

Tool Tutorials

Dr. Michele C. Weigle

<http://www.cs.odu.edu/~mweigle/CS795-F12/>

Outline

- ▶ Excel
- ▶ R
- ▶ Inkscape
- ▶ Tableau
- ▶ Processing.js
- ▶ Protovis
- ▶ D3.js
- ▶ jQuery Sparklines
- ▶ Resources
 - ▶ Lots of online references (URIs are included and posted in Links webpage)

Goal

<http://www.cs.odu.edu/~mweigle/cs795/barchart.html>

- ▶ Take a single dataset and show how to visualize it in each of our tools
- ▶ Just a preview of each tool
 - ▶ in-depth learning is up to you
- ▶ All of the web-based tools allow you to code sophisticated interaction. The examples today are all static.
- ▶ Note: Don't get tied to a particular tool -- use the one that's appropriate for the task at hand

▶ 3

CS 795/895 - Fall 2012 - Weigle

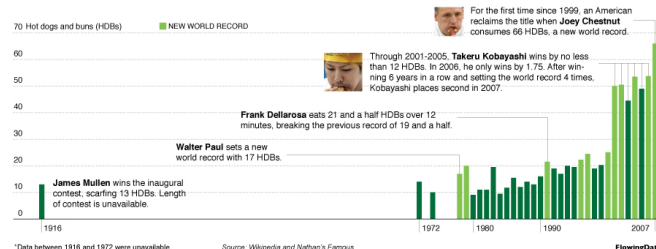
Data Set

<http://datasets.flowingdata.com/hot-dog-contest-winners.csv>

- ▶ From *Visualize This*, Ch 4
- ▶ Results from the past 3 decades of Nathan's Hot Dog Eating Contest
 - ▶ Takeru Kobayashi
- ▶ Source: Wikipedia
- ▶ Columns:
 - ▶ year
 - ▶ winner
 - ▶ hot dogs eaten
 - ▶ country
 - ▶ new record (0 or 1)

Winners from Nathan's Hot Dog Eating Contest

Since 1916, the annual eating competition has grown substantially attracting competitors from around the world. This year's competition will be televised on July 4, 2008 at 12pm EDT live on ESPN.



<http://flowingdata.com/2009/07/02/whos-going-to-win-nathans-hot-dog-eating-contest/>

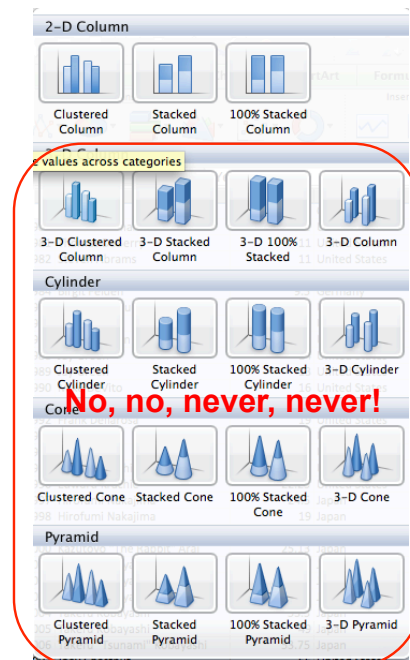
Actually going for Fig 4-5 in *Visualize This*

▶ 4

CS 795/895 - Fall 2012 - Weigle

Excel

- ▶ Start with a bar chart
 - ▶ plain, 2D bars
 - ▶ no 3D!
 - ▶ no cylinders, no cones, no pyramids!



