



Actionable visualisation principles and guidance for a foundational data science course

David Sterrett
13 June 2024

Acknowledgements



Kobi Gal



Narges Rohani



Anna Hajitofi



Domas Linkevicius



Filippo Ferrari

and other members of the Foundations of Data Science course team

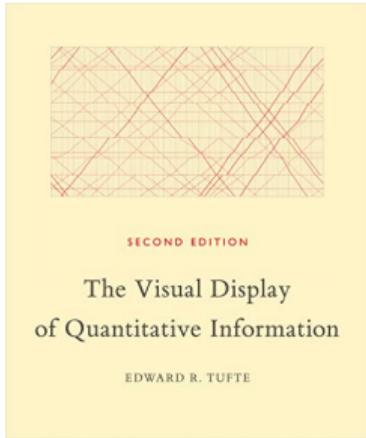
Informatics 2 - Foundations of Data Science



FOUNDATIONS
OF
DATA
SCIENCE

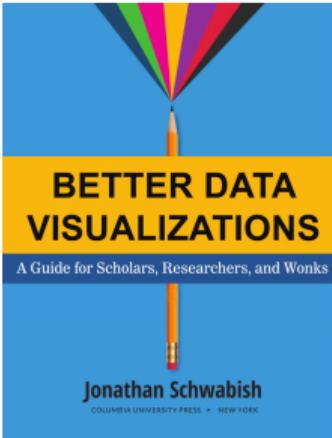
- Level 8, 2nd year (Scotland) undergraduate course, run since 2020/21
- Compulsory for all Computer Science & AI students (~250 students)
- Learning outcomes
 1. **Describe and apply good practices** of data storage, manipulation and **visualisation**
 2. Use standard packages and tools such as **Python and LaTeX**.
 3. Apply basic techniques from descriptive and inferential statistics and machine learning; analyse and interpret output
 4. Critically evaluate data-driven methods and claims from case studies, in order to identify and discuss a) potential ethical issues and b) the extent to which stated conclusions are warranted given evidence provided.
 5. Data science project and report

Visualisation principles & guidelines

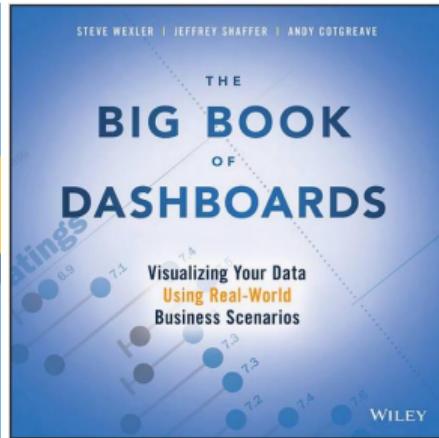


The Visual Display
of Quantitative Information

EDWARD R. TUFTÉ



Jonathan Schwabish



7 Key Principles of Effective Data Visualization

Counts 1 · Follow · Published in Godspeed AI: E-commerce Magazine · 4 min read · Oct 3, 2019

···



Source

Design Principles

Data Visualization Tips

- Top Ten Chart Books and Tools by Angelo De Luca (University of Illinois)
- Data Visualization Checklist by Stephanie Evergreen (check out her free year-long visualization tools)
- Best Chart Rules to Follow from FlowingData
- 7 Basic Rules for Making Charts and Graphs from FlowingData
- Eight Principles of Data Visualization by Ryan Bell
- 10 Rules for Designing Infographics from DataVizDojo (articles by Andy Kirk)
- Helpful Articles by DataVizReport's Christopher King
- Six Principles for Designing Any Chart by Michael Unte (plus links to great guides)

Designing Data Visualizations

• Data Visualization Principles: Lessons from Tufte by Mike Friedman

- A nice, quick summary of some key practical advice from Edward Tufte's seminal book, *Visual Inquiry of Complex Systems*.

• 10 Reasons of Information Chasing the Visualization Prerequisites

- Edward Tufte often writes a number of short posts on data visualization and this work is still relevant to today. These posts provide a quick overview of his main ideas, which are self-referential and can reinforce data visualization principles.

• Data vs. Info vs. Info

- One of Tufte's ideas is data to info ratios, generally stating that when creating a visualization, you should try to minimize the amount of data and maximize the amount of information. This is a good principle, but it's important to remember nothing else is important for understanding. Tufte is a strong advocate of clean, minimal design, this can be a good rule of thumb; however, this Info vs. Info page provides a lot more discussion around this idea.



