



Week 5: Practical Coding and Statistics in Astronomy

SESSION 2: DATA VISUALIZATION

STARFISH SCHOOL 2021

Visualizing your Data

You have your data, now what?

Ask yourself:

- Do I need to visually represent/plot this data?
- Is this plot answering the question I'm asking?
 - Is this looking for specific values of something, or is this some form of comparison?
 - Are the quantities that I'm plotting the right ones?
- Is this plot hiding something that I should be looking at?

Graphical Perception

Graphical Perception: Theory, Experimentation,
and Application to the Development of
Graphical Methods

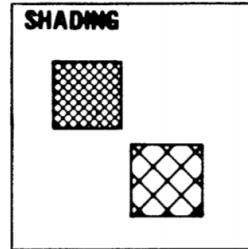
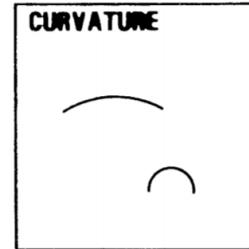
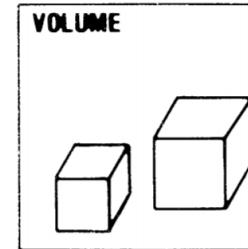
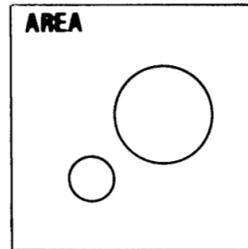
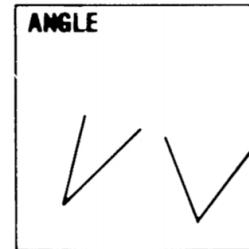
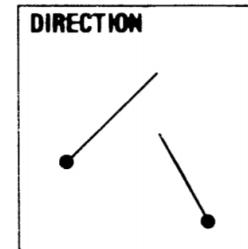
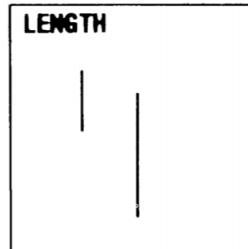
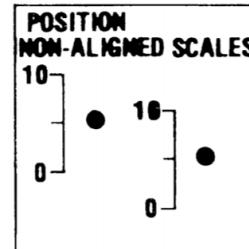
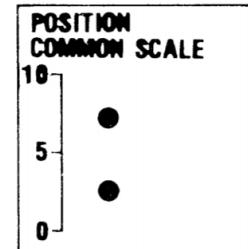
WILLIAM S. CLEVELAND and ROBERT McGILL*

Source: *Journal of the American Statistical Association*, Sep., 1984, Vol. 79, No. 387
(Sep., 1984), pp. 531-554

Published by: Taylor & Francis, Ltd. on behalf of the American Statistical Association

Stable URL: <https://www.jstor.org/stable/2288400>

Journal of the American Statistical Association, September 1984



COLOR SATURATION

Figure 1. Elementary perceptual tasks.

Graphical Perception

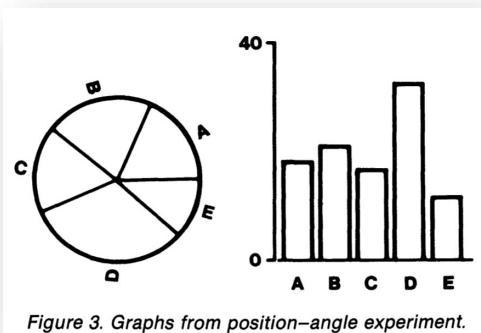
Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods

WILLIAM S. CLEVELAND and ROBERT McGILL*

Source: *Journal of the American Statistical Association*, Sep., 1984, Vol. 79, No. 387
(Sep., 1984), pp. 531-554

Published by: Taylor & Francis, Ltd. on behalf of the American Statistical Association

Stable URL: <https://www.jstor.org/stable/2288400>



Journal of the American Statistical Association, September 1984

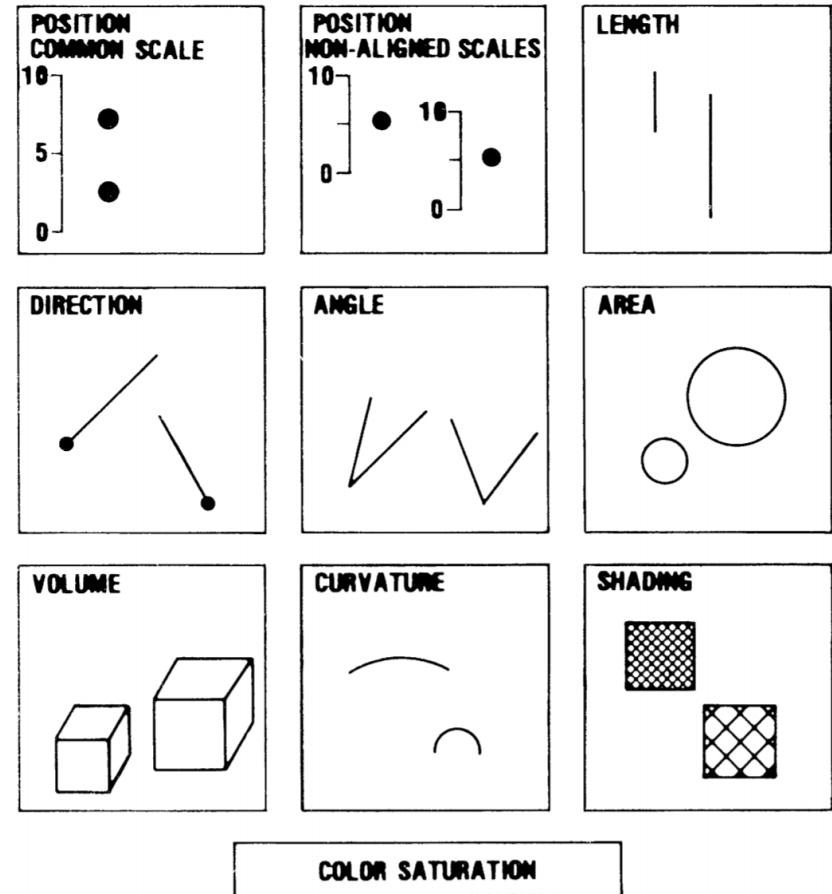


Figure 1. Elementary perceptual tasks.

Graphical Perception

Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods

WILLIAM S. CLEVELAND and ROBERT McGILL*

Source: *Journal of the American Statistical Association*, Sep., 1984, Vol. 79, No. 387
(Sep., 1984), pp. 531-554

Published by: Taylor & Francis, Ltd. on behalf of the American Statistical Association

Stable URL: <https://www.jstor.org/stable/2288400>

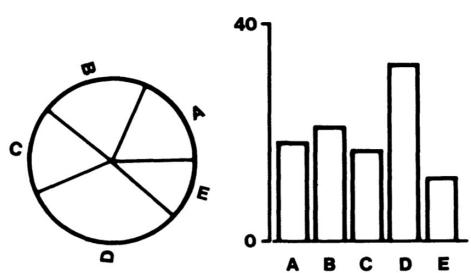


Figure 3. Graphs from position-angle experiment.

The following are the 10 elementary tasks in Figure 1, ordered from most to least accurate:

1. Position along a common scale
2. Positions along nonaligned scales
3. Length, direction, angle
4. Area
5. Volume, curvature
6. Shading, color saturation

Journal of the American Statistical Association, September 1984

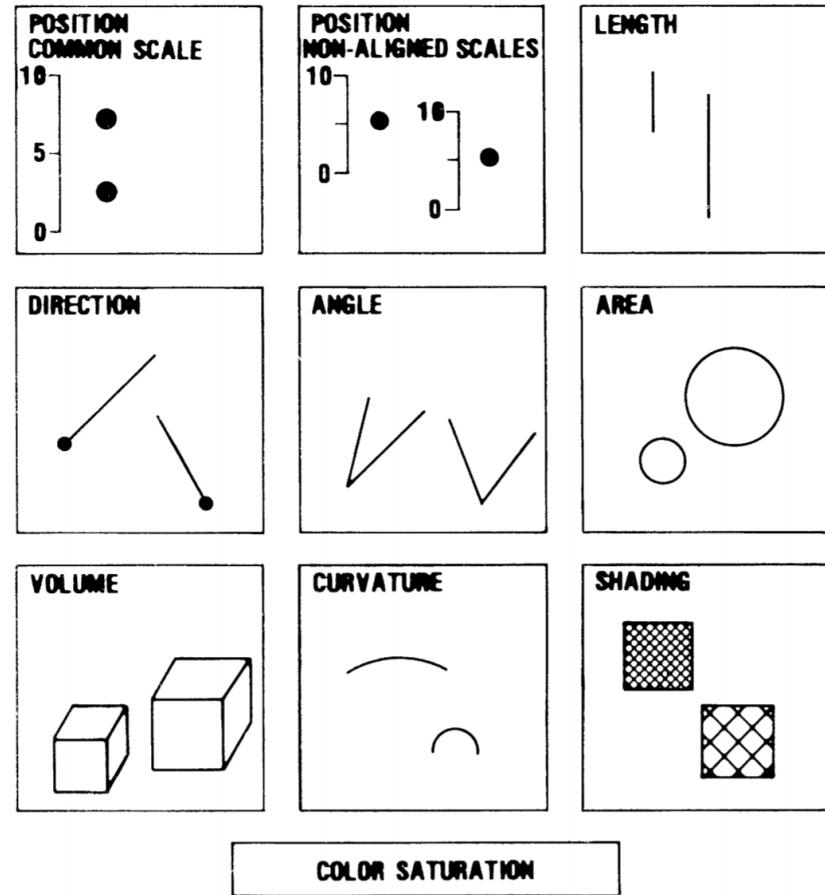
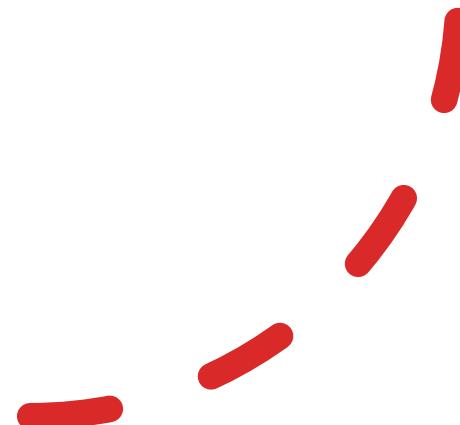


Figure 1. Elementary perceptual tasks.

