Intro to R

Why use R?

- Open source and free
  - R is available for free, making it accessible to everyone
  - A large, active community contributes to continuous improvement and knowledge sharing
  - Strong in statistical analysis
- R has a wide array of packages dedicated to statistical analysis
  - R is great at producing complex plots
  - R has specialised tools for various fields, such as econometrics and randomisation for research and evaluation
- Reproducibility
  - Script based workflow: R uses scripts for analysis, which can be shared and re-executed to reproduce results
  - R works well with Git for version control, helping track changes and collaborate efficiently
  - Tools like Quarto allow for the integration of code, results, and text into a single document, ensuring analyses are transparent and reproducible

Workflow

- RStudio: An IDE (integrated development environment) for R
  - RStudio provides a user friendly interface
- Working directories
  - The folder where R looks for files and saves output
  - Avoid using relative paths - use R projects
- Creating an R project
  - R Projects help organise all related files, scripts, and data in a single directory
  -
- Scripts
  - Scripts are written in `.R` files, where you can save commands and code for later use
  - You can run line-by-line or in chunks, making it easy to test and debug your scripts
  - Use comments within scripts to explain the purpose of code sections, improving readability and collaboration.
- Packages
  - Packages can be installed using `install.packages()`
  - Once installed packages are loaded into the R session using `library()`
  - Make sure to install and load tidyverse
    * `install.packages("tidyverse")`
    * `library(tidyverse)`

Try it!

Download and load packages. You only need to install it once. Tidyverse is a super useful collection of packages designed to make data analysis easier.

```r
#install.packages("tidyverse")
#install.packages("devtools")
#devtools::install_github("jakubkuzilek/oulad")


library(tidyverse)
library(oulad)
```

Load the data.

```r
data("student", package = "oulad")

# Or use Excel
student <- read.csv("student.csv")
```

Create a count.

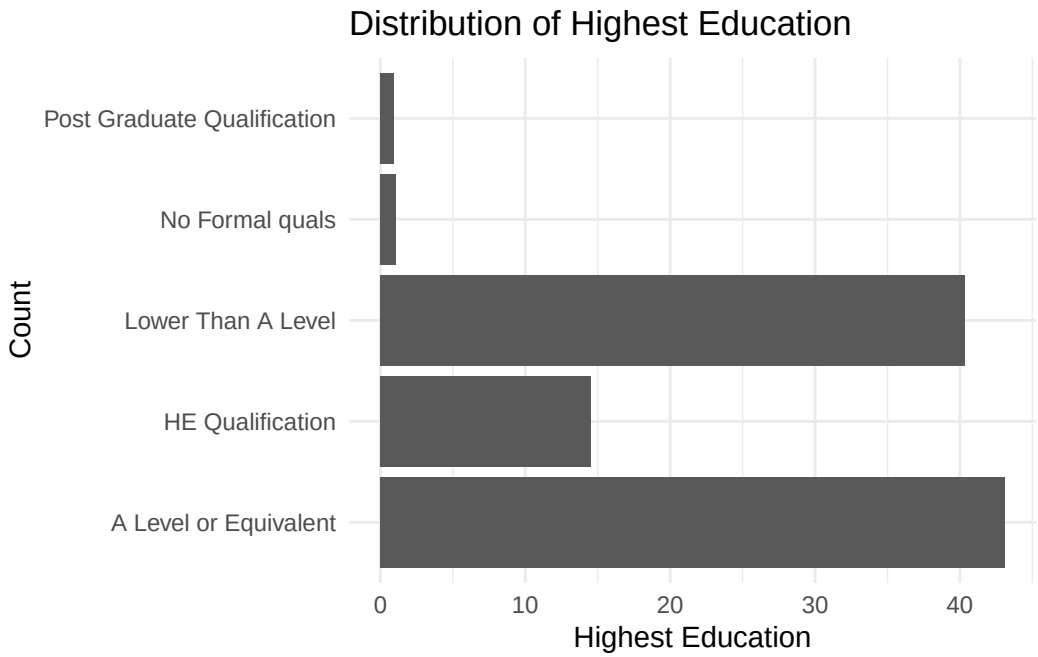
```r
student_summary <- student %>%
  count(highest_education) %>%
  mutate(percentage = round(n / sum(n) * 100, 2)) %>%
  arrange(desc(n))

student_summary
```

```
            highest_education     n percentage
1       A Level or Equivalent 14045      43.09
2          Lower Than A Level 13158      40.37
3            HE Qualification  4730      14.51
4             No Formal quals   347       1.06
5 Post Graduate Qualification   313       0.96
```

Make a basic chart.

