

# Kurs i dataeditering: Imputering

ASLAUG FOSS HURLEN

2025



**Statistisk sentralbyrå**  
Statistics Norway

# Plan for Kurset

**10:00 – 10.40** Introduksjon til imputering og regelbasert imputering

**10.40 - 11.15** *Øvelser i R med R-pakken dcmodyfy*

**11:15 – 11:45** Lunsj

**11:45 - 12:25** Imputering med regresjon, nærmeste nabo og andre modeller

**12:25 – 13.15** *Øvelser i R med R-pakken simulation*

**13.15 – 13.30** Logging og kvalitetsindikatorer for imputering

**13.30 – 13.45** *Øvelse i R med pakken lumberjack*

**13.45 – 14.00** Oppsummering



# Læringsmålet

Målet er:

- Kjenne til de mest kjente metodene for imputering
- Kjenne til prosessdata og kvalitetsindikatorer for imputering/editering
- Kunne bruke R til å gjennomføre imputeringen.



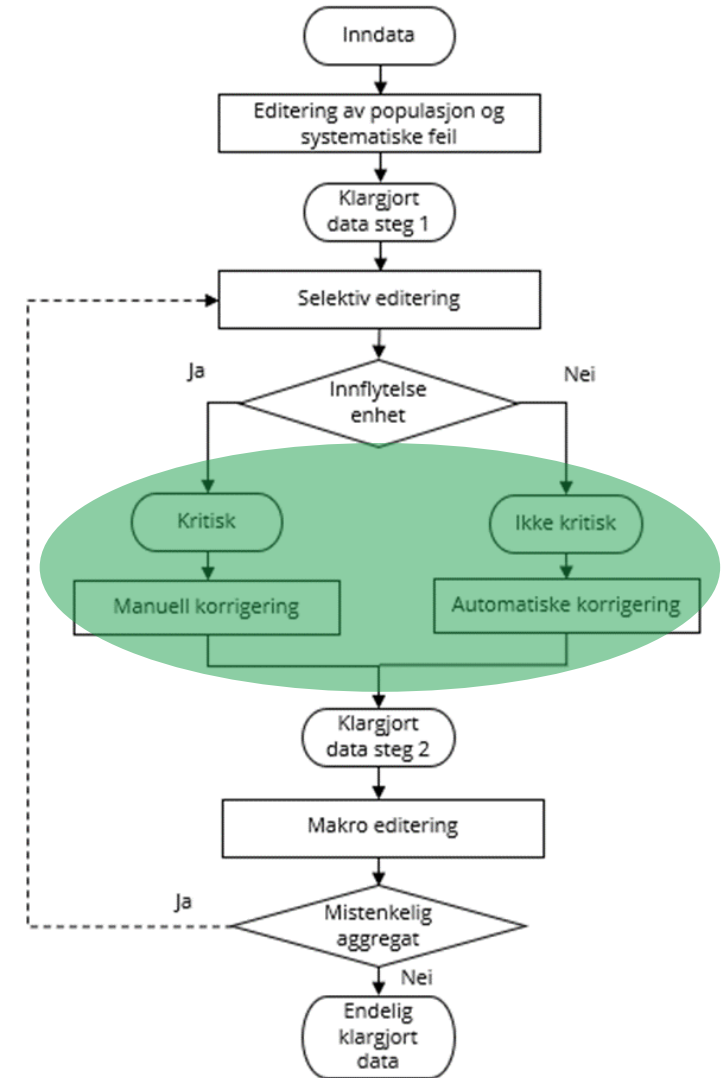
# Materialet for kurset

- **Github:** <https://github.com/statisticsnorway/kurs-metode-imputere>
- **Byrånettesiden «dataeditering»:** lenke til materialet og lenker til all bakgrunnsliteratur  
<https://ssbno.sharepoint.com/sites/Metodikkistatistikkproduksjonen/SitePages/Dataeditering.aspx>



# Imputering og prosessmodell

- korrigerer av mistenkelige verdier og erstatte manglende verdier
- Generic Statistical Data Editing Model
- <https://statswiki.unece.org/display/sde/GSDDEM>



# Imputering

- Imputering er prosessen der verdier i et datasett som mangler eller er mistenkelige erstattes av kjente akseptable verdier.
- Vi vil imputere med formålet å redusere frafallsskjevhet og lage et «fullt» datasett.



# Hva er imputering og hvorfor det trengs?

# Datasett – partielt frafall

Enhet	Variabel 1	Variabel 2	Variabel 3	Variabel 4	Variabel 5
1					
2					
3					
4					
5					
6					





# Datasett – enhetsfrafall

Enhet	Variabel 1	Variabel 2	Variabel 3	Variabel 4	Variabel 5
1					
2					
3					
4					
5					
6					



# Datasett – mistenkelige og feil verdier

Enhet	Variabel 1	Variabel 2	Variabel 3	Variabel 4	Variabel 5
1					
2					
3					
4					
5					
6					



# Resultat hvis vi ikke gjør noe med frafall

- Sum av det som er rapportert
- Vanskelig å sammenligne over tid

*Aslaug Hurlen Foss og Liv Taule*

**Museumsstatistikken**

En gjennomgang av definisjoner, kvalitet og populasjon



# Hva kan man gjøre med frafall?

- Imputering: lage “fullt” datasett
- Vekting – vanlig i utvalgsundersøkelser

# Typer av imputering:

- *Manuell*: ekspertkunnskap, tilleggsopplysninger, rekontakt
- *Regelbasert imputering*: imputering basert på logiske regler
- *Modellbasert imputering*: gjennomsnitt, regresjon, decision tree, osv
- *Donor imputering*: får en verdi fra en annen enhet eller periode.

Nærmeste nabo-imputering

# Regelbasert imputering for åpenbare og systematiske feil

# Systematiske feil og åpenbare feil

Åpenbare feil - er observasjoner som har klart uriktige verdier

Systematiske feil - er gjennomgående feil som trekker i en retning og forekommer for mange enheter i undersøkelsen



# Regelbasert imputering

- Ofte **logisk forhold** eller basert på **ekspertkunnskap**.

- 'if - then'-type påstander:

**if** Alder < 0 **then** Alder = «-1»\*Alder

**if** lønn < 10 000 **then** kjønn = «kvinne»





# Fagkunnskap - emnekunnskap

- Reglene settes ut fra kunnskap om datasett
- Det er viktig å kunne vurdere holdbarhet av reglene over tid



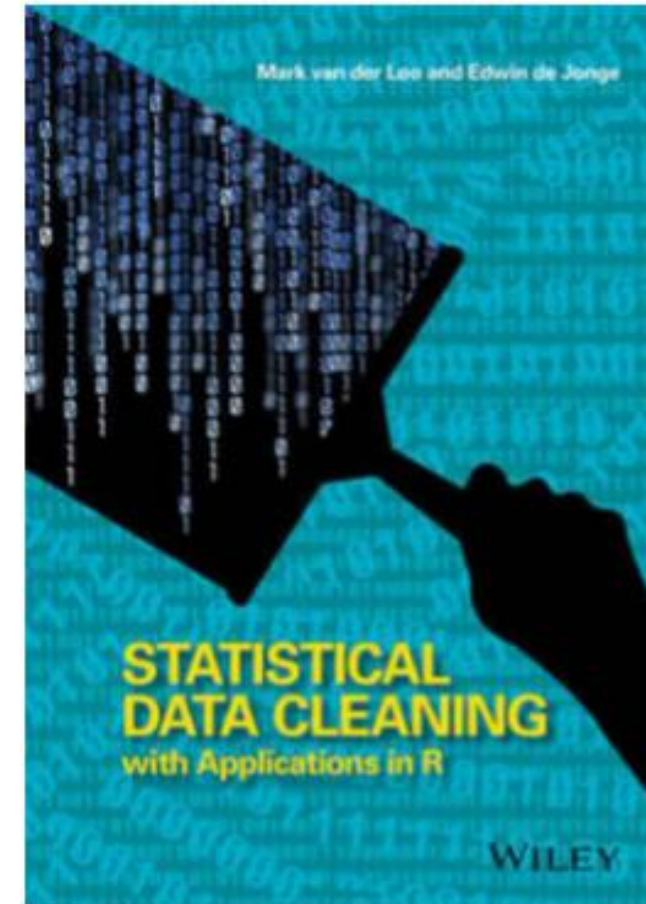
# Historisk imputering

- En enhet er mest lik seg selv
- Veksten mellom periodene kan ignoreres
- Eksempel:
  - Foreløpige tall kommuneregnskap blir historisk imputert. Små kommuner
  - Km med snøskuterløper. Lite endring fra år til år. Vedtak for endring.



# Pakken *dcmodyfy*

- Mark van der Loo og Edwin de Jonge, statistics Nederlands
- Introduksjon:
  - <https://cran.r-project.org/web/packages/dcmodyfy/vignettes/introduction.html>



# Hvorfor en pakke for regelretting?

- Samle og vedlikeholde regler for korrigerings et sted
- Kan legges på en egen fil
- Kan enkelt legge til logging av endring



# Grunnleggende arbeidsflyt

- **data:** Det er ditt datasett (data formate: data.frame).
- **modifier:** Object som er laget for modifiseringsregler.
- **modify:** Funksjon som anvender modifiseringsregler på data.

```
modify( data, modifier(modifiseringsregler) )
```



# *modifier* – definere og lagre regler

Object

Modifiseringsregler

```
library(dcmodyfy)
m <- modifier( if (var1 < 0) var1 <- abs(var1),
               if (var1 > 1000 * var2) var1 <- var1/1000 )
```

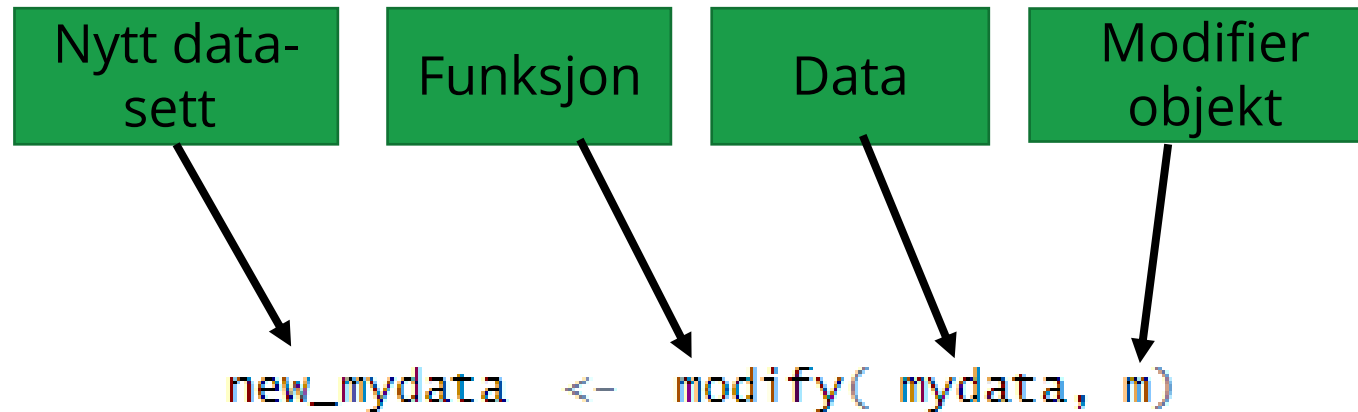
Funksjon

```
> m
Object of class modifier with 2 elements:
M1:
  if (var1 < 0) var1 <- abs(var1)

M2:
  if (var1 > 1000 * var2) var1 <- var1/1000
```



# *modify* data med regler



```
> new_mydata
  ID var1 var2
1  1    2    9
2  2    9    1
3  3    1    4
4  4    7    8
```

```
> mydata
  ID var1 var2
1  1    2    9
2  2    9    1
3  3   -1    4
4  4    7    8
```



# Vurdering av imputering

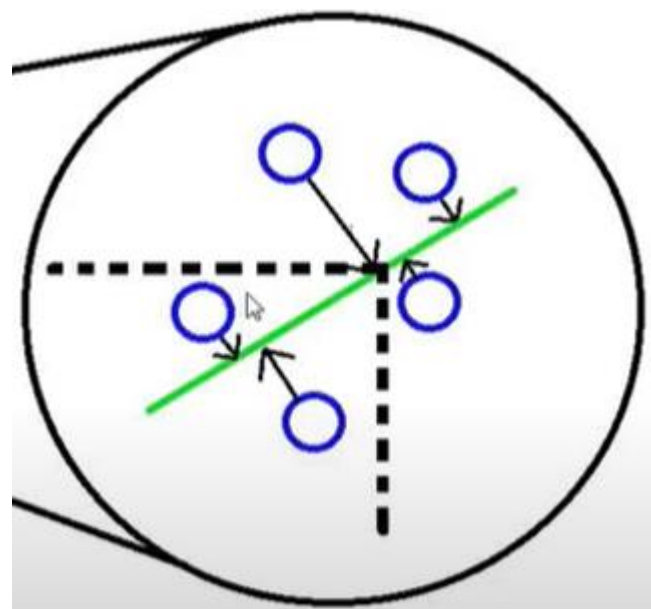
- Grafikk
- Størrelse på feil

$$RMSE = \sqrt{\frac{\sum_{i=1}^N \|y(i) - \hat{y}(i)\|^2}{N}},$$

Et tall som forteller hvor god modellen er – jo mindre jo bedre

## Data til vurdering av resultat

- Automatisk korrigert mot manuelt editert
- Foreløpige tall mot endelige tall
- Lager testdata





# Eksempel

DATASETT WOMEN - BMI



**Statistisk sentralbyrå**  
Statistics Norway

# Gruppe oppgaver

- Korrigerer dere verdier manuelt?
- Hvordan finner dere «riktig verdi»?
- Er det mulig å lage regelretting istedenfor manuell endring?



