Using Visualisation Toolkits

David Trye | COMPX532 | DSIGN532

1

Objectives

1. To introduce a selection of powerful **visualisation tools**, and to highlight their strengths and limitations
2. To provide some practical **tips and tricks** for using these tools

2

Disclaimer: A Subjective Selection

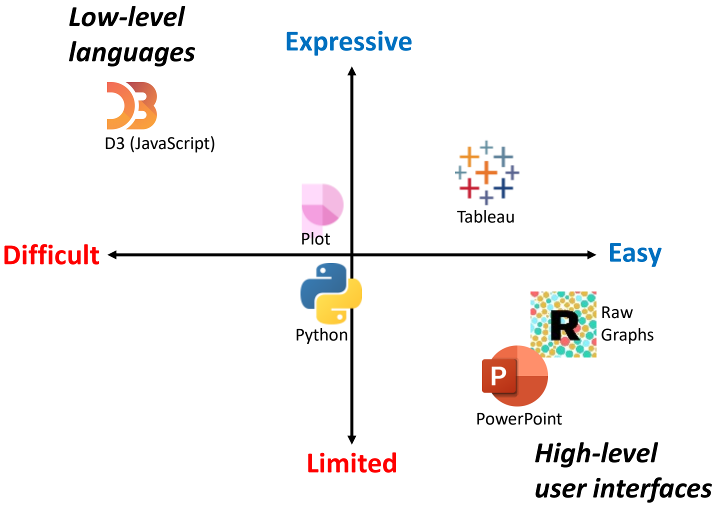
- There are many other useful visualisation tools available!
- 15 Free Data Visualization Tools for 2023

3

Expressiveness vs Usability

Mike Bostock, 2017 Keynote:

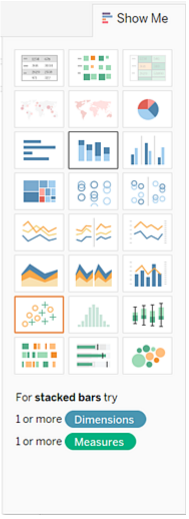
- [Code-free] visualisation tools sacrifice **generality** for greater **efficiency** performing particular tasks
- Code has **unparalleled expressiveness**; it is the most general tool we have
- A medium for discovery *must* be general. Creativity requires composition



4

Tableau

- As a student, you are eligible for a free, one-year *Tableau Desktop* license
  - https://www.tableau.com/academic/students
- Can create **interactive dashboards** in minutes
  - Connect to one or more data sources
  - Manipulate dimensions by dragging and dropping them
  - Use spreadsheet-like formulas to derive new variables
  - Support for operations like filtering and aggregation
- Thousands of examples online:
  - https://public.tableau.com/app/discover
- Easy to share/publish your visualisations online



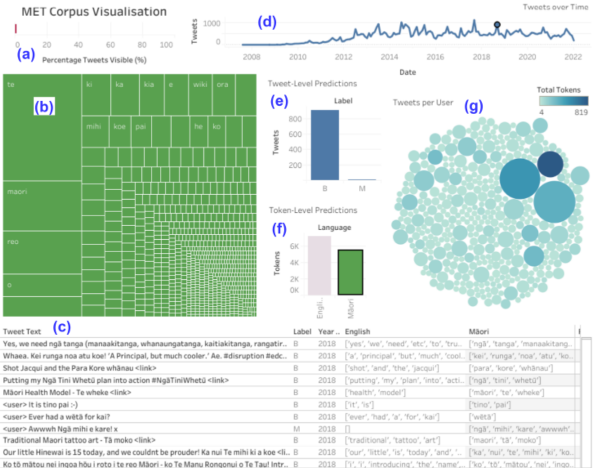
For **stacked bars** try

1 or more Dimensions

1 or more Measures

Tableau Example: MET Corpus Demo

- The MET Corpus is a collection of 75,000 mixed Māori-English tweets
- These were labelled at both the word and token level
- Provides insights into borrowing and code-switching in NZ



