**Study Guide Unit 1 Name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
DATA 110 Date \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Saidi**

**Intro to DATA 110 – Data Communication and Visualization**

*“The simple graph has brought more information to the data analyst’s mind than any other device.”* John Tukey

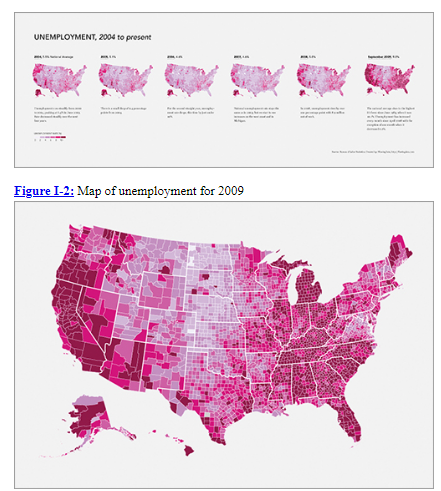
* Intro to Data Vis
* Size of Data
* Hacking
* Data Vis in Journalism
* Flowing Data / Data Underload Examples
* Gapminder
* Shape of Data
* Categorical vs Continuous
* R Markdown
* Scatterplo

## Why Data Visualization?

Visualizations tell stories, find hidden patterns in data, and make compelling cases for making decisions. These stories can help you solve real-world problems, like decreasing crime, improving healthcare, and moving traffic on the freeway. They can shed light on social justice issues and disparities, and graphics can help politicians and companies make important decisions.

John Tukey

1. The greatest value of a picture is when it forces us to notice what we never expected to see.
2. The simple graph has brought more information to the data analyst’s mind than any other device.
3. Numerical quantities focus on **expected** values, graphical summaries on **unexpected** values.



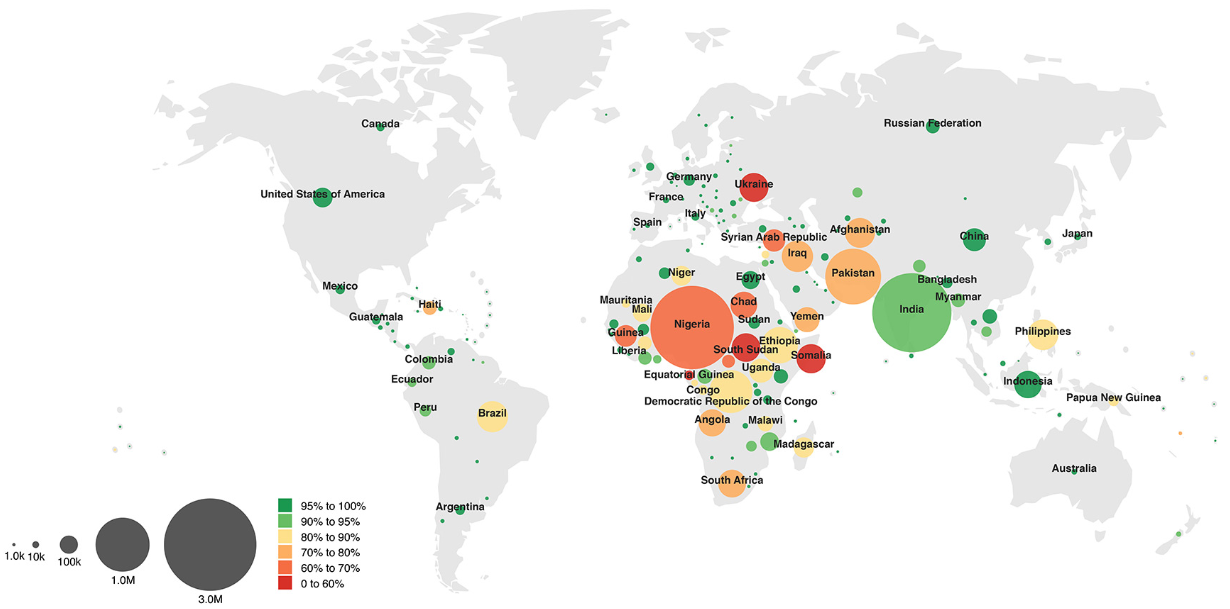
Look at **Figure 1-1 and 1-2** Maps of U.S. unemployment for 2004 to 2009. Discuss these answers with your neighbor.

1. How is this map broken up?
2. What do the darker colors mean?
3. What regions had the most dark areas?
4. Do you see a pattern or trend from 2004 to 2009? Be specific.
5. Is there additional information you would have liked included on these maps?

