Properties of Data

Digging into Data

University of Maryland

February 11, 2013

ggplot2 material adapted from Karthik Ram
Roadmap

- Getting and cleaning data
  - Unavoidable step
  - Example of how I do it
- Goal
  - Not to teach you how
  - What end results you need to tell stories from data
  - Telling those stories with pictures
  - Same thing necessary for making predictions and clustering
  - Homework 1
- `ggplot2`
- `CaBi`
Outline

1. Data Terminology
2. Testbed: Capital Bikeshare
3. Visualizing and Summarizing Data in Rattle
4. ggplot2
5. ggplot2 with "real" data
6. Wrapup
(Confusing) Terminology

- A dataset has different components
- Input: what you always know
  - Sometimes called independent variable
  - Sometimes called regressor
  - Sometimes called feature
- Output: what you’re trying to learn
  - Sometimes called independent variable
  - Sometimes called the regressand
  - Sometimes called the response variable
  - Sometimes called the “label”
A dataset has different components

Input: what you always know
- Sometimes called independent variable
- Sometimes called regressor
- Sometimes called feature

Output: what you’re trying to learn
- Sometimes called independent variable
- Sometimes called the regressand
- Sometimes called the response variable
- Sometimes called the “label”
- Does not exist for **unsupervised** learning
Terminology

- But not all data are usable
- Most data also have an **identifier**
- Could also be metadata
  - When data was collected
  - Who collected it
  - How much it cost
- Often important to exclude such data from your algorithms
## Terminology

<table>
<thead>
<tr>
<th><strong>Discrete Data</strong></th>
<th><strong>Continuous Data</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Also called categoric</td>
<td>Also called numeric</td>
</tr>
<tr>
<td>Bins that you group data into</td>
<td>Numeric values that represent data</td>
</tr>
<tr>
<td>There is no “in between”</td>
<td>There is an “in between”</td>
</tr>
<tr>
<td>You can ask most frequent value</td>
<td>You can take the average</td>
</tr>
<tr>
<td></td>
<td>It makes sense to ask questions like what if this were 10% more $X$</td>
</tr>
</tbody>
</table>
Outline

1. Data Terminology

2. Testbed: Capital Bikeshare

3. Visualizing and Summarizing Data in Rattle

4. ggplot2

5. ggplot2 with "real" data

6. Wrapup
Capital Bikeshare

- Until this year, largest bikeshare system in US
- Publicly share data
- Important problems:
  - Where should new stations be?
  - Rebalancing
  - Pricing
  - Coordinating with other transit
CSV File

http://www.capitalbikeshare.com/trip-history-data

```
Duration,Start date,Start Station,End date,End Station,Bike#,Subscription Type
0h 7m 28s,12/31/2012 23:58,Eastern Market Metro / Pennsylvania Ave & 7th St SE,1/1/2013 0:05,
0h 6m 24s,12/31/2012 23:56,14th & V St NW,1/1/2013 0:02,Massachusetts Ave & Dupont Circle NW,
0h 6m 58s,12/31/2012 23:56,14th & V St NW,1/1/2013 0:03,Massachusetts Ave & Dupont Circle NW,
2h 23m 50s,12/31/2012 23:51,Lincoln Park / 13th & East Capitol St NE ,1/1/2013 2:15,Lincoln P
,W00704,Casual
```
What story do you want to tell?

- What data are there?
- What information do you want?
- How to get from point A to point B?
What story do you want to tell?

- What data are there?
- What information do you want?
- How to get from point A to point B?
  - More art than science
  - No right answers
Adding it to Google Docs

Import into Google Spreadsheet
Adding it to Google Docs

Loads nicely into columns

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Duration</td>
<td>Start date</td>
<td>Start Station</td>
<td>End date</td>
<td>End Station</td>
</tr>
<tr>
<td></td>
<td>2m 7s</td>
<td>12/31/2013 23:59:00</td>
<td>Eastern Market Metro / Pennsylvania Ave &amp; 7th St. SE</td>
<td>1/1/2013 0:05:00</td>
<td>14th St SE</td>
</tr>
<tr>
<td>2</td>
<td>5m 6s</td>
<td>12/31/2013 23:59:00</td>
<td>14th &amp; V St NW</td>
<td>1/1/2013 0:02:00</td>
<td>14th &amp; V St NW</td>
</tr>
<tr>
<td>3</td>
<td>5m 6s</td>
<td>12/31/2013 23:59:00</td>
<td>14th &amp; V St NW</td>
<td>1/1/2013 0:03:00</td>
<td>14th &amp; V St NW</td>
</tr>
</tbody>
</table>
Adding it to Google Docs

