken *sjPlot.* Funksjonen *plot_model* viser da alderseffektene for laveste verdi og høyeste verdi på variabelen storby. Ettersom variabelen storby er en dummy, så vises alderseffekten for de som bor i storby og alderseffekten for de som ikke bor i storby. Hvis begge variablene hadde vært kontinuerlige, så hadde denne kommandoen vist aldersfordelingene for de med laveste og de med høyeste verdi på den andre uavhengige variabelen i kommandoen. Når vi bruker argumentet *type = "int"* inne i denne funksjonen, så vil R klare å identifisere de to samspillsvariablene også når det er flere uavhengige variabler i modellen.

```
plot_model(modell6, type = "int")
```



I dette marginalplottet ser vi at det er bare små forskjeller i aldersgruppenes homoaksept i storbyene, mens det er en klar tendens til mindre aksept meg økt alder blant de som bor andre steder. Det vil si at den generelle nedgangen i aksept som vi fant i modell 5 ikke gjelder for storbyene, og derfor ble heller ikke aldersvariabelen signifikant når vi kontrollerte for samspillet mellom alder og bosted.

Logistisk regresjon

Tema:

- *Logistisk regresjon i R*
- *Tolkning av logistisk regresjon*
    - *Logit*
    - *Odds*
    - *Predikert sannsynlighet*

Det siste vi skal gå gjennom i dette innføringsheftet er logistisk regresjon. Logistisk regresjon skiller seg fra lineær regresjon ved at den avhengige variabelen er en dikotom dummyvariabel, altså at svarkategoriene er kodet som enten 1 eller 0. Dette husker vi fra tidligere, der vi blant annet har dummykodet kjønn og bosted (**gndr** og **domicil)**.

Når avhengig variabel er dikotom, blir måten vi tolker regresjonen på ganske annerledes fra lineær regresjon. Vi måler ikke lenger hvordan forandringer i verdier på uavhengig(e) variabler påvirker plasseringen på avhengig variabel; i stedet måler vi hvordan forandringer i uavhengig variabel påvirker *sannsynligheten* for at avhengig variabel blir målt til 1 (og ikke 0).

Hvis vi ønsker å undersøke hva som påvirker sannsynligheten for å stemme på et bestemt politisk parti må vi bruke logistisk regresjon. I dette tilfellet skal vi se på sannsynligheten for å stemme på Frp. Tidligere har vi omkodet variabelen prtvtcno til en dikotom variabel som måler om man stemte på Frp (=1) eller ikke Frp (=0). Nå skal vi bruke denne variabelen som avhengig variabel.
Som uavhengige variabel bruker vi først bare utdanningsnivå (eduyrs) som uavhengig variabel, og etterpå utvider vi modellen ved å også bruke kvinne, agea og imwbcnt. Den siste av disse variablene måler holdninger til innvandrere ved å spørre om innvandrere gjør landet til et bedre eller dårligere sted å bo på en skala fra 0-10, der 0 betyr at innvandring gjør det dårligere å bo i landet mens 10 betyr at innvandring gjør det bedre å bo i landet.

Nå er vi klare til å estimere logistiske regresjonsmodeller. Først estimerer vi en bivariat logistisk regresjonsmodell:

```
modell_logistisk1 <- glm(frp ~ eduyrs, data = ess, family = binomial)
summ(modell_logistisk1, digit=3)
```

```
MODEL INFO:
Observations: 1033 (304 missing obs. deleted)
Dependent Variable: frp
Type: Generalized linear model
  Family: binomial
  Link function: logit

MODEL FIT:
χ²(1) = 17.491, p = 0.000
Pseudo-R² (Cragg-Uhler) = 0.041
```

```
Pseudo-R² (McFadden) = 0.032
AIC = 529.417, BIC = 539.297

Standard errors: MLE
--------------------------------------------------
                      Est.    S.E.   z val.       p
---------------- -------- ------- -------- -------
(Intercept)         -0.602   0.449   -1.340   0.180
eduyrs              -0.134   0.032   -4.233   0.000
--------------------------------------------------
```