# Data visualization is part art and part science. The challenge is to get the art right without getting the science wrong and vice versa. A data visualization first and foremost has to accurately convey the data. It must not mislead or distort. If one number is twice as large as another, but in the visualization they look to be about the same, then the visualization is wrong. At the same time, a data visualization should be aesthetically pleasing. Good visual presentations tend to enhance the message of the visualization. If a figure contains jarring colors, imbalanced visual elements, or other features that distract, then the viewer will find it harder to inspect the figure and interpret it correctly (Fundamentals of Data Visualization, Claus O. Wilke)

A shift in release of government data came in mid-2009 with the launch of [Data.gov](file:///C:\Users\rsaidi\Dropbox\Rachel\MontColl\DATA%20110\Notes\data.gov), a comprehensive catalog of data provided by federal agencies and represents transparency and accountability of groups and officials. Now all this information is in one place and better formatted for analysis and visualization.

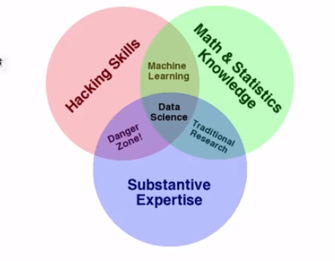
Check out the website, <https://stamen.com/work/> to see what a company that specializes in data visualization and analytics does with data projects. Below is one example:



## The Size of Data (in case you didn’t know….)

|  |  |  |
| --- | --- | --- |
| Potential database volumes in bytes for some typical applications (volumes estimated to the nearest order of magnitude). Strictly, bytes are counted in powers of 2 – 1 kilobyte is 1024 bytes, not 1000. | | |
| 1 megabyte | 1 000 000 | Single data set in a small project database |
| 1 gigabyte | 1 000 000 000 | Entire street network of a large city or small country |
| 1 terabyte | 1 000 000 000 000 | Elevation of entire Earth surface recorded at 30 m intervals |
| 1 petabyte | 1 000 000 000 000 000 | Satellite image of entire Earth surface at 1 m resolution |
| 1 exabyte | 1 000 000 000 000 000 000 | A possible 3-D representation of the entire Earth at 10 m resolution |
| 1 zettabyte | 1 000 000 000 000 000 000 000 | One-fifth of the capacity (in 2013) of U.S. National Security Agency Utah Data Center |

## One of the key data science traits (Drew Conway)



Hacking generally refers to unauthorized intrusion into a computer or a network. The person engaged in hacking activities is known as a hacker. This hacker may alter system or security features to accomplish a goal that differs from the original purpose of the system.

Hacking can also refer to non-malicious activities, usually involving unusual or improvised alterations to equipment or processes.

Hacking Skills

You have to be able to synthesize different skills from all over the place, be willing to do the hard work to search for the information you need.

## Key Characteristics of Hackers (from JHU Coursera – The Data Scientist Toolbox)

* Willing to find answers on their own
* Knowledgeable about where to find answers on their own
* Unintimidated by new data types or packages
* Unafraid to say they don’t know the answer
* Polite but relentless

## Principles of Analytic Graphics

* Show comparisons
* Show causality, mechanism, explanation
* Show multivariate data
* Integrate multiple modes of evidence
* Describe and document the evidence
* Content is king

**Telling Stories with Data**

## Some Reputable Journalism Sites with Visualizations

* Vox
* New York Times
* Wall Street Journal
* Bloomberg News
* Fivethirtyeight.com
* Washington Post
* Gapminder.org
* FlowingData (<https://flowingdata.com/>) by Nathan Yau or go to Data Underload: <https://flowingdata.com/category/projects/data-underload/>

Check out my presentation on [Data Visualizations Used for Social Justice.](file:///C:\Users\rsaidi\Dropbox\Rachel\MontColl\Mathtalk\Social%20Justice%20Through%20Statistics.pptx)