Tweet Text	Label	Year ..	English	Māori
Yes, we need ngā tanga (manaakitanga, whanaungatanga, kaitiakitanga, rangatir..	B	2018	['yes', 'we', 'need', 'etc', 'to', 'tru..	['ngā', 'tanga', 'manaakitang..
Whaea. Kei runga noa atu koe! 'A Principal, but much cooler.' Ae. #disruption #edc..	B	2018	['a', 'principal', 'but', 'much', 'cool..	['kei', 'runga', 'noa', 'atu', 'ko..
Shot Jacqui and the Para Kore whānau <link>	B	2018	['shot', 'and', 'the', 'jacqui']	['para', 'kore', 'whānau']
Putting my Ngā Tini Whetū plan into action #NgāTiniWhetū <link>	B	2018	['putting', 'my', 'plan', 'into', 'acti..	['ngā', 'tini', 'whetū']
Māori Health Model - Te wheke <link>	B	2018	['health', 'model']	['māori', 'te', 'wheke']
<user> It is tino pai :-)	B	2018	['it', 'is']	['tino', 'pai']
<user> Ever had a wētā for kai?	B	2018	['ever', 'had', 'a', 'for', 'kai']	['wētā']
<user> Awwwh Ngā mihi e kare! x	M	2018	[]	['ngā', 'mihi', 'kare', 'awwwh'..
Traditional Maori tattoo art - Tā moko <link>	B	2018	['traditional', 'tattoo', 'art']	['maori', 'tā', 'moko']
Our little Hinewai is 15 today, and we couldnt be prouder! Ka nui Te mihi ki a koe <li..	B	2018	['our', 'little', 'is', 'today', 'and', '..	['ka', 'nui', 'te', 'mihi', 'ki', 'ko..
Ko tō mātou nei ingoa hōu i roto i te reo Māori - ko Te Manu Rongonui o Te Tau! Intr..	B	2018	['i', 'i', 'introducing', 'the', 'name',..	['ko', 'tō', 'mātou', 'nei', 'ingo..

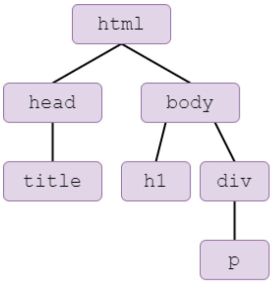
Figure 3: Interactive tool for exploring the *MET Corpus*: (a) percentage of corpus visible, (b) selected tokens by frequency, (c) tweet table, (d) tweets by year, (e) tweet predictions, (f) token predictions, (g) tweets by user.

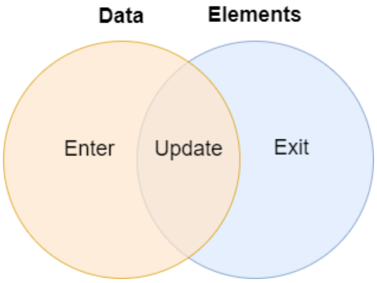
D3.js

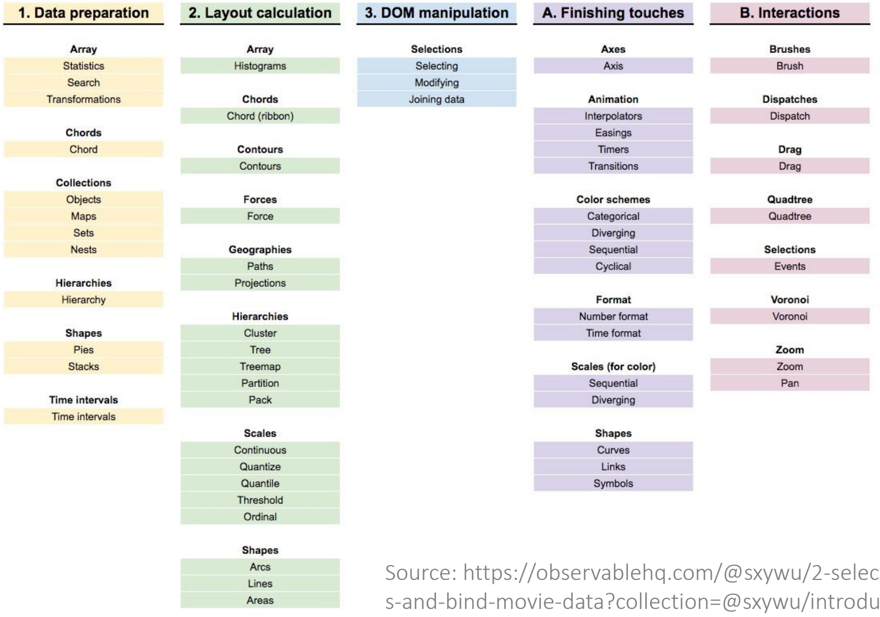
What is D3 ?