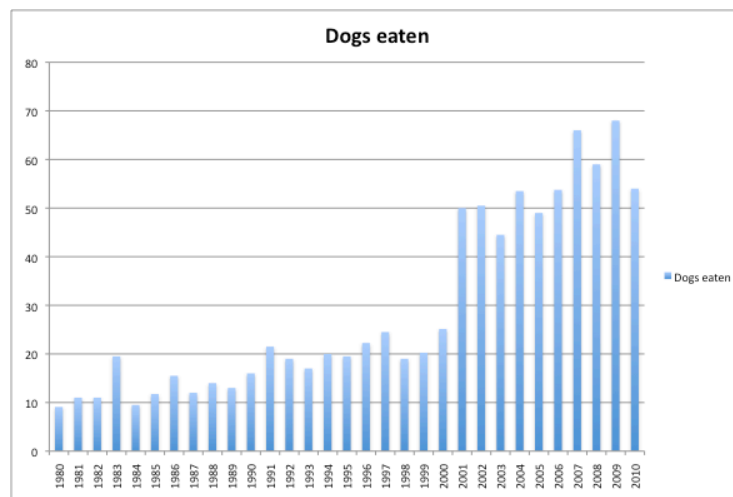
▶ 5

CS 795/895 - Fall 2012 - Weigle

Excel

Default Bar Chart

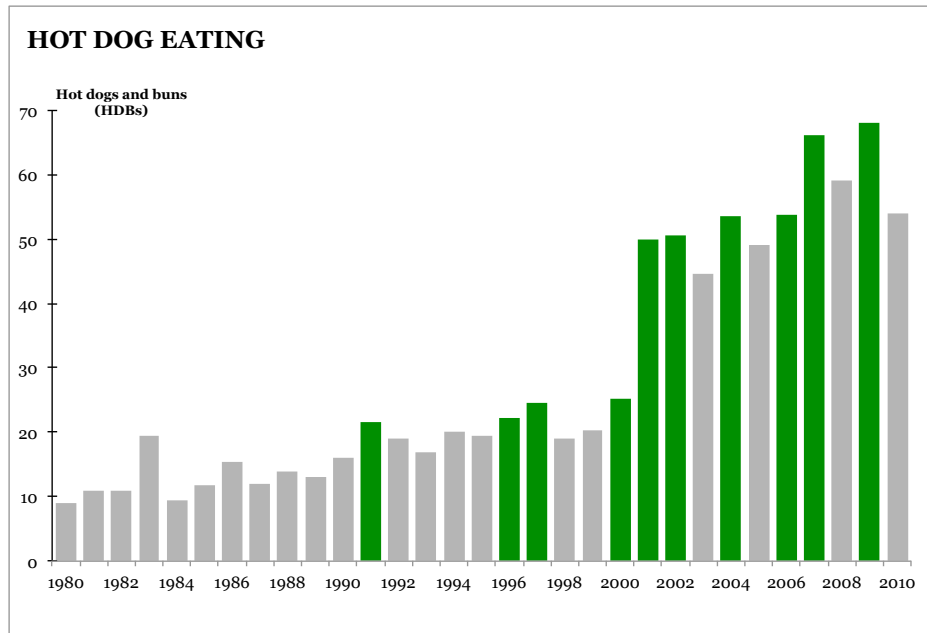
- ▶ Once correct data columns are picked for the axes
- ▶ Problems?



▶ 6

CS 795/895 - Fall 2012 - Weigle

Excel



► 7

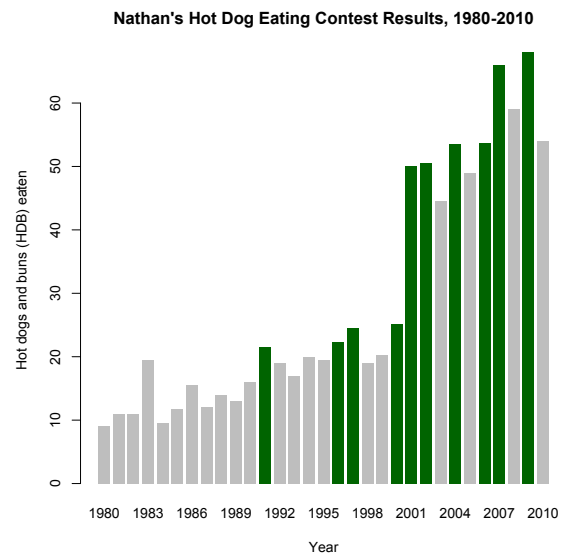
CS 795/895 - Fall 2012 - Weigle

R

```
fill_colors <- c()
for (i in 1:length(hotdogs$New.record)) {
  if (hotdogs$New.record[i] == 1) {
    fill_colors <- c(fill_colors, "dark green")
  } else {
    fill_colors <- c(fill_colors, "gray")
  }
}

barplot(hotdogs$Dogs.eaten, names.arg=hotdogs
$Year, col=fill_colors, border=NA, space=0.3,
xlab="Year", ylab="Hot dogs and buns (HDB)
eaten")

title("Nathan's Hot Dog Eating Contest Results,
1980-2010")
```