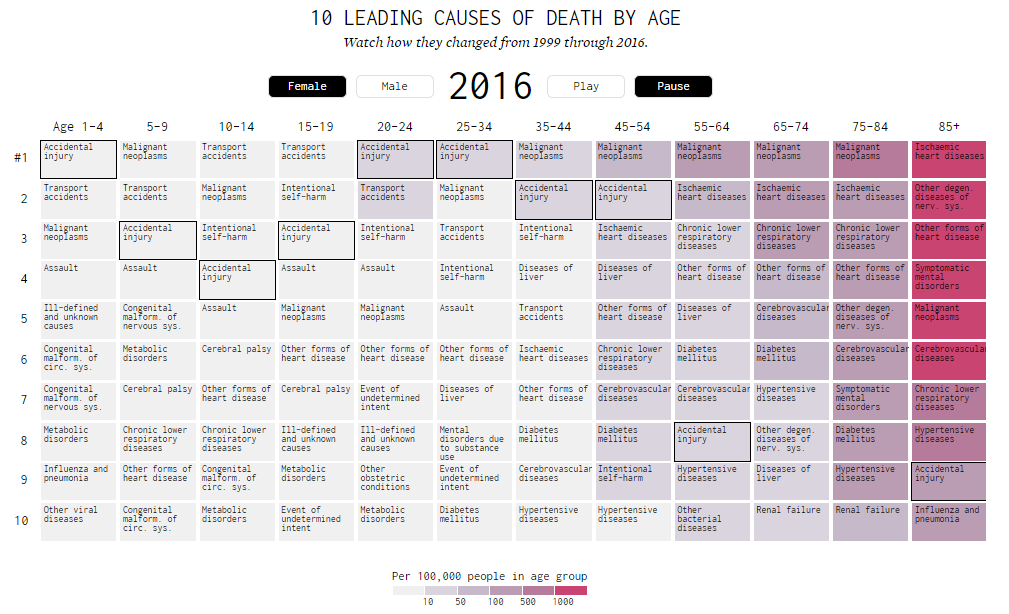
## Intersection of Data, Art and Entertainment

Data can be both informative and artful. Check out Nathan Yau’s many examples on <https://flowingdata.com/>. Look at the image of Nathan Yau’s “10 Leading Causes of Death by Age” <https://flowingdata.com/2018/10/02/shifting-death/> (see screen capture below). Discuss these answers with your neighbor.

1. Name all of the variables Yau considers.

1. What do the colors represent?

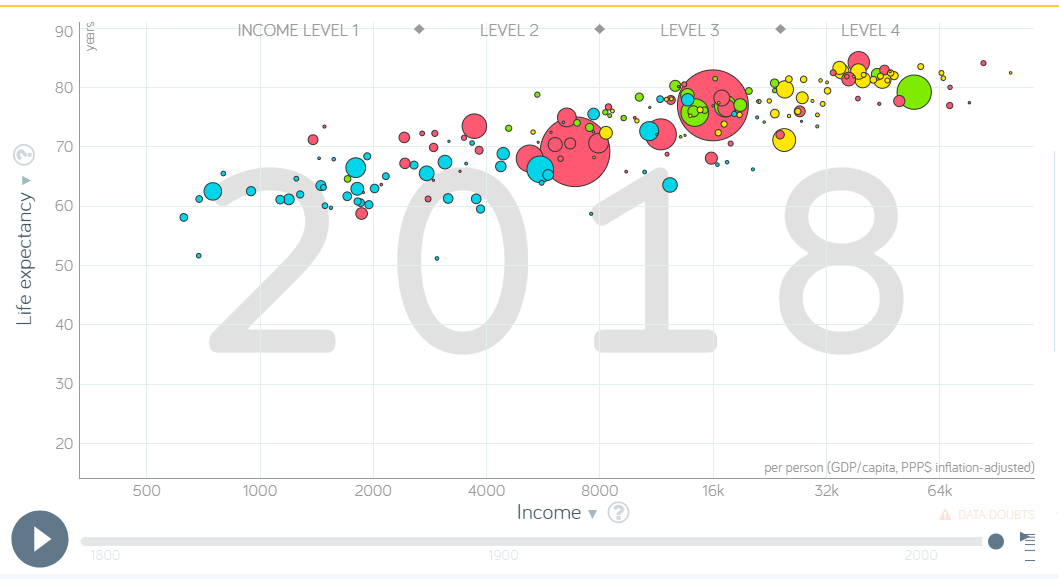
1. What is Yau’s intended effect in using animation?
2. Do you believe this is an effective data visualization? Why or why not?



## **Compelling Stories Through Data Exploration**

[200 Years, 200 Countries](https://www.youtube.com/watch?v=jbkSRLYSojo), by Hans Rosling

[Gapminder.org](https://www.gapminder.org/) (You should explore this entire site, but be sure to check out the Tools and Data links)



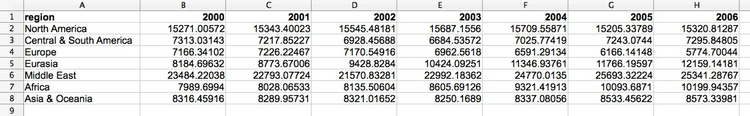
## Better Visualizations

Note the following sections in ***Visualize This***:

* Patterns
* Relationships
* Design
* Coding (legends and labels)
* Geometrical scaling
* Include source
* Consider your audience

## What shape is your data?

Particularly when data shows a time series for a single variable, it is often provided like this data on trends in international oil production by region, in “wide” format:



(Source: Peter Aldhous, from [U.S. Energy Information Administration](http://www.eia.gov/cfapps/ipdbproject/IEDIndex3.cfm?tid=5&pid=53&aid=1) data)

Here, all of the numbers represent the same variable, and there is a column for each year. This is good for people to read, but most software for data analysis and visualization does not work well with data in this format.