Her ble det bare estimert en parameter, koeffisienten for eduyrs, og da er antallet enheter 1033. Merk at det er betydelig færre observasjoner i denne modellen enn i andre modeller du har kjørt, fordi de som ikke stemte ikke er inkludert. Deretter vises en χ^2-test (17,49) med et statistisk signifikant utfall (0.000). Dette betyr bare at modellen er signifikant bedre enn en modell uten forklaringsvariabler (uavhengige variabler). Til slutt har vi to ulike mål på Pseudo R^2. De to målene er en parallell til R^2 i lineær regresjon, men kan **ikke** tolkes på samme måte. Mens vi i lineær regresjon snakket om prosent forklart varians, snakker vi her om hvor mye den fullstendige modellen forklarer sammenlignet med en modell uten variabler. De interesserte kan lese mer i Ringdal (2024).

Vi går videre til å tolke modellen. Når man tolker logistisk regresjon, er det tre ulike nivåer å tolke ut ifra. Disse kan sees på som tre ulike vanskelighetsgrader.

Tre tolkninger:

1) Logit

Den negative logiten (B-koeffisienten) viser at antall år med utdanning reduserer sannsynligheten for å stemme Frp (å ha verdi 1 på avhengig variabel). Dette er signifikant på både 5%- og 1%-nivået, og kan dermed generaliseres til populasjonen.
Logiten er den naturlige logaritmen av oddsen for å stemme Frp, og tolkningen av selve koeffisienten blir derfor ganske abstrakt. For å kunne si noe mer om resultatene må vi gå videre til å tolke oddsratio.

2) Oddsratio

Ettersom informasjonen om den logistiske regresjonsmodellen allerede ligger i objektet *logit*, så trenger du ikke å estimere modellen på nytt. Det er nok om du ber R beregne oddratioene fra dette objektet.

```
oddsratio1 <- exp(coef(modell_logistisk1))
print(oddsratio1)
```

```
(Intercept)      eduyrs
  0.5477675   0.8747549
```

Oddsratioet for utdanning er lavere enn 1, og da synker oddsen for å stemme Frp med antall år med utdanning. Hvis vi setter inn oddsratioet i formelen for å finne den prosentvise forskjellen i odds for hvert år med utdanning får vi

(OR-1)*100
(0,8747549-1)*100 = -12,52451 ≈ -12,5

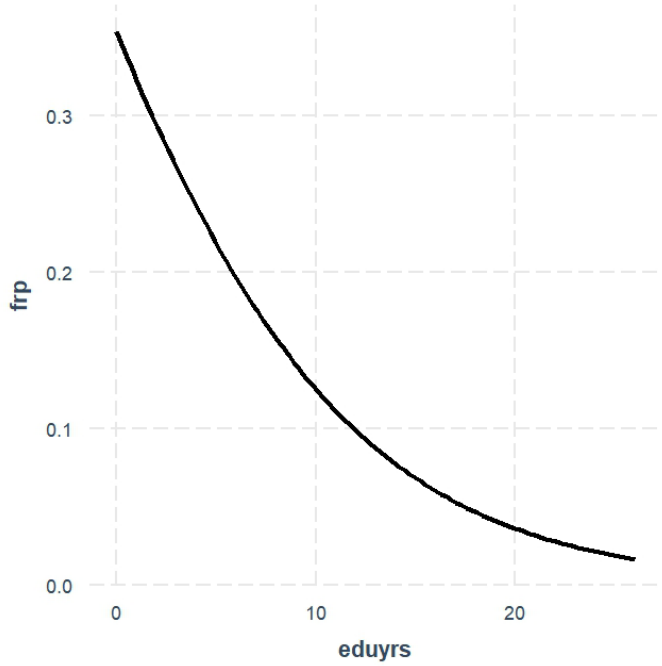
For hvert år med utdanning man fullfører, reduseres oddsen for å stemme Frp med 12,5 prosent.

