TALL SOM FORTELLER

Introduction course

SUSIE JENTOFT, ASLAUG FOSS

DAY 2: 28TH FEBRUARY 2025

Statistisk sentralbyrå
Statistics Norway

Agenda: Part 1

	Wendensday **26th February**	**Friday** **28th February**	**Wednesday** **5th Mars**
12:00/13:00	• Introduction • Github • Basic calculations • Objects	• Review • Data manipulation	• Review • Validate - Outlier detection
12:45/13:45	Exercise 1	Exercise 3	Exercise 5
13:30/14:30	• Logical statements • Read in data	• Merging datasets • Graphics	• Imputation
14:00/15:00	Exercise 2	Exercise 4	Exercise 6
14:50/15:50			Summary

Statistisk sentralbyrå
Statistics Norway

• Create objects
• Create vectors
• Fetch packages
• Read in data wi

Review (day 1)

- Write code in RStudio source files. Run using ctrl + enter
- Create objects with: `<-`
- Create vectors with : `c( )`
- Fetch packages with: `library()`
- Read in data with `read_csv()`

Exercise 2 review

Data manipulation with tidyverse

Base R:

```
leave_house(get_dressed(get_out_of_bed(wake_up(me))))
```

- Tidy and easy(er) way to write code
- Pipelines with pipe operator %>%

tidyverse:

```
me %>%
  wake_up() %>%
  get_out_of_bed() %>%
  get_dressed() %>%
  leave_house()
```

Create new variable: mutate()

- Can be used with a pipeline

```
dataset %>%
  mutate(newvariable = 1000)
```

Give variable a name

```
dataset %>%
  mutate(newvariable = oldvariable * 1000)
```

Give variable a name

Existing variable

What to do

Create new variable: mutate()

- Allocate with <- to save variable
- Variables can be replaced/overridden
- Several variables can be created (use , to separate)
- Change variable type (`as.character()`, `as.numeric()`)

```
dataset %>%
  mutate(variable_name = as.character(variable_name))
```

Create new variable: mutate - ifelse()

```
ifelse(test, yes, no)
```

```
dataset %>% mutate(new_variable = ifelse( variable>100 ,  “1” ,  “0” )
```

Change variable name: rename()

```
dataset %>%
  rename(new_name = old_name)
```

Select rows: filter()

- To select rows using a condition use: filter()
- Write a logical statement inside the brackets
- Several logical statements can be used (separate with ,)

```
dataset %>%
  filter(condition)
```

- Again: Nothing is saved without using <-

Select variables: select()

```
dataset %>%
  select(variable_name)
```

- Write variable name in brackets
- Several variables can be specified (separate with ,)
- Combine in pipeline with other functions (eg filter( ))

```
dataset %>%
  filter(condition) %>%
  select(variable_name)
```

Statistisk sentralbyrå
Statistics Norway

Summary/count: summarise()

- Create summary and descriptive statistics
  - Total, average, minimum, maximum, count etc

```
dataset %>%
  summarise(summary_name = mean(variable_name))
```

Give summary a name

What to do:
- mean()
- median()
- sum()
- n()

Variable to action on

Statistisk sentralbyrå
Statistics Norway

Summary and missing values: is.na /na.rm=TRUE

- Counting missing and summarizing variables with missing values

```
summarise(
n_na = sum(is.na(variabel_name)),
med = median(variabel_name, na.rm = TRUE))
```

Counting missing values

Remove missing values in calculation

Discretise numeric data into categorical: cut()

- Making groups from a numeric variabel

```
mutate(group=cut(variable_name,
                 breaks = c(0, 50, 100, 150, 200),
                 labels = c("gr1", "gr2", "gr3", "gr4")))
```

Value for where to cut the variable

Labels for each groups

Summary by group: group_by()

- Choose a variable(s) to group by for further processes

```
dataset %>%
  group_by(grouping_variable) %>%
  summarise(summary_name = mean(variable_name))
```