The Grammar of Visual Elements

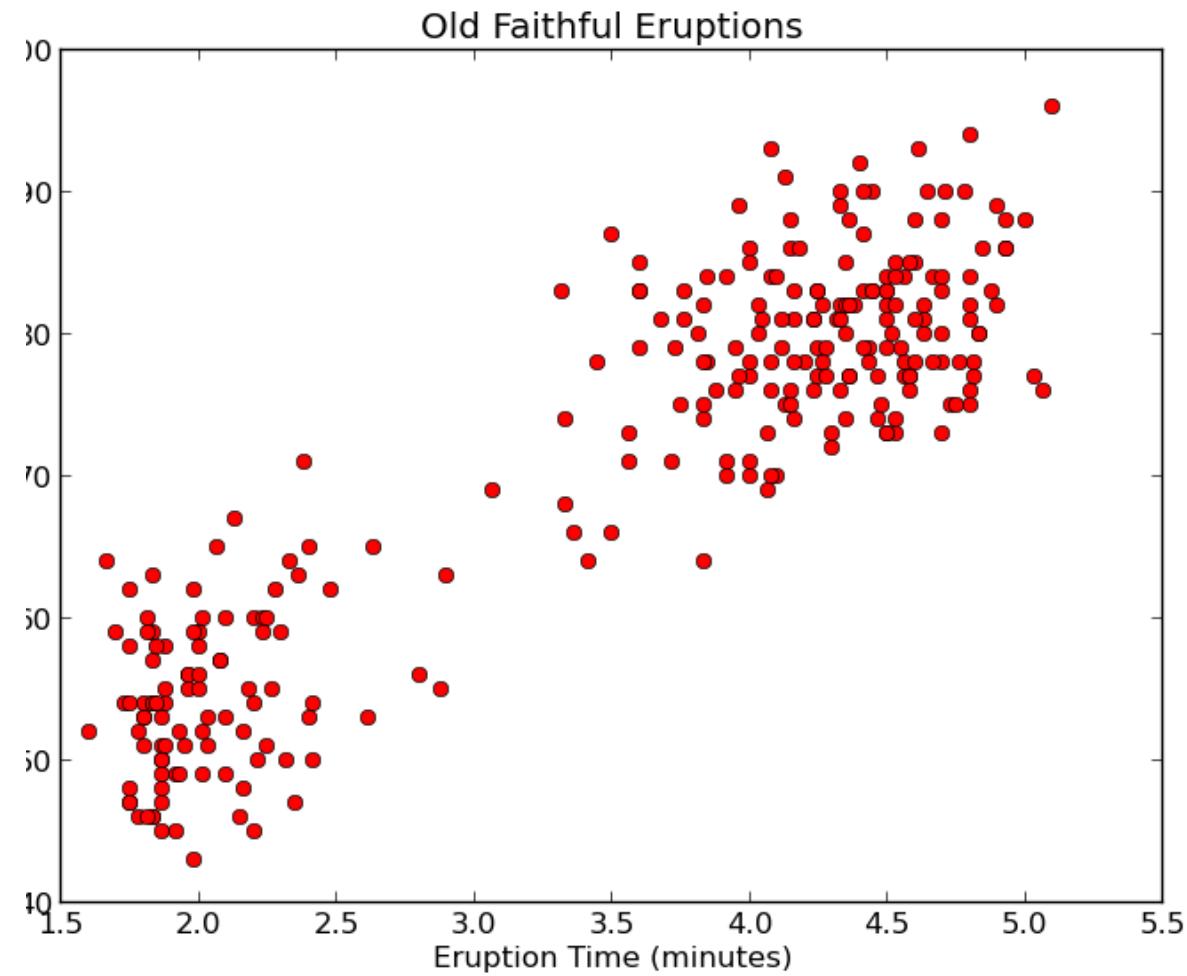
Ways of encoding information in visual form