# Datasett til øvelse i R

- Avtalte årsverk fysioterapi i kommunen – reelt datasett fra 2021!
- Ved foreløpige tall 15. mars mangler en del kommuner – disse blir imputert
- Variabler:
  - Kommune
  - arsverk\_2020
  - arsverk\_2021\_for
  - Brutto\_driftsutgifter\_helse\_2021
  - Folkemengde\_2021
  - arsverk\_2021\_end

# Kursmaterialet

## https://github.com/statisticsnorway/kurs-metode-imputere

← → ↻ 🏠 [github.com/statisticsnorway/R\\_kontrollfunksjoner](https://github.com/statisticsnorway/R_kontrollfunksjoner)

Search or jump to... / Pull requests Issues Marketplace Explore

[statisticsnorway / R\\_kontrollfunksjoner](#) Public Edit Pins

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

[main](#) 2 branches 0 tags [Go to file](#) [Add file](#) [Code](#)

**Your main branch isn't protected**  
Protect this branch from force pushing, deletion, or require status checks before merging

**aslaugfoss** Add files via upload

Eksempler.R	Add files via upload
Losninger.R	Add files via upload
Oppgaver.R	Add files via upload
Presentasjon.pdf	Add files via upload

**Clone** ⓘ

HTTPS SSH GitHub CLI

[https://github.com/statisticsnorway/R\\_kontrollfunksjoner](https://github.com/statisticsnorway/R_kontrollfunksjoner)

Use Git or checkout with SVN using the web URL.

Open with GitHub Desktop

Download ZIP

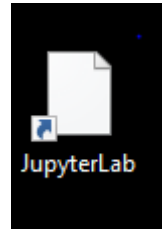
1 / months ago



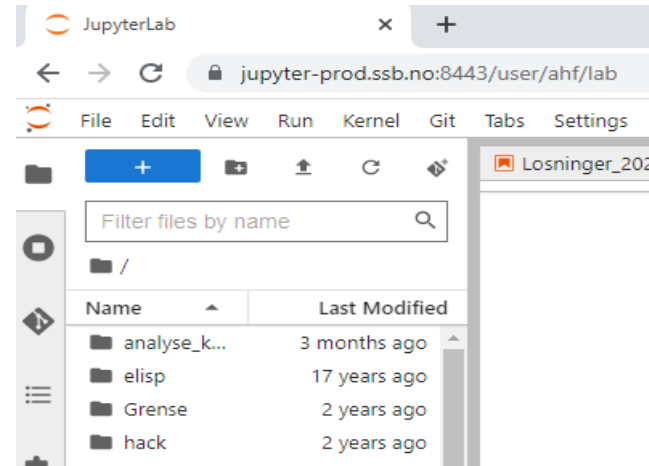
**Statistisk sentralbyrå**  
Statistics Norway

# Starte opp Jupyter i produksjonssonen

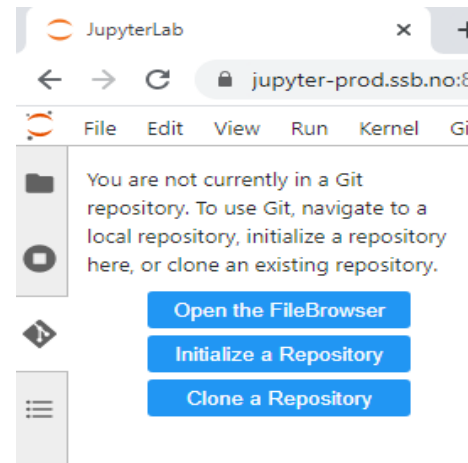
- Trykk på ikonet:



- Stå i «filutforsker»



- Trykk på github ikonet:



## Clone a repo

Enter the Clone URI of the repository

<https://host.com/org/repo.git>

Cancel

CLONE

# Oppgave 1. Frafall og historisk imputering

- 1a) Les inn datasett «fysio» (csv eller RData) og beregn hvor stort frafallet er i foreløpige tall
- 1b) Hva er konsekvensene av å ignorere frafallet?
- 1c) Imputer frafallet i foreløpige tall med forrige års verdi og vurder resultatet. Bruk pakken dcmodify.
- Diskuter resultatet med den du sitter ved siden av!



# Hvordan jobbe med oppgaver

- **Fokuser på metodene:** Kjør programmet «Losninger\_2022» med varierende forklaringsvariabler og med og uten grupper for modellene.
- **Kode metodene selv:** Bruk programmet «Oppgaver\_2022» og kod dine egne løsninger



# Tilfeldig frafall og feil: Modellbasert og donor imputering



# Frafall



Bildets kilde: <https://wiki.ssb.no/display/s880/Imputering+-+seminar?preview=/149954038/149954042/Imputering.pptx>



**Statistisk sentralbyrå**  
Statistics Norway



# Gjennomsnitts-imputering



# Gruppering

- Dele populasjonen inn i homogene grupper (strata)





# Stratifisert gjennomsnitts-imputering



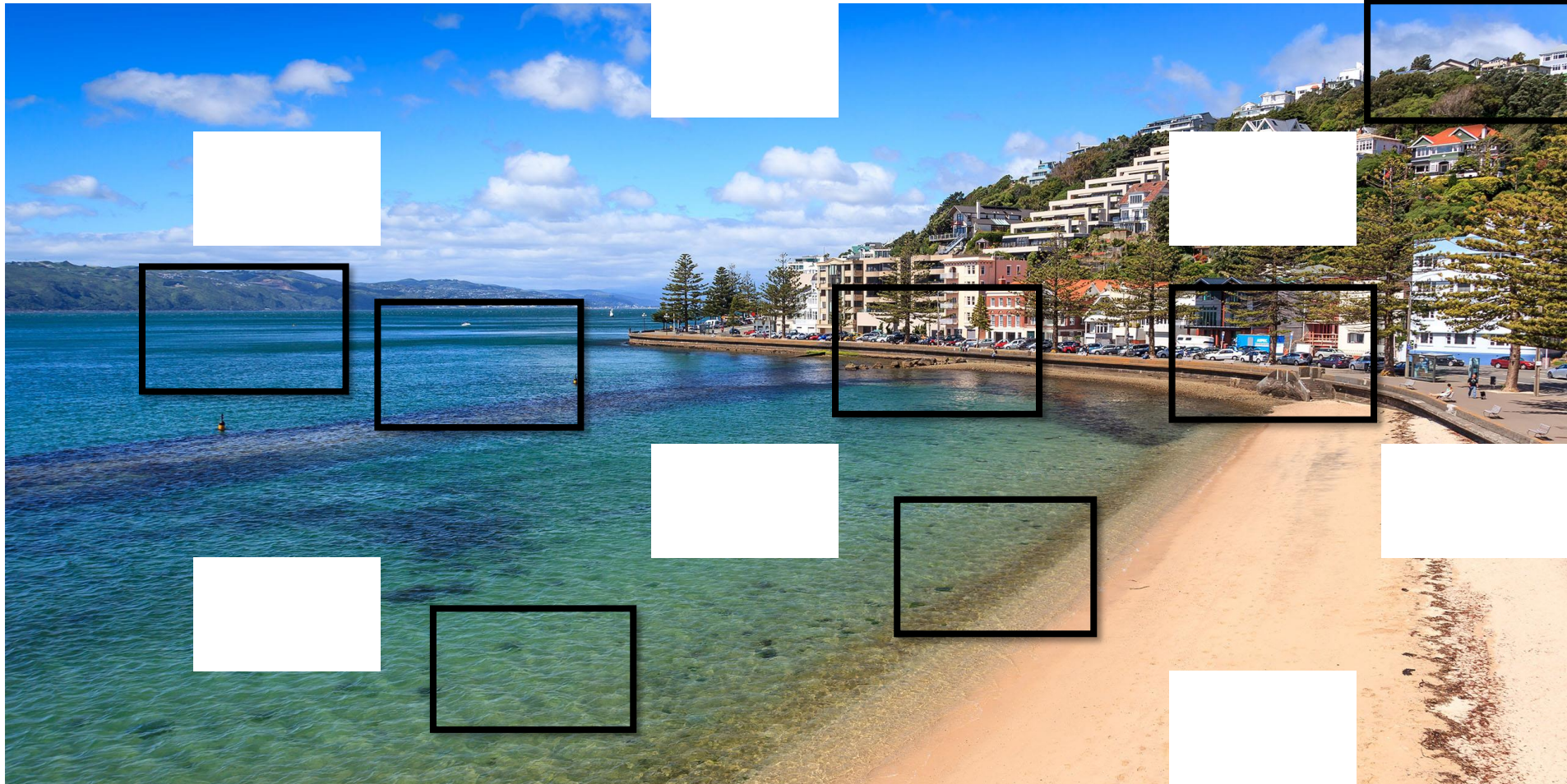


# Stratifisert gjennomsnitts-imputering





# Tilfeldig Hot-deck imputering



Bildets kilde: <https://wiki.ssb.no/display/s880/Imputering+-+seminar?preview=/149954038/149954042/Imputering.pptx>



**Statistisk sentralbyrå**  
Statistics Norway

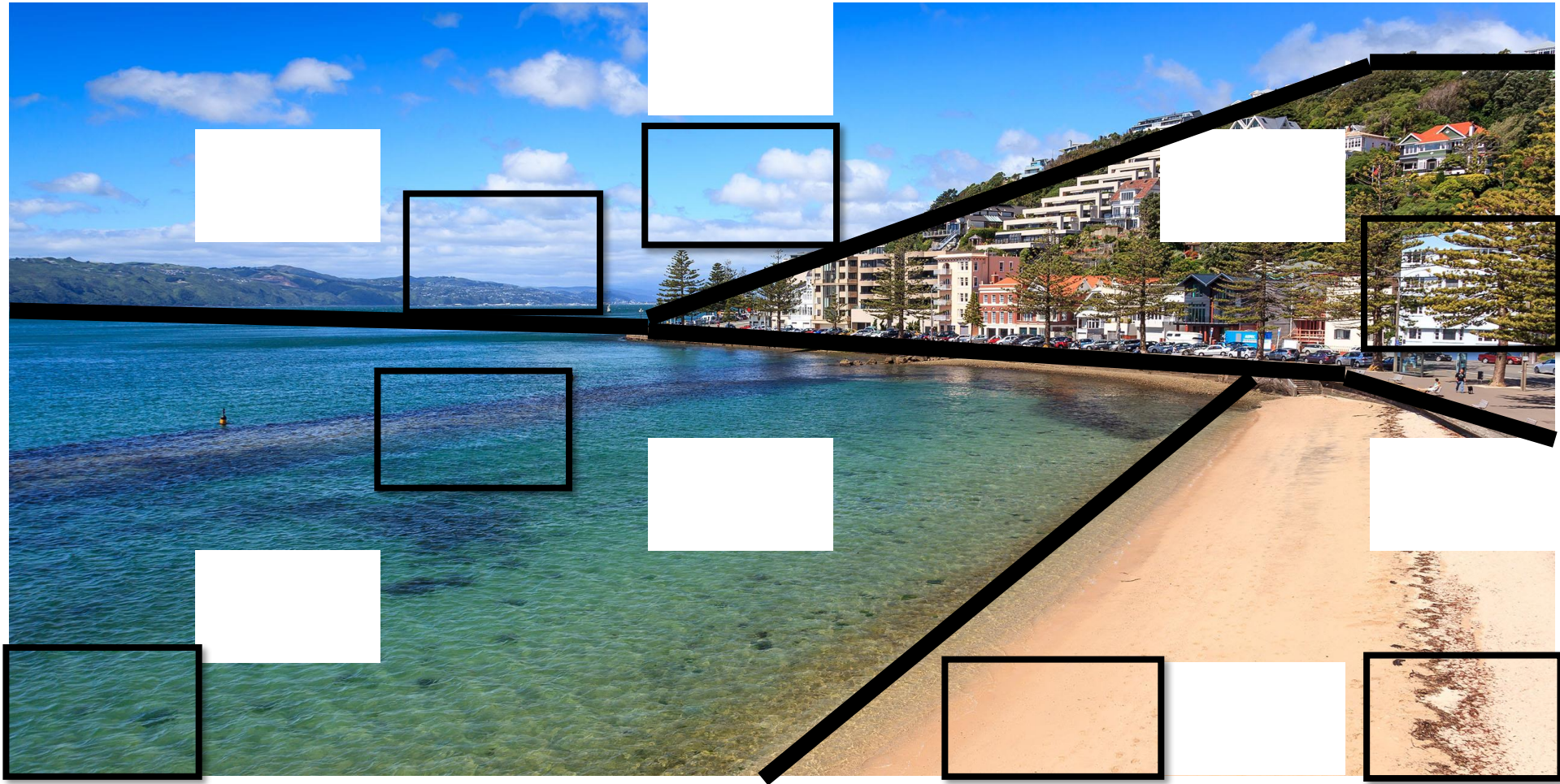


# Tilfeldig Hot-deck imputering





# Stratifisert tilfeldig hot-deck imputering



Bildets kilde: <https://wiki.ssb.no/display/s880/Imputering+-+seminar?preview=/149954038/149954042/Imputering.pptx>