3) Predikert sannsynlighet

Når vi har estimert en logistisk regresjonsmodell kan vi med utgangspunkt i de logistiske regresjonskoeffisientene beregne sannsynligheten for at personer med bestemte kjennetegn på den eller de uavhengige variablene har verdien 1 på den avhengige variabelen. Her kan vi velge å predikere sannsynligheter for alle enhetene, men det er lurt å tenke litt igjennom hva vi ønsker å finne ut av før vi velger hvilke verdier vi skal predikere sannsynlighetene for.

I dette tilfellet velger vi å lage et grafisk plott med sannsynlighetene for å stemme Frp ut fra utdanningsnivå. Du får fram et plott med sannsynligheten for hvert av disse utdanningsårene hvis du kjører denne kommandoen:

```
effect_plot(modell_logistic1, pred = eduyrs)
```



Multiple logistiske regresjonsmodeller

Forskjellen mellom logistiske multiple regresjonsmodellen og lineære multiple regresjonsmodeller er at de logistiske modellene har en dummy som avhengig variabel mens de lineære modellene har en avhengig variabel med minst rangerte 5 kategorier. Det er ingen forskjell i hvordan vi setter inn de uavhengige variablene. Også i logistisk regresjon kan vi teste om de uavhengige variablenes effekt på den avhengige variabelen er lineær, kurvelineær eller inngår i et samspill. Vi kan estimere en multippel logistisk regresjonsmodell med disse kommandoene:

```
modell_logistisk2 <- glm(frp ~ eduyrs + kvinne + inntekt + agea + bykommune, data = ess,
family = binomial)
summ(modell_logistisk2, digit=3)
```

```
MODEL INFO:
Observations: 1001 (336 missing obs. deleted)
Dependent Variable: frp
Type: Generalized linear model
  Family: binomial
  Link function: logit

MODEL FIT:
χ²(5) = 38.242, p = 0.000
Pseudo-R² (Cragg-Uhler) = 0.091
Pseudo-R² (McFadden) = 0.072
AIC = 506.683, BIC = 536.135

Standard errors: MLE
------------------------------------------------------
                      Est.    S.E.   z val.       p
----------------- -------- ------- -------- -------
(Intercept)          0.446   0.763    0.584   0.559
eduyrs              -0.112   0.037   -3.004   0.003
kvinne              -1.078   0.275   -3.920   0.000
inntekt             -0.001   0.000   -2.222   0.026
agea                -0.006   0.007   -0.751   0.453
bykommune           -0.001   0.257   -0.005   0.996
------------------------------------------------------
```

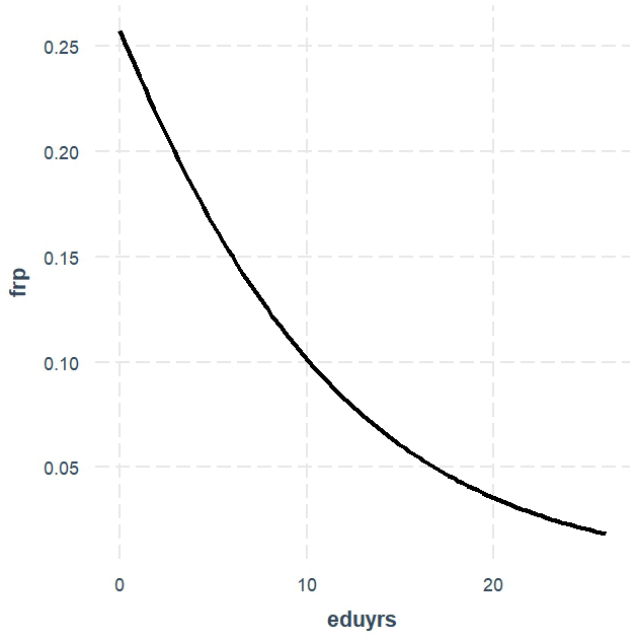
Og vi kan få beregnet oddsratioer med disse kommandoene:

```
oddsratio2 <- exp(coef(modell_logistisk2))
print(oddsratio2)
```

```
(Intercept)      eduyrs      kvinne     inntekt        agea    bykommune
  1.5619715   0.8940991   0.3402306   0.9991426   0.9945054   0.9986982
```