- Position
- Size
- Shape
- Thickness
- Colour
- Opacity



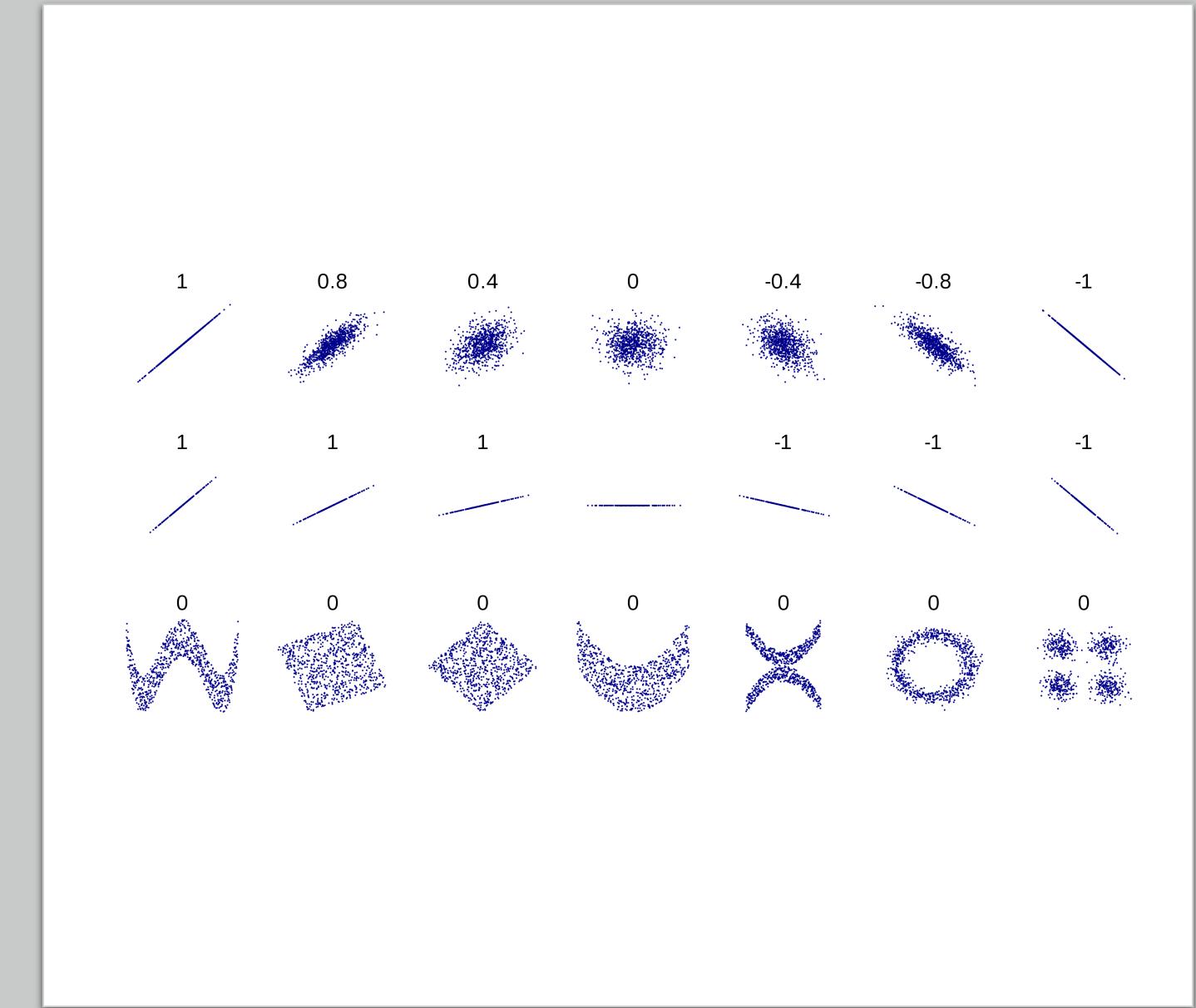
Position

Representation of two dimensions
easily
Strong cultural convention



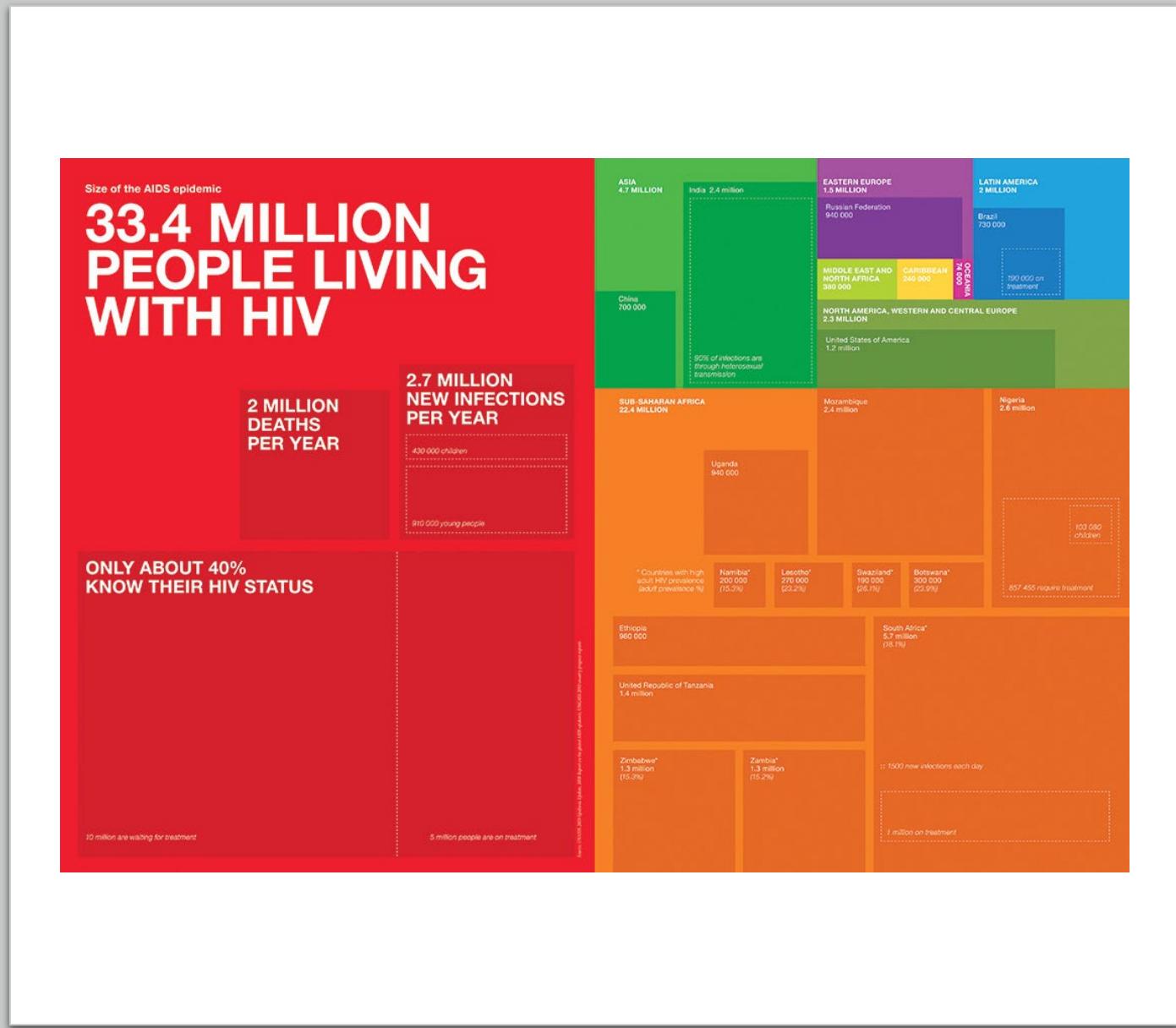
Position

Representation of two dimensions easily
Strong cultural convention



Size

Area as an indication of magnitude
Easy Comparison between elements

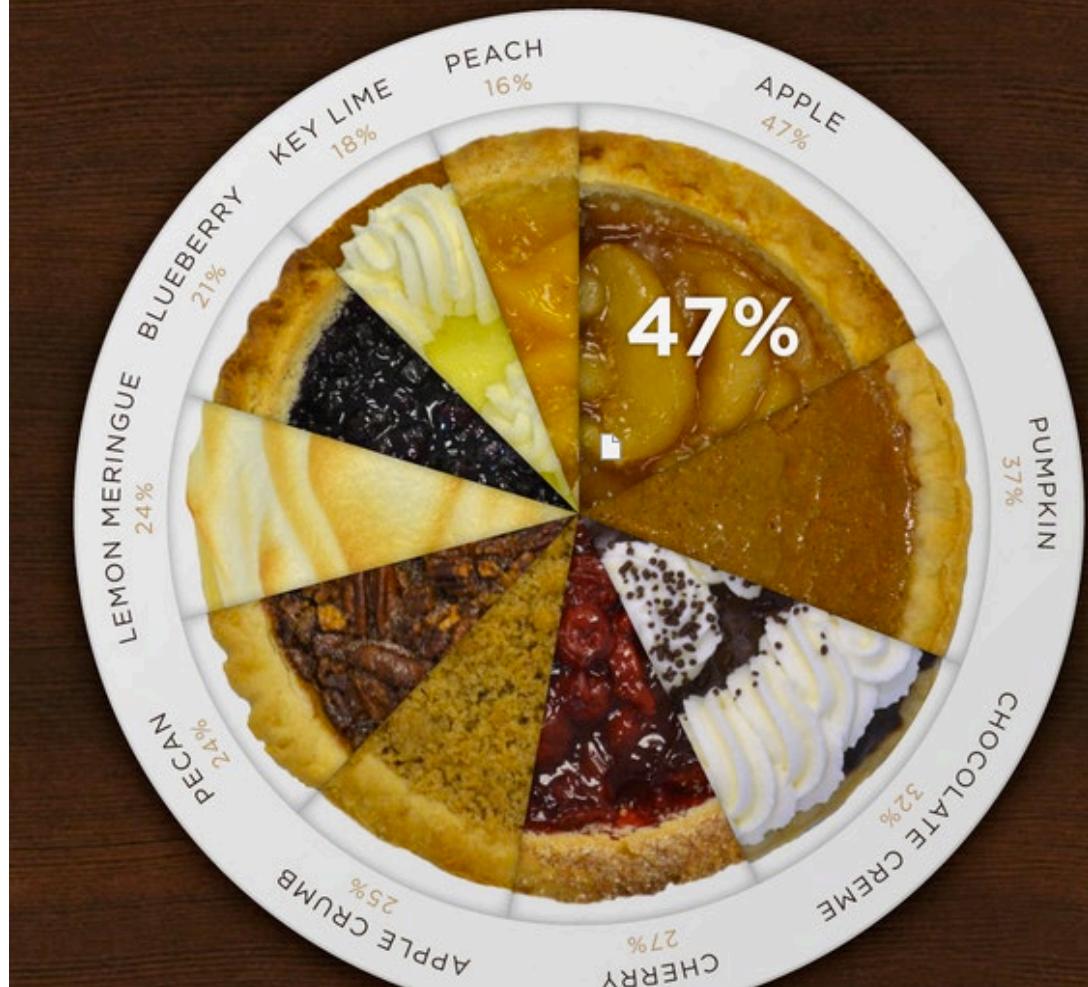


Size

Area as an indication of magnitude

Easy Comparison between elements

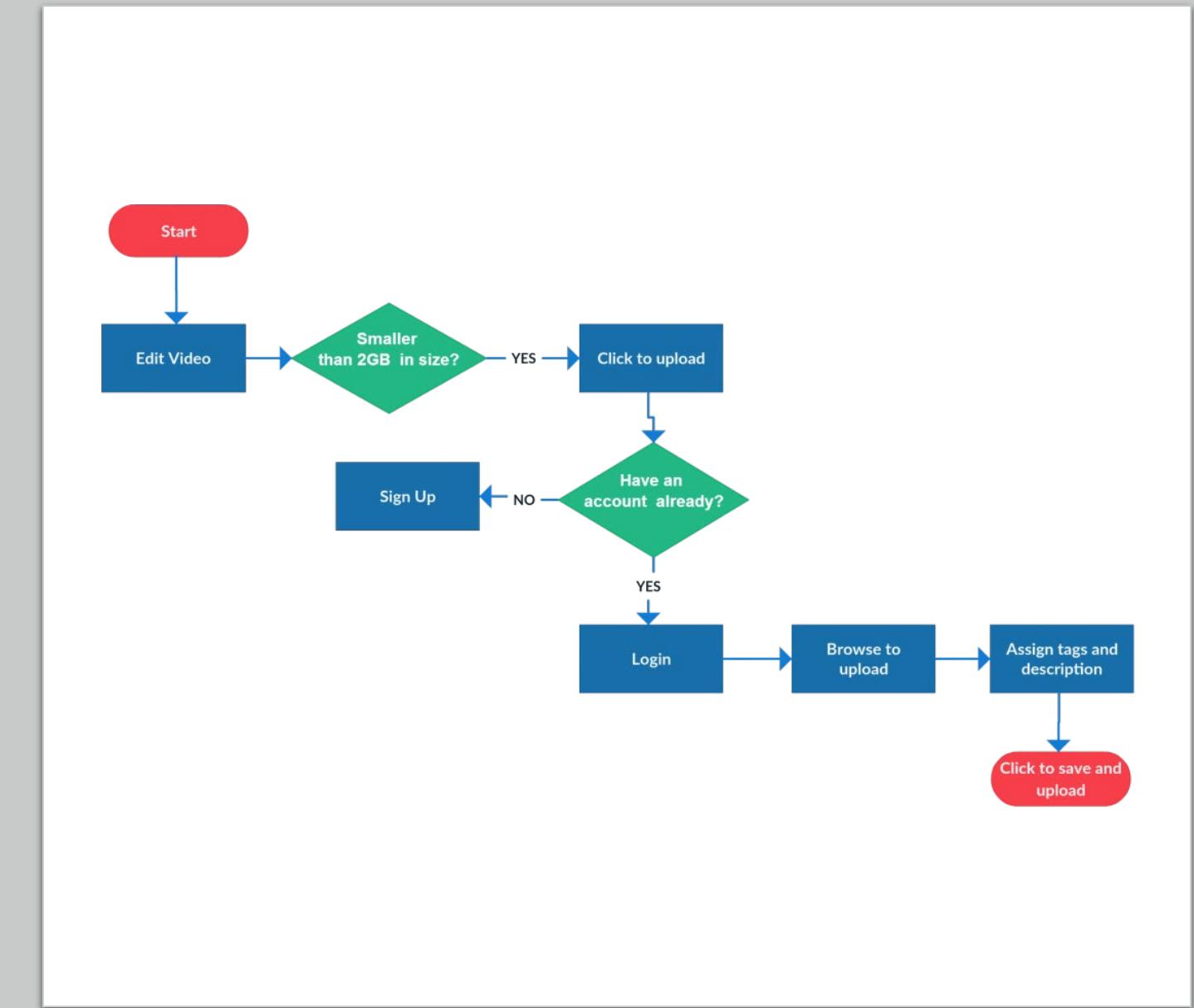
What are your favorite 3 pies?