**Statistisk sentralbyrå**  
Statistics Norway



# Stratifisert tilfeldig hot-deck imputering



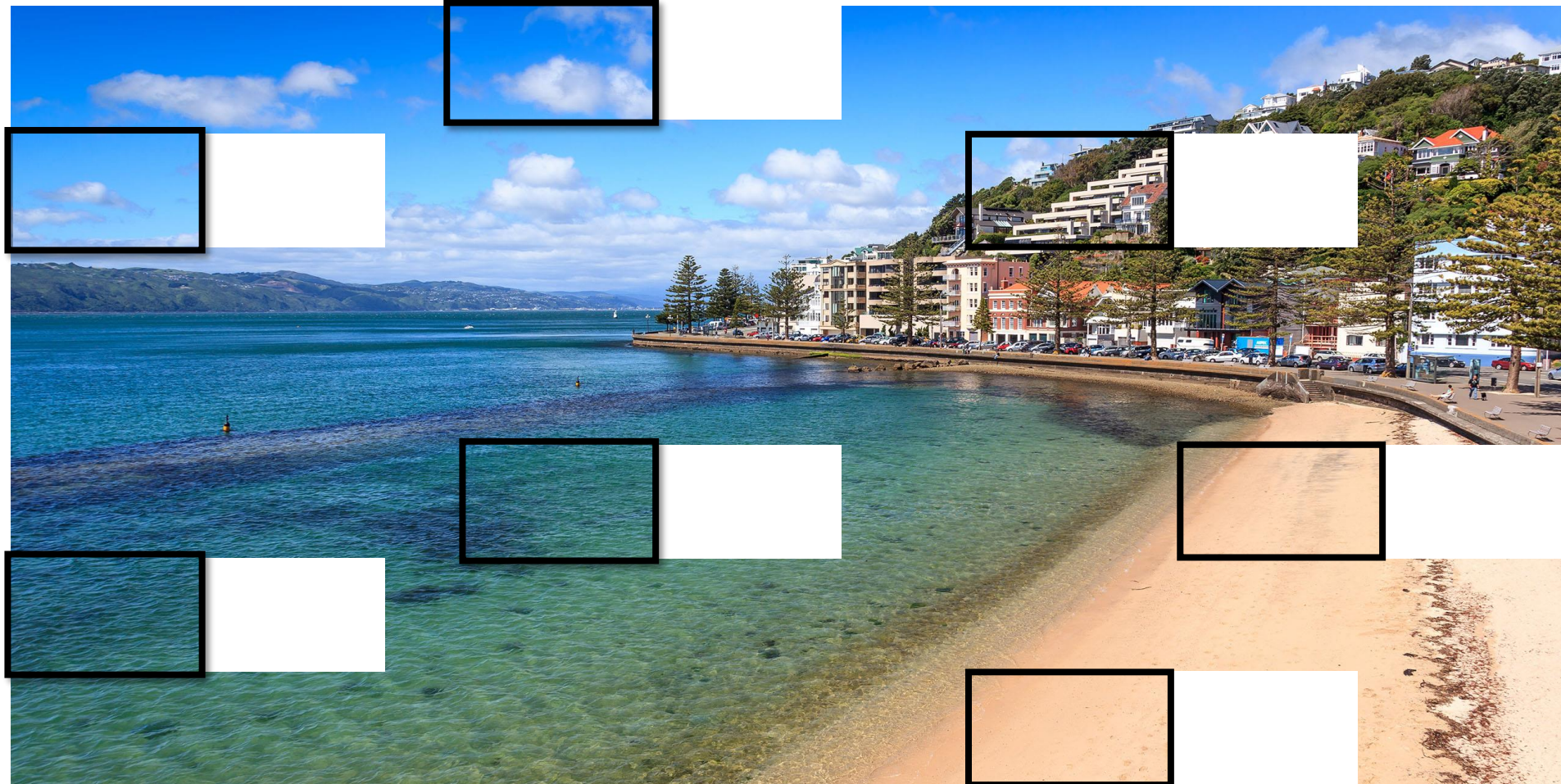
Bildets kilde: <https://wiki.ssb.no/display/s880/Imputering+-+seminar?preview=/149954038/149954042/Imputering.pptx>



**Statistisk sentralbyrå**  
Statistics Norway



# Sekvensiell Hot-deck imputering





# Sekvensiell Hot-deck imputering





# Stratifisert nærmeste nabo imputering



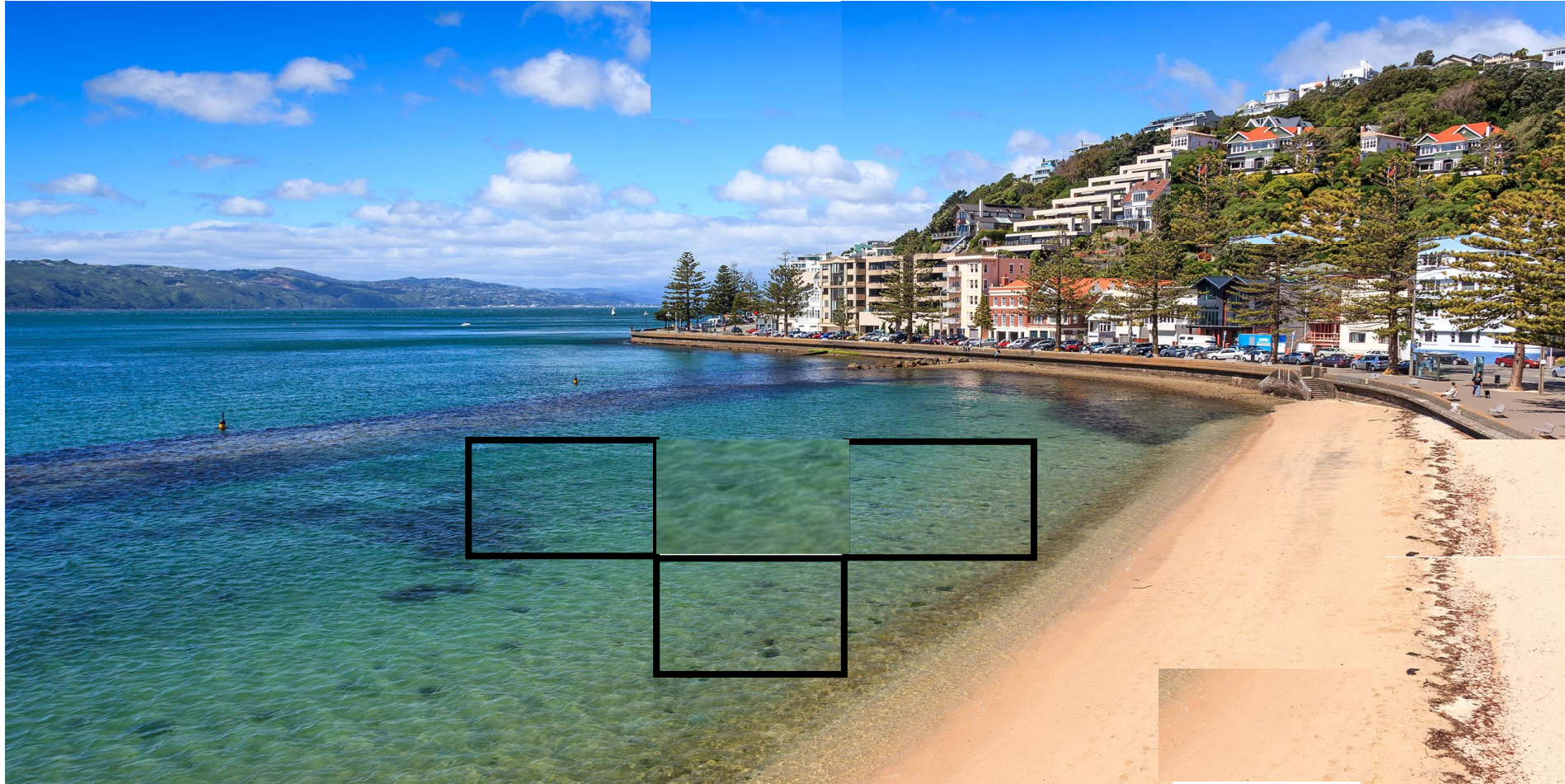


# Stratifisert nærmeste nabo imputering



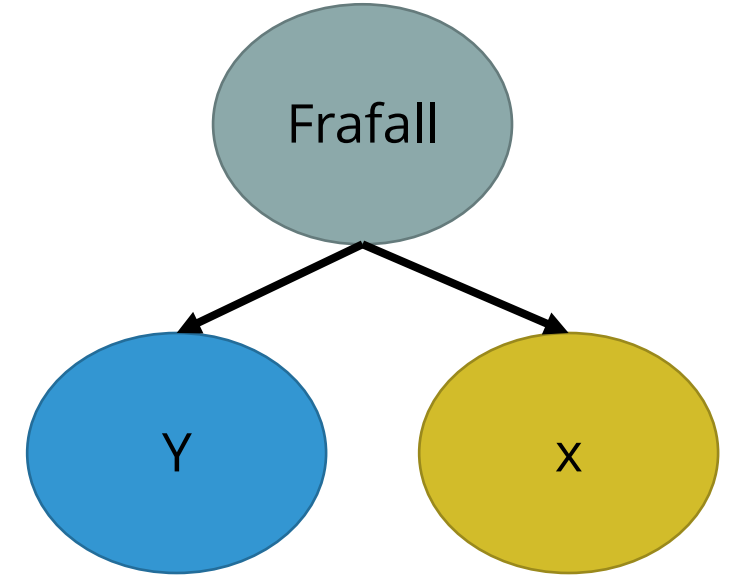


# K-nærmeste nabo imputering



# Tilfeldig frafall eller feil

- Missing Completely At Random (MCAR): frafall avhenger ikke av intereesvariabel  $y_i$  eller hjelpevariabel  $x_i$ 
  - Svar-frafall kan ignoreres i utvalgsundersøkelser
- Missing At Random (MAR): frafall avhenger av hjelpevariabel  $x_i$ , men ikke av interessevariabel  $y_i$ 
  - Vi kan modellere svar-frafall
- Not Missing At Random (NMAR): frafall avhenger av både  $y_i$  (variable av interesse) og  $x_i$  (hjelpevariabel)
  - Modellering ønskelig, men kan ikke forvente en perfekt modell
  - Mest vanlig i virkeligheten. Vanlig behandlet som MAR



# Typer av imputering:

- *Multivariat* imputering: imputerer **mange** variable samtidig
- *Univariat* imputering: imputerer **en og en** variabel separat
- ***Enkel*** imputering: bruke resultater fra et “rimelig” datasett
- ***Multippel*** imputering: kombinere resultater fra flere “rimelige” datasett





# Regresjons-imputering

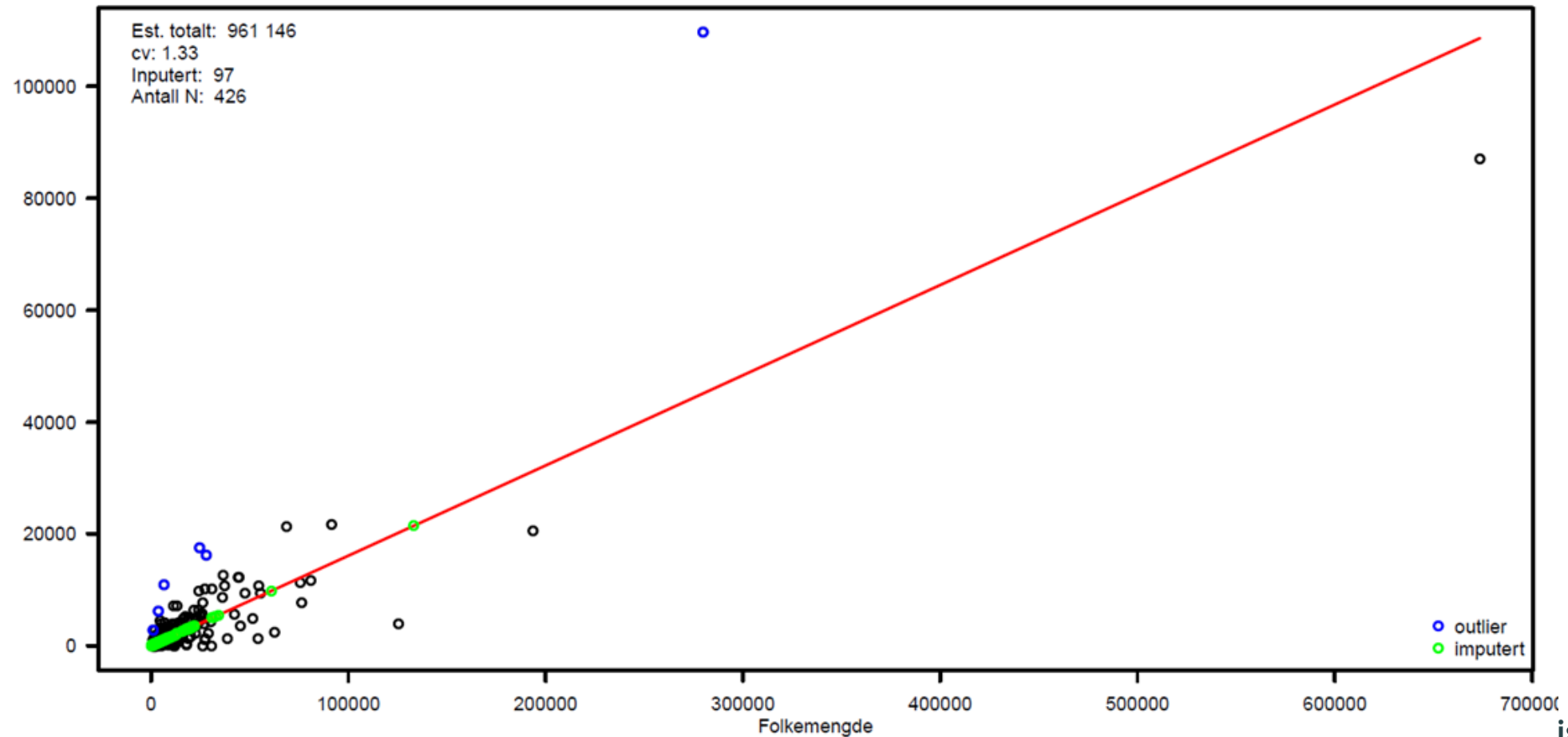
$y_i^* = f(x_i) + e_i$ , der  $f$  ble bygget basert på  $\{(x_i, y_i): i \in s_r\}$