- D3 (or D3.js) is a JavaScript library that stands for **data-driven documents**
- Allows you to dynamically manipulate the **Document Object Model (DOM)**
  - Key concepts include selections, data-binding & the enter-update-exit pattern
  - Also supports data preparation, layout calculation, scales, shapes, animation, interaction and more
- Interactive, flexible and expressive!
- See http://shirleywu.studio/d3intro/

Document Object Model







1. Data preparation

- **Array**: Statistics, Search, Transformations
- **Chords**: Chord
- **Collections**: Objects, Maps, Sets, Nests
- **Hierarchies**: Hierarchy
- **Shapes**: Pies, Stacks
- **Time intervals**: Time intervals

2. Layout calculation

- **Array**: Histograms
- **Chords**: Chord (ribbon)
- **Contours**: Contours
- **Forces**: Force
- **Geographies**: Paths, Projections
- **Hierarchies**: Cluster, Tree, Treemap, Partition, Pack
- **Scales**: Continuous, Quantize, Quantile, Threshold, Ordinal
- **Shapes**: Arcs, Lines, Areas

3. DOM manipulation

- **Selections**: Selecting, Modifying, Joining data

A. Finishing touches

- **Axes**: Axis
- **Animation**: Interpolators, Easings, Timers, Transitions
- **Color schemes**: Categorical, Diverging, Sequential, Cyclical
- **Format**: Number format, Time format
- **Scales (for color)**: Sequential, Diverging
- **Shapes**: Curves, Links, Symbols

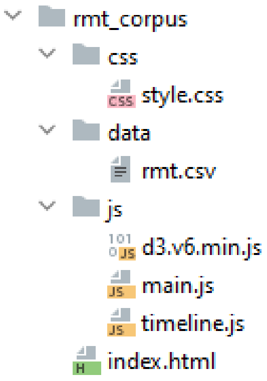
B. Interactions

- **Brushes**: Brush
- **Dispatches**: Dispatch
- **Drag**: Drag
- **Quadtree**: Quadtree
- **Selections**: Events
- **Voronoi**: Voronoi
- **Zoom**: Zoom, Pan

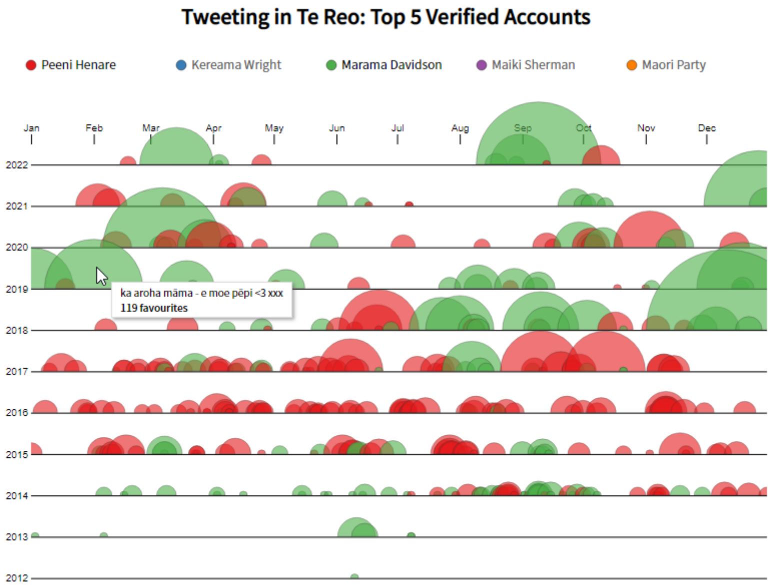
Source: https://observablehq.com/@sxywu/2-select-existing-petal-s-and-bind-movie-data?collection=@sxywu/introduction-to-d3-js