Best Practices for Data Visualisation

Actionable visualisation principles and guidance for a foundational data science course



THE UNIVERSITY of EDINBURGH
informatics

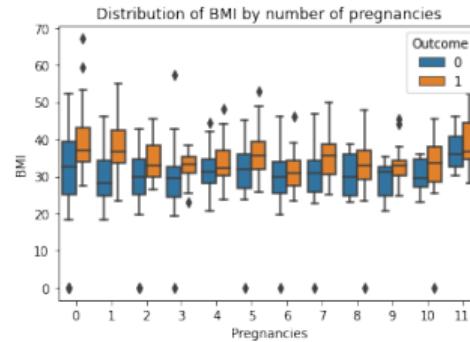
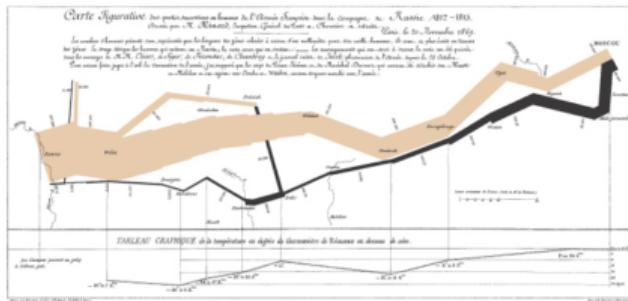
Visualisation principles & guidelines

Tufte objectives (<i>Visual display of quantitative information</i> , p. 13)	Schwabisch <i>Better data visualizations</i>	Tufte 6 principles of graphical integrity (<i>Visual display of quantitative information</i> , p. 77)
Show the data	Show the data	—
Make viewer focus on substance rather than methodology	Reduce the clutter “Start with Gray”	Show data variation, not design variation
Avoid distorting what the data have to say	—	Graphics proportional to numbers (lie factor)
Present many numbers in a small space	Avoid spaghetti charts Use multiples	—

Visualisation principles & guidelines

No single set of principles/guidelines that is:

- appropriate to the level of the course and static visualisations we expect students to produce using Matplotlib and Seaborn

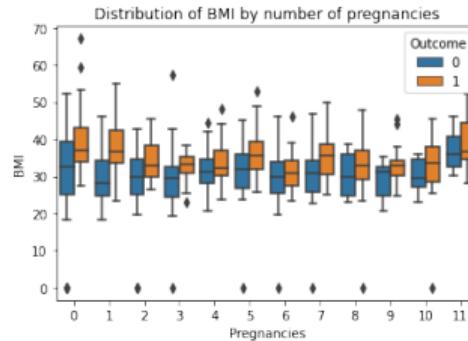
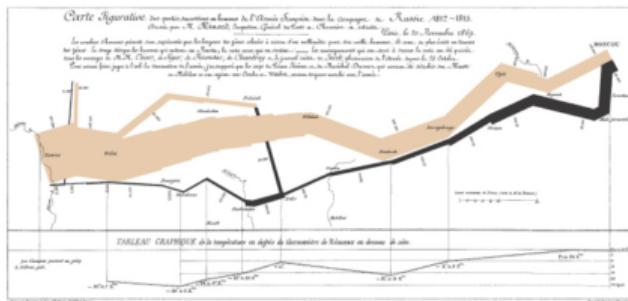


- actionable in the sense that students and markers could assess visualisations against the criteria.

Visualisation principles & guidelines

No single set of principles/guidelines that is:

- appropriate to the level of the course and static visualisations we expect students to produce using Matplotlib and Seaborn



- actionable in the sense that students and markers could assess visualisations against the criteria.

Informatics 2 – Foundations of Data Science: Visualisation principles and guidance

Principle 1: Show the data

Aim to show as much of the data as possible without leading to a confusing visualisation. There are often multiple ways of representing the same dataset, and no "right" answer. The following guidance should help show as much of the data as possible:

- **Choose an appropriate plot type.** Basic types include:

- Bar charts: for plotting numeric variables associated with categorical or ordinal variables, e.g. the mean weight (numeric variable) of male and female (categorical variable) squirrels.
- Line charts: for showing trends of numerical variables over time (a numerical variable).
- Scatterplots: show the relationship between two numeric variables.
- Boxplots: represent the distribution of a numeric variable for multiple categories, e.g. the weights of male and female squirrels.
- Histograms and density plots: good for showing the distribution of a single variable.

- **Show multiple variables by using length, shape, size and colour:**

- Use shape and colour to create extra dimensions for categorical variables. E.g. in a scatterplot of squirrel weight versus length, indicate sex using colour, thus displaying 3 variables. In addition, indicate age categories (ordinal) by changing the size or the shape of the markers (4 variables). But take care that the plot is not too complex to read.
- Barcharts can be extended to two categorical variables and one numerical variable by using colour.

- **Use colour effectively.** (Wexler et al., 2017, pp. 14–18)

- Choose an appropriate colour scale, depending on if the data is sequential (numeric), diverging (numeric with a zero point in the scale) or categorical.
- Colour can also be used to highlight features in the plot, e.g. the largest two bars in a bar plot.

- **Encourage the eye to compare several pieces of data,** e.g. by using multiple plots with the same scale.

– Wexler et al. (2017), p. 31, is a nice example of how this can work better than using multiple symbols on a plot (p. 30).

- **Present many numbers in a small space**

– A boxplot takes up as much space as a barplot, but conveys more information. For example, a boxplot of the squirrel's weight versus sex shows information about the distribution of the weight as well as the median weight.