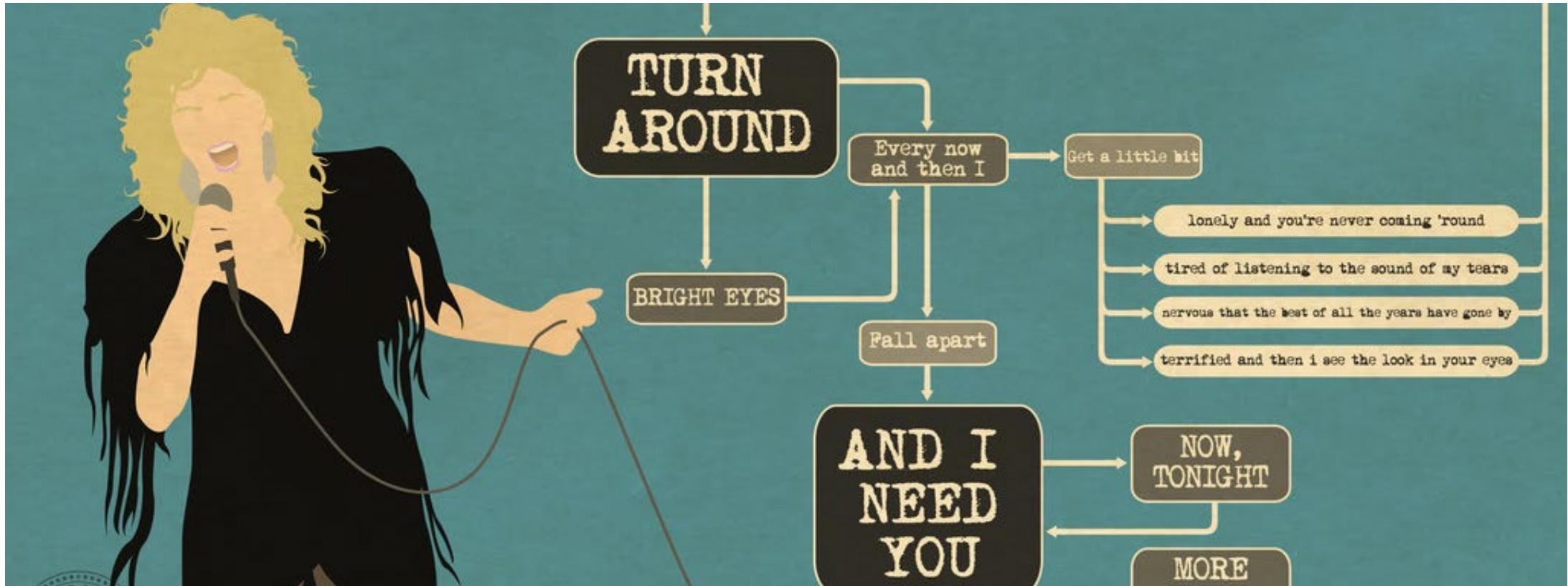


Shape

Can encode categorization information

Can use mappings or cultural conventions





Shape

Can encode categorization information
Can use mappings or cultural conventions



Streetcar Map



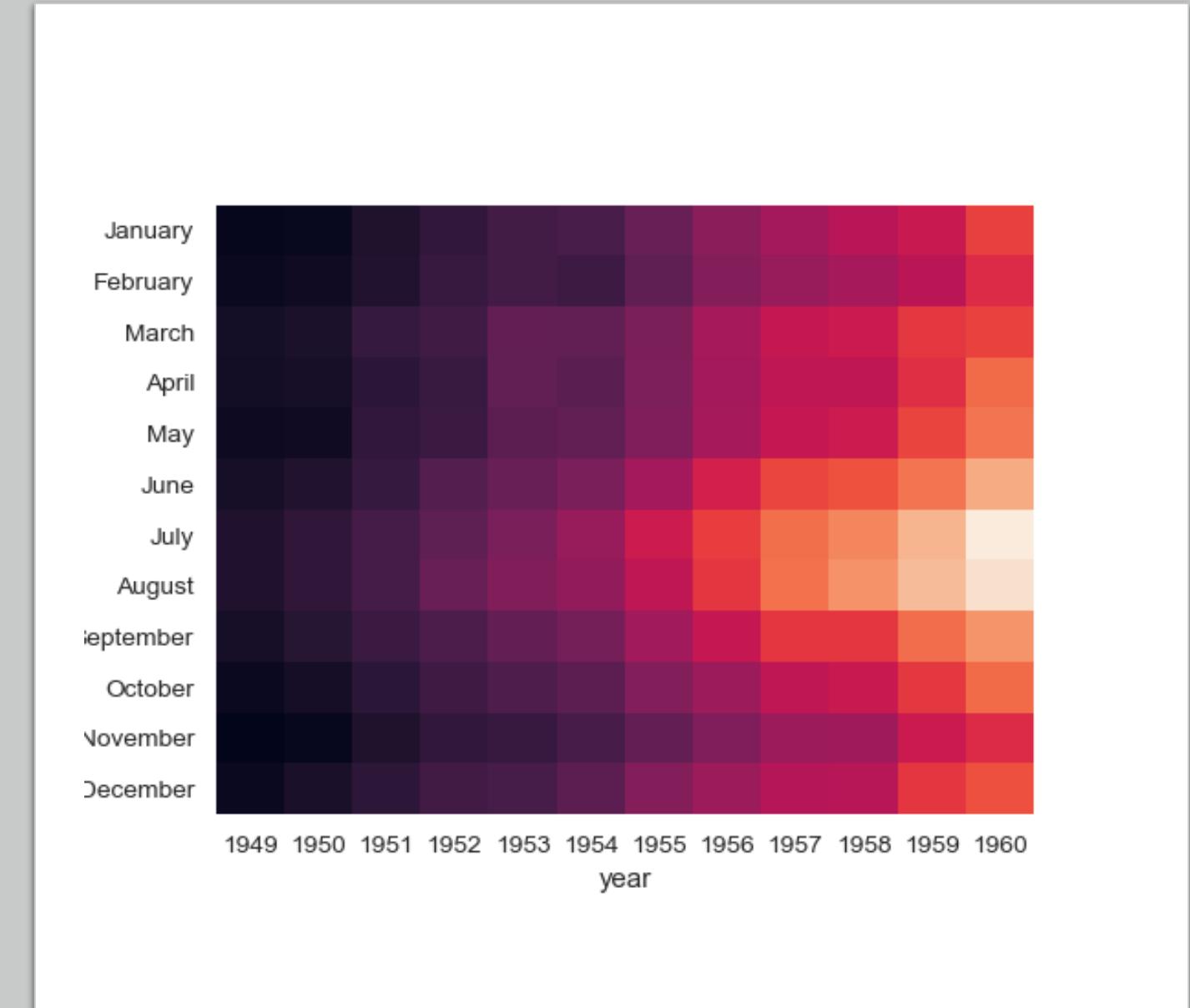
Thickness

Underutilized but highly effective
Provide a means to categorize visual elements

Colour

Encodes magnitude with natural comparative properties

Potential to encode large ranges and types of variation



Colour

Encodes magnitude with natural comparative properties

Potential to encode large ranges and types of variation

Trends in Polar Bear Subpopulations

Subpopulation size

No. of Bears

- <500
- 500-1,000
- 1,000-2,000
- 2,000-3,000
- ?(?) Unknown

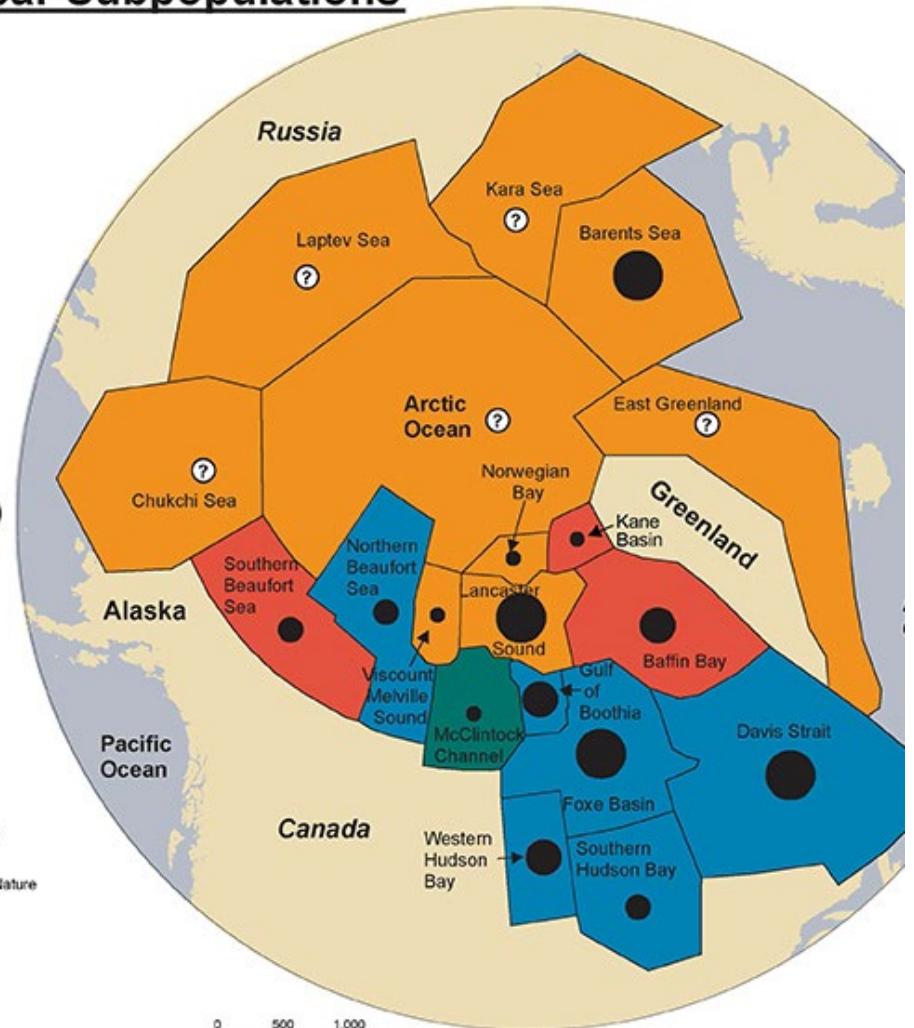
Population Trend (2015)

- Stable
- Increasing
- Declining
- Data deficient

Produced by World Wildlife Fund Canada, May 2015
Sources: Polar Bear Specialists Group, January 2015
Range Boundaries IUCN, 2012.
Projection: North Pole Stereographic.
© 1986 Panda symbol WWF-World Wildlife Fund for Nature
(also known as World Wildlife Fund)
© "WWF" is a WWF Registered Trademark



0 500 1,000 km



Opacity

- Often used in conjunction with colour
- Provides crude information on magnitude



The Three Layers of a Plot

Highest Level: Getting the basic story out quickly

Intermediate Level: Justifying/confirming the story

Deepest Level: Being able to reproduce the story

Visualization Accessibility

“When we don’t intentionally include,
we unintentionally exclude.”

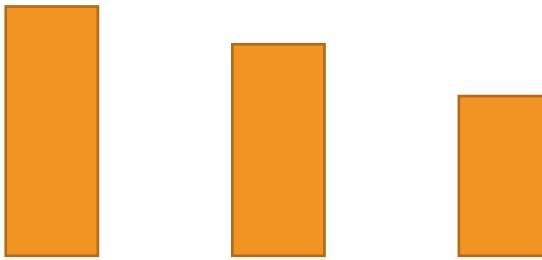
Visualization Accessibility

“When we don’t intentionally include,
we unintentionally exclude.”



