# Data Analytics in R
# Session 6

Maria Kunevich

# Homework feedback

- Your feedback for the course and course assignments is greatly appreciated!
- *https://miro.com/welcomeonboard/SUw3RHpXWjBpUDF2V0dwTWhkOFVVUnlUTnZ4Qm1oVGtSTVN0SmNCUmJyODhydU9hUzA3VUpNZVZHRnNBenhqVHwzMDc0NDU3MzY2NDE0OTE3Njc4?share_link_id=308470562965*

- Let's look at our DataCamp leaderboard: https://app.datacamp.com/groups/data-analytics-in-r-db1ae4f4-62a1-4da2-b5d9-94616b38d5d0/leaderboard

# Homework feedback

- Feedback from me is in the comments to your gists
- Review table: https://docs.google.com/spreadsheets/d/1JyX2fQArbhfkkMf_HTly-FDLDvI-C_-B1DAi8IiSfyc/edit?usp=sharing
- Any questions?

# Plan for today

1. Revision of basic concepts in descriptive statistics.

   https://quizizz.com/join?gc=43268579
2. ‘**Tidyverse**’ packages.
3. The **readr** package. ‘**Tibbles’** as data frames
4. Documentation and reports in R  (‘knitr’ package)
5. Data visualisation (‘ggplot2’ package)

# ‘Tidyverse’ packages

**Tidyverse** - a collection of packages in R with a common design philosophy for data manipulation, exploration, and visualisation.

https://www.tidyverse.org

Tidyverse packages provide a lot of functionality and tend to have code that is easier to read for beginners.

**Resources:** R for Data Science (online book)
https://r4ds.had.co.nz

Download the packages:
install.packages(“tidyverse”)

# ‘Tidyverse’ packages

We’ll explore five packages that are essential for data analysis:

- readr - a quick way to read data to R
- tibble - new data frames
- ggplot2 - data visualisation
- tidyr - data cleaning to make tidy data
- dplyr - data manipulation

```
> library(tidyverse)
── Attaching packages ───────────────────────
✔ ggplot2 3.3.6     ✔ purrr   0.3.4
✔ tibble  3.1.8     ✔ dplyr   1.0.10
✔ tidyr   1.2.1     ✔ stringr 1.4.1
✔ readr   2.1.2     ✔ forcats 0.5.2
```

# The ‘readr’ package

We can import data into R in several  ways (e.g. R base functions), but Tidyverse packages are faster + tibbles are automatically produced that are easier to read and use

The **readr** package allows reading rectangular data (rows and columns), each column refers to a single variable, each row refers to a single observation.

**Resources:** the [data import chapter] in *R for Data Science*.

R cheat sheet for data import (uploaded to GitHub repository)

- `read_csv()` : comma-separated values (CSV) files
- `read_tsv()` : tab-separated values (TSV) files
- `read_delim()` : delimited files (CSV and TSV are important special cases)
- `read_fwf()` : fixed-width files
- `read_table()` : whitespace-separated files
- `read_log()` : web log files

# The ‘tibble’ package

## Overview

A **tibble**, or `tbl_df`, is a modern reimagining of the data.frame, keeping what time has proven to be effective, and throwing out what is not. Tibbles are data.frames that are lazy and surly: they do less (i.e. they don’t change variable names or types, and don’t do partial matching) and complain more (e.g. when a variable does not exist). This forces you to confront problems earlier, typically leading to cleaner, more expressive code. Tibbles also have an enhanced `print()` method which makes them easier to use with large datasets containing complex objects.

**Resources:** the tibbles chapter in *R for data science*

# Data frames -> tibbles

The most important differences:

- **Input type remains unchanged** - data.frame is notorious for treating strings as factors; this will not happen with tibbles
- **Variable names remain unchanged** - in base R, creating data.frames will remove spaces from names, converting them to periods or add “x” before numeric column names. Creating tibbles will not change variable (column) names.
- **There are no row.names() for a tibble** - Tidy data requires that variables be stored in a consistent way, removing the need for row names.
- **Tibbles print first ten rows** and columns that fit on one screen - printing a tibble to screen will never print the entire huge data frame out. By default, it just shows what fits to your screen.

(from *Tidyverse Skills for Data Science* https://jhudatascience.org/tidyversecourse/)