<http://www.cs.odu.edu/~mweigle/cs795/barchart-r.html>

► 8

CS 795/895 - Fall 2012 - Weigle

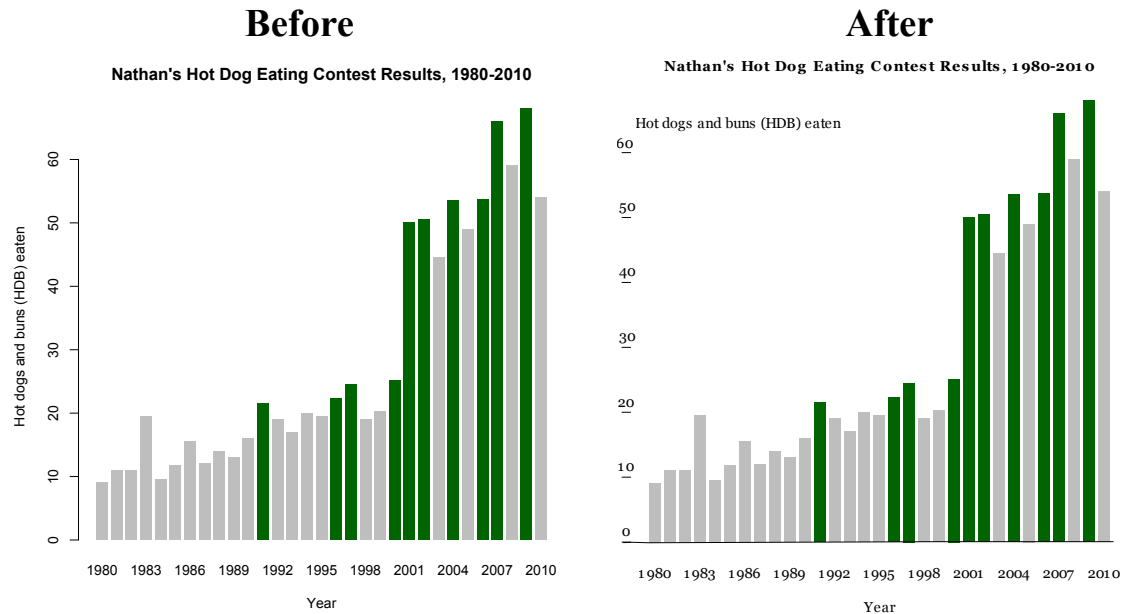
Let's Clean This Up in Inkscape

- ▶ Save as PDF, open PDF in Inkscape
- ▶ Change font to Georgia
 - ▶ use “edit paths by nodes” selector (aka Direct Selection tool) to choose text boxes
 - ▶ click on T in right side box
- ▶ Rotate number labels on y-axis
 - ▶ select all of the numbers, choose Object > Transform... from top menu
 - ▶ choose Rotate tab and enter -90 for the Angle
 - ▶ check “Apply to each object separately”
- ▶ Remove y-axis line
 - ▶ select the line, right-click, and choose Delete

Let's Clean This Up in Inkscape

- ▶ Move y-axis numbers
 - ▶ with all numbers selected, choose Move tab in Transform box
 - ▶ uncheck “Apply to each object separately”
 - ▶ uncheck “Relative move”
 - ▶ my settings – horizontal: 60 px, vertical: 102 px
- ▶ Move y-axis label
 - ▶ select the label
 - ▶ rotate -90 degrees
 - ▶ move to horizontal: 80 px, vertical: 525 px
- ▶ Add x-axis lines and tick marks
 - ▶ choose Bezier and line tool and draw x-axis line
 - ▶ double-click to end the line

Let's Clean This Up in Inkscape



▶ 11

CS 795/895 - Fall 2012 - Weigle

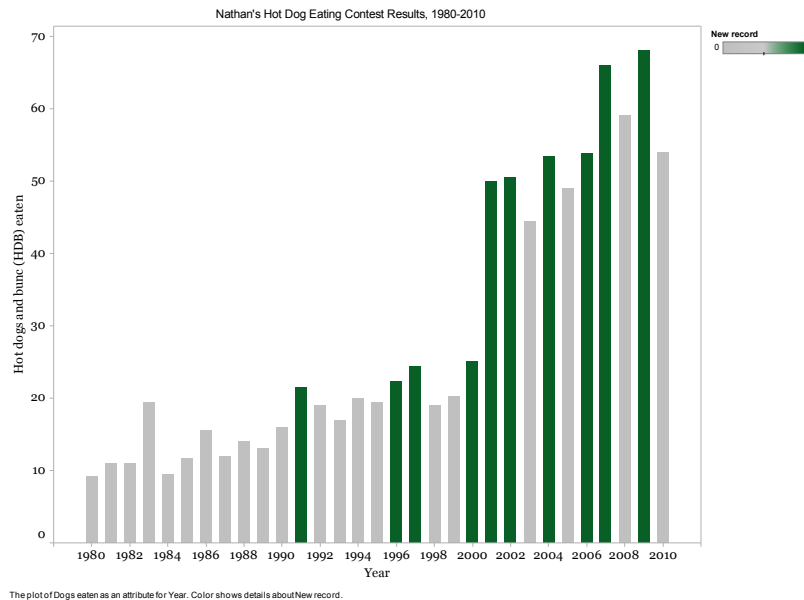
Tableau

- ▶ Load CSV
- ▶ Drag Year to Columns
 - ▶ choose Dimension
- ▶ Drag Dogs eaten to Rows
 - ▶ choose Attribute
- ▶ Choose Bar graph
- ▶ Drag New record to Color
 - ▶ edit colors to get gray and green
- ▶ Edit y-axis label
- ▶ Format fonts
- ▶ Edit y-axis tick marks