It would be nice to have more

- Real world locations
- Elevation
- CaBi has some of this information
- Google (Maps) knows the rest . . .
Adding it to Google Docs

http://www.capitalbikeshare.com/data/stations/bikeStations.xml
Adding it to Google Docs

Creating a new sheet just for stations
Adding it to Google Docs

Load columns from the xml file

We now have columns for lat, long for every station
Adding it to Google Docs

Now we can attach a location to each row in the original sheet

<table>
<thead>
<tr>
<th>Duration</th>
<th>Start date</th>
<th>Start Station</th>
<th>End date</th>
<th>End Station</th>
<th>Bike#</th>
<th>Subscription Type</th>
<th>LatStart</th>
<th>LongStart</th>
</tr>
</thead>
<tbody>
<tr>
<td>0h 7m 28s</td>
<td>12/31/2013 23:58:00 SE</td>
<td>Eastern Market Metro / Pennsylvania Ave &amp; 17th St</td>
<td>1/1/2013 00:05:00</td>
<td>14th &amp; D St SE</td>
<td>W01301</td>
<td>Subscribe</td>
<td>38.884</td>
<td></td>
</tr>
</tbody>
</table>
Adding it to Google Docs

Now we’ve added neat new columns to the spreadsheet; time to download
Loading a dataset

```r
rides <- read.csv("data/cabi-rides.ext.csv")
```

- Creates a “data frame”
- This is the basic unit of R data (Rattle creates these automatically for you)
- Very easy to add columns
- Use the $ to access columns
Outline

1. Data Terminology
2. Testbed: Capital Bikeshare
3. Visualizing and Summarizing Data in Rattle
4. ggplot2
5. ggplot2 with "real" data
6. Wrapup
Summarizing Data

Getting Output Directly

- “Explore” tab
- Click: “summary”

<table>
<thead>
<tr>
<th>duration</th>
<th>startStation</th>
<th>count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min.</td>
<td>0.0000 Massachusetts Ave &amp; Dupont Circle NW</td>
<td>116</td>
</tr>
<tr>
<td>1st Qu.</td>
<td>0.1000 15th &amp; P St NW</td>
<td>97</td>
</tr>
<tr>
<td>Median</td>
<td>0.1667  Columbus Circle / Union Station</td>
<td>94</td>
</tr>
<tr>
<td>Mean</td>
<td>0.2418  Thomas Circle</td>
<td>79</td>
</tr>
<tr>
<td>3rd Qu.</td>
<td>0.2667  Eastern Market Metro / Pennsylvania Ave &amp; 7th St SE</td>
<td>74</td>
</tr>
<tr>
<td>Max.</td>
<td>13.5667 17th &amp; Corcoran St NW</td>
<td>70</td>
</tr>
<tr>
<td>NA’s</td>
<td>2.0000 (Other)</td>
<td>3629</td>
</tr>
</tbody>
</table>
### Getting Output Directly

- **“Explore”** tab
- **Type:** “summary”

<table>
<thead>
<tr>
<th>endStation</th>
<th>distance</th>
<th>startHour</th>
</tr>
</thead>
<tbody>
<tr>
<td>Massachusetts Ave &amp; Dupont Circle NW</td>
<td>148</td>
<td>0.0</td>
</tr>
<tr>
<td>15th &amp; P St NW</td>
<td>103</td>
<td>0.1333</td>
</tr>
<tr>
<td>Thomas Circle</td>
<td>94</td>
<td>921.5</td>
</tr>
<tr>
<td>17th &amp; Corcoran St NW</td>
<td>86</td>
<td>1515.5</td>
</tr>
<tr>
<td>Columbus Circle / Union Station</td>
<td>82</td>
<td>2402.2</td>
</tr>
<tr>
<td>North Capitol St &amp; F St NW</td>
<td>74</td>
<td>13166.5</td>
</tr>
<tr>
<td>(Other)</td>
<td>3572</td>
<td>NA’s 1.0000</td>
</tr>
</tbody>
</table>