D3 Example: RMT Corpus Visualisation

- The RMT Corpus contains predominantly Māori-language tweets
- 2D calendar visualisation of top 5 verified accounts
  - Position = Date tweeted
  - Radius = Number of favourites (likes + quotes + retweets)
  - Colour = User (filterable legend)
- Code Editor: WebStorm (free student account)
  - Visual Studio Code is a popular alternative
- For debugging, used Browser Developer Tools and ChatGPT



```
rmt_corpus
  css
    style.css
  data
    rmt.csv
  js
    d3.v6.min.js
    main.js
    timeline.js
  index.html
```



Inspired by: https://www.nytimes.com/interactive/2017/09/01/upshot/cost-of-hurricane-harvey-only-one-storm-comes-close.html

11

Observable Plot

12

Observable Notebooks

https://observablehq.com/explore

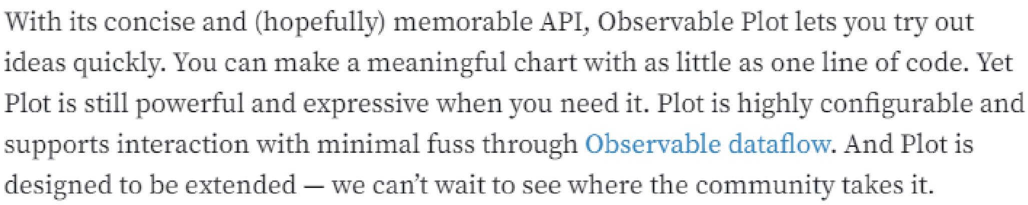
- "Observable helps you use data to think"
  - Explore and visualise data
  - Get ideas/inspiration from existing notebooks
  - Share your insights with the vis community
- Reactive (live) programming
  - Interactive JavaScript notebooks (similar to Jupyter Notebook for Python)
  - Experiment with (big) data in real-time
  - Quickly build and iterate prototypes, re-use code, etc.
  - Supports creation of highly interactive visualisations

Melody Meckfessel,
CEO & Co-founder

Mike Bostock,
CTO & Co-founder
D3 Creator

What is Observable Plot?

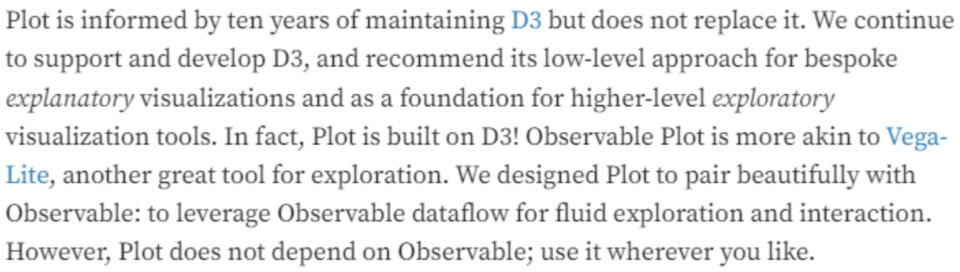
- A **higher-level** JavaScript library for **exploratory** data visualisation
- Built on top of D3 (but more concise and convenient)
- Centred around the idea of "using vision to think"



With its concise and (hopefully) memorable API, Observable Plot lets you try out ideas quickly. You can make a meaningful chart with as little as one line of code. Yet Plot is still powerful and expressive when you need it. Plot is highly configurable and supports interaction with minimal fuss through Observable dataflow. And Plot is designed to be extended — we can't wait to see where the community takes it.

Plot vs D3

- Plot not intended as a “replacement” for D3



Plot is informed by ten years of maintaining D3 but does not replace it. We continue to support and develop D3, and recommend its low-level approach for bespoke *explanatory* visualizations and as a foundation for higher-level *exploratory* visualization tools. In fact, Plot is built on D3! Observable Plot is more akin to Vega-Lite, another great tool for exploration. We designed Plot to pair beautifully with Observable: to leverage Observable dataflow for fluid exploration and interaction. However, Plot does not depend on Observable; use it wherever you like.