Exercise 3

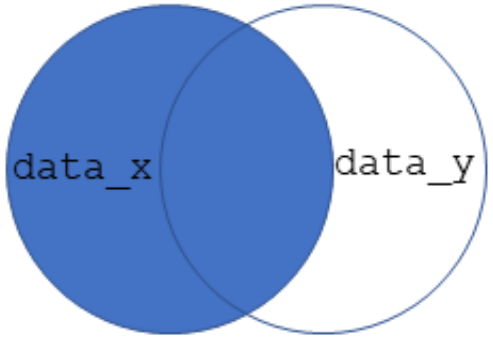
- Exercise 3 is in the file: **Exercises_day2.R**

Add a row : add_row()

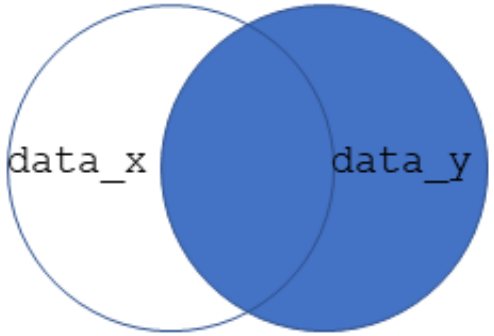
- Add a row to an existing dataset
- Rows must have the same length and type as the dataset

```
dataset %>%
  add_row(variable_name1 = "Kyiv", variable_name2 = 57733)
```

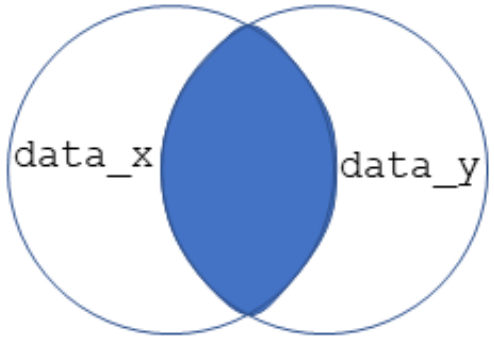
Join two datasets



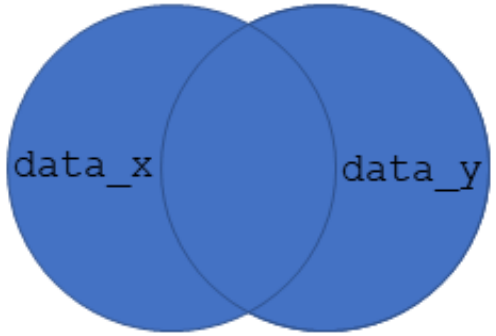
left_join(data_x, data_y)



right_join(data_x, data_y)



inner_join(data_x, data_y)



full_join(data_x, data_y)

Join two datasets

- Use **`by =`** to specify key variable to join on

by = c("Date" = "year")

```
merged_data <- left_join(dataset_1, dataset_2, by = variable_name)
```

- Several variables can be used for joining (as a vector)

Statistisk sentralbyrå
Statistics Norway

Key variable – doublets and missing:

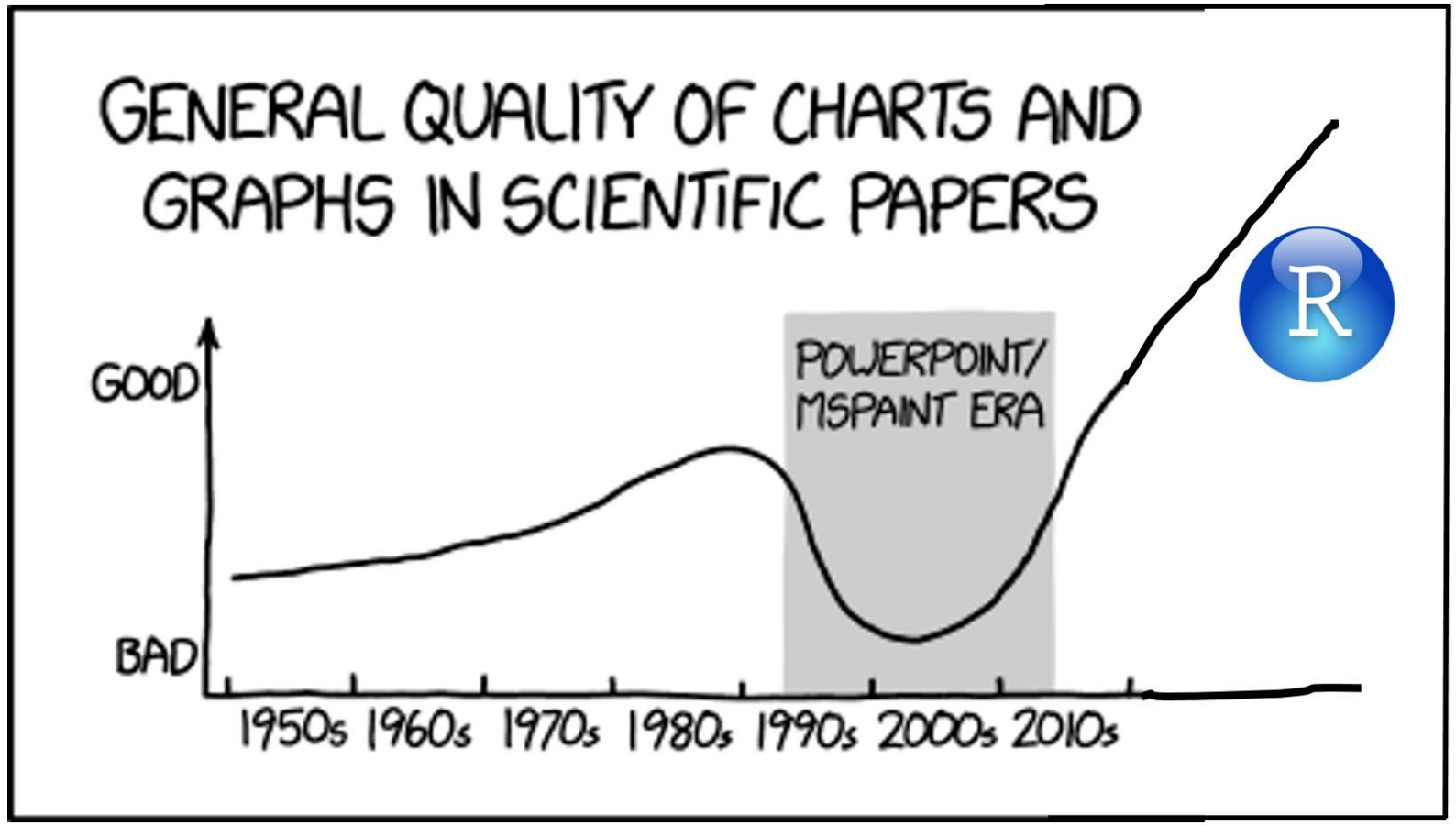
- Units with same ID (by variables) or missing values on ID can create trouble
- Check data before and after joining of datasets

```
summarise(
dist=n_distinct(by_variabel),
n=n(by_variabel),
na = sum(is.na(by_variabel)))
```

Count different values

Statistisk sentralbyrå
Statistics Norway

Graphs


GENERAL QUALITY OF CHARTS AND GRAPHS IN SCIENTIFIC PAPERS
GOOD
BAD
POWERPOINT/ MSPAINT ERA
1950s 1960s 1970s 1980s 1990s 2000s 2010s
R

Plots med ggplot()

- **aes** : aesthetics, which variables
- **geom_** : what type of plot
- **stat** : what type of summary to present

Table 18-1 A Selection of Geoms and Associated Default Stats

Geom	*Description*	*Default Stat*
`geom_bar()`	Bar chart	`stat_bin()`
`geom_point()`	Scatterplot	`stat_identity()`
`geom_line()`	Line diagram, connecting observations in order by *x*-value	`stat_identity()`
`geom_boxplot`	Box-and-whisker plot	`stat_boxplot()`
`geom_path`	Line diagram, connecting observations in original order	`stat_identity()`
`geom_smooth`	Add a smoothed condi-tioned mean	`stat_smooth()`
`geom_histogram`	An alias for `geom_bar()` and `stat_bin()`	`stat_bin()`