- **Choose appropriate transforms**

– Transforming data can make features of it clearer. For example, plotting the value of Bitcoin over time shows very little detail about the early history of the currency, when it was not valuable. However, plotting the log of the value of Bitcoin on the y -axis allows this detail to be seen.

Principle 2: Make the meaning of the data clear

A visualisation is meaningless if it's not labelled. Every plot should have:

- Title or caption
- Axis labels as English words
- Units given, where appropriate (e.g. "Length (mm)" not just "Length")
- All variables labelled – e.g. a legend indicating the colours used to represent squirrel sex
- Use graphical and textual annotation – e.g. it can be helpful to highlight a time series with events that you know about

Principle 3: Avoid distorting what the data have to say

Choices in visualisation design can lead to the instant impression given by preattentive processing of the visualisation being quite different to the numbers in the dataset. Tufte (1982) measures the level of distortion in a visualisation by the "Lie factor":

$$\text{Lie factor} = \frac{\text{size of effect shown in graphic}}{\text{size of effect in data}}$$

The following guidelines help to avoid distorting the data:

- **Use appropriate scales and baselines**

– A very common problem is that the baseline (i.e. the lowest point on the y -axis) in a barchart is not zero. This can lead to small differences appearing large.

- **Be aware of limitations of our perception of size**

– Although marker area can be useful for indicating categories, humans are not very good at relating the area to a quantity – we are much better at comparing lengths.

Principle 4: Make the data accessible

A visualisation is meaningless if it's illegible and loses impact if it's difficult to read. To ensure data is accessible:

- **Make sure text is legible**, i.e. font size of minimum 8 points in a PDF, or about 20 points in a presentation. (It is surprising how often talks are given in which it's impossible to read the labels on plots even from the front row.)
- **Use colours that work for people with colour-vision deficiency.** Wexler et al. (2017), Chapter 1 has an excellent introduction to using colour in visualisations.

Principle 5: Focus on the content

Give the viewer's brain as little work to do as possible.

- No chartjunk – e.g. colours that don't have any meaning.
- Reduce clutter
- Consistent colours between plots in a study
- Correct spelling

References

Tufte, E. (1982). *The visual display of quantitative information*. Graphics Press, Cheshire, Connecticut.

Wexler, S., Shaffer, J., and Cotgreave, A. (2017). *The Big Book of Dashboards: Visualizing Your Data Using Real-World Business Scenarios*. Wiley.

Our own set of visualisation principles & guidance

- Mostly based on Tufte, but drawing on Wexler & al's *Big book of dashboards* for colour guidance
- Fit on A4 sheet
- Used by students in workshop sessions to assess visualisations from previous coursework
- Used in the marking scheme for visualisations in two courseworks
- Not perfect – comments welcome!



<https://github.com/Inf2-FDS/fds-visualisation>

Our own set of visualisation principles & guidance

- Mostly based on Tufte, but drawing on Wexler & al's *Big book of dashboards* for colour guidance
- Fit on A4 sheet
- Used by students in workshop sessions to assess visualisations from previous coursework
- Used in the marking scheme for visualisations in two courseworks
- Not perfect – comments welcome!



<https://github.com/Inf2-FDS/fds-visualisation>

Our own set of visualisation principles & guidance

- Mostly based on Tufte, but drawing on Wexler & al's *Big book of dashboards* for colour guidance
- Fit on A4 sheet
- Used by students in workshop sessions to assess visualisations from previous coursework
- Used in the marking scheme for visualisations in two courseworks
- Not perfect – comments welcome!



<https://github.com/Inf2-FDS/fds-visualisation>

Our own set of visualisation principles & guidance

- Mostly based on Tufte, but drawing on Wexler & al's *Big book of dashboards* for colour guidance
- Fit on A4 sheet
- Used by students in workshop sessions to assess visualisations from previous coursework
- Used in the marking scheme for visualisations in two courseworks
- Not perfect – comments welcome!



<https://github.com/Inf2-FDS/fds-visualisation>

Our own set of visualisation principles & guidance

- Mostly based on Tufte, but drawing on Wexler & al's *Big book of dashboards* for colour guidance
- Fit on A4 sheet
- Used by students in workshop sessions to assess visualisations from previous coursework
- Used in the marking scheme for visualisations in two courseworks
- Not perfect – comments welcome!



<https://github.com/Inf2-FDS/fds-visualisation>

The 5 principles

1. Show the data
2. Make the meaning of the data clear
3. Avoid distorting what the data have to say
4. Make the data accessible
5. Focus on the content

Principle 1: Show the data

Aim to show as much of the data as possible without leading to a confusing visualisation.
There are often multiple ways of representing the same dataset, and no “right” answer.