- See https://observablehq.com/@observablehq/introducing-observable-plot

15

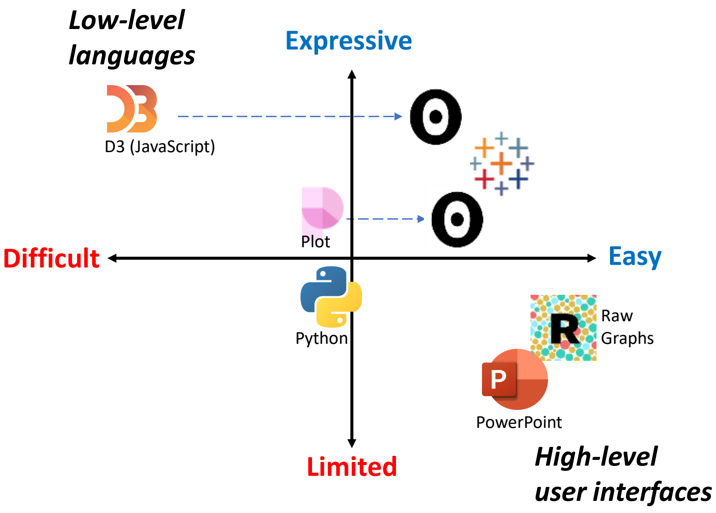
Demonstration: Prototyping with Plot

16

Expressiveness vs Usability Revisited

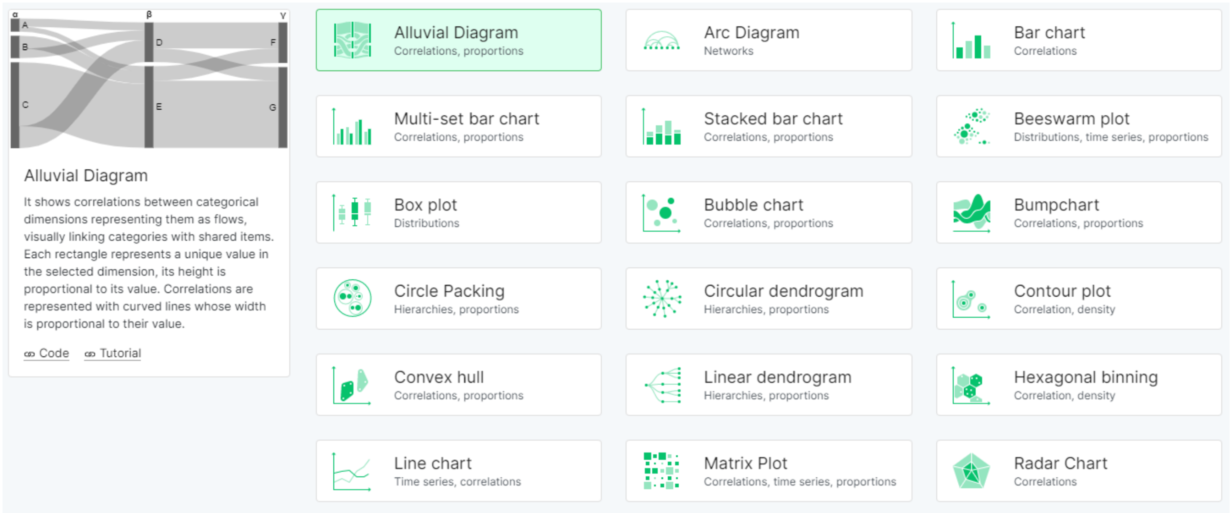
Mike Bostock, 2017 Keynote:

- [Code-free] visualisation tools sacrifice **generality** for greater **efficiency** performing particular tasks
- Code has **unparalleled expressiveness**; it is the most general tool we have
- A medium for discovery *must* be general. Creativity requires composition



17

RAW Graphs



Alluvial Diagram

It shows correlations between categorical dimensions representing them as flows, visually linking categories with shared items. Each rectangle represents a unique value in the selected dimension, its height is proportional to its value. Correlations are represented with curved lines whose width is proportional to their value.

