

How to wow with R

A brief introduction generating satisfying outputs

by Matthew Suderman (slide credits: Luke McGuinness)

Topics we will cover

- Create **beautiful plots** with `ggplot2`
- **Literate programming** with `Rmarkdown`
- Build **interactive apps** with `shiny`
- **Share your work** as a package

Pretty plots: Intro to ggplot2

ggplot2

A package to create **highly customisable publication-ready plots**

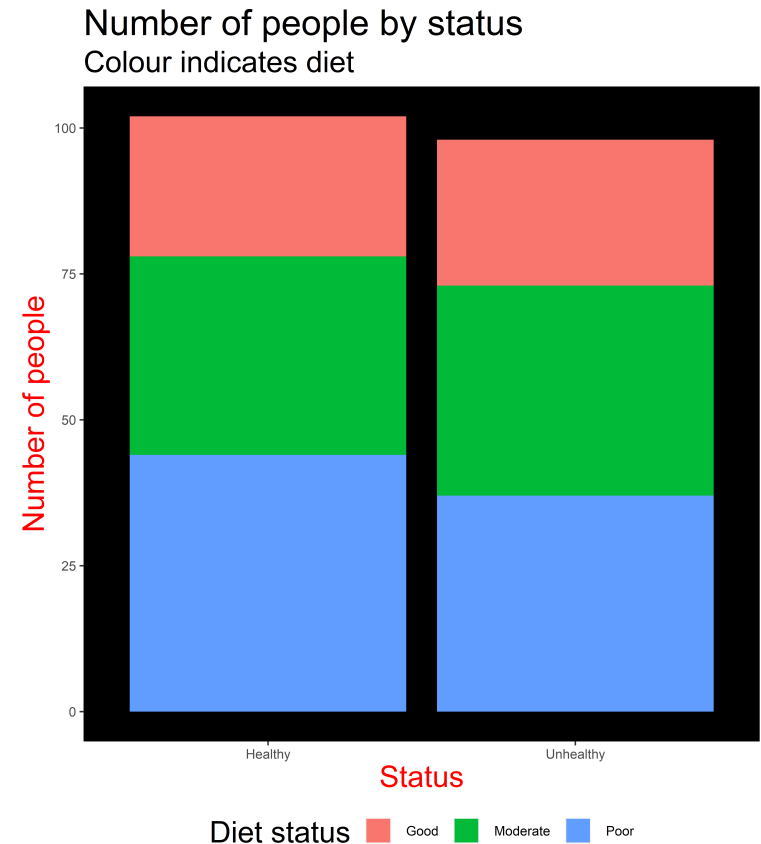
Key elements:

- **aesthetic**: what you want to graph (e.g. x, y, colours, etc)
- **geom**: how you want to graph it (e.g. scatterplot, histogram)
- **options**: optional titles, themes, etc.

```

ggplot(data = df) +
  aes(x = status) +
  geom_histogram(stat = "count") +
  aes(fill = diet) +
  labs(fill = "Diet status") +
  labs(
    title = "Number of people by status",
    subtitle = "Colour indicates diet",
    x = "Status",
    y = "Number of people") +
  theme(
    axis.title=element_text(colour="red"))+
  theme(legend.position = "bottom") +
  theme(panel.grid = element_blank()) +
  theme(
    panel.background=element_rect(fill="black")) +
  theme(title = element_text(size = 20))