Bar plot

```
ggplot(aes(variable_name)) +
  geom_bar()
```

Use + to add plot type

Specify variable

Specify bar plot

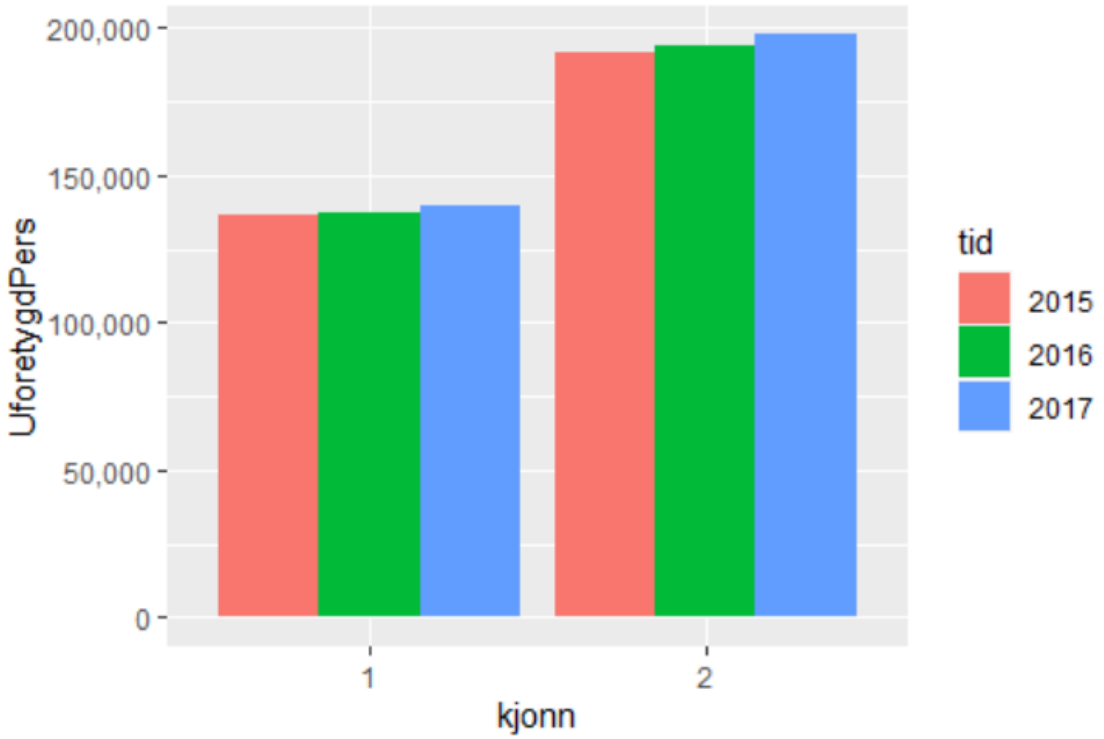
```
ggplot(aes(x = variabelnavn1, y = variabelnavn2)) +
  geom_bar(stat = "identity")
```

Specify x and y variables

Specify to use variable value

Bar plot

- Use fill() in aes to specify a varible to colour by
- Combine with other functions first (eg filter)



Statistisk sentralbyrå
Statistics Norway

Scatter plot

- Compare two numeric variables

```
ggplot(aes(x = variable_name1, y = variable_name2)) +
  geom_point()
```

- Add a regression line with:

```
geom_smooth(method = "lm")
```

- Colour point by group with:

```
geom_point(aes(color = variable_name))
```

Statistisk sentralbyrå
Statistics Norway

Save plots

- Click 

- Or save to working directory (**`getwd()`**)

```
png(file = "plot_name.png")
ggplot(aes(variable_name)) +
                geom_bar()
dev.off()
```

Specify file name

Create plot

Specify we are finished

Exercise 4

- Exercise 4 is in the file: **Exercises_day2.R**