Code Tutorial

Alluvial Diagram
Correlations, proportions

Arc Diagram
Networks

Bar chart
Correlations

Multi-set bar chart
Correlations, proportions

Stacked bar chart
Correlations, proportions

Beeswarm plot
Distributions, time series, proportions

Box plot
Distributions

Bubble chart
Correlations, proportions

Bumpchart
Correlations, proportions

Circle Packing
Hierarchies, proportions

Circular dendrogram
Hierarchies, proportions

Contour plot
Correlation, density

Convex hull
Correlations, proportions

Linear dendrogram
Hierarchies, proportions

Hexagonal binning
Correlation, density

Line chart
Time series, correlations

Matrix Plot
Correlations, time series, proportions

Radar Chart
Correlations

18

RAWGraphs

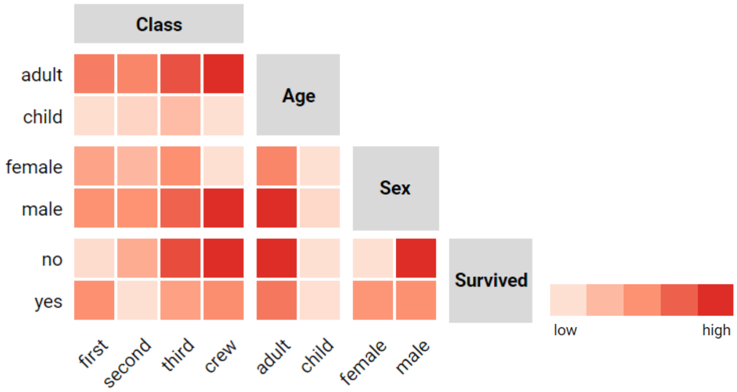
- "The missing link between spreadsheets and data"
  - Instantly maps data variables to visual channels
  - Quick and convenient!
  - Uses D3 under the hood
- No programming knowledge required
  - Friendly user interface
  - However, limited layouts available
    - Doesn't support detailed (low-level) customisation
    - BUT can build custom models with basic understanding of D3

Demonstration: Prototyping with RAW Graphs

Prototyping with Python (Pandas), Sheets & PowerPoint

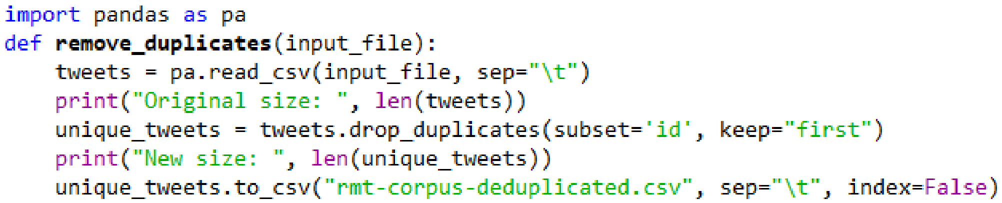
A ‘Heatmap Matrix’ for Categorical Data

Building on Rocha & da Silva, 2018



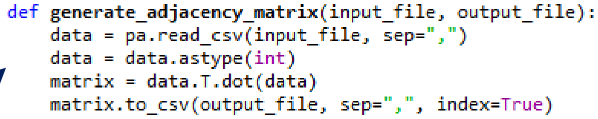
Data Wrangling with Python

- I tend to use Python for cleaning & pre-processing data
  - Popular alternatives include R, JavaScript & Excel/Sheets
- Pandas *DataFrames* are especially useful for handling datasets with many variables (multi-dimensional data)
  - Rows = data items
  - Columns = variables



```
import pandas as pa
def remove_duplicates(input_file):
    tweets = pa.read_csv(input_file, sep="\t")
    print("Original size: ", len(tweets))
    unique_tweets = tweets.drop_duplicates(subset='id', keep="first")
    print("New size: ", len(unique_tweets))
    unique_tweets.to_csv("rmt-corpus-deduplicated.csv", sep="\t", index=False)
```

Process



```
def generate_adjacency_matrix(input_file, output_file):
    data = pa.read_csv(input_file, sep=",")
    data = data.astype(int)
    matrix = data.T.dot(data)
    matrix.to_csv(output_file, sep=",", index=True)
```