```



Virtually endless customisability

Plots on previous slides are **only a taster**

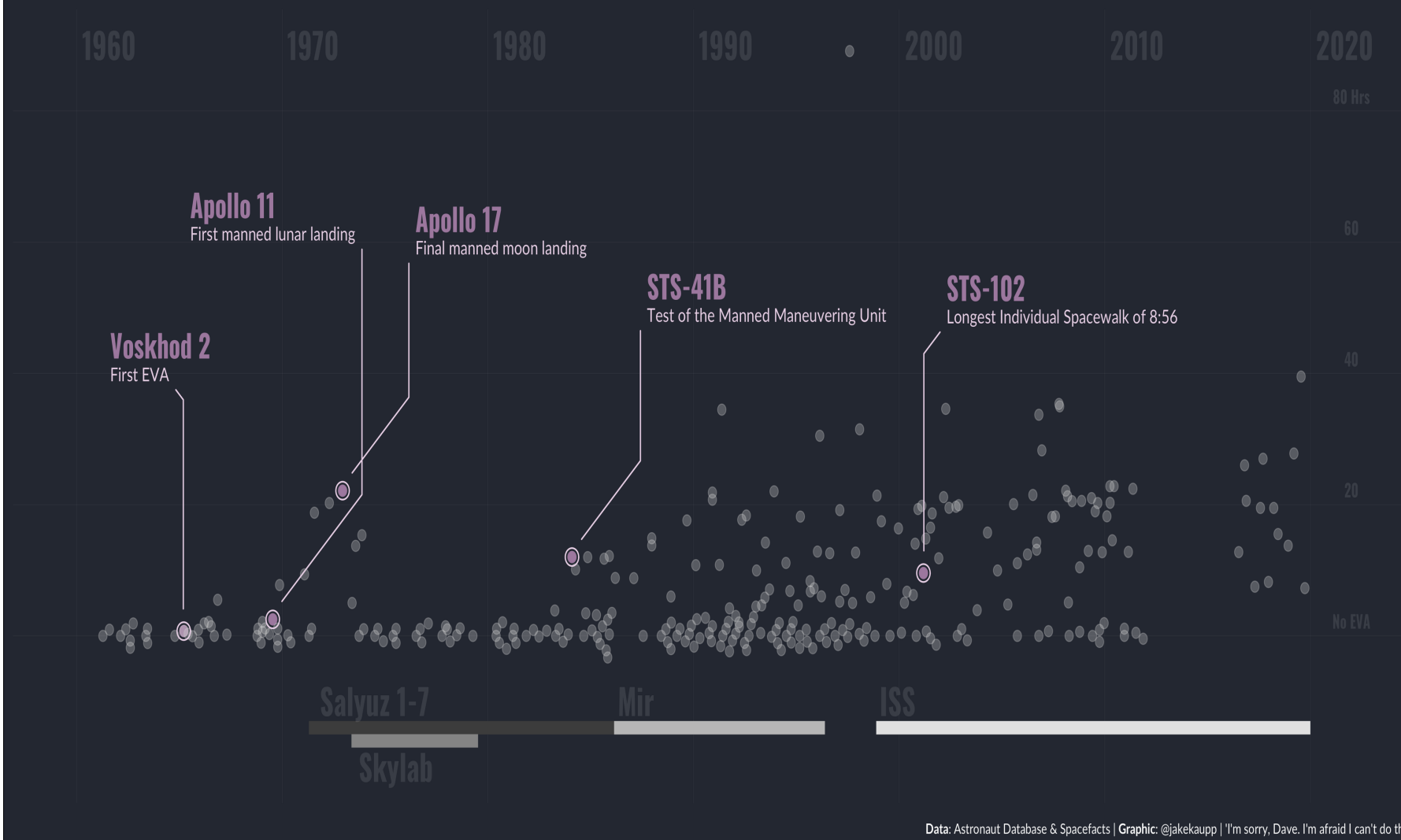
Can make **any plot you can think up**

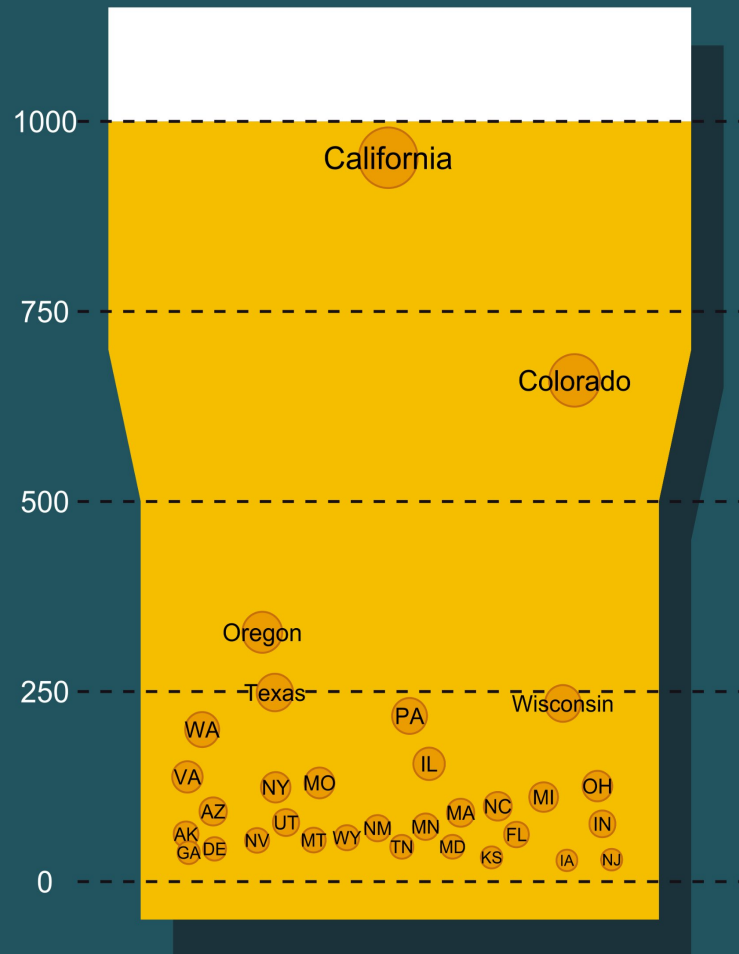
Key resources, all open-source:

- [ggplot2 documentation and cheatsheet](#) - great starting point.
- ["Data visualisation"](#) by Kieran Healy - great introductory text on the science of data visualization, with examples in R.
- [Tidy Tuesday](#) - weekly community-run visualization exercise

Since We Started Making Space Stations, Everybody Just Wanted To Go Outside

Illustrated below are the total EVA hours per American and Russian Space Missions since 1960. Overlaid at the bottom are the lifespans of the 10 space stations constructed by Russia or America. The **annotations** highlight notable missions of interest.





Great American Beer Festival

The Professional Judge Panel awards gold, silver or bronze medals that are recognized around the world as symbols of brewing excellence. These awards are among the most coveted in the industry and heralded by the winning brewers in their national advertising.

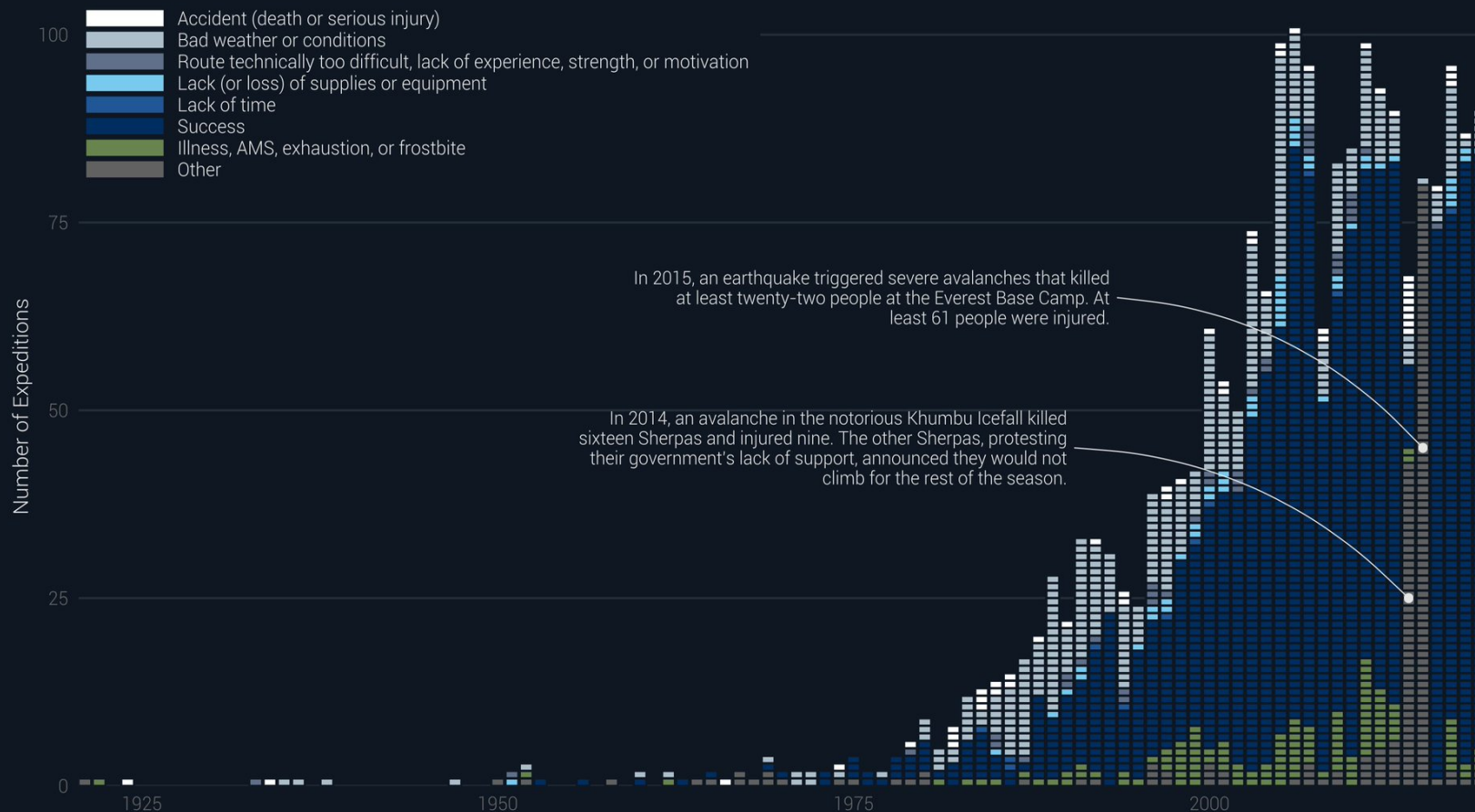
In this beer, each US State with greater than 25 medals since 1987 is represented by a bubble. The higher and larger the bubble in the pint the greater the number of medals recieved. California is way in front, with 962 medals!