Guidance:

- Choose an appropriate plot type – chart types and how to choose them
- Show multiple variables by using length, shape, size and colour
- Use colour effectively
- Encourage the eye to compare several pieces of data – e.g. with “small multiples”
- Choose appropriate transforms

Presented in lecture via examples and in Jupyter notebooks in lab sessions

Principle 2: Make the meaning of the data clear

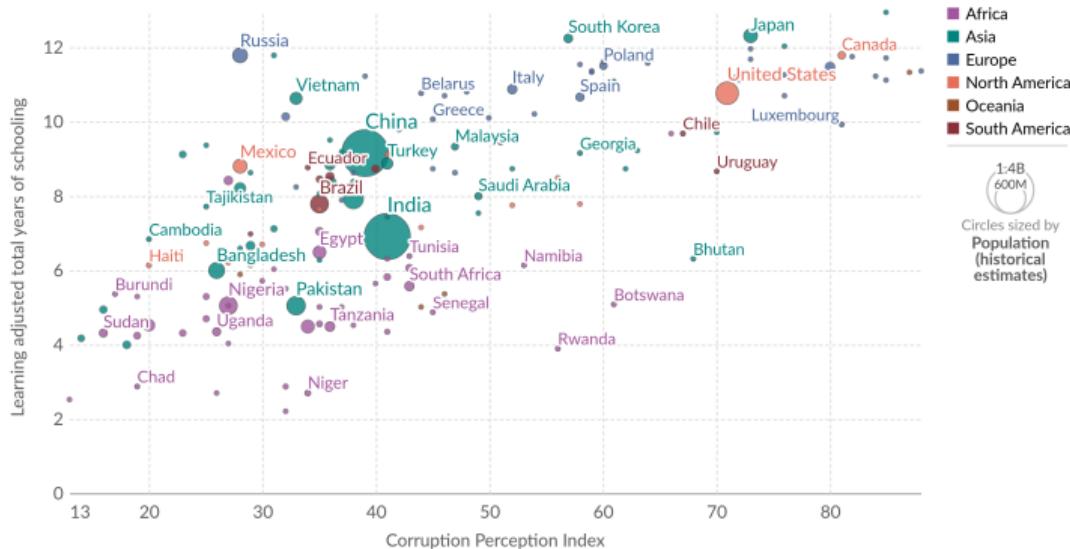
A visualisation is meaningless if it's not labelled. Every plot should have:

- Title or caption
- Axis labels as English words
- Units given, where appropriate
- All variables labelled

Use Graphical and textual annotation

Learning-adjusted years of schooling vs. Corruption Perception Index, 2020
Learning-adjusted years of schooling¹ merge the quantity and quality of education into one metric, accounting for the fact that similar durations of schooling can yield different learning outcomes.

OurWorld
in Data



Data source: Filmer et al. (2018) via World Bank; Transparency International (2018)

Note: Transparency International's Corruption Perception Index lower values reflect higher perceived corruption).

OurWorldInData.org/corruption | CC BY



THE UNIVERSITY of EDINBURGH
informatics

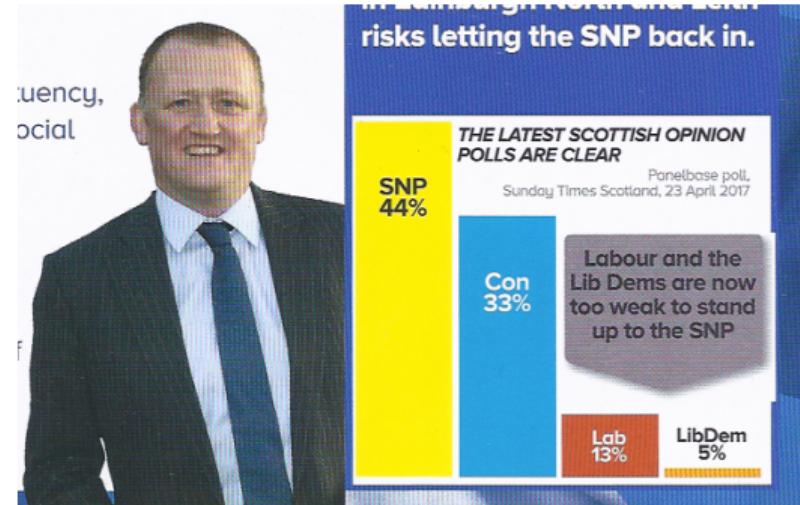
Principle 3: Avoid distorting what the data have to say

Choices in visualisation design can lead to the instant impression given by preattentive processing of the visualisation being quite different to the numbers in the dataset. Tufte (1982) measures the level of distortion in a visualisation by the “Lie factor”:

$$\text{Lie factor} = \frac{\text{size of effect shown in graphic}}{\text{size of effect in data}}$$

The following guidelines help to avoid distorting the data:

- Use appropriate scales and baselines
- Be aware of limitations of our perception of size



Edinburgh Election Leaflet, 2017

Taught in lecture and online quizzes

Principle 4: Make the data accessible

A visualisation is meaningless if it's illegible and loses impact if it's difficult to read. To ensure data is accessible:

- **Make sure text is legible**, i.e. font size of minimum 8 points in a PDF, or about 20 points in a presentation. (It is surprising how often talks are given in which it's impossible to read the labels on plots even from the front row.)
- **Use colours that work for people with colour-vision deficiency.** Wexler et al. (2017), Chapter 1 has an excellent introduction to using colour in visualisations.