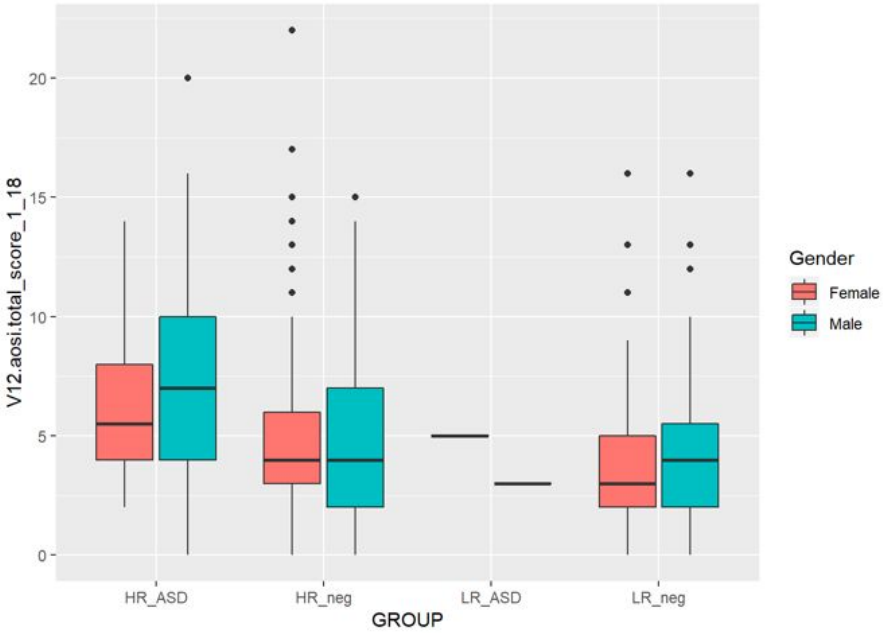
# ‘Tidyverse’ packages

**ggplot2**

helps to create different visualisations by applying visual properties to data variables in R

Basic description of the package:

browseVignettes("ggplot2")



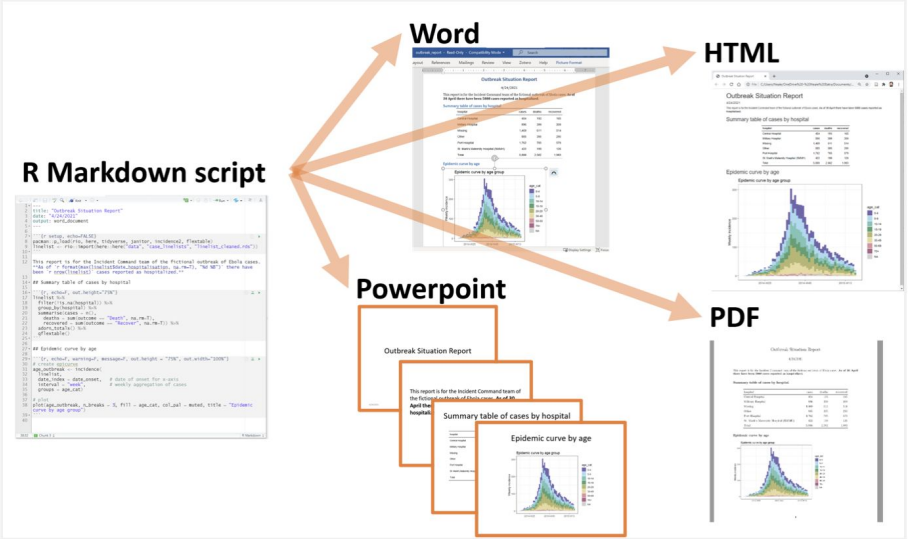
# Data visualisation in R ("ggplot2" package)

Some resources online:

- https://ggplot2.tidyverse.org
- Cheat Sheet: Data Visualization with ggplot2
- R for Data Science: Data Exploration, https://r4ds.had.co.nz/explore-intro.html
- R Graphics Cookbook, 2nd edition, Winston Chang, https://r-graphics.org
- Online webinar: Plotting Anything with ggplot2
  https://www.youtube.com/watch?v=h29g21z0a68
- Data Visualization. A practical introduction by Kieran Healy, https://socviz.co

# Documentation and reports in R Markdown

- **R Markdown** is a tool that combines your code and comments to create an **automated and reproducible** output.
- In **Rmd** file you can include text, figures, tables and dynamically update the output.
- The package **“rmarkdown”** is used to **render** the Rmd file into the desired output.
- The package **“knitr”** reads the code chunks, executes them, and “knit” them back to the document.
- **Pandoc** software (is installed automatically with RStudio) converts the output into word/pdf/powerpoint etc.




# R Markdown components

```
---
title: "Untitled"
author: "Paula Blomquist"
date: "07/04/2021"
output: html_document
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```{r cars}
summary(cars)
```

## Including Plots

You can also embed plots, for example:

```{r pressure, echo=FALSE}
plot(pressure)
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.
```

YAML

Code chunk

markdown text

Code chunk

markdown text

Code chunk

markdown text

# R Markdown documents and “knitr” package

Some resources on how to create different types of documents:

- Official website: https://rmarkdown.rstudio.com
- R markdown Cheat Sheets (uploaded to GitHub)
- *R for Data Science*: R Markdown, https://r4ds.had.co.nz/r-markdown.html
- *Getting started with R Markdown*, https://www.dataquest.io/blog/r-markdown-guide-cheatsheet/
- *Using R Markdown for Class Reports* by Cosma Shalizi https://www.stat.cmu.edu/~cshalizi/rmarkdown/
- *Introduction to Using R Markdown for Class Assignments*, *https://scidesign.github.io/Rmarkdownforclassreports.html*