<https://youtu.be/SWB-KLXN-Ok>

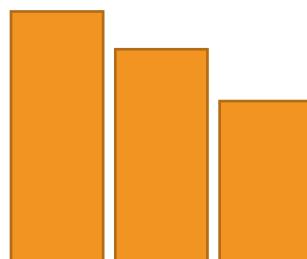
Visualization Accessibility



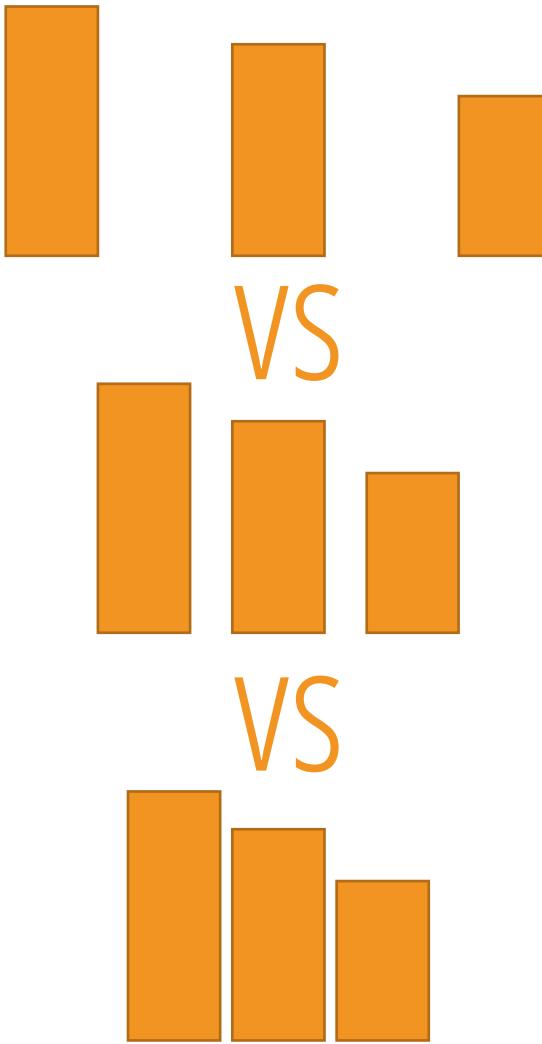
VS



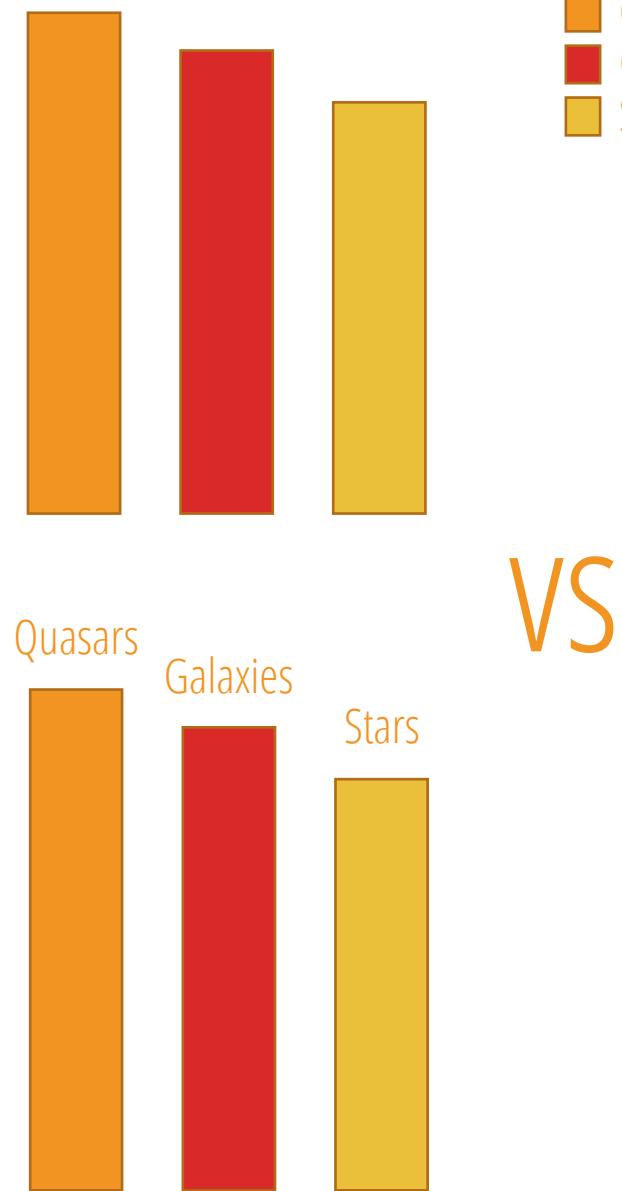
VS



Visualization Accessibility



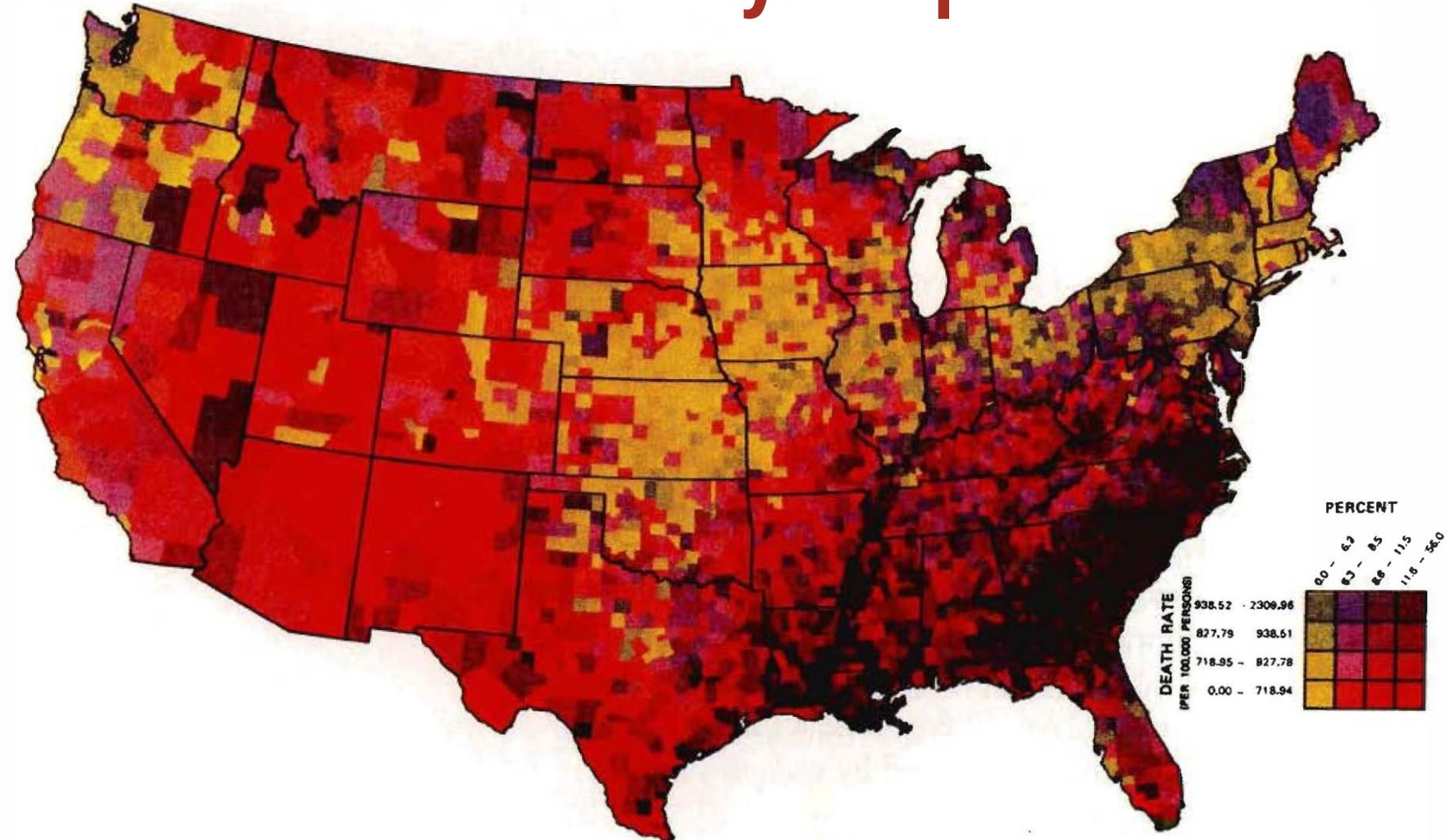
VS

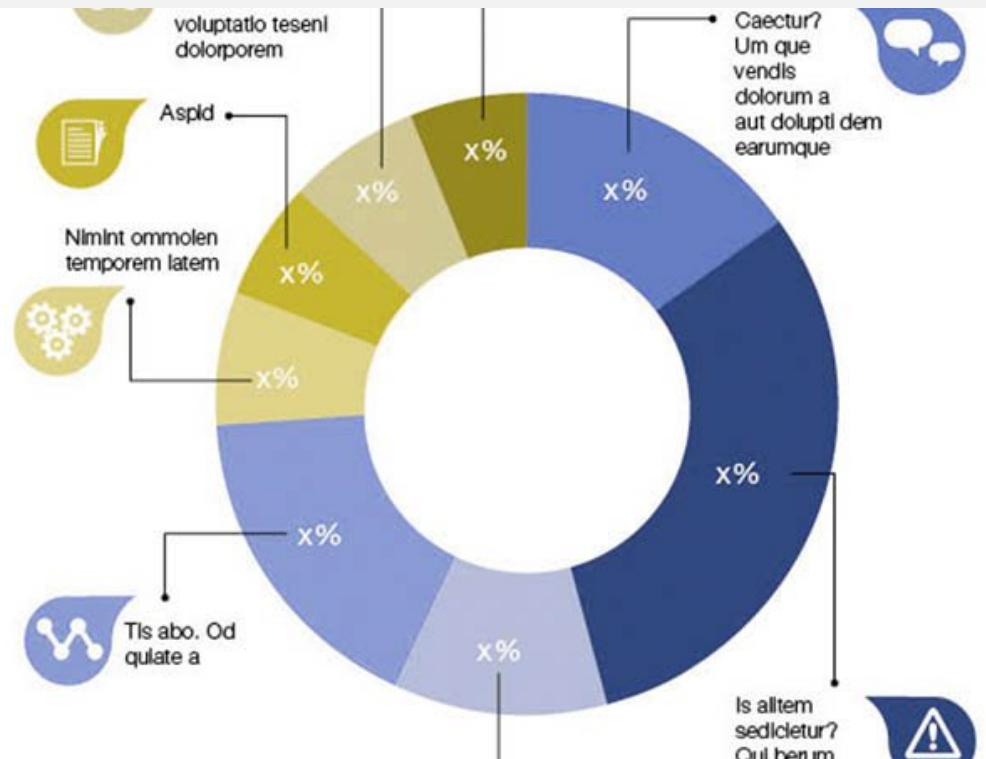
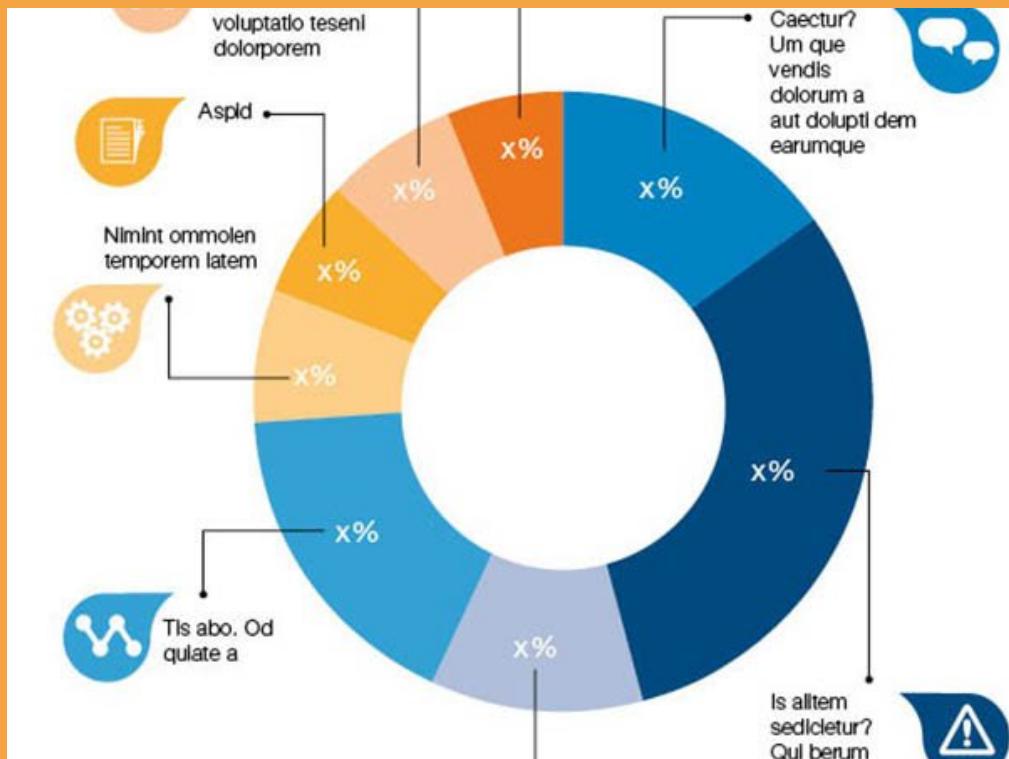