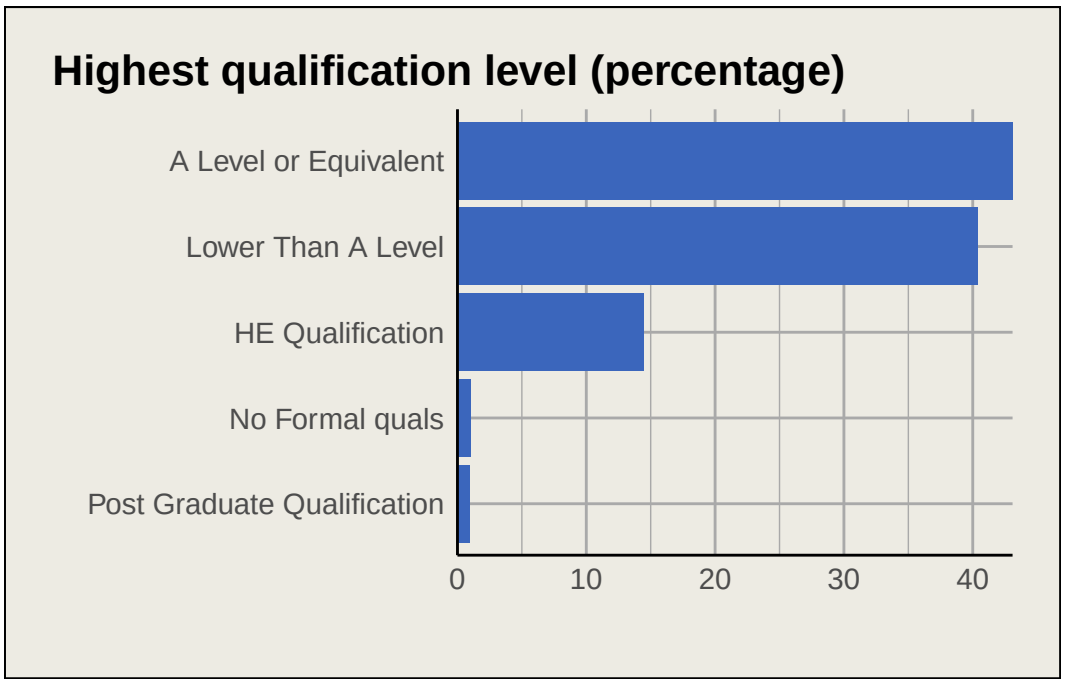
```r
ggplot(student_summary, aes(x = percentage, y = highest_education)) +
  geom_bar(stat = "identity") +
  labs(
    title = "Distribution of Highest Education",
    x = "Highest Education",
    y = "Count"
  ) +
  theme_minimal()
```



Customise it.

```
ggplot(student_summary, aes(x = percentage, y = fct_reorder(highest_education, percentage)))
  geom_bar(stat = "identity", fill = "#3b66bc") +
  labs(
    title = "Highest qualification level (percentage)",
    x = "",
    y = ""
  ) +
  theme_minimal() +
  theme(
    plot.title.position = "plot",
    plot.title = element_text(size = 16, face = "bold"),
    plot.subtitle = element_text(size = 12),
    plot.caption.position = "plot",
    plot.caption = element_text(hjust = 0, size = 8, face = "italic"),
    panel.grid.major = element_line(colour = "darkgrey"),
    panel.grid.minor = element_line(colour = "darkgrey"),
    plot.background = element_rect(fill = "#EDEBE3"),
    plot.margin = margin(0.25, 0.25, 0.25, 0.25, "in"),
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.line = element_line(colour = "black", linewidth = 0.5),
```

```
    legend.position = "none"
  ) +
  scale_x_continuous(expand = c(0, 0))
```



Reporting with Quarto

- Avoid copy and pasting results and charts into reports - inefficient and error prone
- With Quarto you can run code and write your report, all in one place - no need to copy and paste anything
- Everything will be fully reproducible - a source of truth
- Collaborate with colleagues and use version control
- Use interactive features like interactive charts and tables
- Case study: TASO Technical Guide

Resources for further learning

- R for Data Science
- TASO's coding good practice

- TASO data visualisation style guide
- Productive R Workflow (paid for course)