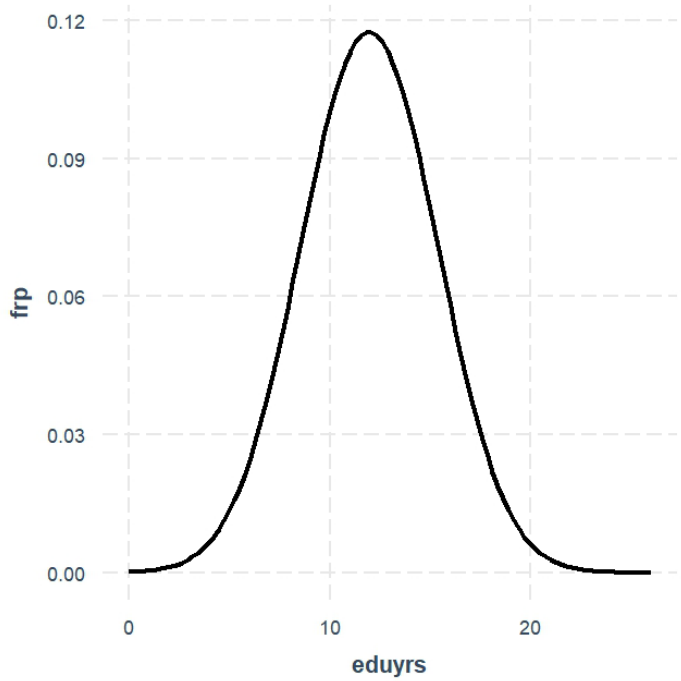
Hvis vi vil se effekten av f.eks. utdanning fra denne modellen, så må verdiene på de andre uavhengige variablene settes til sine respektive gjennomsnittsverdier. Dette blir gjort automatisk med denne kommandoen:

```
effect_plot(modell_logistisk2, pred = eduyrs)
```



Hvis du vil ha et tilsvarende plott med en kurvelineær utdanningseffekt kan du skrive:

```
modell_logistisk3 <- glm(frp ~ eduyrs+I(eduyrs^2)+kvinne+inntekt+agea+bykommune, data = ess_reg, family = binomial)
summ(modell_logistisk3, digits=3)
effect_plot(modell_logistisk3, pred = eduyrs)
```



Logistisk regresjon presentert som AME

Average Marginal Means (AME) angir den gjennomsnittlige endringen i sannsynligheten for å stemme FrP når en uavhengig variabel øker med én enhet (eller skifter kategori), mens alle andre variabler holdes konstante.

summary(margins(modell_logistisk3))

```
 factor     AME     SE       z      p   lower   upper
   agea -0.0002 0.0005 -0.3767 0.7064 -0.0011  0.0008
     by  0.0025 0.0168  0.1513 0.8797 -0.0304  0.0355
 eduyrs -0.0102 0.0025 -4.0430 0.0001 -0.0151 -0.0052
inntekt -0.0000 0.0000 -1.9126 0.0558 -0.0001  0.0000
 kvinne -0.0679 0.0186 -3.6493 0.0003 -0.1043 -0.0314
```

Effektene uttrykkes i prosentpoeng, men i R rapporteres de som *sannsynlighetsendringer*. Her blir f.eks. den gjennomsnittlige endringen i sannsynligheten for å stemme FrP lik -0.0102 = –1,02 prosentpoeng for hvert økte trinn på utdanningsskalaen.

Referanser

Ringdal, Kristen (2024). *Enhet og mangfold*. *Samfunnsvitenskapelig forskning og kvantitativ metode*. 5. utg. Bergen: Fagbokforlaget.