VS

Quasars
Galaxies
Stars

Visualization Accessibility: Graphical Puzzles



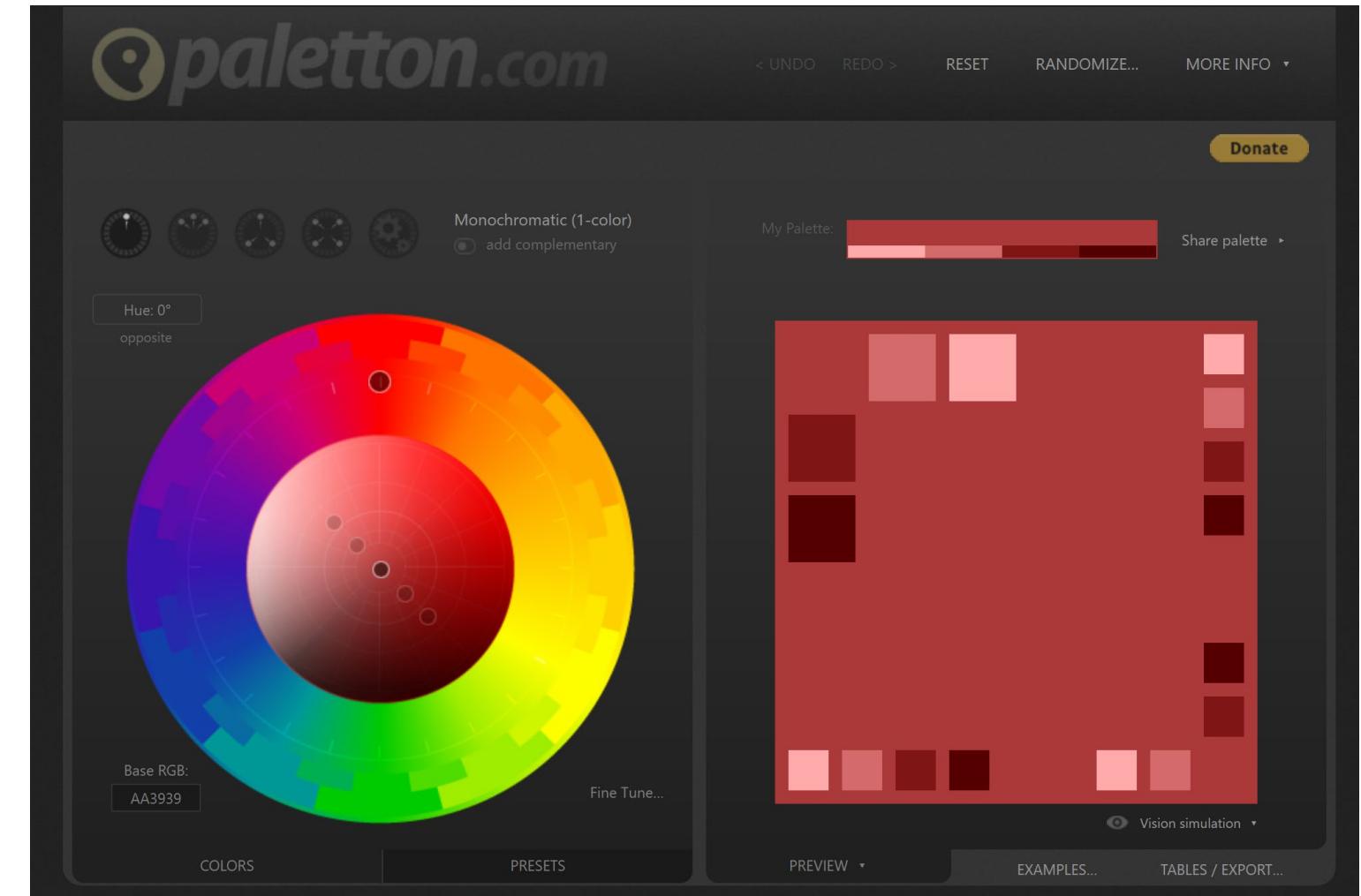


Visualization Accessibility: Colour Blindness



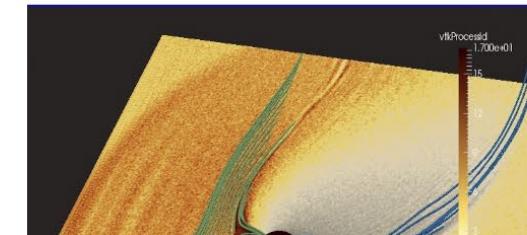
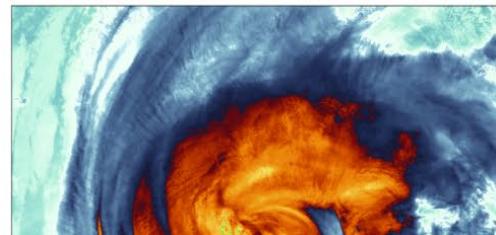
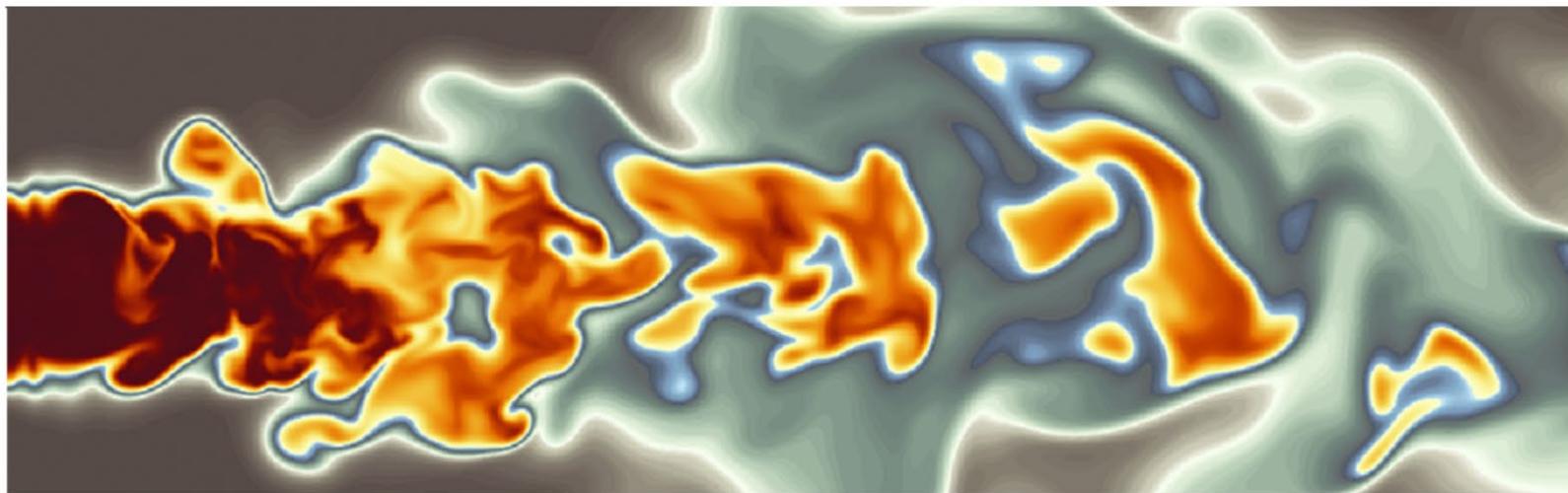
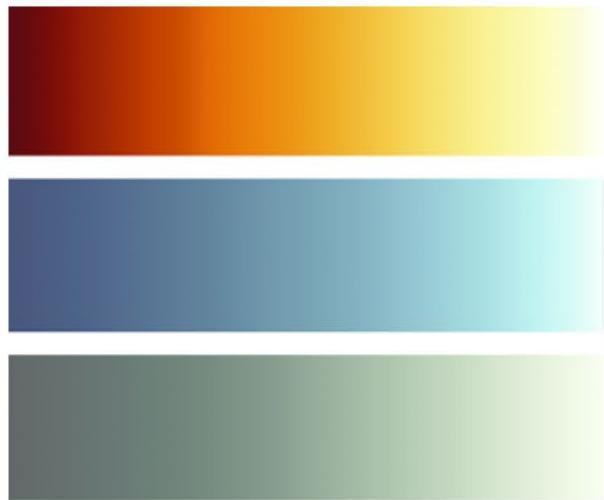
Colours are relative

Choosing Harmonious Colours



[HOME](#)[Colormaps](#)[ColorMoves](#)[Color Sets](#)[3-D Color](#)[Gallery](#)[Publications](#)[People](#)[Resources](#)

SciVisColor: Color Tools and Strategies for Scientific Visualization



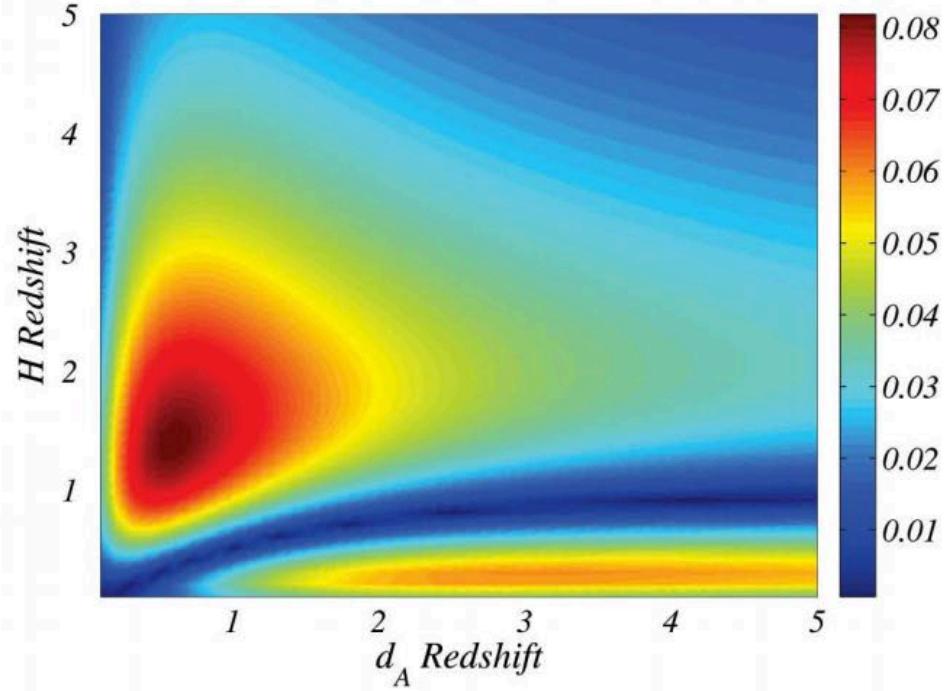
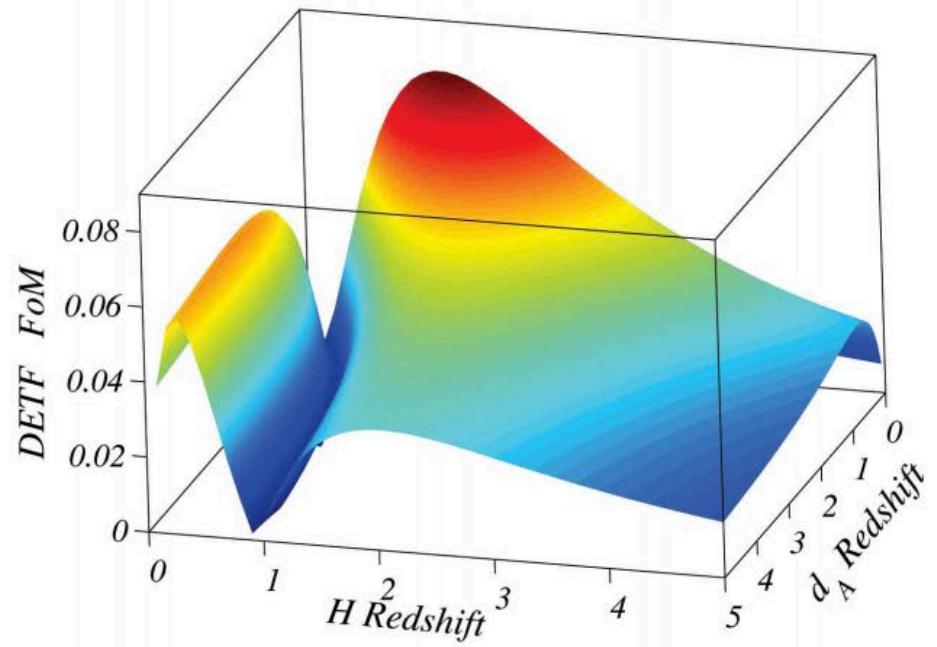
Colour Theory is complex

Which is brightest? Which is darkest?