**Note:** Median and mean distances are provided for comparison.
Descriptive Statistics: Quartiles

- Order your data
- Find the middle data point - this is your median
  - If even number of data points, average points in the middle
- Repeat on two halves on either side of median - these are your first and third quartiles
Descriptive Statistics

- min - smallest data point
- max - largest data point
- mean - sum of all data divided by number of data points

\[
\mu = \frac{\sum x_i}{N}
\]

Properties of Data
February 11, 2013 17 / 53
Descriptive Statistics

- min - smallest data point
- max - largest data point
- mean - sum of all data divided by number of data points

\[ \mu = \frac{\sum x_i}{N} \]  

\[ \text{median of all data, second quartile} \]
\[ \text{median of lower part, first quartile} \]
\[ \text{median of upper part, third quartile} \]

65, 65, 70, 75, 80, 80, 85, 90, 95, 100
What to look for . . .

- Are the min / max reasonable?
- Is there a lot of missing data (NA)?
- Do the most frequent levels for categorical data make sense?
Box Plots

- Show median, mean, Q1, Q2, max and min
- Show if distributions are skewed
- Easier to see than reading off numbers
- Introduced by Tukey
- Under “Explore”, “Distributions”
Outline

1. Data Terminology
2. Testbed: Capital Bikeshare
3. Visualizing and Summarizing Data in Rattle
4. ggplot2
5. ggplot2 with "real" data
6. Wrapup
Some housekeeping

Install some packages (make sure you also have recent copies of reshape2 and plyr)

install.packages("ggplot2", dependencies = TRUE)
- Ugly, laborious, and verbose
- There are better ways to describe statistical visualizations.
Why ggplot2?

- Follows a grammar, just like any language.
- It defines basic components that make up a sentence. In this case, the grammar defines components in a plot.
- Grammar of graphics originally coined by Lee Wilkinson
Why ggplot2?

- Supports a continuum of expertise.
- Get started right away but with practice you can effortlessly build complex, publication quality figures.
- Common pitfall:
  - Never use `qplot` - short for quick plot.
  - You’ll end up unlearning and relearning a good bit.
Some terminology

- **ggplot** - The main function where you specify the dataset and variables to plot
- **geoms** - geometric objects
  - `geom_point()`, `geom_bar()`, `geom_density()`, `geom_line()`, `geom_area()`
- **aes** - aesthetics
  - shape, transparency (alpha), color, fill, linetype.
- **scales** Define how your data will be plotted
  - `continuous`, `discrete`, `log`
The iris dataset

```
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1     5.1         3.5         1.4       0.2     setosa
## 2     4.9         3.0         1.4       0.2     setosa
## 3     4.7         3.2         1.3       0.2     setosa
## 4     4.6         3.1         1.5       0.2     setosa
## 5     5.0         3.6         1.4       0.2     setosa
## 6     5.4         3.9         1.7       0.4     setosa
```
Let’s try an example

```r
ggplot(data = iris, aes(x = Sepal.Length, y = Sepal.Width)) + geom_point()
```
Basic structure

```r
ggplot(data = iris, aes(x = Sepal.Length, y = Sepal.Width))
  + geom_point()
myplot <- ggplot(data = iris, aes(x = Sepal.Length, y = Sepal.Width))
myplot + geom_point()
```

- Specify the data and variables inside the `ggplot` function.
- Anything else that goes in here becomes a global setting.
- Then add layers of geometric objects, statistical models, and panels.
Scatter Plots: Increase the size of points

```
ggplot(data = iris, aes(x = Sepal.Length, y = Sepal.Width)) +
geom_point(size = 3)
```
Scatter Plots: Add some color

```r
ggplot(iris, aes(Sepal.Length, Sepal.Width, color = Species)) + geom_point(size = 3)
```
Scatter Plots: Differentiate points by shape

```r
ggplot(iris, aes(Sepal.Length, Sepal.Width, color = Species)) + geom_point(aes(shape = Species), size = 3)
```
Boxplots

See ?geom_boxplot for list of options

```r
library(MASS)
ggplot(birthwt, aes(factor(race), bwt)) + geom_boxplot()
```
Histograms