- Lineær regresjon
- robust lineær regresjon



# Regresjons-imputering

AVL\_KALK\_AVSKR\_350 Regresjonsimputering

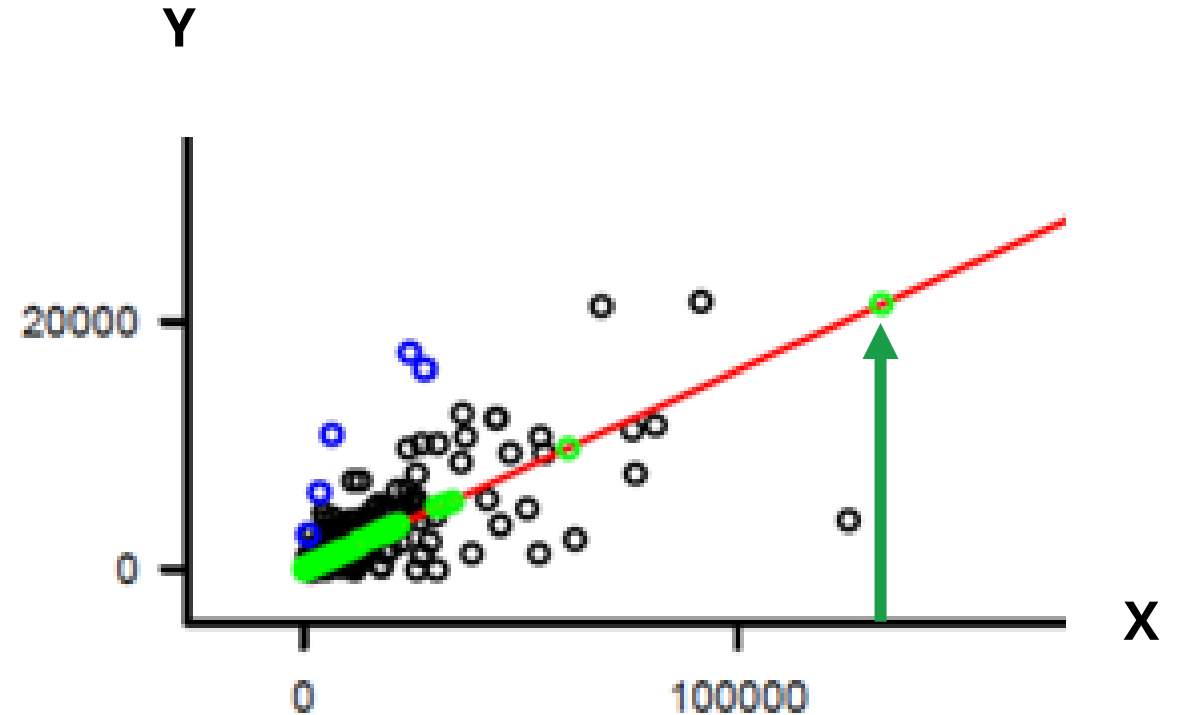


# Regresjons-imputering

- Lager en lineær modell av data:

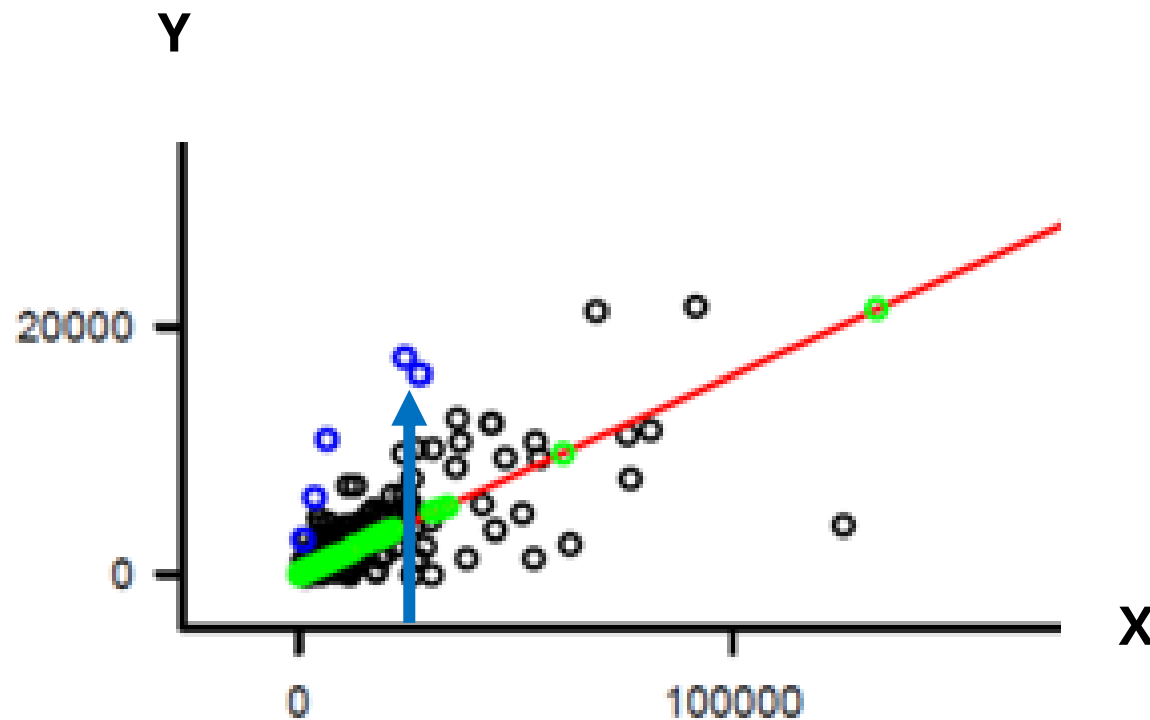
$$y_i = \beta_1 + \beta_2 x_i + e_i$$

- $\hat{\beta}$  ble estimert basert på data som har svart  $\{(x_i, y_i): i \in s_r\}$
- Vi predikerer  $y_i^* = \beta_1 + \beta_2 x_i^*$



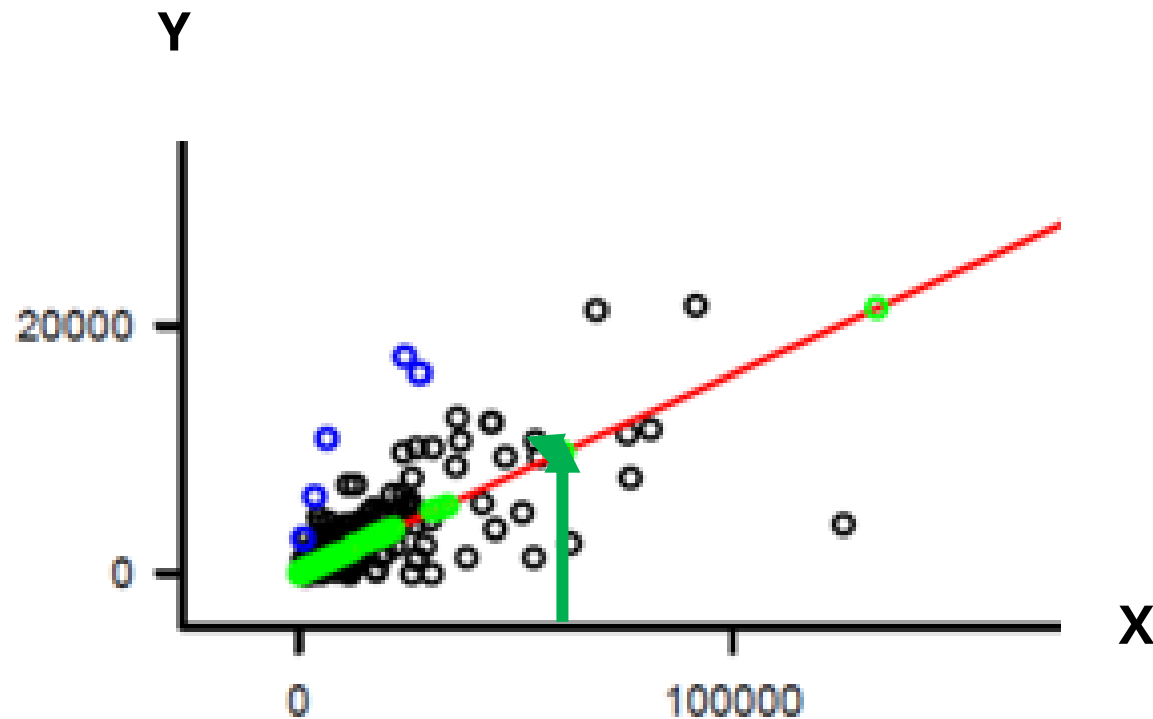
# Robust regresjons-imputering

- Kaster ut outliere når den lineære modellen skal estimeres
- Gir en mindre vekt til outliere når den lineære modellen estimeres
  - $(y_i - w_i y_i^*)^2 \rightarrow \min$
  - For least squares alle  $w_i = 1$
  - For Robust regresjon vekt  $w_i$  er mindre for «influential points»



# Predictive mean matching

- Lager en lineær modell av data
- Den nærmeste observasjonen på regresjonslinja donerer sin y verdi



Bildets kilde: <https://www.youtube.com/watch?v=tUuS10HtadQ>



**Statistisk sentralbyrå**  
Statistics Norway

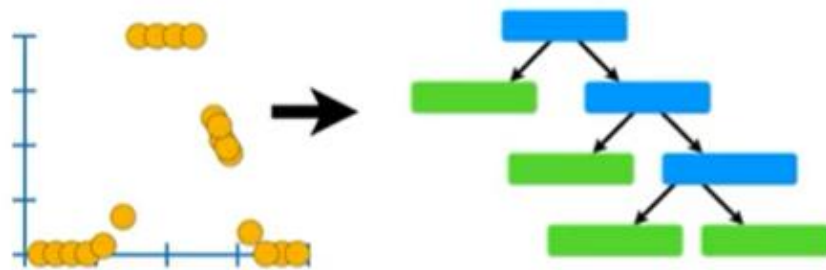
# Imputering med kostra-pakke

- Robust regresjon
    - kaster ut ekstremverdier iterativt – store standardiserte residualer
    - Kan velge flere modeller
    - Beregner usikkerhet – variasjonskoeffisient
  - Historisk imputering
    - Finner den siste observerte verdien
    - Logger hvilken periode den er fra
    - Beregner usikkerhet - variasjonskoeffisient
- `ImputeRegression()`
  - `ImputeHistory()`



# Ikke lineær sammenheng?

## Regression Trees....

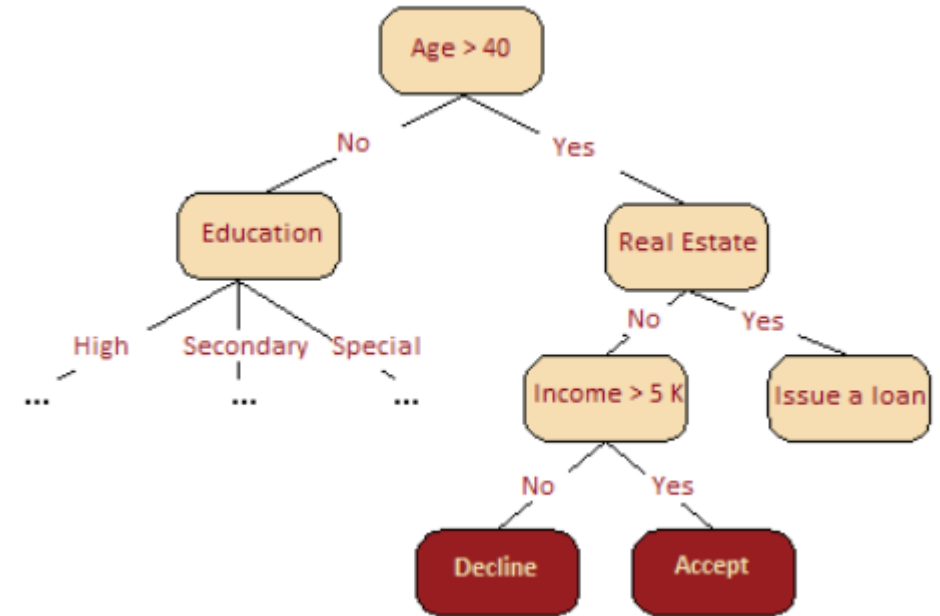


- CART Classification and Regression Trees models
- Random forest

Fin forklaring av modellene:

<https://www.youtube.com/watch?v=g9c66TUylZ4>

[https://www.youtube.com/watch?v=J4Wdy0Wc\\_xQ](https://www.youtube.com/watch?v=J4Wdy0Wc_xQ)



# Hvordan velge imputeringsmetode?

- Bruk fagkunnskap og vurder metodene
- Beregne feilen – RMSE Treningsdata – testdata
- Se på makronivå
- Se på grafikk - plot mot forrige år, hjelpe variabler
- Se på variasjonskoeffisient  $cv = \frac{\sigma}{\mu}$





# R-pakken *simputation*

Flere pakker for imputering (mice, VIM, Amelia, mi, ...), men:

- Simputation gir et *uniformt grensesnitt* for ofte brukt metoder
- Simputation er en pakke for å gjøre imputering enklere!