Colour Theory is complex

Which is brightest? Which is darkest?

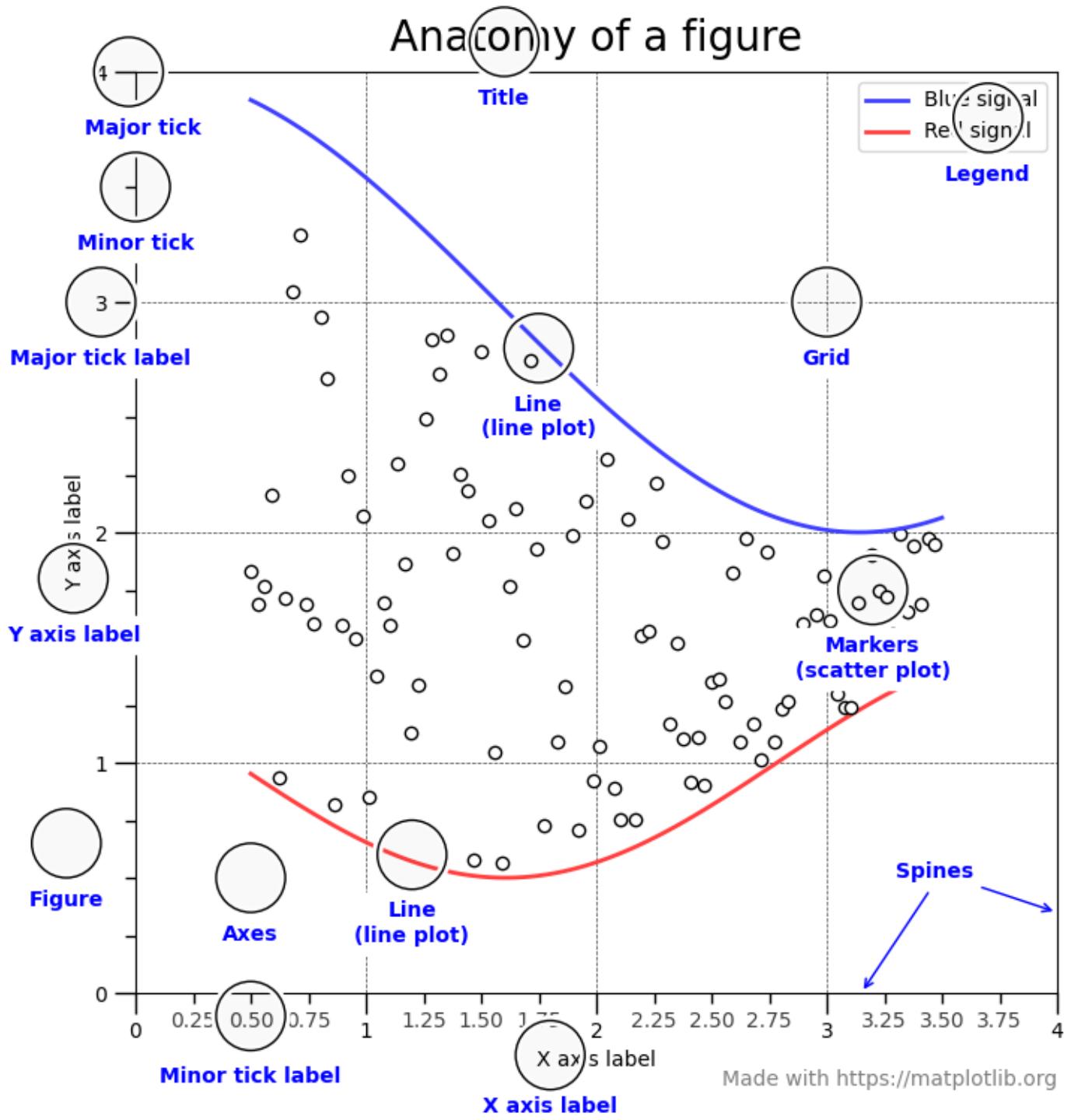


Just because you can,
doesn't mean you
should

- <https://arxiv.org/pdf/0906.0993.pdf>

Making Matplotlib do your bidding!

Just need to look up what the element is called, and you can edit it!



Matplotlib Styles

Create your own style file to make your own defaults:

1. Make a text file called `<stylename>.mplstyle` and put it in
`~/.config/matplotlib`
2. Use it by calling `plt.style.use("<stylename>")`

You can also change your global defaults by creating/editing a file called
`matplotlibrc`

Details: <https://matplotlib.org/3.2.1/tutorials/introductory/customizing.html>

Typography Matters

Critical Tool for visualization

At its best, provides a visual hierarchy and communicates

At its worst, adds chaos and confusion to the visualization



She Ji: The Journal of Design, Economics, and
Innovation

Volume 2, Issue 1, Spring 2016, Pages 59-87



Using Typography to Expand the Design Space of Data Visualization

Richard Brath Ebad Banissi

Show more

<https://doi.org/10.1016/j.sheji.2016.05.003>

Under a Creative Commons license

Get rights and content

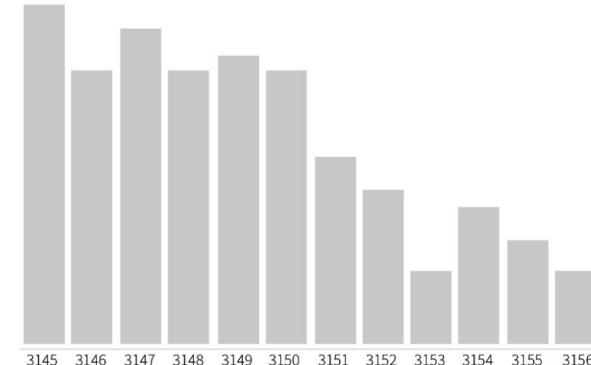
open access

Highlights

- The design space of data visualization can be expanded with

Lore ipsum Dolor

Lore ipsum dolor sit amet, consectetur adipiscing elit.



Notes: wikipedia and census.org

Header Text
PT Sans, 22px/24px, bold

Subheader Text
Noto Sans, 14px/20px, light

Callout Text
Noto Sans, 14px/20px, light

Label/Axis Text
Noto Sans, 14px/20px, light

Notes/Sources Text
Noto Sans, 14px/20px, light

National Geographic/ Cartography

They are masters at using and executing visual hierarchies with font faces



Good Open Fonts to use for Visualization

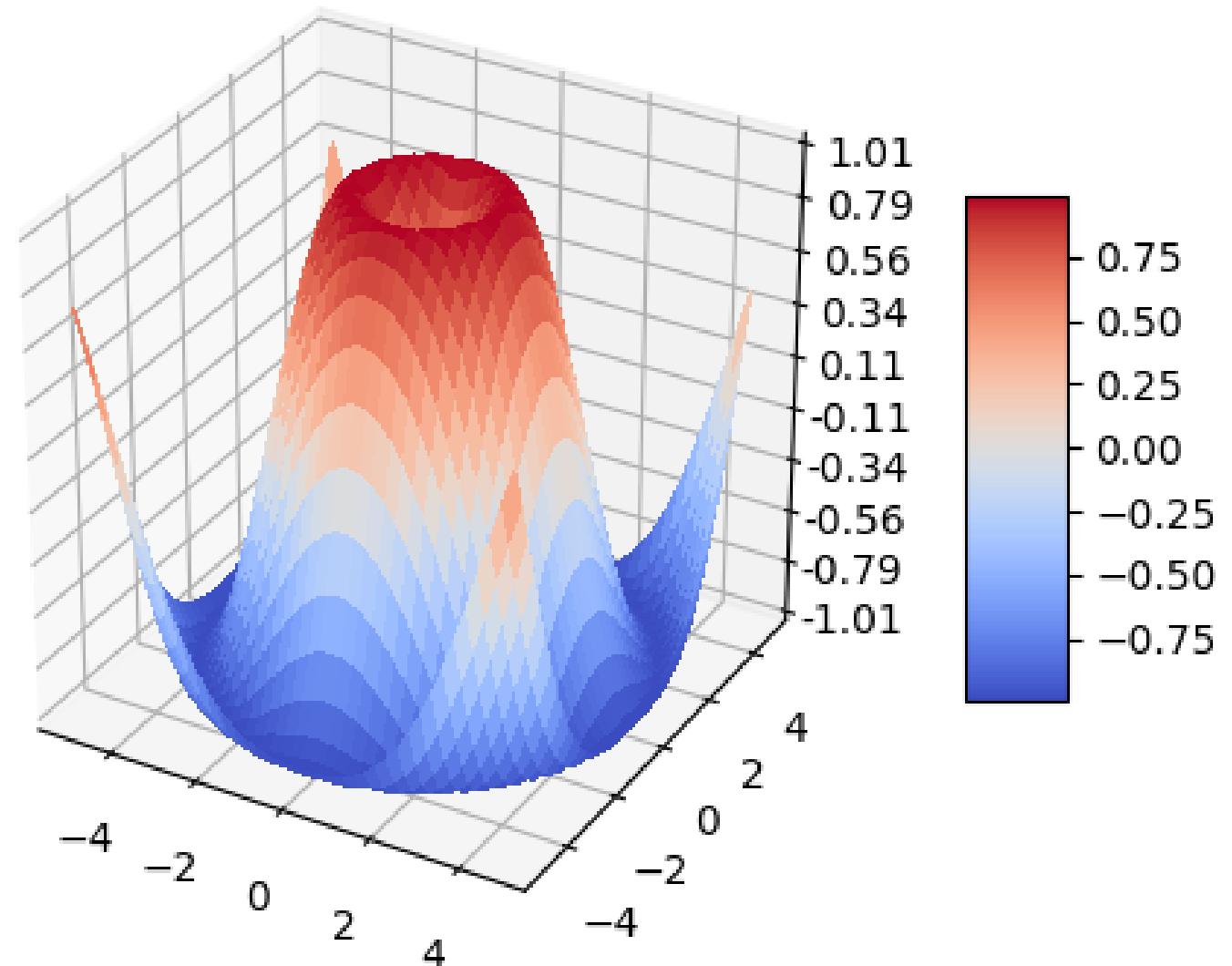
All professionally designed, provide good range of weights and are free to use (open source licensing)