Data from greatamericanbeerfestival.com/the-competition/winners
Visualisation by Jack Davison (Twitter @JDavison_)
Code found at github.com/jack-davison

EVEREST EXPEDITIONS

Journeys to the top of the world end for many reasons. This graphic looks at the results of all Everest expeditions since 1921.

Everest has had a colorful history, but even still most expeditions reach the peak.



Recap

Key concept is that you are **adding elements together**:

- first the data
- then the elements of the data you want to display (`aesthetics`)
- then the way you want to display them (`geoms`)
- then extra options (`colour/themes/titles`)

Literate programming: RMarkdown

RMarkdown

A single file (with `.Rmd` extension) that incorporates

- text
- code
- output (results and figures)

Can be converted to multiple file formats:

- HTML (web-page)
- Word
- PDF

Components of an RMarkdown file

(Optional) YAML header

- Defines metadata, such as the title, author and date
- Defines the output format (Word, PDF, HTML, etc.)

Code chunks: R code chunks

- This is where the core computation happens
- Can choose to show the underlying code or just the results

Text with inline code

- Great for presenting results as part of a sentence

(Optional) References

- Can cite published materials as in an academic report or publication

RMarkdown example

<https://github.com/MRCIEU/introduction-to-r/tree/main/how-to-wow/rmd-example>

Why bother?

Improves **reproducibility** and **transparency**, as the origin of every single result in the paper can be examined

Makes a lot of tasks more **straightforward**:

- Updating a manuscript with new **data/analyses**
- **Citing** other work
- Controlling **figure placement**

Things you can produce using RMarkdown

- Academic papers ([example](#))
- Theses ([bristolthesis](#))
- Supervisory reports (great if format is similar each time)
- Slides (this slide deck was made using RMarkdown!)

Resources

[RMarkdown tutorial](#) from RStudio

[RMarkdown: The Definitive Guide](#) by Yihui Xie

Making R accessible:

Intro to shiny

What is shiny?

`shiny` is an R package that allows users to **build interactive web applications** ("apps") straight from R.

Shiny apps are web-pages that users can interact with to:

- Explore data
- Perform analyses
- Create plots



Structure of a shiny app

A shiny app is a special type of R file that has three components:

- User interface
- Server
- Call to the shinyApp

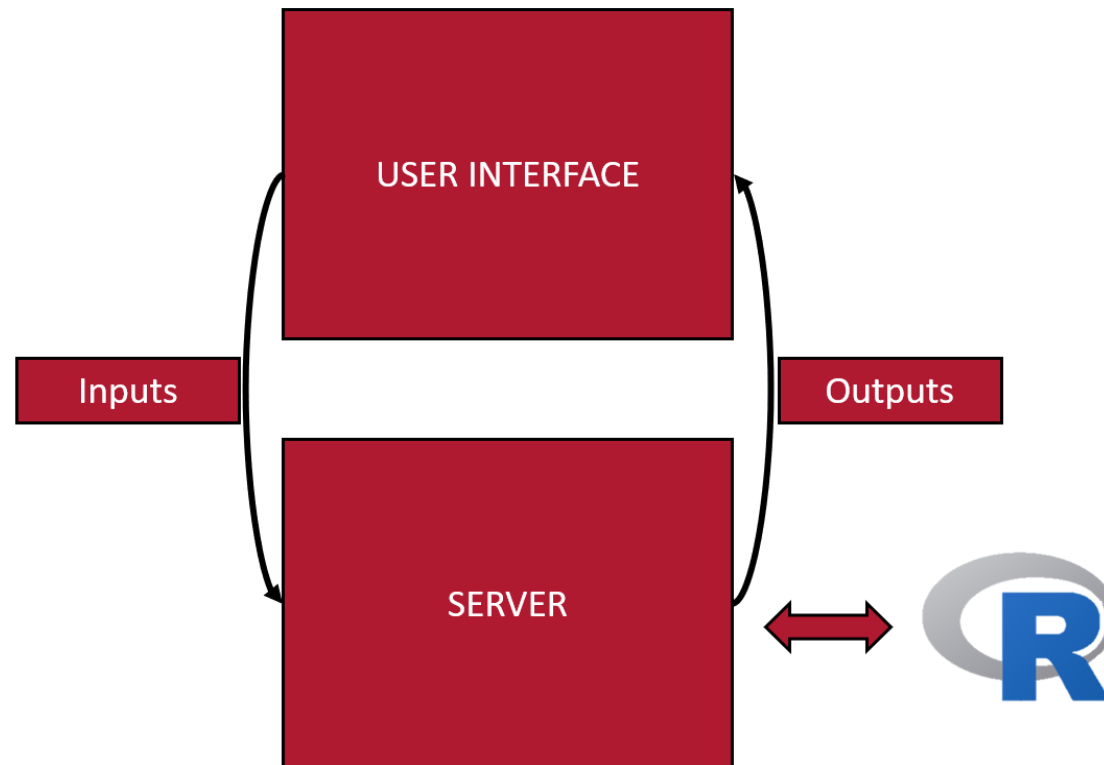
Example

<http://epidelta.mrcieu.ac.uk/>

Source code: <https://github.com/perishky/epidelta/blob/master/app/app.R>

How it works

Essentially using a **webpage** rather than **console** to instruct R.



Why bother?

- R has a **high barrier to entry** for new users
- Apps allow those without any experience of R to **explore** and **benefit** from your work
- The more people who can use it, the **better known/cited** it will be!

Resources

[shiny gallery](#) - collection of example apps, with the underlying R code

[shiny cheatsheet](#) - 2-page summary of functions

[shiny contest](#) - annual competition with prizes for the most impressive/most imaginative app

[Bristol short course on data visualisation and web applications](#) has a half-day dedicated to building shiny apps

Packaging your code

Why package?

"I wish I'd left this code across scattered .R files instead of combining it into a package" said no one ever #rstats

-- David Robinson (@drob) June 19, 2015

- **Reliable** and **universal** way to share code/data
- Makes it possible for others to **cite your work**
- Well-developed **testing framework**

Example

<https://github.com/MRCIEU/aries>

```
├── DESCRIPTION
├── docs
│   ├── aries.pdf
│   └── tutorial
│       ├── tutorial.html
│       └── tutorial.rmd
├── man
│   ├── aries.copy.release.Rd
│   ├── aries.info.Rd
│   └── ...
├── NAMESPACE
├── R
│   ├── detectionp.r
│   ├── info.r
│   └── ...
├── README.md
├── tests
│   ├── test-info.r
│   └── ...
```

Where to get/share packages?

CRAN

- Official repository
- Strict submission process to guarantee quality
- `install.packages("packagename")`

Bioconductor

- Topic specific repository, with a focus on bioinformatics
- Strict submission requirements, similar to CRAN
- `BiocManager::install("packagename")`

GitHub

- Popular for open source projects
- Wild west of packages
- `devtools::install_github("username/packagename")`

Resources

R Packages book by Hadley Wickham (<http://r-pkgs.had.co.nz/>)

Writing an R package from scratch by Hilary Parker (hilaryparker.com/2014/04/29/writing-an-r-package-from-scratch/)

Wrapping up

- Create **beautiful plots** with `ggplot2`
- Perform **literate programming** with `Rmarkdown`
- Build **interactive apps** with `shiny`
- **Share your work** as a package

Slide credits

Luke McGuinness

- Email: luke.mcguinness@bristol.ac.uk
- Twitter: [@mcguinlu](https://twitter.com/mcguinlu)
- GitHub: [@mcguinlu](https://github.com/mcguinlu)

Original slide deck:

<https://mcguinlu.github.io/slides/intro-to-r-phd/index.html>