Data Visualization and R

Theory and Implementation
May 28, 2013, Köln, Germany

Ryan Womack (rwomack@rutgers.edu)
Data Librarian, Rutgers University

IASSIST 2013

Introduction

What this workshop IS:

- Focuses on standard techniques of data visualization, the day-to-day power tools for understanding data
- Reviews various graphical techniques, from early to recent, from simple to advanced
- Discusses principles of good data presentation, and show the R implementation of many functions

Introduction

What this workshop is NOT:

- It is not about "infographics", the beautiful, heavily customized products of expert graphic designers
- It is not a complete introduction to R, even though R is used
- It is not an introduction to other software beyond R, or the use of R with scripting languages to produce interactive graphics on the web
- It is not necessarily a balanced survey of all data visualization. In particular, it is light on graph networks, clustering, and trees (not my expertise)

Introduction

What you can hope to gain:

- Familiarity with the basic principles and history of data visualization, and recent developments
- Exposure to a wide-range of plotting techniques and R packages
- A sampling of interactive and big data methods
- Understanding of the power and potential of combining appropriate graphics techniques with data

Setup

- Workshop materials, including R scripts, supplemental images and data, are available for download from http://www.rci.rutgers.edu/~rwomack/IASSIST/Dataviz/
- The R script file contains working demonstrations of the concepts mentioned here.
- You will have to install any packages not already on your system.

Why Data Visualization?

Data visualization can:

- provide clear understanding of patterns in data
- detect hidden structures in data
- condense information

Some examples of good data visualization can be found at:

- Information Aesthetics
- Chart Porn
- Eagereyes
- DataVis.ca
- Visualizing.org
- VizWiz
- US Census Data Visualization Gallery

Bad Graphs

- Pie Charts are known to be problematic
- Clutter and other issues can ruin graphics

For more bad ideas, try:

- Junk Charts
- Ten Worst Graphs

R basics

- install.packages, read.table, read.csv, library(foreign), library(xlsx)
- help via help.start(), ?, library(help="package"), vignette
- Cookbook for R and Quick-R are good jumping off points
- Guerilla Guide to R or R cheat sheets or Rutgers R Libguide
- search Stack Overflow for R graphics to get specific coding tips on graphics
- R Graph Gallery has many examples of graphs with code, several adapted for this workshop
- Task Views or Crantastic can be use to discover pacakges. In addition to help and online documentation, packages are often described in articles published in places like the *Journal of Statistical Software*.
- This workshop will allow you to use the R language, but glosses over many details of structure and syntax to focus on the graphical elements

Playfair

- Astronomical observations, charts, and maps led in graphical innovation prior to 1800.
- William Playfair is the pioneer of the line chart, bar chart, time series plots, and pie chart.
- Playfair, W. (1786). *Commercial and Political Atlas: Representing, by Copper-Plate Charts, the Progress of the Commerce, Revenues, Expenditure, and Debts of England, during the Whole of the Eighteenth Century*,
- Playfair, W. (1801). *Statistical Breviary*.
- Both republished in *The Commercial and Political Atlas and Statistical Breviary*, 2005, Cambridge University Press.

Minard

Charles Joseph Minard was the next influential data graphic creator after Playfair.

- Minard's flow map of Napoleon's Russian campaign is celebrated by Tufte and others as one of the greatest information graphics.
- It embodies an ideal of highly compressed informative elements, presented with style
- Six variables: size, location in 2 dimensions, the direction of the army, temperature, date [and group]
- However, this is a one-off design that crosses into Infographics, but it can be reproduced in R and other software.

Fisher and Tukey

- Statisticians such as Ronald Fisher and John Tukey continued to advance graphical methods for the analysis of data.
- Fisher emphasized plotting the data to understand relationships.
- Tukey's *Exploratory Data Analysis* emphasized the use of graphics to understand the data during analysis, rather than the final presentation to an outside audience.
- Tukey created the stem and leaf plot.

Tufte

Edward R. Tufte's series of books, beginning with *The Visual Display of Quantitative Information*, have become the most widely know works on data visualization.

- There is considerable overlap between the various publications
- Tufte's ideal is highly compressed, elegant, and informative data, as expressed in dense printed graphics
- Tufte sometimes emphasizes beauty and design to the detriment of simplicity and clarity [e.g., train schedules]
- "Graphical elegance is often found in simplicity of design and complexity of data."
- "Beautiful graphics do not traffic with the trivial."