Source Serif Pro Source Sans 3

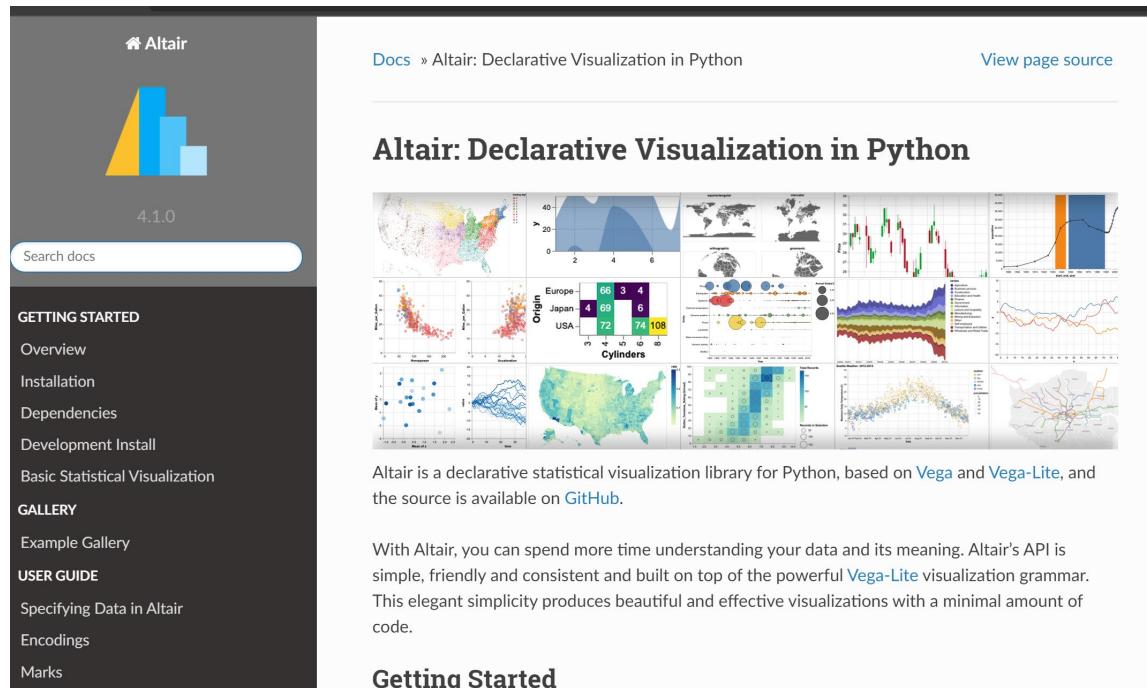
Open Sans

Going Beyond the Basics: 3D

mplot3d gives you built-in 3D capabilities.



Going Beyond the Basics: Interactive Plots



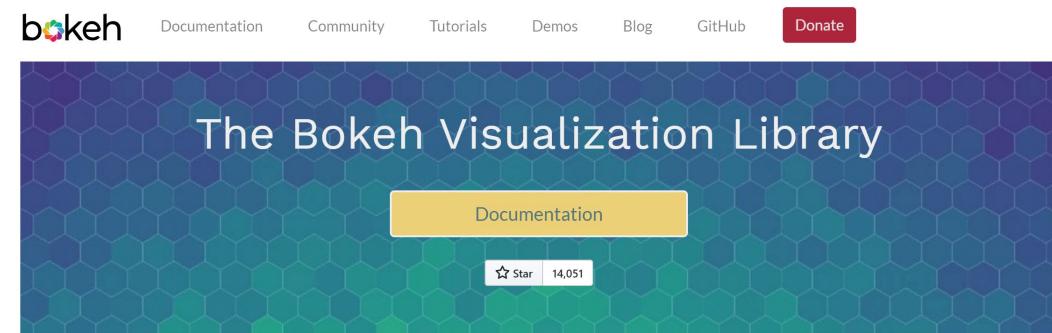
The screenshot shows the Altair documentation homepage. At the top left is the Altair logo (a stylized bar chart) and the version number 4.1.0. A search bar is located below the logo. The main navigation menu includes sections for **GETTING STARTED**, **GALLERY**, and **USER GUIDE**. The **GETTING STARTED** section contains links for Overview, Installation, Dependencies, Development Install, and Basic Statistical Visualization. The **GALLERY** section has links for Example Gallery. The **USER GUIDE** section includes Specifying Data in Altair, Encodings, and Marks. Below the navigation is a large grid of various Altair visualizations, such as maps, scatter plots, and line charts.

Altair: Declarative Visualization in Python

Altair is a declarative statistical visualization library for Python, based on Vega and Vega-Lite, and the source is available on GitHub.

With Altair, you can spend more time understanding your data and its meaning. Altair's API is simple, friendly and consistent and built on top of the powerful Vega-Lite visualization grammar. This elegant simplicity produces beautiful and effective visualizations with a minimal amount of code.

Getting Started



The screenshot shows the Bokeh documentation homepage. At the top right is the Bokeh logo and a navigation bar with links for Documentation, Community, Tutorials, Demos, Blog, GitHub, and a red **Donate** button. The main title is "The Bokeh Visualization Library". Below the title is a yellow callout button labeled "Documentation". To the right of the callout is a badge showing "Star 14,051".

The Bokeh Visualization Library

Documentation

Star 14,051

Bokeh at a Glance

ggplot2



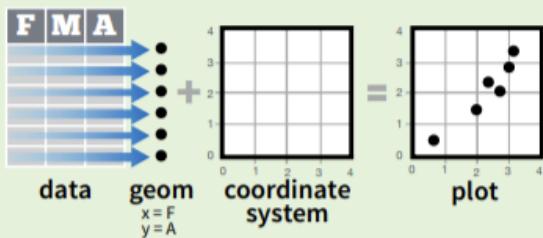
The *ggplot2* package is extremely popular for data visualization
(you've probably seen plots made in R with ggplot2 without knowing it!)



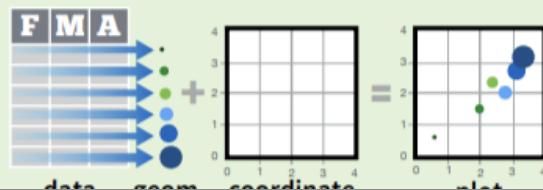
ggplot2 cheat sheet

Basics

ggplot2 is based on the **grammar of graphics**, the idea that you can build every graph from the same components: a **data** set, a **coordinate system**, and **geoms**—visual marks that represent data points.



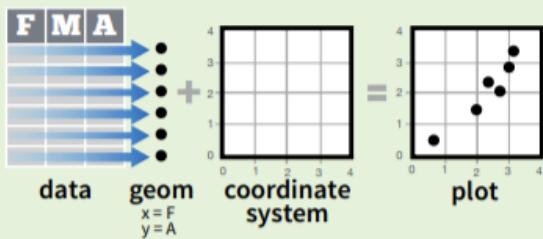
To display values, map variables in the data to visual properties of the geom (**aesthetics**) like **size**, **color**, and **x** and **y** locations.



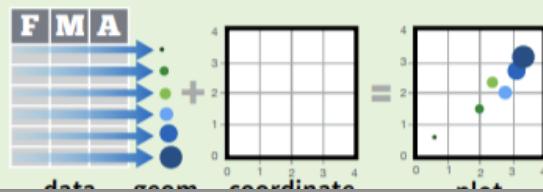
ggplot2 cheat sheet

Basics

ggplot2 is based on the **grammar of graphics**, the idea that you can build every graph from the same components: a **data** set, a **coordinate system**, and **geoms**—visual marks that represent data points.



To display values, map variables in the data to visual properties of the geom (**aesthetics**) like **size**, **color**, and **x** and **y** locations.



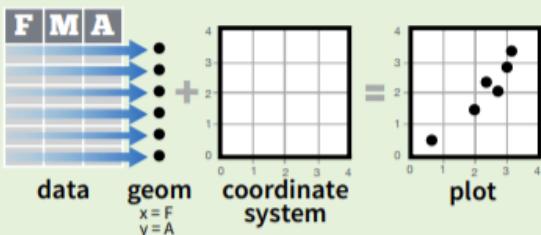
ggplot(data = mpg, aes(x = cty, y = hwy))

Begins a plot that you finish by adding layers to.
Add one geom function per layer.

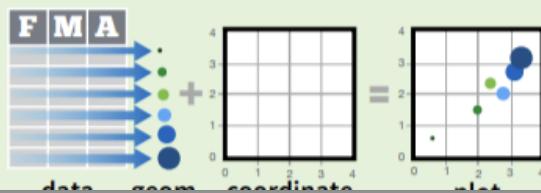
ggplot2 cheat sheet

Basics

ggplot2 is based on the **grammar of graphics**, the idea that you can build every graph from the same components: a **data** set, a **coordinate system**, and **geoms**—visual marks that represent data points.



To display values, map variables in the data to visual properties of the geom (**aesthetics**) like **size**, **color**, and **x** and **y** locations.

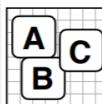


ggplot(data = mpg, aes(x = cty, y = hwy))

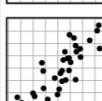
Begins a plot that you finish by adding layers to.
Add one geom function per layer.

Two Variables

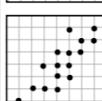
Continuous X, Continuous Y
`e <- ggplot(mpg, aes(cty, hwy))`



`e + geom_label(aes(label = cty), nudge_x = 1, nudge_y = 1, check_overlap = TRUE)`
x, y, label, alpha, angle, color, family, fontface, hjust, lineheight, size, vjust



`e + geom_jitter(height = 2, width = 2)`
x, y, alpha, color, fill, shape, size



`e + geom_point()`
x, y, alpha, color, fill, shape, size, stroke

Continuous Bivariate Distribution
`h <- ggplot(diamonds, aes(carat, price))`



`h + geom_hex(binwidth = c(0.25, 500))`
x, y, alpha, color, fill, linetype, size, weight



`h + geom_bin2d(binwidth = c(0.25, 500))`
x, y, alpha, colour, group, linetype, size

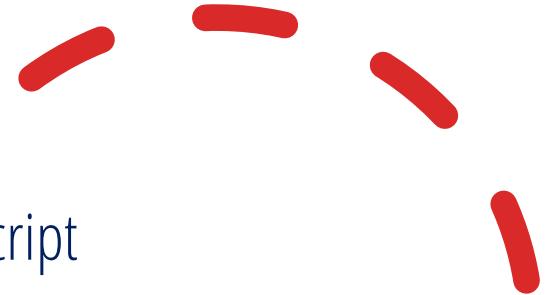


`h + geom_hex()`
x, y, alpha, colour, fill, size

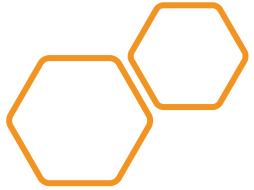
Exercise: Plot Data Multiple Ways

- Using the 'Summary_info.txt' from yesterday's exercise, make two plots of the data to show different things. For each plot, include *at least three* bit of information (e.g. ra, dec, z or, z, flux, SNR etc.)
- As you make the plots, develop *common styles* to make the data consistent between plots. Experiment with colour, gradients, symbols. What story are you trying to tell? Which pieces of information teach you something?

Optional Exercise: ggplot2 in R



- Go back to the week5_exercise.R script
- Breakdown and understand the code that is making the ggplot
- Make a new ggplot, using the SDSS quasar data set
 - Try setting the theme to something different!
 - Change the labels
 - Try making a multi-panel plot (see `cut_interval`)
- Make a ggplot using the spectra data set



Further Resources

- From data to visualization:
 - <https://www.data-to-viz.com/>
- The R Graph Gallery
 - <https://www.r-graph-gallery.com/>