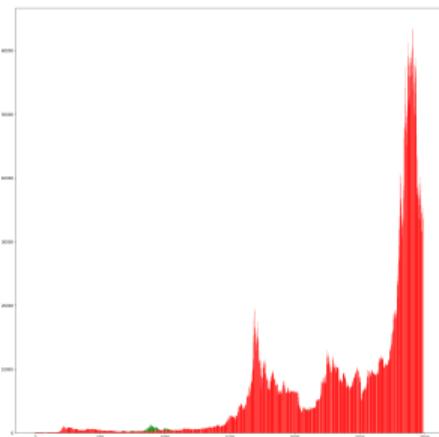
Principle 5: Focus on the content

Give the viewer's brain as little work to do as possible.

- No chartjunk – e.g. colours that don't have any meaning.
- Reduce clutter
- Consistent colours between plots in a study
- Use an appropriate number of decimal places
- Correct spelling

Principles in summative assessment: Entry question

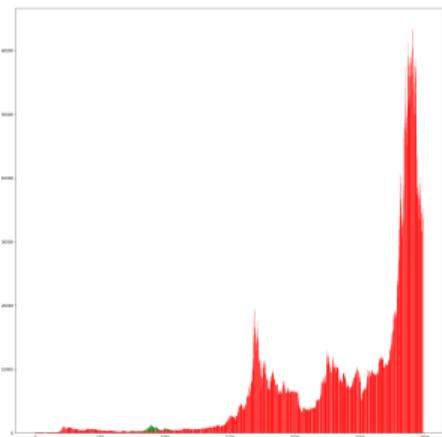
1. We give data and code to produce a plot:



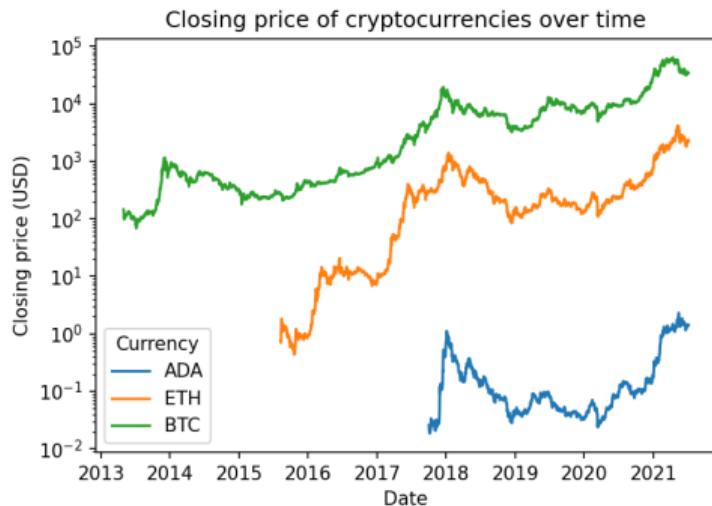
List the problems with this plot. For each problem, state: **which principle** is broken **why** and **how you could fix the problem?**

Principles in summative assessment: Entry question

1. We give data and code to produce a plot:



2. Implement the changes proposed in (1)



List the problems with this plot. For each problem, state: **which principle** is broken **why** and **how you could fix** the problem?

Principles in summative assessment: Intermediate & advanced Questions

- **Visualisation coursework: Visualisation with goal set:** e.g. “Compare the number of deaths and admissions between council areas using a visualisation of your choice.”
- **Visualisation coursework: Create your own visualisation:** “generate a question, and create a plot that addresses the question.”
- **Rubric:**
 - “Overall effectiveness” (mostly “Show the data”) (70% of marks)
 - “Meaning of the data”
 - “Data not distorted”
 - “Accessibility: Text size”
- **Final project:** Rubric entries that relate to the visualisation principles

Principles in formative assessment

- Before coursework, get students to work in groups of about 5 to mark visualisations from students in previous years (with permission)
- Shared spreadsheets to record marks

Student view during most of exercise:

A	B	C	D	E	F	G	H	I	J		
1											
2	Question	Paper	Visualisation: Show the data	Visualisation: Clear meaning	Visualisation: Avoid distortion	Visualisation: Accessible	Visualisation: Focus on content	Explanation quality	Code readability	Total	Fee
3	Q6	Sample1	Inadequate - 1	Fair - 2	Good - 3	Inadequate - 1	Fair - 2	Good - 3	Fair - 2	14	
4		Sample3	Inadequate - 1	Inadequate - 1	Inadequate - 1	Good - 3	Inadequate - 1	Fair - 2	Excellent - 4	13	
5		Sample2	Good - 3	Inadequate - 1	Fair - 2	Good - 3	Fair - 2	Inadequate - 1	Good - 3	15	
6											

< > Group A (David) Group B Group C Group D Group E Group F All Results By Sample All Results By Group +