Tufte's principles

Tufte has developed and popularized numerous principles and terminology:

- **Graphics reveal data** - show the data without distorting it - "above all else show the data"
- **Small multiple** - understanding one slice makes understanding others easier
- **Lie factor** - effect shown/effect in reality
- **Graphical Integrity** - no lies, let data vary, not design
- **Data density** - maximize data/ink ratio
- **Sparklines** - seems they haven't caught on
- **chartjunk** - self-explanatory
- **Powerpoint** is responsible for most of the world's sorrows [*The Cognitive Style of Powerpoint*]

Back to Pies

Why is the pie chart bad?

- Low data density
- Failure to order numbers along a visual dimension
- Perception difficulty in judging area
    - Stacked bar charts also pose perceptual problems

Or in the terms of Gary Klass:

- Data Ambiguity
- Data Distortion
- Data Distraction

Cleveland

- William Cleveland's *Elements of Graphing Data* and *Visualizing Data* pioneered systematic considerations of data legibility
- Cleveland is particularly known for promoting the *dot plot* as a alternative to bars and pies.
- The dot plot provides clarity and easy comparison of data.
- Cleveland also pioneered Trellis graphics
- Trellis graphics emphasizes comparison of multiple panels of data
- The `lattice` package implements Trellis graphics in R
- See Cleveland.pdf for a summary of Cleveland's recommendations

Cleveland - Techniques

TECHNIQUES

- logs, % change, residuals
- point graph [2d histogram], histogram, percentile graph [and with comparisons/reference line], box plot [Tukey]
- dot charts - best way to attach label to quantity 2-way dot chart {multiway} grouped dot chart
- overlap is dealt with by jitter, distinguishable symbols {+sunflowers}, taking log or other transformation
- box plots for high multiples
- visually distinguish curves and points [this has gotten easy by now]

Cleveland - 3 or more variables

THREE OR MORE VARIABLES

- Framed-rectangle graphs
- scatterplot matrices
- interaction/brushing
- 3d wireframe or stereogram (points)

Cleveland - Perception

PERCEPTION

- pie v. dot chart
- distance and detection
- length in a stacked bar
- 45 degree banking [Tufte also recommends 1.5:1 horizontal to vertical ratio]
- strive for clarity

LATTICE

- The `lattice` package implements Trellis graphics in R
- `lattice` excels at comparative plotting
- uses similar syntax to base graphics, but with greater sorting and manipulative power

The Grammar of Graphics

The Grammar of Graphics, by Leland Wilkinson, was extremely influential in thinking about graphics

- Grammar means ”rules for art and science”
- The Grammar of graphics specifies rules both mathematical and aesthetic
- Earlier graph producers focused on aesthetics of static content
- Dynamic graphics and scientific visualization, by contrast, require sophisticated designs to enable brushing, drill-down, zooming, linking
- The Grammar of Graphics is easily adapted to this approach

Grammar of Graphics - pt I

- DATA - weighting, reshaping, counting, bootstrapping
- VARIABLES - transform, sort, log, ranking, residuals, quantiles
- ALGEBRA - nesting or blending data
- SCALES - nominal, ordinal, interval, ratio must be specified
- STATISTICS - static methods available to all graph types e.g, mean, sd, smoothing

Grammar of Graphics - pt II

- GEOMETRY - line, area, etc., along with modifiers like jitter and dodge
- COORDINATES - refers to the coordinate system of the graph (cartesian, polar, etc.)
- AESTHETICS - color, texture, size, position, etc. of the data points. Includes using color to classify.
- FACETS - subgroups, multiway tables
- GUIDES - legends, axes, color scales, keys

ggplot2

- `ggplot2` was developed by Hadley Wickham as an implementation of the Grammar of Graphics
- Relatively complete and powerful graphics package
- Can do many things, but not 3D
- See ggplot2 Help Docs and the ggplot2 book for complete descriptions
- Other short introductions to `ggplot2` are available, such as these from Sharp Statistics and inundata

A Miscellany of Visualizations

- The Cleveland dot chart
  - use to compare labeled quantities, ordered lists
- Kernel Density plot
  - visualize the distribution of data with more precision than a histogram
- Scatterplot Matrix
  - study relationships between all variable combinations

Visualizing Distributions of Data

- Box and Whiskers Plot
  - illustrate quantiles and outliers. There is also a Tufte version.
- Stem and Leaf Plot
  - see precise quantities associated with distribution of data
- Violin plot
  - Blends density information with box and whiskers style (in an artistic manner)
- Dot plot
  - plot distribution point-by-point

Categorizing Data

Many techniques are available to automatically identify related data:

- A *tree* illustrates a categorical classification of the data based on its own characteristics
  - one implementation is the `party` package
- *Self-organizing maps* are a form of neural network that derives characteristics from the data and plots patterns
  - see `kohonen` and `som` packages
- *Clustering* of data can be accomplished by numerous algorithms
- `hclust` and `pvclust` are some methods described at Quick-R's Cluster Analysis page.
- Other graphs and network analysis tools are available [not explored here]

Visualizing Categorical Data

- The *mosaic plot* allows multiple categories to be displayed on the same graph, but can be complicated to interpret.
- The *spineplot* is a variant of the mosaic plot, plotting proportions in 2 dimensions.
- You can also do a timeline.