Vedlegg

```
# Introduksjon til R

# Her leser du inn de pakken vi får bruk for i kurset
library(tidyverse)
library(haven)
library(summarytools)
library(expss)
library(gmodels)
library(car)
library(psych)
library(jtools)
library(sjPlot)
library(margins)
# Hvis du mangler noen av pakkene, så må du installere dem

# Hvis du har PC: Les inn datasettet du har hentet fra ESS fra din PC-katalog
# ess <- read_dta("C:/katalognavn/ESS11-subset.dta")
# Hvis du har Mac: Les inn datasettet du har hentet fra ESS fra din Mac-katalog
# ess <- read_dta("/Volumes/navn/katalognavn/ESS11-subset.dta")
```

```
# Lag en enkel tabell for variabelen polintr
table(ess$polintr)
# Lag en vakrere tabell med pakken summarytools
freq(ess$polintr)

# Hvordan presenterer vi den kontinuerlige aldersvariabelen agea?
freq(ess$agea)
# Kontinuerlige variabler bør heller presenteres med statistiske mål
descr(ess$agea)


# Klargjøring av data
# Endrer missing (NA) i variabelen happy til medianverdien 8.
freq(ess$happy)
ess$happy <- recode(ess$happy, "NA=8")
freq(ess$happy)

# Tabell med variabelen bosted (domicil)
freq(ess$domicil)
# Denne kan kodes om til den nye dummyen bykommune, der 1=by og 0=landsbygd
ess$bykommune <- recode(ess$domicil, "1:3=1; 4:5=0")
freq(ess$bykommune)
# Her sere dere at merkelappene henger igjen i variabelen med tomme verdier
# Dette unngår dere ved å legge in as.numeric i rekodingen
ess$bykommune <- recode(as.numeric(ess$domicil), "1:3=1; 4:5=0")
freq(ess$bykommune)

# Tabell med kjønnsvariabelen gndr
freq(ess$gndr)
# Denne variabelen kan også kodes om til dummyen kvinne
ess$kvinne <- recode(as.numeric(ess$gndr), "1=0; 2=1")
freq(ess$kvinne)

# Tabell med partibarometer
freq(ess$prtvtcno)
# Partibarometeret kan gjøres om til en dummy for FrP
ess$frp <- recode(as.numeric(ess$prtvtcno), "1:7=0; 8=1; 9:11=0")
freq(ess$frp)

# Tabell for samlet husholdsinntekt i desiler
freq(ess$hinctnta)
# Slik kan du interpolere inntektsvariabelen fra ordinal- til forholdstallsnivå
ess$inntekt <- recode(ess$hinctnta, "1=148;2=358;3=475;4=583;5=692;6=808;7=930;8=1073;9=1288;10=1423")
descr(ess$inntekt)

# Men denne variabelen er nå kodet slik at den kan tolke gjennomsnittsverdien
descr(ess$inntekt)

# Lage en indeks for politisk tillit
freq(ess$trstprl)
freq(ess$trstplt)
freq(ess$trstprt)
matrise <- data.frame(ess$trstprl, ess$trstplt, ess$trstprt)
```

```
cor_matrix <- cor(matrise, use = "complete.obs")
print(cor_matrix)
# Her er alle korrelasjonene over 0,3 og vi kan slå de tre variablene sammen til en indeks
ess$poltillit <- (ess$trstprl+ess$trstplt+ess$trstprt)
freq(ess$poltillit)
hist(ess$poltillit)

# Nå kan du også bruke det lagrede objektet matrise til å lage objekter for Cronbachs alfa og for en faktoranalyse
# Cronbachs alfa
alfa <- alpha(matrise)
print(alfa)
# Faktoranalyse
faktor <- fa(matrise)
print(faktor)
# Også disse to analysene bekrefter at de tre variablene kan slås sammen til en indeks
ess$politisktillit <- (ess$trstprl+ess$trstplt+ess$trstprt)/3
freq(ess$politisktillit)
descr(ess$politisktillit)
hist(ess$politisktillit)

# Sammenheng mellom variablene
# Lage en bivariat krysstabell med politisk interesse fordelt etter kjønn
ctable(ess$polintr, ess$gndr, prop = "c", useNA = "no")