Workbook Statistics Give Feedback to Microsoft

Principles in formative assessment

- Before coursework, get students to work in groups of about 5 to mark visualisations from students in previous years (with permission)
- Shared spreadsheets to record marks

Student view during most of exercise:

	A	B	C	D	E
1					
2	Question Q6	Paper	Visualisation: Show the data	Visualisation: Clear meaning	Visualisation: Avoid distortion
3		Sample1	Inadequate - 1	Fair - 2	Good - 3
4		Sample3	Inadequate - 1	Inadequate - 1	Inadequate - 1
5		Sample2	Good - 3	Inadequate - 1	Fair - 2
6					Good - 3

< > ≡ Group A (David) Group B Group C Group D Group E Group F All Results By Sample All Results By Group +

Workbook Statistics

Principles in formative assessment

- Before coursework, get students to work in groups of about 5 to mark visualisations from students in previous years (with permission)
- Shared spreadsheets to record marks

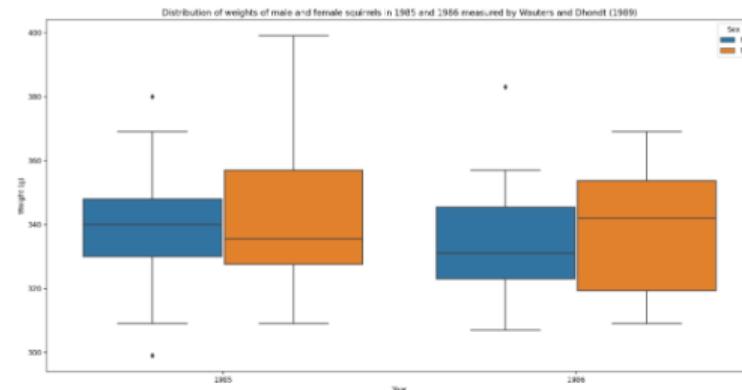
Student view at end of exercise:

1	A	B	C	D	E	F	G	H	I	J	K
2	Question	Sample	Group	Sample	Visualisation: Clear meaning	Visualisation: Avoid distortion	Visualisation: Accessible	Visualisation: Focus on content	Explanation quality	Code readability	Total
3	Q6	Sample 1	A	Fair – 2	Fair – 2	Fair – 2	Inadequate – 1	Good – 3	Fair – 2	Good – 3	15
4			B	Inadequate – 1	Fair – 2	Good – 3	Inadequate – 1	Fair – 2	Good – 3	Fair – 2	14
5			C	Inadequate – 1	Inadequate – 1	Inadequate – 1	Inadequate – 1	Fair – 2	Absent – 0	Absent – 0	6
6			D	Fair – 2	Inadequate – 1	Good – 3	Inadequate – 1	Absent – 0	Fair – 2	Inadequate – 1	10
7			E	Fair – 2	Fair – 2	Good – 3	Inadequate – 1	Good – 3	Good – 3	Inadequate – 1	15
8			F	Fair – 2	Inadequate – 1	Fair – 2	Inadequate – 1	Good – 3	Fair – 2	Fair – 2	13
9											
10		Sample 2	A	Good – 3	Good – 3	Fair – 2	Inadequate – 1	Fair – 2	Fair – 2	Fair – 2	15
11			B	Inadequate – 1	Inadequate – 1	Inadequate – 1	Good – 3	Inadequate – 1	Fair – 2	Excellent – 4	13
12			C	Fair – 2	Good – 3	Good – 3	Fair – 2	Fair – 2	Absent – 0	Absent – 0	12
13			D	Inadequate – 1	Fair – 2	Fair – 2	Fair – 2	Absent – 0	Absent – 0	Absent – 0	7
14			E	Good – 3	Good – 3	Excellent – 4	Fair – 2	Good – 3	Good – 3	Fair – 2	20
15			F	Good – 3	Good – 3	Fair – 2	Absent – 0	Absent – 0	Absent – 0	Absent – 0	8
16											
17		Sample 3	A	Good – 3	Good – 3	Excellent – 4	Fair – 2	Good – 3	Fair – 2	Good – 3	20
18			B	Good – 3	Inadequate – 1	Fair – 2	Good – 3	Fair – 2	Inadequate – 1	Good – 3	15
19			C	Absent – 0	Absent – 0	Absent – 0	Absent – 0	Absent – 0	Absent – 0	Absent – 0	0
20			D	Absent – 0	Absent – 0	Absent – 0	Absent – 0	Absent – 0	Absent – 0	Absent – 0	0
21			E	Good – 3	Good – 3	Good – 3	Good – 3	Good – 3	Fair – 2	Good – 3	20
22			F	Absent – 0	Absent – 0	Absent – 0	Absent – 0	Absent – 0	Absent – 0	Absent – 0	0
23											

The biggest problem for students: creating visualisations with legible text!