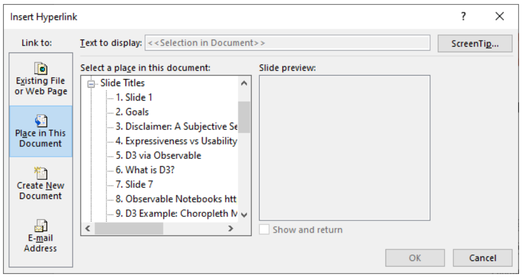
1. Download Titanic dataset
2. Compute pairwise adjacency matrix using **Pandas**, keeping categories grouped by variable
3. Open output file as spreadsheet
4. Remove redundant values
5. Create variable ‘boxes’
6. Colour cells using conditional formatting
7. Create legend
8. Remove numeric values
9. Fine-tune display (borders, etc.)
10. Add tooltips in **PowerPoint**

Sheets

Demonstration: Heatmap Spreadsheet

Adding Interaction with PowerPoint

- Can create Tooltips and Buttons
  - Buttons are just clickable regions that point to different slides
  - Simply create (transparent) shapes, then press ‘Ctrl + K’



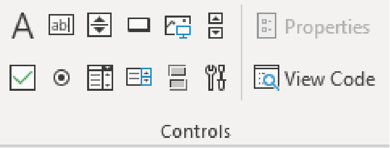
Text to display: <<Selection in Document>>

Select a place in this document:

- Slide Titles
  - 1. Slide 1
  - 2. Goals
  - 3. Disclaimer: A Subjective Se
  - 4. Expressiveness vs Usability
  - 5. D3 via Observable
  - 6. What is D3?
  - 7. Slide 7
  - 8. Observable Notebooks htt
  - 9. D3 Example: Choropleth M

☐ Show and return

- Other controls in ‘Developer’ mode



Choosing the ‘right’ tool

- *There is no single visualisation tool that is optimal for all kinds of users or for all situations!*
- Relevant factors include:
  - Your own skills, preferences and experience (find tools that work for you!)
  - Nature of the data (type, size, shape/structure, overall complexity)
  - Nature of the visualisation (exploratory vs explanatory, static vs interactive, stand-alone vs dashboard, responsive vs fixed dimensions)
- Sometimes you need a mixture of tools
  - For instance, one for pre-processing, a couple for creating different components, another for bringing everything together

Summary – where does each tool shine?

- **Tableau:** Interactive dashboards with multiple connected components
- **D3:** Bespoke visualisations, creative freedom
- **Observable Plot:** Preliminary exploration, common chart types, reusable code snippets
- **RawGraphs:** Rapid prototyping of rarer chart types (e.g. Alluvial diagrams)
- **Microsoft PowerPoint + Google Sheets**: Useful for developing proof-of-concepts, surprisingly versatile
- Important to think about the constraints that are imposed on us, as vis designers, by the tools that we use!

Resources

- Tableau
  - Getting Started
  - Training Videos
  - 2023 Competition Entries
- D3
  - Peter Cook's D3 in Depth
  - Shirley Wu's Introduction
  - UBC's InfoVis Course
  - Mike Bostock's D3 Gallery
  - Prototyping in D3
  - D3 API
- Observable
  - Introduction
  - Twitter
  - YouTube Tutorials
- Plot
  - Introduction
  - Eric Lo's Examples
- Python
  - Python Visualisation Libraries
  - Data Wrangling with Pandas

Academic Papers

- Stolte, C., Tang, D., & Hanrahan, P. (2002). **Polaris**: A system for query, analysis, and visualization of multidimensional relational databases. IEEE Transactions on Visualization and Computer Graphics, 8(1), 52-65.
- Bostock, M., Ogievetsky, V., & Heer, J. (2011). **D3: data-driven documents**. *IEEE transactions on visualization and computer graphics*, *17*(12), 2301-2309.
- Satyanarayan, A., Moritz, D., Wongsuphasawat, K., & Heer, J. (2016). **Vega-lite**: A grammar of interactive graphics. *IEEE transactions on visualization and computer graphics*, *23*(1), 341-350.
- Mauri, M., Elli, T., Caviglia, G., Uboldi, G., & Azzi, M. (2017). **RAWGraphs**: a visualisation platform to create open outputs. In *Proceedings of the 12th biannual conference on Italian SIGCHI chapter* (pp. 1-5).

Thank you!

31