So if you receive **“wide” data,** you will usually need to convert it to **“long” format (**shown here) using some software, such as R or Open Refine:



(Source: Peter Aldhous, from [U.S. Energy Information Administration](http://www.eia.gov/cfapps/ipdbproject/IEDIndex3.cfm?tid=5&pid=53&aid=1) data)

Notice that now there is one column for each variable, which makes it easier for computers to understand.

## Types of data: categorical vs. continuous (from Peter Aldhous)

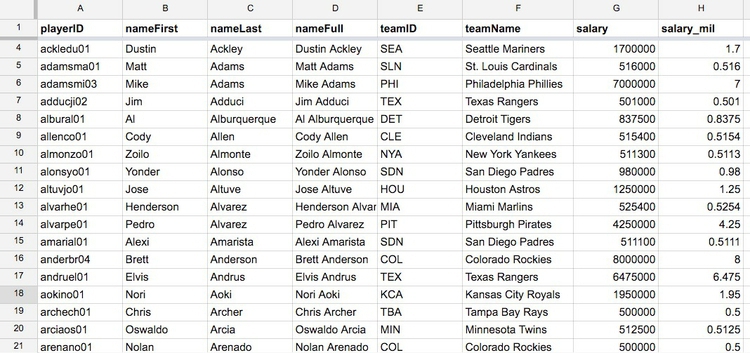
Before analyzing a dataset, or attempting to draw a graphic, it is important to consider what, exactly, you are working with.

Statisticians often use the term “variable.” This simply means any measure or attribute describing a particular item, or “record,” in a dataset. For example, school students might gather data about themselves for a class project, recording their gender and eye color, and height and weight. There is an important difference between gender and eye color, called “categorical” variables, and height and weight, termed “continuous.”

* **Categorical** variables are descriptive labels given to individual records, assigning them to different groups. The simplest categorical data is dichotomous, meaning that there are just two possible groups — in an election, for instance, people either voted, or they did not. More commonly, there are multiple categories. We analyze categorical data differently than continuous or discrete quantitative data.
* **Continuous** data is richer, consisting of numbers that can have a range of values on a sliding scale. Continuous variables might include temperature and amount of rainfall, age, height, weight, blood pressure, etc.

There is a third type of data we often need to consider: **date and time**. Perhaps the most common task in data journalism is to consider how a variable or variables have changed over time.

Datasets will usually contain a mixture of categorical and continuous variables. Here, for example, is a small part of a spreadsheet containing data on salaries for Major League Baseball players at the opening of the 2014 season:



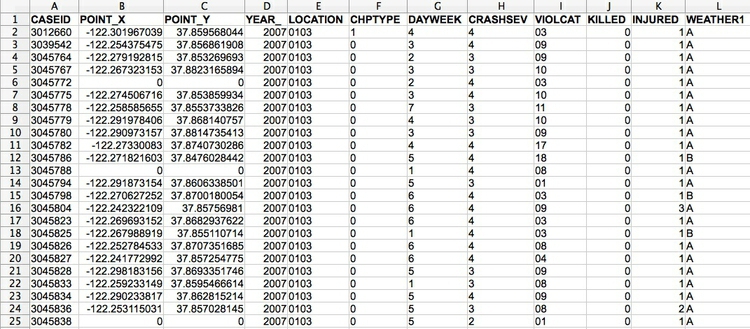
(Source: Peter Aldhous, data from [Lahman Baseball Database](http://www.seanlahman.com/baseball-archive/statistics/) data)

This is a typical data table layout

* the players --- form the rows
* variables --- arranged in columns

It is easy to recognize the categorical variables of teamID and teamName because they are each entered as text. The numbers for salary, expressed in full or in millions of dollars (salary\_mil), are continuous variables.

**Never** assume that every **number** in a dataset represents a **continuous variable.** Text descriptions can make datasets unwieldy, so database managers often adopt simpler codes such as letting numbers store categorical data.   
  
You can see this in the following example, showing data on traffic accidents resulting in injury or death in Berkeley, downloaded from a database maintained by researchers on the Berkeley campus.



