

Lecture 3:

Data Cognition and Histograms

February 5, 2018

Overview

Course Administration

Good, Bad and Ugly

Few, Chapters 5 and 7

Charts in R

Course Administration

1. Collect proposals
2. Rosa has graded problem sets – thank you
3. And after this class, I'll figure out how to post grades
4. I have revised readings on readings tab – we were already behind!
5. Anything else?

Next Week's Good Bad and Ugly

Monday by 9 am. Earlier is ok.

- Kelsey Wilson
- Nathan Rupp
- Haley Dunn

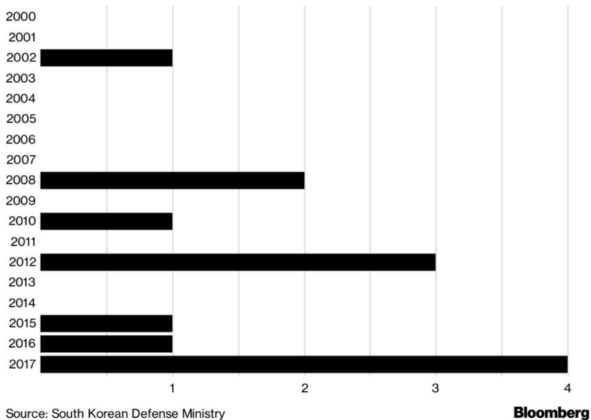
This Week's Good Bad and Ugly

- Meryl Howard
- Lilia Ledezma

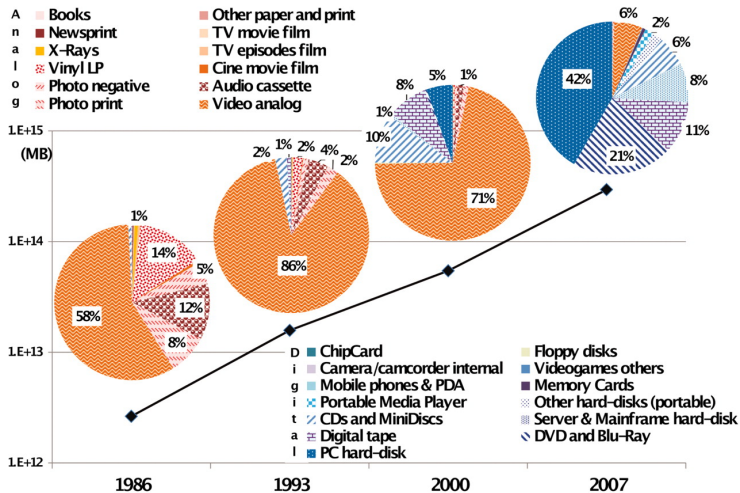
Meryl's Example

Runs Across the Border

Four North Korean soldiers have defected south this year -- the most since at least 2000



Lilia's Example



Non-R Portion of the Lecture

- *Big Numbers*, from the WSJ
- Few, Chapters 5 and 7

Big Numbers

- What is the problem with people and big numbers?
- What's one suggested strategy to help?

This seems to be one of the broad challenges of conveying numbers

Few:

Visual Perception and Graphical Communication

Tufte's Six Rules of Graphic Integrity, 1 to 3 of 6

1. The representation of numbers, as physically measured on the surface of the graphic itself, should be directly proportional to the numerical quantities represented.
2. Clear, detailed, and thorough labeling should be used to defeat graphical distortion and ambiguity. Write out explanations of the data on the graphic itself. Label important events in the data.
3. Show data variation, not design variation.

Tufte's Six Rules of Graphic Integrity, 4 to 6

4. In time-series displays of money, deflated and standardized units of monetary measurement are nearly always better than nominal units.
5. The number of information-carrying (variable) dimensions depicted should not exceed the number of dimensions in the data.
6. Graphics must not quote data out of context.