Laget av Mark van der Loo and Edwin de Jonge, Statistics Netherlands

Mer info: <https://cran.r-project.org/web/packages/simputation/vignettes/intro.html>

og: <https://cran.r-project.org/web/packages/simputation/simputation.pdf>



# Tilgjengelige imputeringsmetoder

## Regresjons-imputering

- linear regression (**\_lm**)
- robust linear regression (**\_rlm**)
- ridge/elasticnet/lasso regression (**\_en**)
- CART models (decision trees) (**\_cart**)
- Random forest (**\_rf**)

## Multivariate imputering

- Imputation based on the expectation-maximization algorithm (**\_em**)
- missForest (=iterative random forest imputation) (**\_mf**)

## Hot-deck imputering

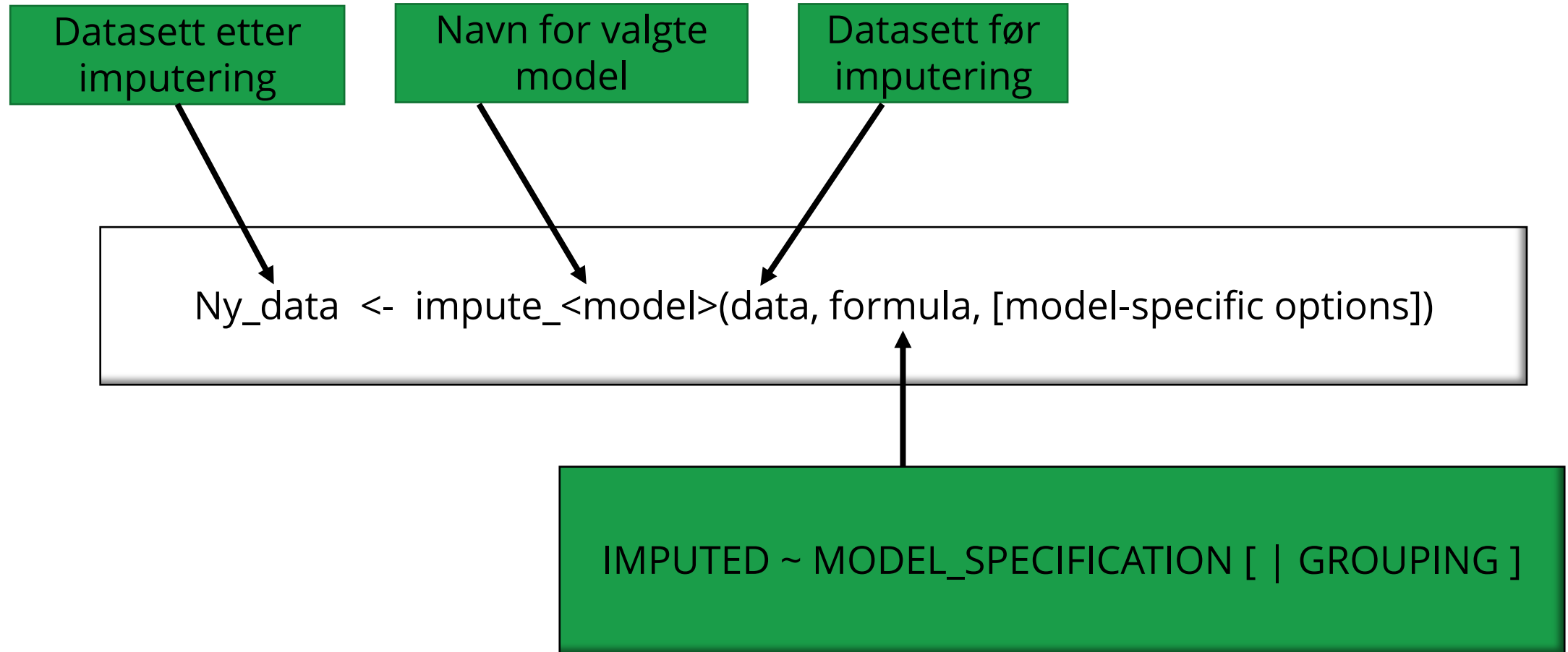
- k-nærmest nabo (based on gower's distance) (**\_knn**)
- sequential hotdeck (LOCF, NOCB) (**\_shd**)
- random hotdeck (**\_rhd**)
- Predictive mean matching (**\_pmm**)

## Andre

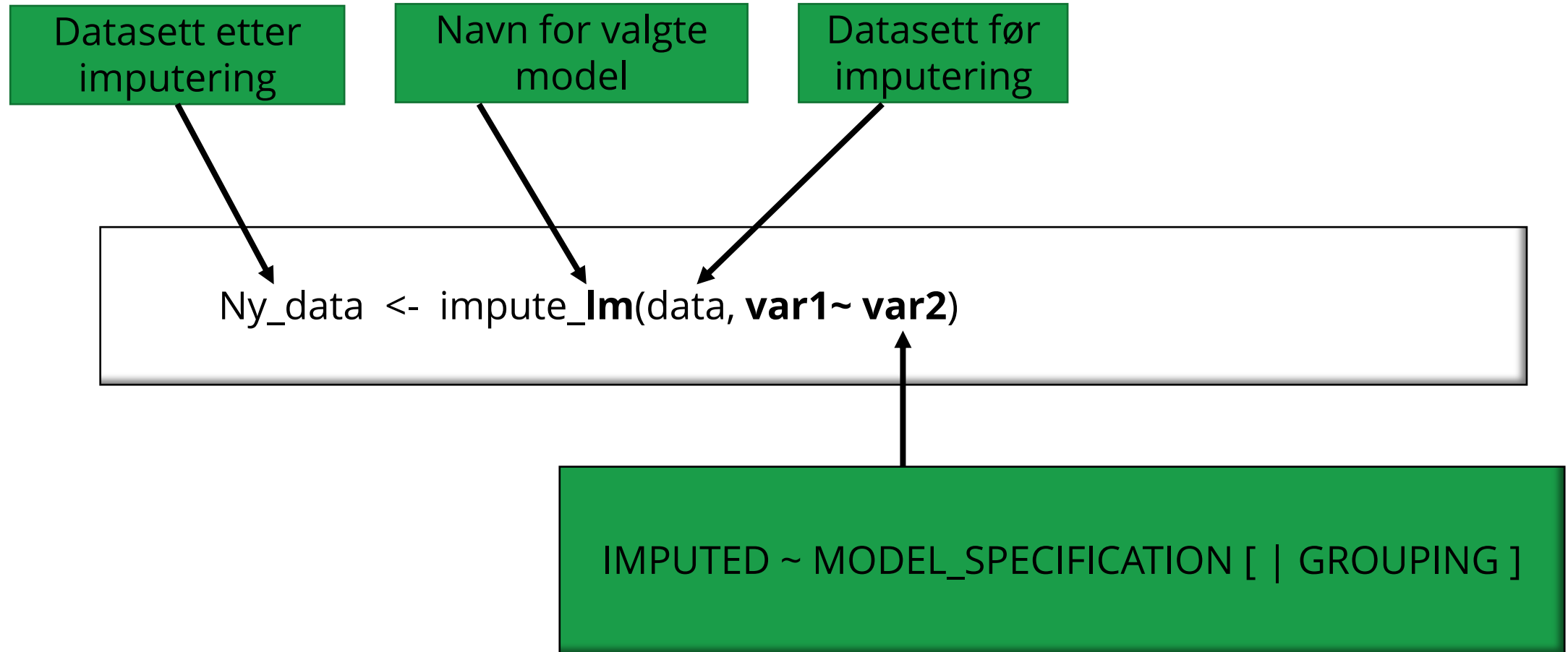
- (groupwise) median imputation (optional random residual) (**\_median**)
- Proxy imputation: copy another variable or use a simple transformation to compute imputed values. (**\_proxy, \_constant**)



# Imputation interface



# Simputation grensesnitt: lineær regresjon



# Imputeringskjede

Skrive flere imputeringer i pipeline

```
library(magrittr)

newdata<- mydata %>%
  impute_lm(var1 ~ var2) %>%
  impute_median(var1) %>%
  impute_cart(var3 ~ .)
```



# Imputerer flere variabler samtidig

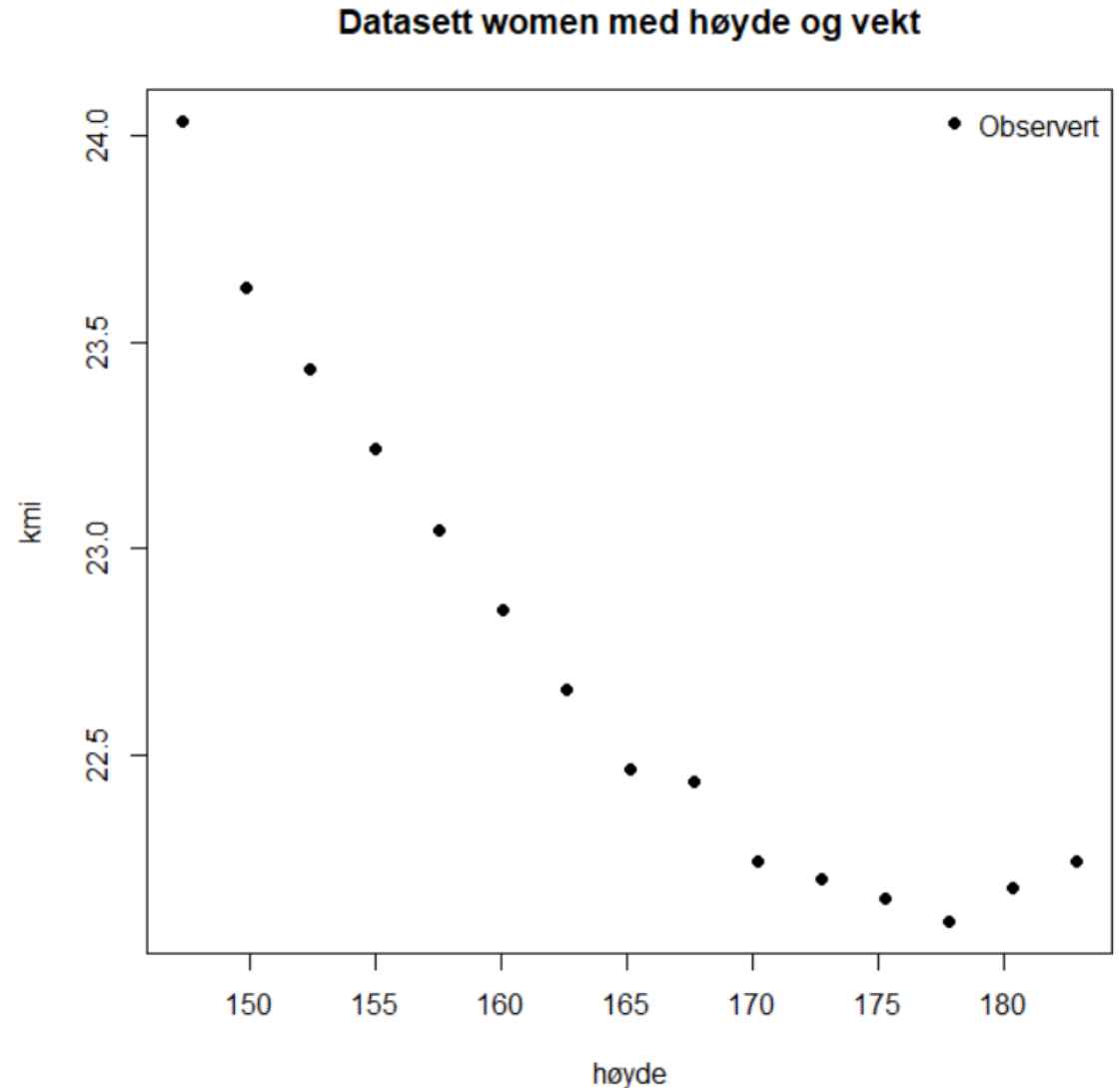
Imputere flere variable samtidig med lik modell

```
newdata <- impute_rlm(mydata, var1 + var2 ~ var3)
```



# Eksempler – datasett women

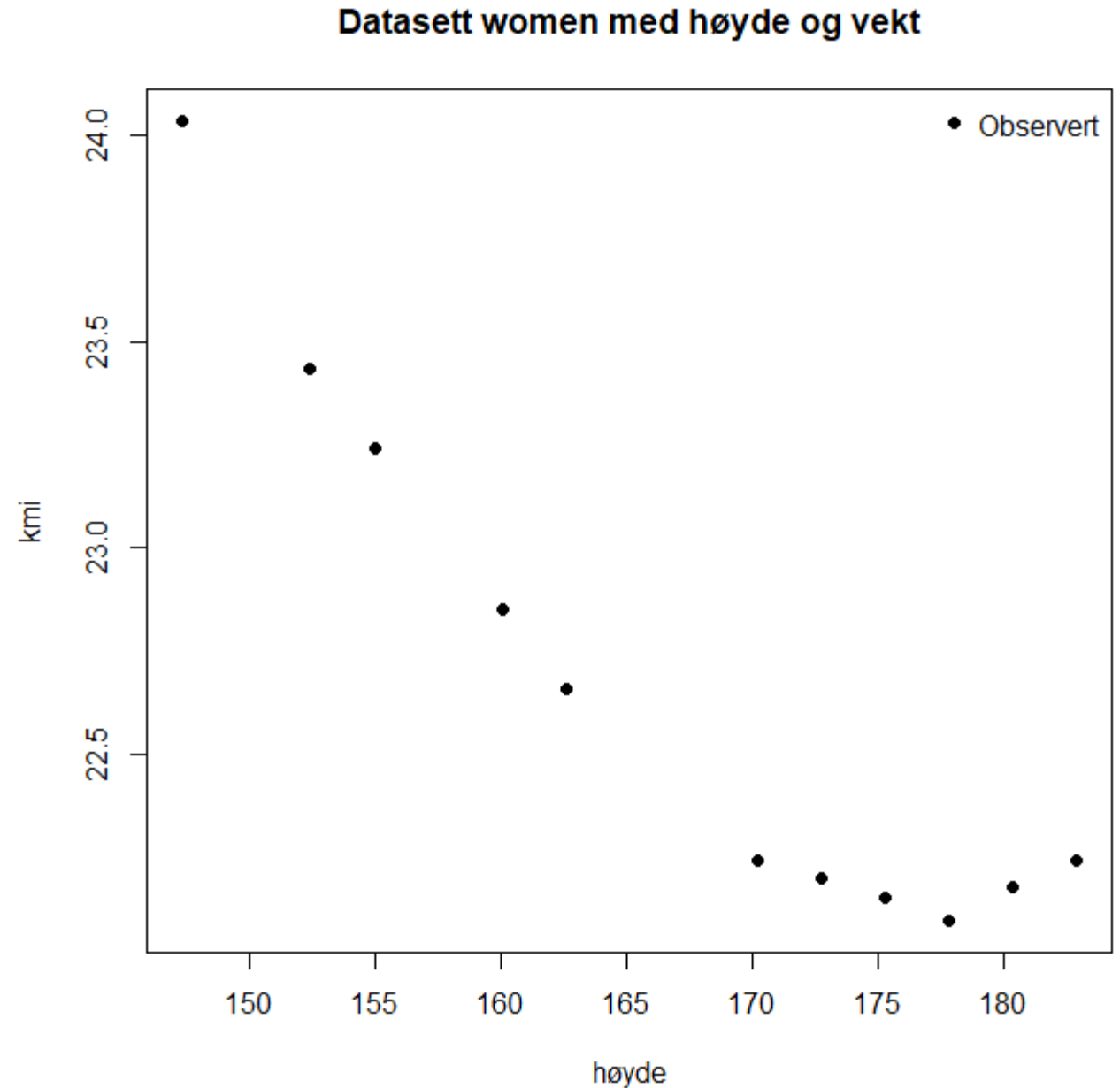
- 15 personer
- Variabler høyde og vekt
- Beregner KMI (BMI)
- Tar ut verdien for kmi for 4 personer som vi skal imputere



# Vurdering av modell

- Grafikk
- Størrelse på feil

$$RMSE = \sqrt{\frac{\sum_{i=1}^N \|y(i) - \hat{y}(i)\|^2}{N}},$$





# Eksempel i Jupyter med datasett women



# Oppgave 2 testing av imputeringsmetoder

- a) Imputer med gjennomsnittet innen hver kostragruppe og vurder resultatet.
  - Hva blir totalen nå?
  - Hvor stor blir feilen - RMSE? sammenlign med endelige tall
  - Bruk grafikk til å vurder hvor god metoden er
- b) Random hotdeck
- c) Nærmeste nabo
- d) Regresjon
- e) Prediktiv mean matching



# Hvordan jobbe med oppgaver

- **Fokuser på metodene:** Kjør programmet «Losninger\_2022» med varierende forklaringsvariabler og med og uten grupper for modellene.
- **Kode metodene selv:** Bruk programmet «Oppgaver\_2022» og kod dine egne løsninger



# Gruppeoppgave

- Kan noen av metodene som er testet i dag bli brukt?
- Hva slags krav må metode tilfredsstille for at det skal bli godt nok?