(Source: Peter Aldhous, from [Transportation Injury Mapping System](http://tims.berkeley.edu/) data)

Of the numbers seen here, only the YEAR, latitudes and longitudes (POINT\_Y and POINT\_X) and numbers of people KILLED or INJURED actually represent continuous variables. (Look carefully, and you will see that these **numbers are justified right within each cell**. **The other numbers are justified left**, like the text entries, because they were imported into the spreadsheet as text values.)

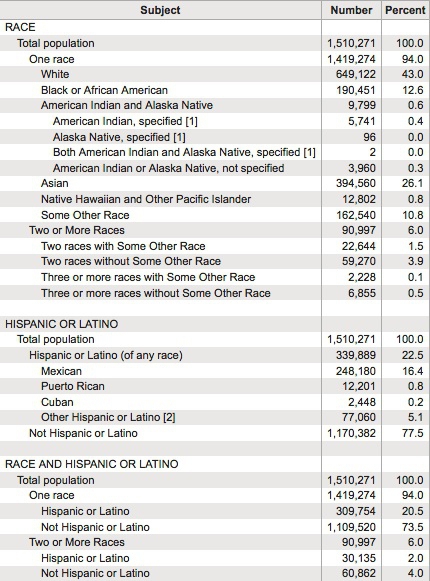
Like this example, many datasets are difficult to interpret without their supporting documentation. So each time you acquire a dataset, if necessary make sure you also obtain the “codebook” describing all of the variables/fields, and how they are coded. [Here is the codebook](http://paldhous.github.io/ucb/2016/dataviz/data/SWITRS_codebook.pdf) for the traffic accident data.

## Working with categorical data

You might imagine that there is little that you can do with categorical data alone, but it can be powerful, and can also be used to create new continuous variables.

The most basic operation with categorical data is to aggregate it by counting the number of records that fall into each category. This gives a table of **frequencies.**  These are often divided by the total number of records, then multiplied by 100 to show them as percentages of the total.

Here is an example, showing data on the racial and ethnic identities of residents of Alameda County, from the 2010 US Census:



(Source: [American FactFinder](http://factfinder2.census.gov/faces/nav/jsf/pages/index.xhtml), U.S. Census Bureau)

Creating frequency counts from categorical data creates a new continuous variable — what has changed is the level of analysis. In this example, the original data would consist of a huge table with a record for each person, noting their racial/ethnic identity as categorical variables; in creating the frequency table shown here, the level of analysis has shifted from the individual to the racial/ethnic group.

We can ask more interesting questions by considering two categorical variables together — as pioneering data journalist Philip Meyer showed when he collected and analyzed survey data to examine the causes of the 1967 Detroit Riot. In July of that year, one of the worst riots in U.S. history raged in the city for five days, following a police raid on an unlicensed after-hours bar. By the time calm was restored, 43 people were dead, 467 injured and more than 2,000 buildings were destroyed.

At the time, Detroit was regarded as a leader in race relations, so local racial discrimination was not initially seen as one of the main underlying causes of what happened. One popular theory at the time was that the riots were led by black residents who had moved to Detroit from the rural South. Meyer demolished this idea by examining data on whether or not the people surveyed had rioted, and whether they were brought up in the South or the North. He combined these results into a “contingency table” or “cross-tab”:

|  | **South** | **North** | **Total** |
| --- | --- | --- | --- |
| **Rioters** | 19 | 51 | 70 |
| **Non-rioters** | 218 | 149 | 367 |
| **Total** | 237 | 200 | 437 |

It certainly looks from these numbers as if Northerners were more likely to have participated in the riot.

But Meyer’s team only interviewed a sample of people from the affected neighborhoods, not everyone who lived there. If they had taken another sample, might they have obtained different results? This is one example where some more sophisticated statistical analysis can help. For contingency tables, a method known as the **chi-squared test** asks the relevant question: if Southerners and Northerners were in fact equally likely to have rioted, what is the likelihood of obtaining a sample as biased as this by chance alone? In this case, the chi-squared test told Meyer that the probability was less than one in a thousand. So Meyer felt confident writing in the newspaper that Northerners were more likely to have rioted. His work won a Pulitzer Prize for the Detroit Free Press and shifted the focus of political debate about the riot to racial discrimination in policing and housing in Detroit.

## Bar Charts for Categorical Data (from Nicole Radziwill)

To create a bar chart/bar plot in R with the barplot function using categorical data, which is a collection of numbers that represent frequencies (or counts) of events or outcomes that fall into different groups or categories. [Note: If you are trying to display distributions of quantitative data, choose a histogram instead. Bar charts are for categorical data only. BAR CHARTS ARE NOT THE SAME AS HISTOGRAMS!]