▶ 12

CS 795/895 - Fall 2012 - Weigle

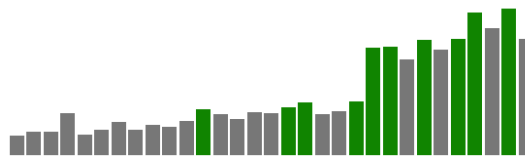
Tableau



<http://public.tableausoftware.com/views/HotDogExample/Sheet1?:embed=y>

Processing.js

- Processing is Java-based, so much of the code is Java-like
- Write a separate Processing sketch to create the visualization
- Processing.js just compiles the Processing code to Javascript
- Load the data
- Setup the canvas
- Draw the bars

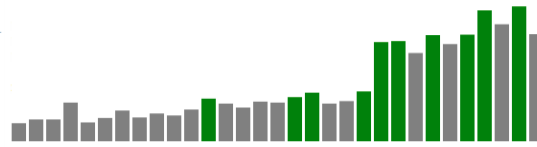


```
void drawDataBars() {  
  Data d;  
  float x, y;  
  noStroke();  
  rectMode(CORNERS);  
  
  for (int row = 0; row < rowCount; row++) {  
    d = (Data) data.get(row);  
    x=map(d.year, yearMin, yearMax, plotX1, plotX2);  
    y = map(d.dogs, 0, dataMax, plotY2, plotY1);  
    if (d.record == 1) { fill (#198300); }  
    rect(x-barWidth/2, y, x+barWidth/2, plotY2);  
    fill (#777777);  
  }  
}
```

<http://www.cs.odu.edu/~mweigle/cs795/barchart-processing.html>

Protovis

- ▶ Javascript-based
- ▶ Load your data
- ▶ Draw marks based on the data
- ▶ Functions return the object they are called on, so you can do function chaining

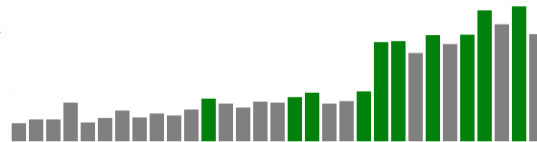


```
var vis = new pv.Panel().width(width).height(height)
    .add(pv.Bar)
    .data(data)
    .bottom(0).width(barWidth)
    .height(function(d) yScale(d.dogs))
    .left(function() xScale(this.index))
    .fillStyle(function (d) (d.record==1)?"green":"gray")
    .root.render();
```

<http://www.cs.odu.edu/~mweigle/cs795/barchart-protovis.html>

D3.js

- ▶ Similar in syntax to Protovis
 - ▶ written by the same folks
- ▶ Code example draws the exact same figure as Protovis
- ▶ Uses SVG, so exact procedure to draw items is a little more involved



```
// draw the rectangles
chart.selectAll("rect")
    .data(data)
    .enter().append("rect")
    .attr("x", function(d, index) {return x(index);})
    .attr("y", function(d) { return height - y(d.dogs);})
    .attr("height", function(datum) {return y(d.dogs);})
    .attr("width", barWidth)
    .attr("style", function(d) {
        if (d.record == 1) return "fill: green";
        else return "fill: gray";});
```

<http://www.cs.odu.edu/~mweigle/cs795/barchart-d3.html>

jQuery Sparklines



```
<!-- Load the libraries in the header -->
<script type="text/javascript" src="jquery-1.7.2.min.js"></script>
<script type="text/javascript" src="jquery.sparkline.min.js"></script>

...

<!-- In the body, make the span for the chart -->
<span class="barchart"></span>

<!-- After the span, write the Javascript -->
<script type="text/javascript">
var myvalues = [9.1,11,11,19.5,9.5,11.75,15.5,12,14,13,16,21.5,19,
17,20,19.5,22.25,24.5,19,20.25,25.13,50,50.5,44.5,53.5,49,53.75,66,59,68,54];

$('.barchart').sparkline(myvalues, {type: 'bar', barColor: 'green'});
</script>
```

<http://www.cs.odu.edu/~mweigle/cs795/barchart-sparklines.html>

► 17

CS 795/895 - Fall 2012 - Weigle

Outline

- Excel
- R
- Inkscape
- Tableau
- Processing.js
- Protovis
- D3.js
- jQuery Sparklines

Next Up:
Design Principles and Graph
Types

Reading:

- Few, "Effectively Communicating Numbers"
- Heer et al., "A Tour Through the Visualization Zoo"

► 18

CS 795/895 - Fall 2012 - Weigle

Your Turn

- ▶ Tell us about the tutorials you did.
- ▶ R
- ▶ Tableau
- ▶ Processing.js
- ▶ Protovis
- ▶ D3.js
- ▶ Inkscape