# Logging og kvalitetsindikatorer

# Dokumentasjon av imputering

- Lag en variabel som dokumenter hvilken verdi som er endret
- Logg gammel og ny verdi

key	variable	old	new
<dbl>	<chr>	<int>	<int>
1.003464e+14	varighet3	3	NA
1.003844e+13	KvpStonad	335846	212798

id	hoyde	vekt	kmi	kmi_org	imp	
1	147.32	52.16308	24.03476	24.03476	1	g
2	149.86	53.07026	NA	23.63087	2	g
3	152.40	54.43104	23.43563	23.43563	1	g
4	154.94	55.79182	23.24039	23.24039	1	g
5	157.48	57.15259	NA	23.04545	2	g
6	160.02	58.51337	22.85107	22.85107	1	g
7	162.56	59.87414	22.65750	22.65750	1	g
8	165.10	61.23492	NA	22.46493	2	g
9	167.64	63.04929	NA	22.43494	2	g
10	170.18	64.41006	22.24010	22.24010	1	g
11	172.72	66.22443	22.19898	22.19898	1	g
12	175.26	68.03880	22.15088	22.15088	1	g
13	177.80	69.85317	22.09645	22.09645	1	g
14	180.34	72.12113	22.17575	22.17575	1	g
15	182.88	74.38909	22.24215	22.24215	1	g



# Kvalitetsindikatorer for imputering

- Imputeringsrate – editeringsandel
  - «Sum antall imputerte verdier»/ «totalt antall verdier»
  - Eksempel women- bmi IR=4/15=0.267
- Usikkerhet – variasjonskoeffisient -  $cv = \frac{\sigma}{\mu}$ 
  - Usikkerheten skapt av imputering i forhold til estimatet
  - Krever beregning av usikkerheten -lagt inn i kostra-pakken





# Pakken lumberjack for logging av endringer



lumberjack

/ˈlʌmbədʒək/

*noun*

(especially in North America) a person who fells trees, cuts them into logs, or transports them to a sawmill.



**Statistisk sentralbyrå**  
Statistics Norway

# Lagre endringer med pakken *lumberjack*

- Lett å lagre endringer
- Mulig å studere effekt av imputering
- Pipe operator %>>%

```
library(lumberjack)
logger <- cellwise$new(key="ID")

out <- mydata %>>%
  start_log(logger) %>>%
  impute_lm(var1 ~ var2) %>>%
  dump_log(file="mylog.csv", stop=TRUE)
```



# Eksempel: Omsetningsindeksen

```
#rette opp 1000-feil og setter de som har <lik> til missing for å kunne imputere
mod <- modifier(
  if (is.na(OMS)) OMS <- 0,
  if (is.na(NACE)) NACE <- "47111",
  if (is.na(NACE2)) NACE2 <- "47",
  if (OMS_FMND > 0 & OMS > 0 & 750 < OMS/OMS_FMND & OMS/OMS_FMND < 1400) OMS <- OMS/1000,
  if (OMS > 0 & OMS == OMS_FAAR) OMS <- NA,
  if (OMS > 0 & OMS == OMS_FMND) OMS <- NA
)

logger <- cellwise$new(key="ID")

out<- doi %>>%
start_log(logger) %>>%
modify(mod) %>>%
impute_rlm(OMS ~ OMS_FMND + OMS_FAAR) %>>%
impute_rlm(OMS ~ OMS_FMND) %>>%
dump_log(file="minlog.csv", stop=TRUE)
log<-read.csv("minlog.csv")
dim(log)
head(log)
```

step	time	srcref	expression	key	variable	old	new
<int>	<fct>	<lg>	<fct>	<dbl>	<fct>	<int>	<dbl>
1	1	2020-10-15 11:13:14 CEST	NA	modify(mod)	14219230025	OMS	474.146
2	1	2020-10-15 11:13:14 CEST	NA	modify(mod)	14219230026	OMS	213.740
3	1	2020-10-15 11:13:14 CEST	NA	modify(mod)	14219230027	OMS	484.528
4	1	2020-10-15 11:13:14 CEST	NA	modify(mod)	14219230028	OMS	493.670
5	1	2020-10-15 11:13:14 CEST	NA	modify(mod)	14219230029	OMS	529.103
6	1	2020-10-15 11:13:14 CEST	NA	modify(mod)	14219230030	OMS	209.617










# Logger typer

	step	time	srcref	expression	changed
1	1	2021-03-31 13:06:35	NA	start_log(cellwise\$new(key = "id"))	FALSE
2	2	2021-03-31 13:06:35	NA	start_log(expression_logger\$new(mean = mean(height), sd ...	FALSE
3	3	2021-03-31 13:06:35	NA	start_log(filedump\$new(dir = paste0(getwd(), "/filedump_re...	FALSE
4	4	2021-03-31 13:06:35	NA	mutate(women, bmi = weight/height^2)	TRUE
5	5	2021-03-31 13:06:35	NA	mutate(women, height = height * 0.0254)	TRUE

	step	time	srcref	expression	key	variable	old	new
13	4	2021-03-31 13:06:35 CEST	NA	mutate(women, bmi = weight/height^2)	13	bmi	NA	0.03142857
14	4	2021-03-31 13:06:35 CEST	NA	mutate(women, bmi = weight/height^2)	14	bmi	NA	0.03154136
15	4	2021-03-31 13:06:35 CEST	NA	mutate(women, bmi = weight/height^2)	15	bmi	NA	0.03163580
16	5	2021-03-31 13:06:35 CEST	NA	mutate(women, height = height * 0.0254)	1	height	58	1.47320000
17	5	2021-03-31 13:06:35 CEST	NA	mutate(women, height = height * 0.0254)	10	height	67	1.70180000

```
simple$new()
cellwise$new(key = "id")
expression_logger$new(mean=mean(height), sd=sd(height))
filedump$new(dir = paste0(getwd(), "/filedump_res"))
```

	step	srcref	expression	mean	sd
1	1	NA	start_log(expression_logger\$new(mean = mean(height), sd ...	65.000	4.4721360
2	2	NA	start_log(filedump\$new(dir = paste0(getwd(), "/filedump_re...	65.000	4.4721360
3	3	NA	mutate(women, bmi = weight/height^2)	65.000	4.4721360
4	4	NA	mutate(women, height = height * 0.0254)	1.651	0.1135923
5	5	NA	dump_log("simple")	1.651	0.1135923
6	6	NA	dump_log("cellwise")	1.651	0.1135923

<input type="checkbox"/>		._step000.csv	180 B	Mar 31, 2021, 1:06 PM
<input type="checkbox"/>		._step001.csv	180 B	Mar 31, 2021, 1:06 PM
<input type="checkbox"/>		._step002.csv	464 B	Mar 31, 2021, 1:06 PM
<input type="checkbox"/>		._step003.csv	521 B	Mar 31, 2021, 1:06 PM
<input type="checkbox"/>		._step004.csv	521 B	Mar 31, 2021, 1:06 PM
<input type="checkbox"/>		._step005.csv	521 B	Mar 31, 2021, 1:06 PM
<input type="checkbox"/>		._step006.csv	521 B	Mar 31, 2021, 1:06 PM



# Kvalifiseringsprogrammet - automatisk korrigering

med pakken dcmodyfy, simputation og logging med pakken lumberjack

```
library(dcmodyfy)
library(simputation)
library(lumberjack)
kval3$varighet3<-kval3$varighet
G<-106399
#Barnetillegg 27 kr itdager i uken per barn
barnt<-27

regler <- modifier( if (KvpStonad > (2*G) + Antbu18*barnt*52*5 + 70000)
                    KvpStonad<-2*G + 52*5*Antbu18*barnt,
                    if (varighet != varighet2) varighet3<- NA
                    )

#Logfil
logfile1 <- tempfile(fileext=".csv")
logfile2 <- tempfile(fileext=".csv")

kval3$ID<- as.character(paste(kval3$PersonFodselsnr, kval3$KommuneNr, sep = ""))

out <- kval3 %L>%
  start_log(cellwise$new(key="ID")) %L>%
  start_log(expression_logger$new(tot_stonad=sum(KvpStonad), mean_varighet=mean(varighet3, na.rm=TRUE)) ) %L>%
  modify(regler) %L>%
  impute_pmm(varighet3~ KvpStonad -1) %L>%
  dump_log("cellwise",file=logfile1) %L>%
  dump_log("expression_logger",file=logfile2,stop=TRUE)

a <-data.frame(read.csv(logfile1))
nrow(a)
head(a)
read.csv(logfile2)
```



A data.frame: 6 × 8

	step	time	srcref	expression	key	variable	old	new
	<int>	<chr>	<lgl>	<chr>	<dbl>	<chr>	<int>	<int>
1	2	2021-09-22 12:03:04 UTC	NA	modify(regler)	1.003464e+14	varighet3	3	NA
2	2	2021-09-22 12:03:04 UTC	NA	modify(regler)	1.003844e+13	KvpStonad	335846	212798
3	2	2021-09-22 12:03:04 UTC	NA	modify(regler)	1.008282e+14	varighet3	3	NA
4	2	2021-09-22 12:03:04 UTC	NA	modify(regler)	1.008623e+14	varighet3	5	NA
5	2	2021-09-22 12:03:04 UTC	NA	modify(regler)	1.008761e+14	varighet3	9	NA
6	2	2021-09-22 12:03:04 UTC	NA	modify(regler)	1.008966e+13	KvpStonad	285917	212798

A data.frame: 4 × 5

step	srcref	expression	tot_stonad	mean_varighet
<int>	<lgl>	<chr>	<int>	<dbl>
1	NA	start_log(expression_logger\$new(tot_stonad = sum(KvpStonad), mean_varighet = mean(varighet3, na.rm = TRUE)))	1339229279	7.688193
2	NA	modify(regler)	1295352854	9.029140
3	NA	impute_pmm(varighet3 ~ KvpStonad - 1)	1295352854	7.673370
4	NA	dump_log("cellwise", file = logfile1)	1295352854	7.673370



# Eksempel Jupyter





# Gruppeoppgave

- Vil det være vanskelig å logge endringer som blir gjort i den statistikken du jobber med?
- Blir det laget kvalitetsindikatorer i din statistikk for editering?
- Hva er fordelen med å ha kvalitetsindikatorer?



# Øvelser: del 3

- Oppgave 3. Velg endelig modell for imputering og sett opp logging av endring av verdier og total