Maps and Glyphs

Maps are obviously an important and widespread way of presenting data.

- We examine a few examples of *choropleth* maps, in which shading indicates data levels

Glyphs present iconic representations of data elements as plotted points.

- A dynamic example is here.
- As an R example, consider Chernoff faces and the `aplpack` package. Also, Smiley faces.

3-D

- 3-D scatterplots
  - `cloud (lattice)`
- contour plots
  - to plot standardized levels of data
- wireframe plots
  - to present a 3-D surface representation of data
- `rgl` (a separate package containing several 3d plotting functions and animation)

Interactive DataViz - Principles

- Why aren't all of our graphs interactive?
- *Brushing* is used to select data points and track them through various analyses.
- Drilling down, zooming, and subsetting are also interactive techniques.
- Data displays can be linked so that a selection in one panel modifies the output displayed in another panel.
- Interactivity is especially useful for data exploration, studying multidimensional relationships.

Linked Data Panels and Vis packages

In many contexts, visualizing the relationships between data elements is made easier by viewing related data panels simultaneously.

- One example of this occurs in time series data with decomposition into trend, seasonal, and random components
- The tableplot (`tabplot` package) implements another linked data view across all variables.
- googleVis and other "Vis" packages, e.g. healthvis.

Interactive Data in Practice

There are many R packages that allow for interactive data work in a graphical user interface, including:

- `playwith` - versatile package that works with any graphics function. Graphics can be explored, edited, and exported.
  - requires separate installation of GTK+ on your computer [method varies by OS]
- `rggobi` - powerful 3-D tool for brushing, identifying and manipulating data with a book and online companion site.
  - requires separate installation of GGobi on your computer [method varies by OS]
- `rattle` - package designed for data mining, includes graphics options alongside other statistical functions
- `latticist` - allows complex linking of plots

Big Data

- Big data presents special issues for data visualization
- While many techniques and graphics are the same, exploration and plotting must be optimized for the size of the data set
- Representation of the complexity of the data may require special techniques
- `hexbin`
- `bigvis`

Airline Data

For this exercise, we will use Airline on-time performance data.

- This data contains information on every flight in the United States from 1987 to 2008, including arrival and departure times, delays, and other attributes.
- The link above allows access to the full dataset. These are large. For example, the full 2008 data contains 7,009,728 records and is 689 MB.
- We will use an extract of selected variables from January 2008 only. This subset contains 605,765 records and is 37.5 MB.

Binning

- `hexbin` resolves overplotting issues in large data sets by showing density
- There are many other binning methods

BIGVIS

BIGVIS is a very new package by Hadley Wickham to deal with the issues of Big Data

- There is a Preprint and R Meetup presentation by Hadley Wickham
- Complete code, including the extracts adapted for this workshop, is available at https://github.com/hadley/bigvis-infovis
- Target: process 100 million observations in under 5 seconds.
- Fundamental principle: No need for more data points than there are pixels on the screen.

BIGVIS STEPS

- Condense (`bin`, `condense`)
- Smooth (`smooth`, `best_h`, `peel`)
- Visualize (`autoplot` plus standard methods)

Infographics links

Although not covered here, the following links are a sampling of infographics sites for your later enjoyment:

- Data Storytelling in Video
- Art of Data Visualization - in spite of its title, more on the infographics side
- Parisian Subway Traffic and New York Subway Inequality
- Tulp Interactive
- Information Aesthetics
- Mapping London and London Riots + Twitter
- YouTube Trends Map
- Global Burden of Disease Visualizations

Other considerations

These are not illustrated in the code, but represent future topics for exploration.

- confidence intervals
- missing data [discuss]
- color can be used to indicate certainty
- "scagnostics", borderlining scatterplots (`scagnostics` packages)
- edge blur, crisp vs. fuzzy edges

References

There is also an online bibliography of references to accompany this presentation on my home page.