From Few

1. working memory
2. preattentive processing
3. applying to design
4. gestalt principles of visual perception

1. Working Memory

We don't have much of it

- people can remember 3 to 4 visual encodings for a chart
- therefore, more than about 4 colors as identification are distracting
- good visuals can stick in long-term memory

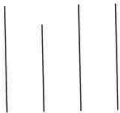
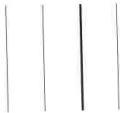
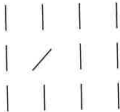
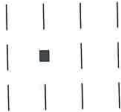

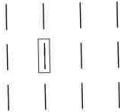
2. Preattentive Processing

- Form
- Color
- Spatial Position

987349790275647902894728624092406037070570279072
803208029007302501270237008374082078720272007083
247802602703793775709707377970667462097094702780
927979709723097230979592750927279798734972608027

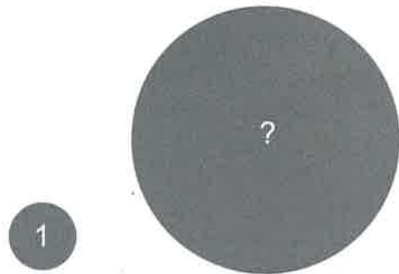
987349790275647902894728624092406037070570279072
803208029007302501270237008374082078720272007083
247802602703793775709707377970667462097094702780
927979709723097230979592750927279798734972608027

Form

Length	Width
	
Orientation	Shape
	
Size	Enclosure
	

But Beware of 2-D Size

- People have a very hard time judging the relative size of 2-D objects
- Changing both length and width is a 2-D change
- Avoid unless you have a specific reason to do this – maybe you're drawing building sizes



Color

1. Hue

- What you think of as “color”
- Blue, Green, etc

2. Intensity

- make it less intense: add a little gray

Color

1. Hue

- What you think of as “color”
- Blue, Green, etc

2. Intensity

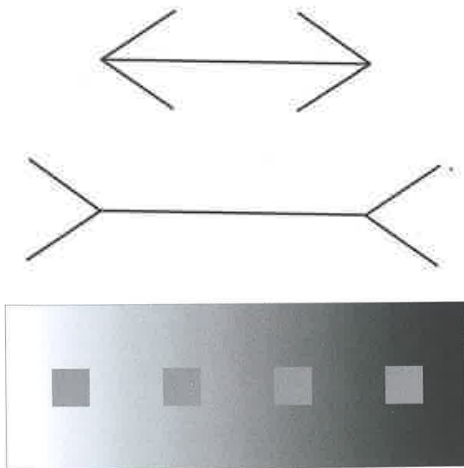
- make it less intense: add a little gray

Contrasting hues stand out. Intense colors stand out.

3. How Do We Perceive Them?

Type	Attribute	Quantitatively Perceived?
Form	Length	Yes
	Width	Yes, but limited
	Orientation	No
	Size	Yes, but limited
	Shape	No
	Enclosure	No
Color	Hue	No
	Intensity	Yes, but limited
Position	2-D Position	Yes

Context Matters



4. Gestalt Principles of Visual Perception

- Proximity
- Similarity
- Enclosure
- Closure
- Continuity

These all generate meaning, whether you intend it or not!

Using this Knowledge to Communicate

1. Highlight
2. Organize

Highlight

- Reduce non-data ink
 - Subtract unnecessary non-data ink
 - De-emphasize and regularize non-data ink
- Enhance data ink
 - Subtract unnecessary data ink
 - Emphasize the remaining data ink

Organize

- Group
- Prioritize
- Sequence

Ways to Call Attention

Attribute	Graphs and Objects
2-D Position	Top, left or center
Shape	Any symbol different from the norm
Enclosure	Border or shading

R

Today's Goals

- A few non-graph commands
 - `summary`
 - `ddply`
- Make histograms
 - Introduce `ggplot`
 - Work on histogram variations
 - We ignore the easier `qplot`
 - On your own: make some histograms of your own

An Easy Start: Summary

- You calculate a variable, call it `counties$shr.mort`
- You'd like to get a sense of its variation
`summary(counties$shr.mort)`
- This reports

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
##	0.0000	0.6193	0.7477	0.7222	0.8551	1.0000	289

A More Challenging But Powerful Command: `ddply`

- Many times in making graphics, you need to modify the original data
- Summarizing data is one way to create meaning
- To go from one unit of analysis – we've used counties and block groups – to another, you need to summarize
 - counties → 9 Census divisions
 - block groups → counties
 - counties by year → year

Make a small dataset

- Make an example dataset

```
set.seed(1)
d <- data.frame(year = rep(2000:2002, each = 3),
  count = round(runif(9, 0, 20)))
print(d)
```