The lengths of the bars are proportional to the values they represent, and the bars can be oriented vertically or horizontally.

* Good bar charts are labeled nicely, with a clear description of the categories that are being counted on the horizontal (x) axis, and a label on the vertical (y) axis that indicates whether frequencies or counts are displayed.
* Membership into each category should be mutually exclusive. That is, you don’t want an observation to appear in multiple bars.
* Pie charts should be avoided. A bar chart is a better way to display your data. If you are trying to illustrate a collection of items that naturally add up to 100%, a pie chart may be appropriate. However, if there are multiple categories where it may be difficult to distinguish which slice is bigger (such as one observation of 28% and another observation of 29%) a bar chart may be more appropriate.
* If you want to display your data in terms of TWO categorical variables, choose a segmented bar chart (described in a separate chapter). Even More Caution: There is a BIG DIFFERENCE between a bar chart and a histogram! Even though a bar chart looks really similar to a histogram at first glance, take a close look at what kind of data is on the horizontal (x) axis. In a bar chart, the horizontal axis lists categories. In a histogram, the horizontal axis will contain ranges of numbers that represent a continuum (e.g. 0-10, 10-20, 20-30 and so forth). Also, in a bar chart, there will be some space between the bars indicating that the categories are separate from one another - whereas with a histogram, there will be no space between the bars! The bars will be very cozy in a histogram, mashed up against one another like they're at a crowded party, whereas the bars in a bar chart need a little more breathing room, and thus are distanced from one another.

## **Bar Charts**

For small data sets, you may have already tallied up your observations, and you don’t need to load a whole file in to create your bar chart. Here is an example of data generated by opening one package of regular M&Ms to look at the distribution of colors, working with your data as a vector. Here are the counts: 12 blue ones, 6 brown ones, 8 green ones, and so forth:

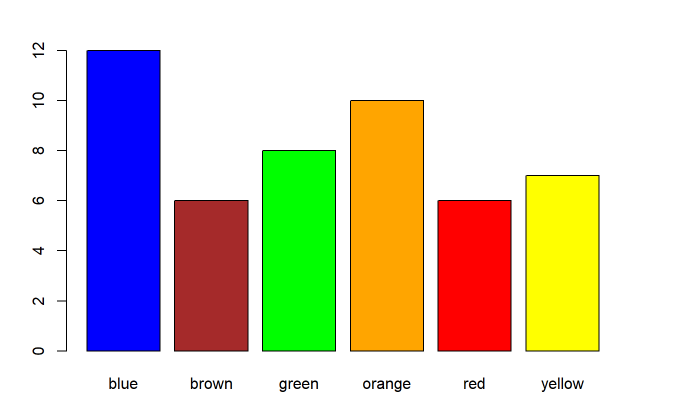
mm.counts <- c(12,6,8,10,6,7)

names(mm.counts) <- c("blue","brown","green","orange",

"red","yellow")

mm.colors <- c("blue","brown","green","orange","red","yellow") *# creates a vector of the palette of colors to be used in the bar chart (that match the m & m colors)*

barplot(mm.counts,col=mm.colors)



## Introducing R and R Studio

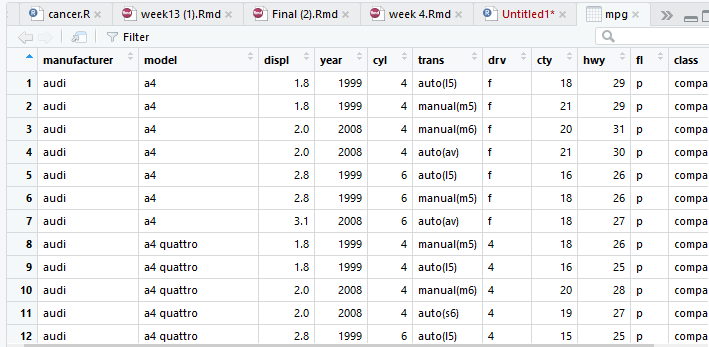
For homework, you will install R and R Studio using provided documents. Here are some shortcuts you will be able to use one you start to code:

|  |  |
| --- | --- |
| Operation | Shortcut |
| <- (assignment operator) | Alt hyphen |
| %>% (piping operator) | Ctrl Shift M |
| Comment out code blocks | Ctrl Shift C |
| Get info on function | Put cursor on function; F1 |
| Find and replace | Ctrl F |
| Clear | Ctrl L |

For more information on keyboard shortcuts, click on the **Tools tab** in **R Studio** and select **Keyboard Shortcut Help**

## **Example using Data “mpg”**

Below are a few rows from the data frame **mpg**, which is a dataset built into R.