See ?geom_histogram for list of options

```r
h <- ggplot(faithful, aes(x = waiting))
h + geom_histogram(binwidth = 30, colour = "black")
```
```r
h <- ggplot(faithful, aes(x = waiting))
h + geom_histogram(binwidth = 8, fill = "steelblue", colour = "black")
```
climate <- read.csv("climate.csv", header = T)

ggplot(climate, aes(Year, Anomaly10y)) + geom_line()
ggplot(climate, aes(Year, Anomaly10y)) +
geom_ribbon(aes(ymin = Anomaly10y - Unc10y, ymax = Anomaly10y + Unc10y),
fill = "blue", alpha = .1) +
geom_line(color = "steelblue")
Bar Plots

ggplot(iris, aes(Species, Sepal.Length)) + geom_bar(stat = "identity")
Density vs. Line Plots

```r
ggplot(faithful, aes(waiting)) +
geom_density(fill = "blue", alpha = 0.1)
```
ggplot(faithful, aes(waiting)) + geom_line(stat = "density")
• Raster graphics (bmp, jpeg, png) don’t scale well
• Preparing graphics for publication requires vector graphics (pdf, eps)
• Much easier to provide publication-quality images with ggplot2
Saving Plots

- If the plot is on your screen
  
  \texttt{ggsave("~/path/to/figure/filename.png")}

- If your plot is assigned to an object
  
  \texttt{ggsave(plot1, file = "~/path/to/figure/filename.png")}

- Specify a size
  
  \texttt{ggsave(file = "/path/to/figure/filename.png", width = 6, height =4)}

- or any format (pdf, png, eps, svg, jpg)
  
  \texttt{ggsave(file = "/path/to/figure/filename.eps")}
  \texttt{ggsave(file = "/path/to/figure/filename.jpg")}
  \texttt{ggsave(file = "/path/to/figure/filename.pdf")}
Outline

1. Data Terminology
2. Testbed: Capital Bikeshare
3. Visualizing and Summarizing Data in Rattle
4. ggplot2
5. ggplot2 with "real" data
6. Wrapup
ggplot2 maps

Get an outline of DC

all_states <- map_data("state")
states <- subset(all_states, region %in%
    c( "district of columbia" ) )

Draw it

p <- ggplot(stations)
p <- p + geom_polygon( data=states, aes(x=long, y=lat))
ggplot2 maps
ggplot2 maps

```r
p <- p + geom_point( data=stations,
                     aes(x=long, y=lat, size = count),
                     color="gold2") +
    scale_size(name="Bikes")
```

![Map of bike stations with size and color scale for bike count](chart.png)
ggplot2 facets

```r
p <- p + facet_grid(type ~ time)
```

Digging into Data (UMD)  Properties of Data  February 11, 2013  46 / 53
stations$time <- factor(stations$time, levels =
c("EARLYMORN","LATEMORN","AFTERNOON", 
"EVENING", "NIGHT", "LATENIGHT"))
```r
p <- ggplot(rides)
p <- p + geom_smooth(aes(x=startHour, y=distance))
p <- p + coord_cartesian(ylim=c(1000,2500))
```
ggplot2 histograms

```r
p <- ggplot(rides)
p <- p + geom_histogram(aes(x=duration), binwidth = .1)
p <- p + scale_y_sqrt()
p <- p + facet_grid(subscription ~ .)
p <- p + scale_x_continuous(limits=c(0, 4))
```

![Histograms showing the distribution of ride durations for casual and subscriber categories.](image)
Outline

1. Data Terminology
2. Testbed: Capital Bikeshare
3. Visualizing and Summarizing Data in Rattle
4. ggplot2
5. ggplot2 with "real" data
6. Wrapup
We’ve done a lot

- You don’t have to be able to do everything we did today
- You have to be able to do some of it
- Play around with the way of manipulating data you feel most comfortable with
Further help

- You’ve just scratched the surface with ggplot2.
- Practice
- Read the docs (either locally in R or at http://docs.ggplot2.org/current/)
- Work together
First assignment

- Find some data
- Edit it so it is in a usable form
- Find interesting relationships in your data
- Use Rattle/ggplot2 to display those relationships (be creative and thorough!)