# Oppsummering

# Takk!

<https://github.com/SNStatComp/awesome-official-statistics-software>

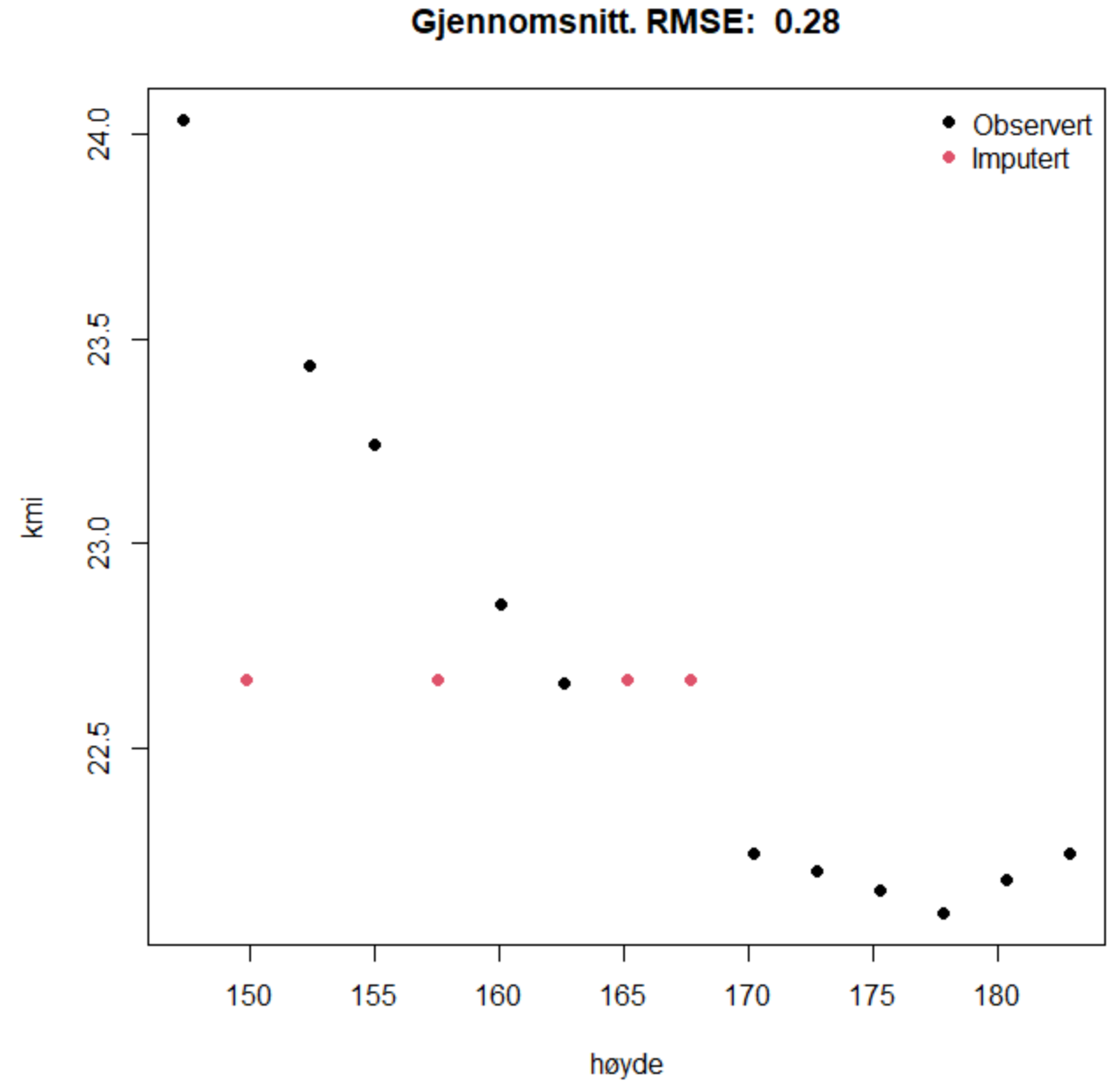


**Statistisk sentralbyrå**  
Statistics Norway

# Gjennomsnitt

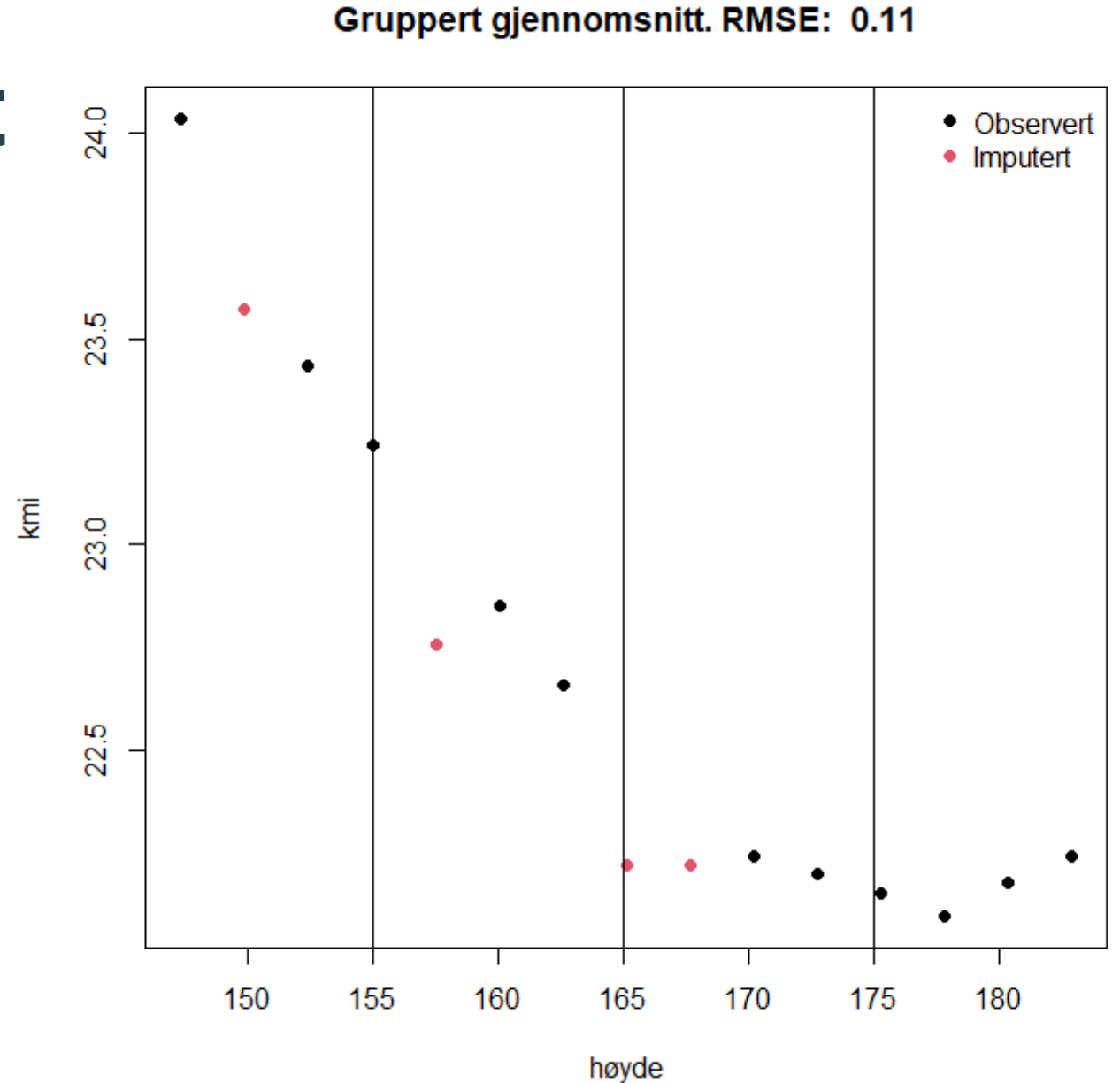
- Kode:

```
impute_proxy(kmi ~ mean(kmi, na.rm = TRUE))
```



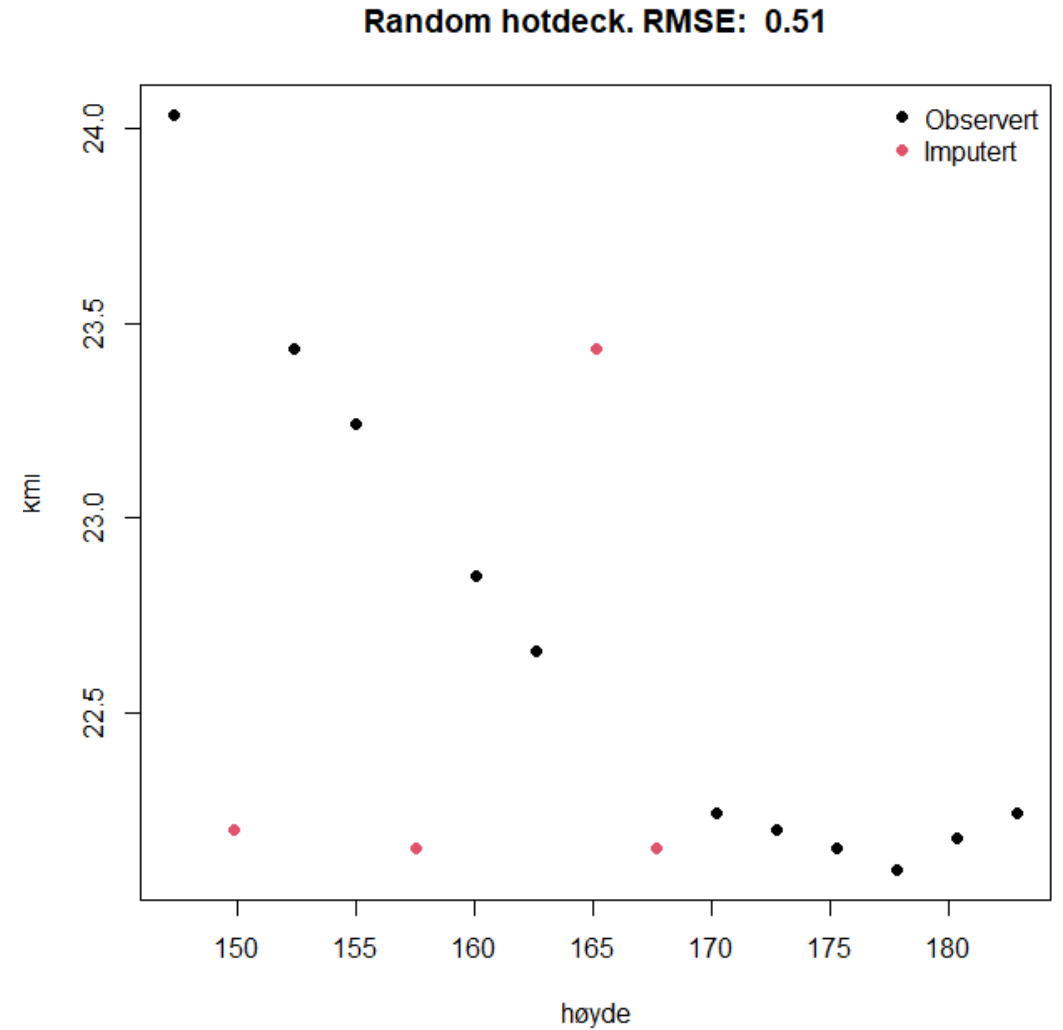
# Gruppert gjennomsnitt

- Kode:
- `impute_proxy(kmi ~ mean(kmi, na.rm = TRUE)|gruppe)`
- `gruppe <- cut(women$hoyde, breaks = c(0, 155, 165, 175, 190), labels = c("gr1", "gr2", "gr3", "gr4"))`



# Random hotdeck

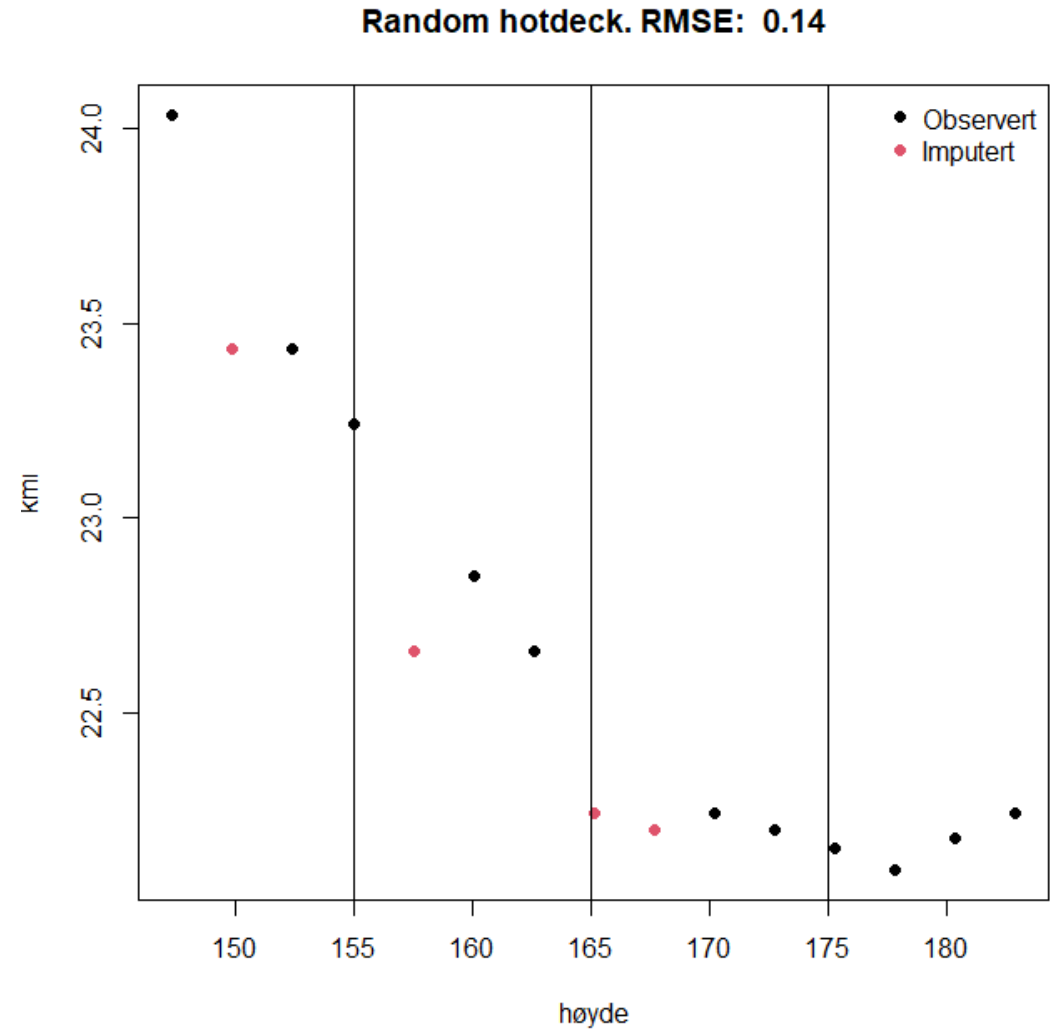
- `impute_rhd(kmi ~ 1, pool = "complete" )`