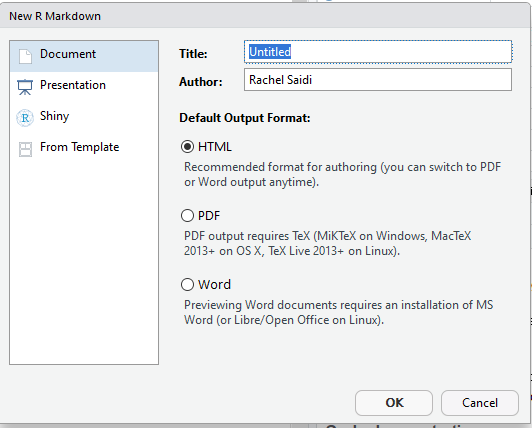


* Variables: Top row of data frame: manufacturer, model, etc.
* Observations: each row of data is a separate record
* Values: audi, a4, 2008, compact, etc.
* Note that R is case sensitive; i.e. **Year** and **year** are different
* There should be no empty cells; missing data should be represented by NA
* Avoid spaces in variables; i.e. use **no\_spaces**, not **no spaces**

**R Markdown to Explore Relationships**

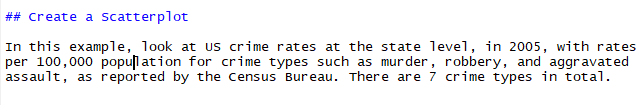
We will learn to code within the context of using R Markdown. Click here for more information on [R Markdown](https://rmarkdown.rstudio.com/articles_intro.html).

Start by opening a new R Markdown file. Select output format to be HTML or Word or PDF.

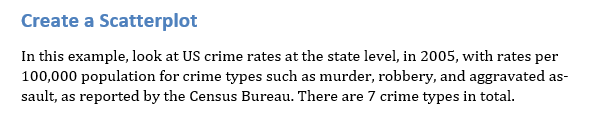


In the white space on the Markdown page, ## will give you subtitles – you know you have done it correctly because they are in blue

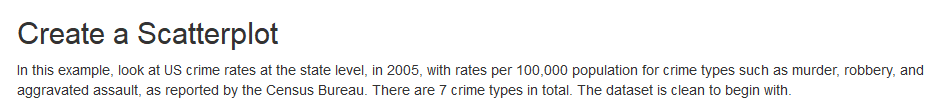
When you do not include hashtags, this gives text (in black). You can use #, ##, ###, etc. for different levels of subtitles, or no hashtags at all for plain text.



Here is how it looks when it is knitted in Word:



Here is how it looks when knitted to HTML



# **Create Chunks**

You can create chunks of code with the shortcut control+ alt+ I. Here is a chunk:



Once you have code typed in, to run just that chunk, click on the right green arrow.

Be sure to leave a line or two of space after a chunk before you add your next subtitle

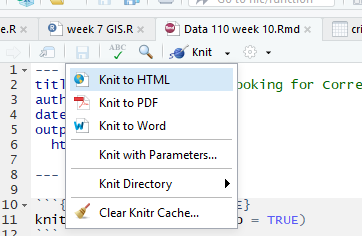
## Subtitle

If you don’t leave a space, it may not knit properly.

Once you have written all your subtitles, notes, and chunks, run the code to be sure it works.

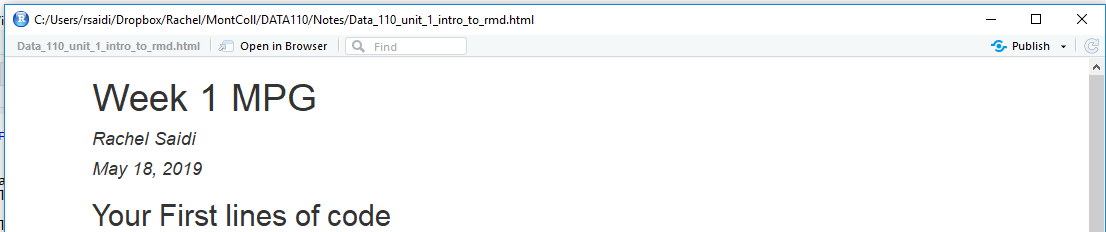
## Knitting

You can knit at any time to test how it looks. You can switch from knitting to Word, HTML, pdf at any time using the knit button at the top of your Markdown code.