##	year	count
## 1	2000	5
## 2	2000	7
## 3	2000	11
## 4	2001	18
## 5	2001	4
## 6	2001	18
## 7	2002	19
## 8	2002	13
## 9	2002	13

Two Things To Do

1. Summarize

- ▶ calculate some kind of statistic by group
- ▶ output a dataset at the level of the new group

2. Transform

- ▶ calculate some kind of statistic by group
- ▶ keep the dataset at the original unit of analysis
- ▶ statistics may repeat within group

Summarize

```
library(plyr)
ddply(d, "year", summarize, mean.count = mean(count))
```

```
##   year mean.count
## 1 2000    7.666667
## 2 2001   13.333333
## 3 2002   15.000000
```

Transform

```
ddply(d, "year", transform, mean.count = mean(count))
```

##	year	count	mean.count
## 1	2000	5	7.666667
## 2	2000	7	7.666667
## 3	2000	11	7.666667
## 4	2001	18	13.333333
## 5	2001	4	13.333333
## 6	2001	18	13.333333
## 7	2002	19	15.000000
## 8	2002	13	15.000000
## 9	2002	13	15.000000

Or, a Different Transform

```
ddply(d, "year", transform, total.count = sum(count))
```

##	year	count	total.count
## 1	2000	5	23
## 2	2000	7	23
## 3	2000	11	23
## 4	2001	18	40
## 5	2001	4	40
## 6	2001	18	40
## 7	2002	19	45
## 8	2002	13	45
## 9	2002	13	45

And on to ggplot

The Heart of the Course

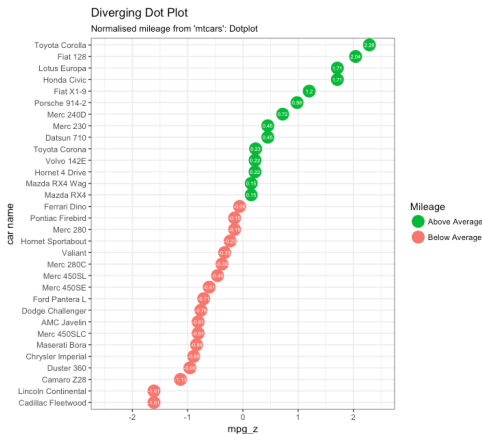
- Major publications use ggplot to make graphs
- After we do some, you'll probably notice it
- There is also a way to publish from R straight to the web, which we won't learn in this course (RShiny)

Great examples for ggplot [here](#).

What is ggplot?

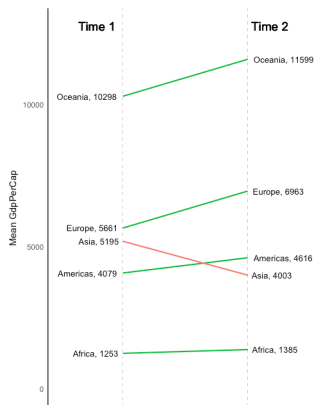
- A graphics language for producing nice and flexible graphics
- Developed by Hadley Wickham, R God
- Has a simpler `qplot` version that we will bypass

Examples of Familiar Graphics



Examples from <http://r-statistics.co/Top50-Ggplot2-Visualizations-MasterList-R-Code.html>

Examples of Familiar Graphics



Examples from <http://r-statistics.co/Top50-Ggplot2-Visualizations-MasterList-R-Code.html>

For today, a ggplot starter

Command looks like

```
ggplot(data, mapping = aes(x=x, y=y)) + geom commands
```

- first term tells R dataframe for chart
- second term is the “aesthetics”
- term tells R what the variables are

Add the graph

- You need a “geom” command to add pics
- “geoms define the basic ‘shape’ of the elements on the plot” (Wickham, 201x)
- the shape can be
 - point
 - line
 - polygon
 - bar
 - text
- Better by example → today’s tutorial

Try Today's Tutorial

- Works up to a reasonable graph
- Pay attention to the output of each bit
- Consider writing a .R script and then moving to .Rmd
- Go forth!

Next Lecture

- Turn in PS 3 and one-page proposal
- Read Few Chapter 6
- R Graphics Cookbook, Chapter 3