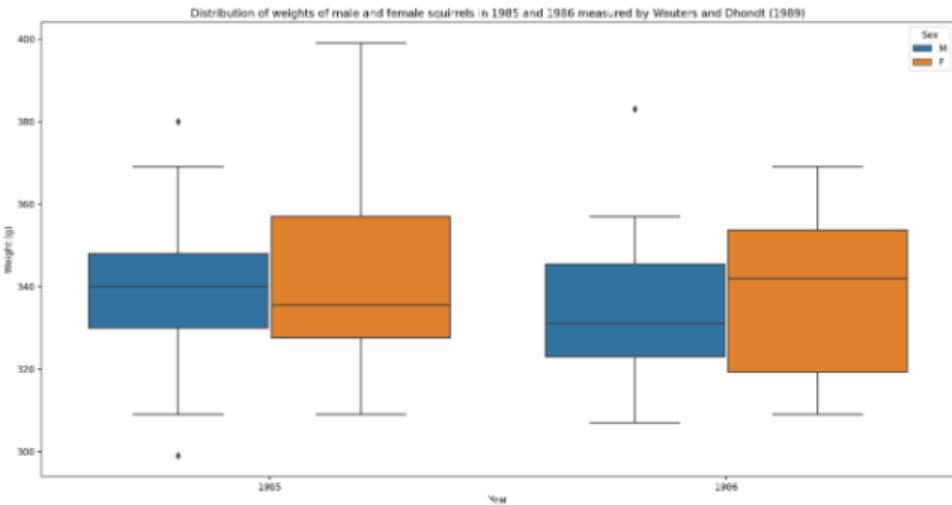
“Too small”:

- Less than a point size or two smaller than the body text, e.g. 9pt if the body size is 11pt
- If I have to zoom into a PDF (or would have to use a magnifying glass if printed out)
- If I can't read the text on slides in a lecture theatre



The curse of figure sizes

```
mpl.rcParams['font.size'] = 10
plt.figure(figsize=(18, 9))
ax = sns.boxplot(x='Year', y='Weight (g)', hue='Sex', data=dat)
plt.title('Distribution of weights of male and female squirrels in 1985 and 1986 measured by Wauters and Dhondt (1989)')
plt.savefig('squirrel-18x9.png')
```

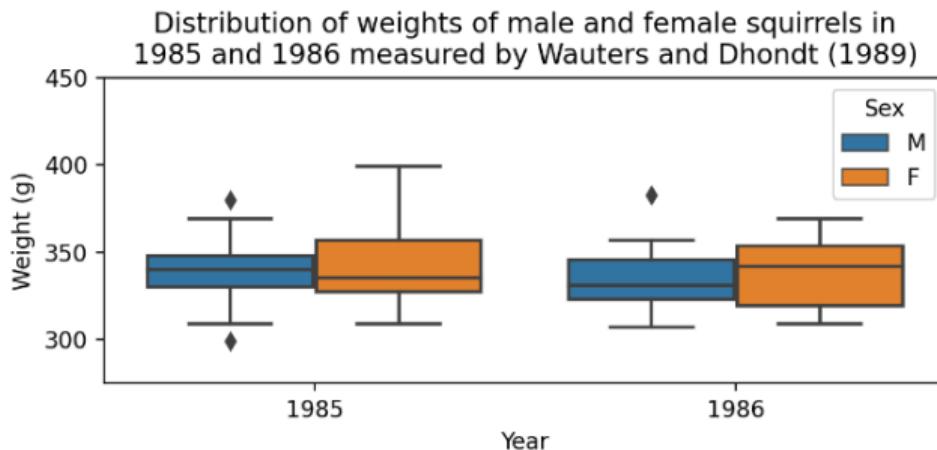


Advice:

- Set fontsize to 10pt
- Make the width of your plot no wider than 6in
- Ensure the title isn't wider than the plot
- plt.tight_layout()
- Work plot around font size constraint

The curse of figure sizes

```
mpl.rcParams['font.size'] = 10
plt.figure(figsize=(6, 3))
ax = sns.boxplot(x='Year', y='Weight (g)', hue='Sex', data=dat)
plt.title('Distribution of weights of male and female squirrels in\n1985 and\n1986 measured by Wauters and Dhondt (1989)')
plt.ylim([275, 450])
sns.move_legend(ax, 'upper right')
plt.tight_layout()
plt.savefig('squirrel-6x3.png')
```

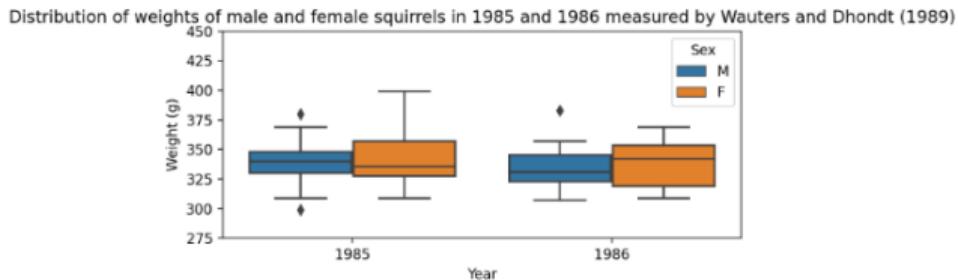


Advice:

- Set fontsize to 10pt
- Make the width of your plot no wider than 6in
- Ensure the title isn't wider than the plot
- plt.tight_layout()
- Work plot around font size constraint

The curse of figure sizes

```
mpl.rcParams['font.size'] = 10
plt.figure(figsize=(6, 3))
ax = sns.boxplot(x='Year', y='Weight (g)', hue='Sex', data=dat)
plt.title('Distribution of weights of male and female squirrels in 1985 and  
1986 measured by Wauters and Dhondt (1989)')
plt.ylim([275, 450])
sns.move_legend(ax, 'upper right')
plt.tight_layout()
plt.savefig('squirrel-6x3-title.png')
```

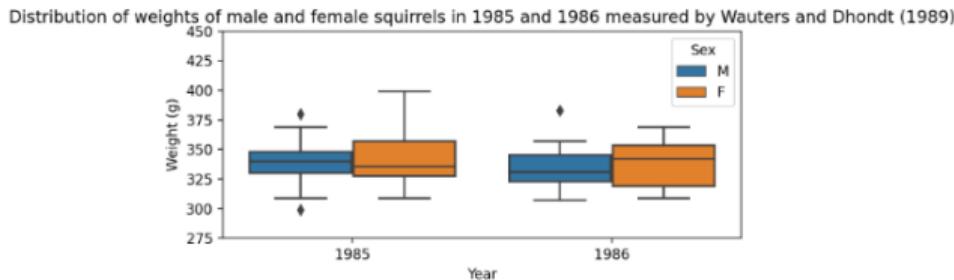


Advice:

- Set fontsize to 10pt
- Make the width of your plot no wider than 6in
- Ensure the title isn't wider than the plot
- `plt.tight_layout()`
- Work plot around font size constraint

The curse of figure sizes

```
mpl.rcParams['font.size'] = 10
plt.figure(figsize=(6, 3))
ax = sns.boxplot(x='Year', y='Weight (g)', hue='Sex', data=dat)
plt.title('Distribution of weights of male and female squirrels in 1985 and 1986 measured by Wauters and Dhondt (1989)')
plt.ylim([275, 450])
sns.move_legend(ax, 'upper right')
plt.tight_layout()
plt.savefig('squirrel-6x3-title.png')
```

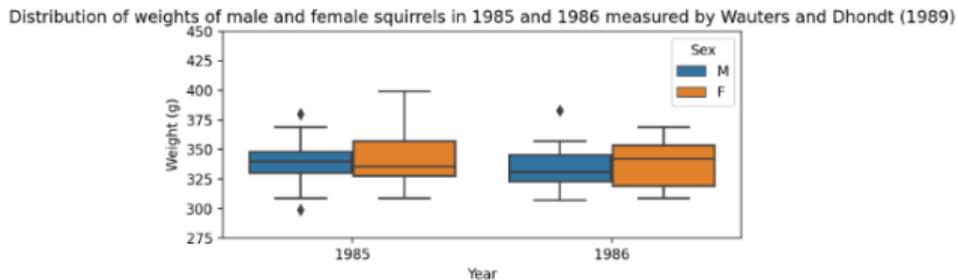


Advice:

- Set fontsize to 10pt
- Make the width of your plot no wider than 6in
- Ensure the title isn't wider than the plot
- plt.tight_layout()
- Work plot around font size constraint

The curse of figure sizes

```
mpl.rcParams['font.size'] = 10
plt.figure(figsize=(6, 3))
ax = sns.boxplot(x='Year', y='Weight (g)', hue='Sex', data=dat)
plt.title('Distribution of weights of male and female squirrels in 1985 and  
1986 measured by Wauters and Dhondt (1989)')
plt.ylim([275, 450])
sns.move_legend(ax, 'upper right')
plt.tight_layout()
plt.savefig('squirrel-6x3-title.png')
```



Advice:

- Set fontsize to 10pt
- Make the width of your plot no wider than 6in
- Ensure the title isn't wider than the plot
- plt.tight_layout()
- Work plot around font size constraint

Evaluation

- Instructor's perspective:
 - Concise set of principles that can be taught fairly straightforwardly
 - And lend themselves to summative assessment and formative work
 - Some principles could be simplified?
 - Overly specific, e.g. Lie Factor?
- Students' perspective:
 - 43/50 (Over two years) report formative workshop exercise is useful
 - Some still mess up on font sizes
 - However, by end-of-course project, some very accomplished visualisations are produced



<https://github.com/Inf2-FDS/fds-visualisation>

Evaluation

- Instructor's perspective:
 - Concise set of principles that can be taught fairly straightforwardly
 - And lend themselves to summative assessment and formative work
 - Some principles could be simplified?
 - Overly specific, e.g. Lie Factor?
- Students' perspective:
 - 43/50 (Over two years) report formative workshop exercise is useful
 - Some still mess up on font sizes
 - However, by end-of-course project, some very accomplished visualisations are produced



<https://github.com/Inf2-FDS/fds-visualisation>