# En enklere tabell med kjikvadrattest
ctable(ess$polintr, ess$gndr, prop = "c", useNA = "no", chisq = TRUE)

# Lineær regresjon
# Bivariat regresjon
freq(ess$wrclmch)
freq(ess$eduyrs)
modell1 <- lm(wrclmch ~ eduyrs, data=ess)
summary(modell1)
summ(modell1)
# Hvis du vil vise resultatene med tre desimaler skriver du:
summ(modell1, digits=3)

# Multippel regresjon
modell2 <- lm(wrclmch ~ eduyrs+kvinne+inntekt+agea+bykommune, data=ess)
summ(modell2, digits=3)

# I modell1 er N=1330 og i modell2 er N=1251. Vi må derfor lage et datasett med lik N i begge modellene?
utvalg <- subset(ess, select = c(wrclmch, eduyrs, kvinne, inntekt, agea, bykommune))
utvalg=na.omit(utvalg)
modell1 <- lm(wrclmch ~ eduyrs, data=utvalg)
summ(modell1, digits=3)
modell2 <- lm(wrclmch ~ eduyrs+kvinne+inntekt+agea+bykommune, data=utvalg)
summ(modell2, digits=3)

# Regresjonsmodeller med dummysett
ess$bosted <- as.factor(ess$domicil)
modell3 <- lm(wrclmch ~ bosted, data=ess)
summ(modell3)
# Hvis du vil endre referansekategori fra 1 til 5 kan du skrive:
```

```
ess$bosted2 <- relevel(ess$bosted, ref=5)
modell3b <- lm(wrclmch ~ bosted2, data=ess)
summ(modell3b, digits=3)

# Kurvelegresjonsmodeller med andregradsledd
# Ny avhengig variabel
freq(ess$imbgeco)
modell4 <- lm(imbgeco ~ agea+I(agea^2), data=ess)
summ(modell4, digits=5)
# Grafisk fremstilling av kurvelineariteten
effect_plot(modell4, pred = agea)

# Regresjonsmodell med samspillseffekt
# Omkoding av variablene homoaksept og bykommune
freq(ess$freehms)
ess$homoaksept <- recode(ess$freehms, "1=5; 2=4; 3=3; 4=2; 5=1")
ess$storby <- recode(ess$domicil, "1:2=1; 3:5=0")
# Estimering av lineær modell uten samspill
modell5 <- lm(homoaksept ~ agea+storby, data=ess)
summ (modell5, digit=3)
# Estimering av lineær modell med samspill
modell6 <- lm(homoaksept ~ agea+storby+agea:storby, data=ess)
summ (modell6, digit=3)
# Grafisk fremstilling av samspillet
plot_model(modell6, type = "pred", terms = c("agea", "storby"))

# logistisk regresjon
# Bivariate logistisk regresjonsmodell
modell_logistisk1 <- glm(frp ~ eduyrs, data = ess, family = binomial)
summ(modell_logistisk1, digit=3)
# Oddsratio
oddsratio1 <- exp(coef(modell_logistisk1))
print(oddsratio1)

# Utdanningseffekten vist grafisk som sannsynligheter
effect_plot(modell_logistisk1, pred = eduyrs)

# Multippel logistisk regresjonsmodell
modell_logistisk2 <- glm(frp ~ eduyrs + kvinne + inntekt + agea + storby, data = ess, family = binomial)
summ(modell_logistisk2, digit=3)
# Oddsratio
oddsratio2 <- exp(coef(modell_logistisk2))
print(oddsratio2)
# Utdanningseffekten kontrollert for andre variabler vist grafisk som sannsynligheter
effect_plot(modell_logistisk2, pred = eduyrs)

# Estimerer den logistiske regresjonsmodellen med kurvelineær utddaningseffekt
modell_logistisk3 <- glm(frp ~ eduyrs + I(eduyrs^2) + kvinne + inntekt + agea + storby, data = ess, family = binomial)
summ(modell_logistisk3, digits=3)

# Plotte den kurvelineære effekten av eduyrs på frp
effect_plot(modell_logistisk3, pred = eduyrs)
```

```
# AME-analyse av modell_logistisk2
summary(margins(modell_logistisk3))
```