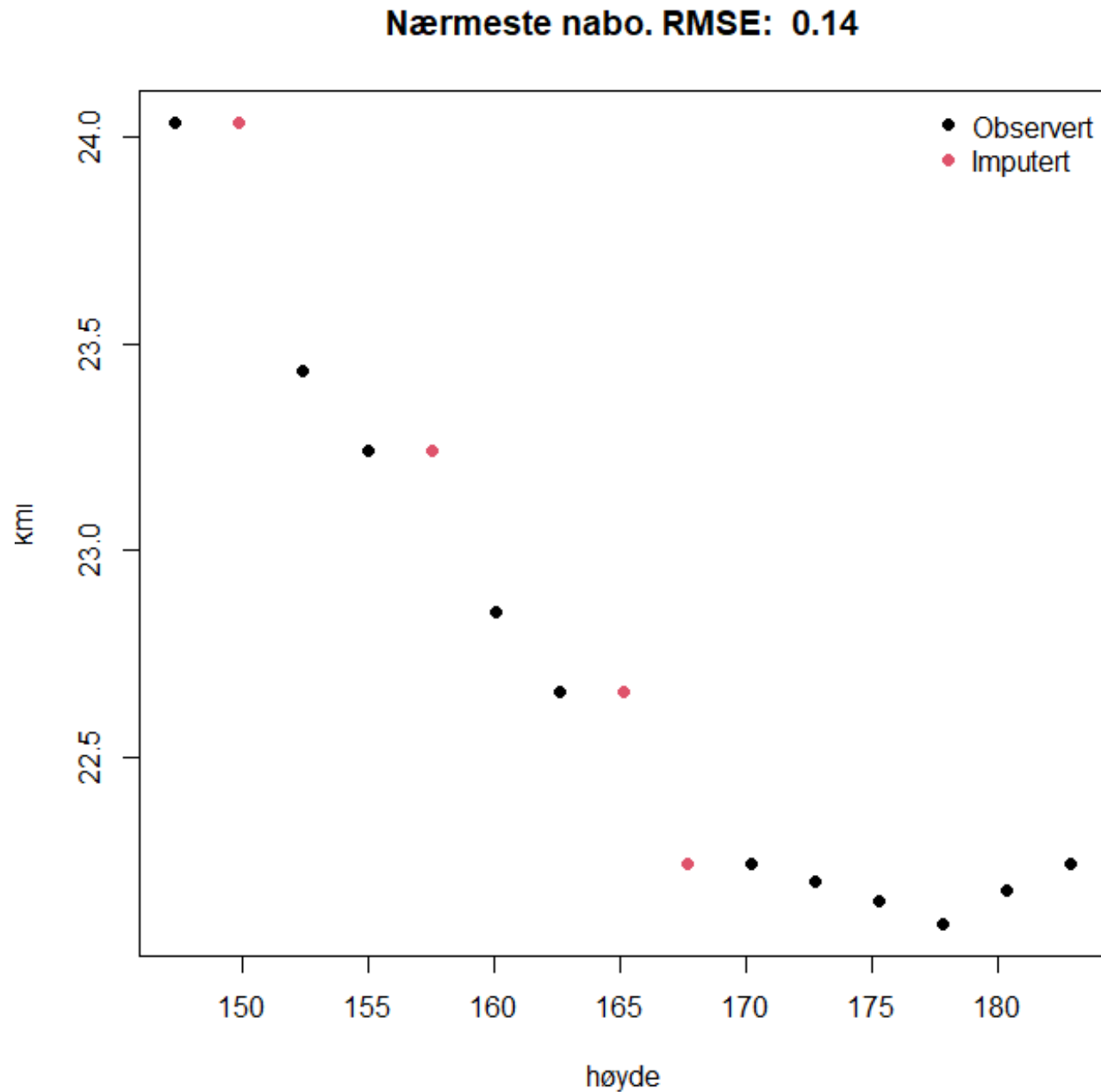
# Random hot deck gruppe

- Kode:
- `impute_rhd(kmi ~ 1 | gruppe, pool = "complete" )`



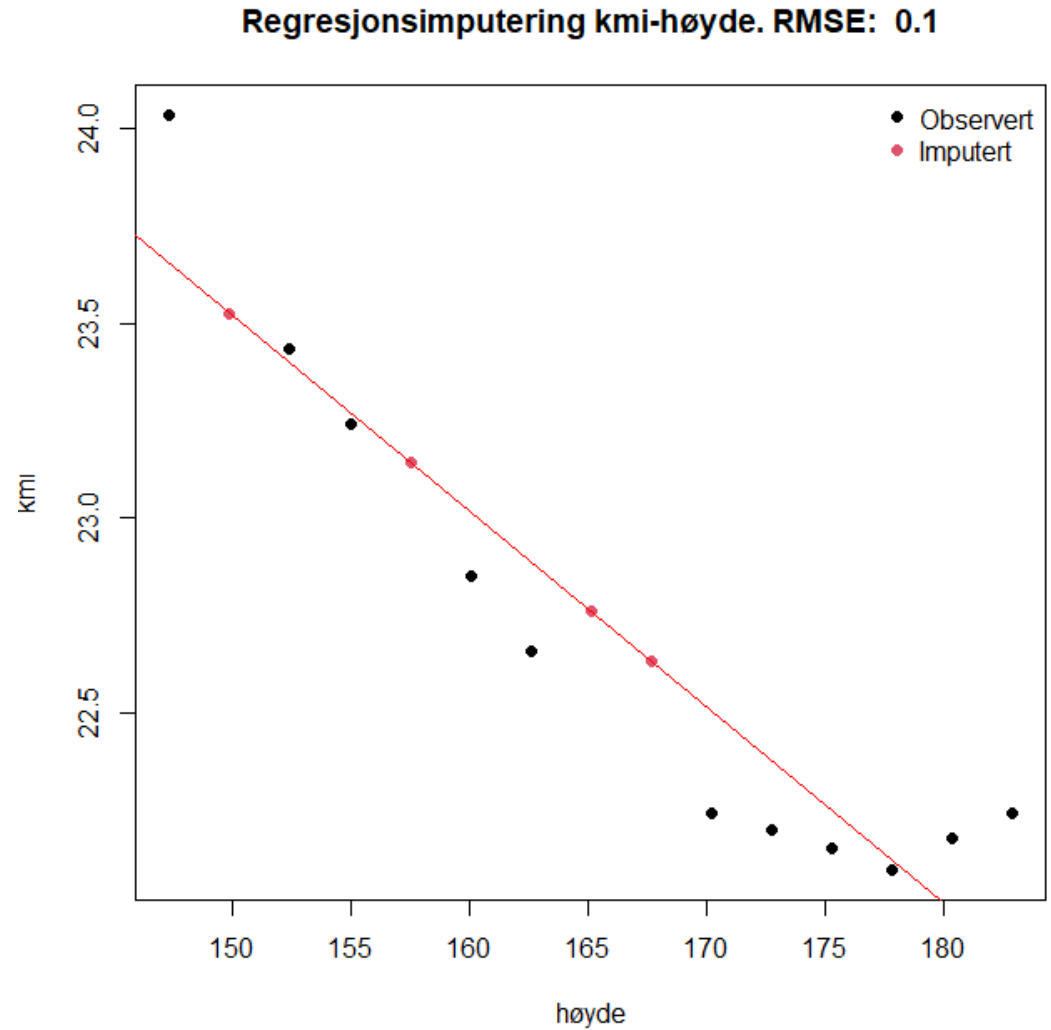
# Nærmeste nabo imputering

- Kode:
- `impute_knn(kmi ~ vekt + hoyde, k = 1)`



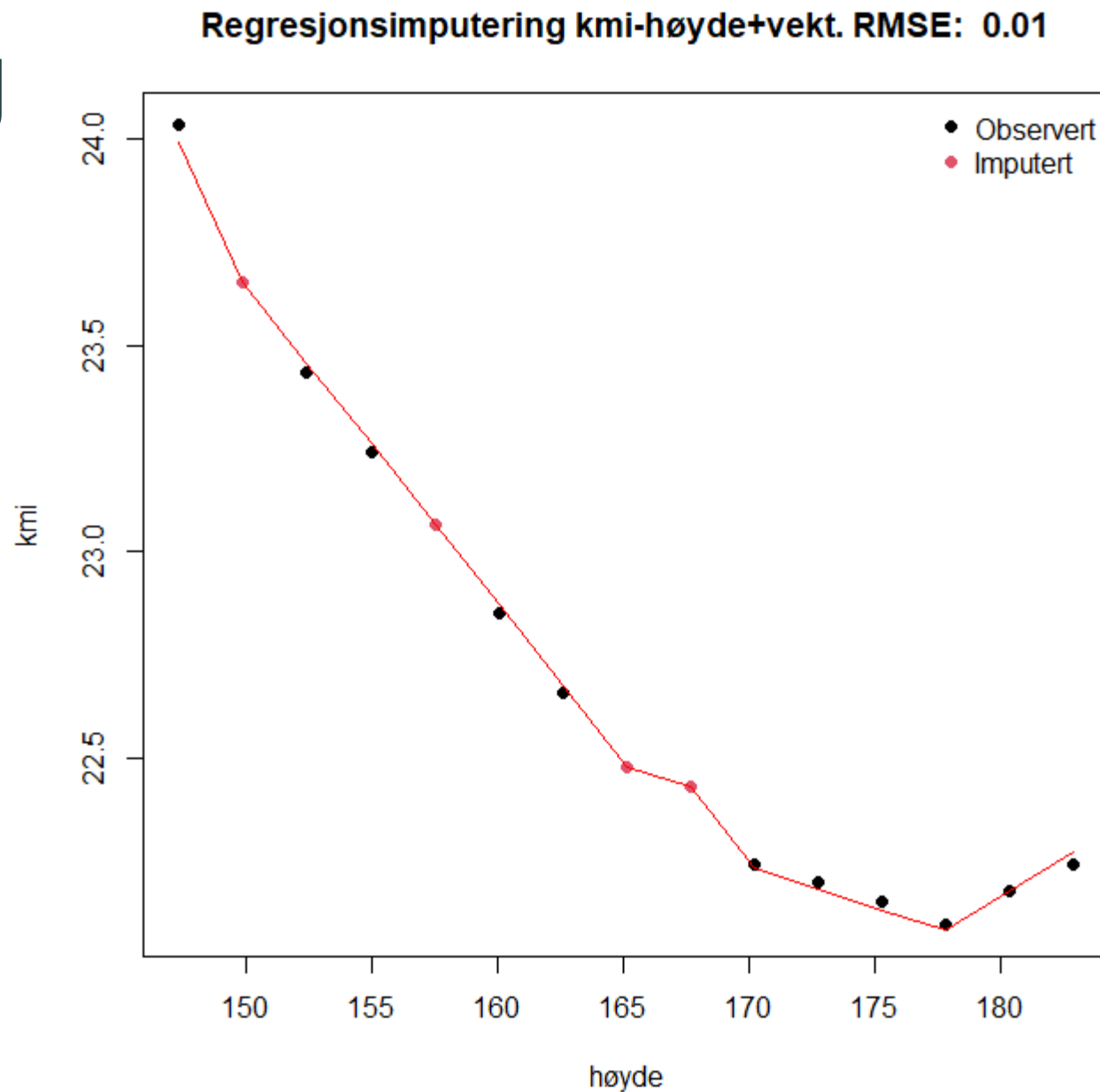
# Regresjonsimputering høyde

- Kode:
- `impute_lm(kmi ~ hoyde )`



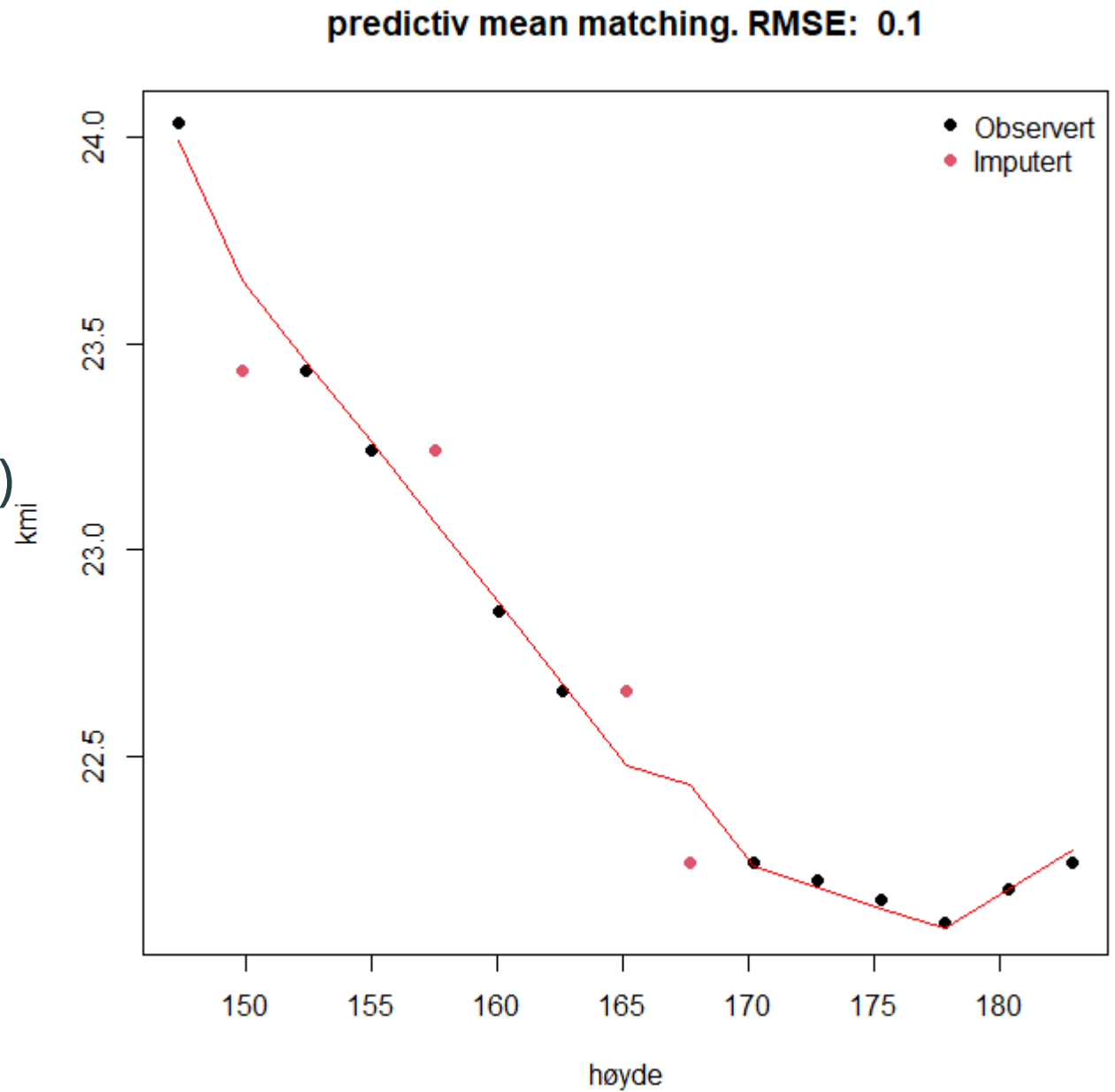
# Regresjonsimputering - høyde og vekt

- Kode:
- `impute_lm(kmi ~ hoyde+vekt )`



# Prediktiv mean matching

- Kode:
- `impute_pmm(kmi ~ vekt + hoyde)`



# Sammenligning av modeller

Modell	RMSE
Gjennomsnitt	0.28
Gjennomsnitt gruppe	0.11
Random hotdeck	0.51
Random hotdeck gruppe	0.14
Nærmeste nabo	0.14
Lineær regresjon - høyde	0.10
Lineær regresjon - høyde+vekt	0.01
Predictiv mean matching - høyde+vekt	0.10