## **Finally, Publish to Rpubs**

When you are done with your Markdown code, or when you just want to see what it looks like knitted to HTML, click the   
Rpubs “Publish” symbol:



You will need to set up a free Rpubs account, but once you do so, you will be able to publish your Markdown as an HTML page and share the link. Alternatively, you can publish your work to Github. We will learn about that later in the class.

# Back to the mpg data in Markdown

[Click here](http://rpubs.com/rsaidi/497334) for my Rpubs file for these notes.

## **Use Side-by-Side Boxplots**

Here is some easy code to create 3 groups of boxplots with some easy-to-access data, filled by group. Since the groups are discrete, you can get rid of the shading

*# install.packages("tidyverse")*

**library**(tidyverse)

## -- Attaching packages ------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.1.0 v purrr 0.3.0

## v tibble 2.0.1 v dplyr 0.8.0.1

## v tidyr 0.8.3 v stringr 1.4.0

## v readr 1.3.1 v forcats 0.4.0

## -- Conflicts ---------------------------------------------- tidyverse\_conflicts() --

## x dplyr::filter() masks stats::filter()

## x dplyr::lag() masks stats::lag()

allscores <- readr::read\_csv("https://goo.gl/MJyzNs")

## Parsed with column specification:

## cols(

## group = col\_double(),

## pre = col\_double(),

## post = col\_double(),

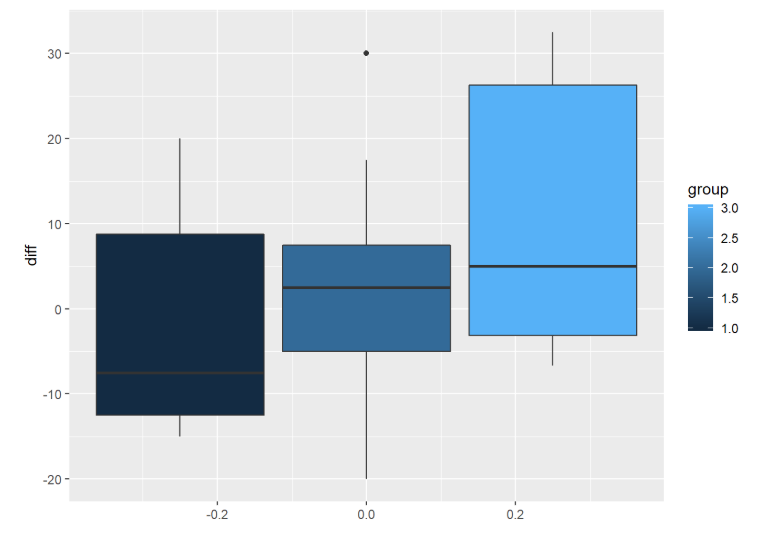
## diff = col\_double()

## )

boxpl <- allscores %>% ggplot() + geom\_boxplot(aes(y=diff, group=group,

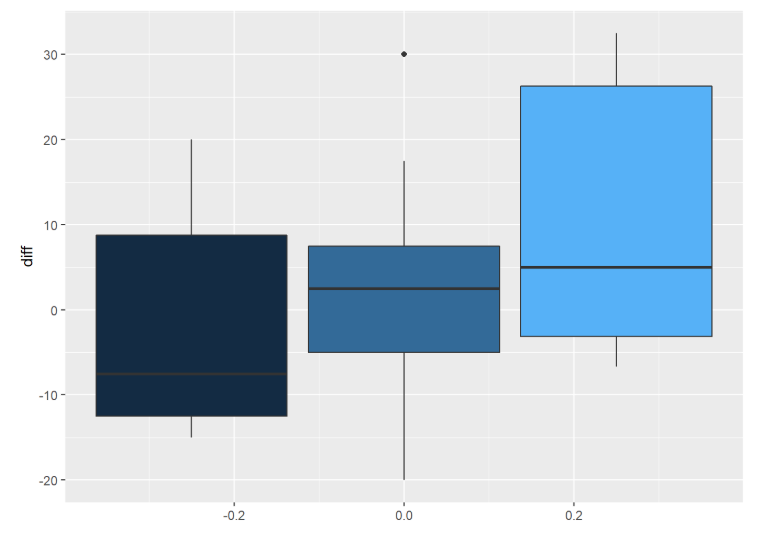
fill=group))

boxpl



boxpl2 <- boxpl + guides(fill=FALSE)

boxpl2



allscores %>%

mutate(group=factor(group, levels=c("1","2","3"), ordered=TRUE)) %>%

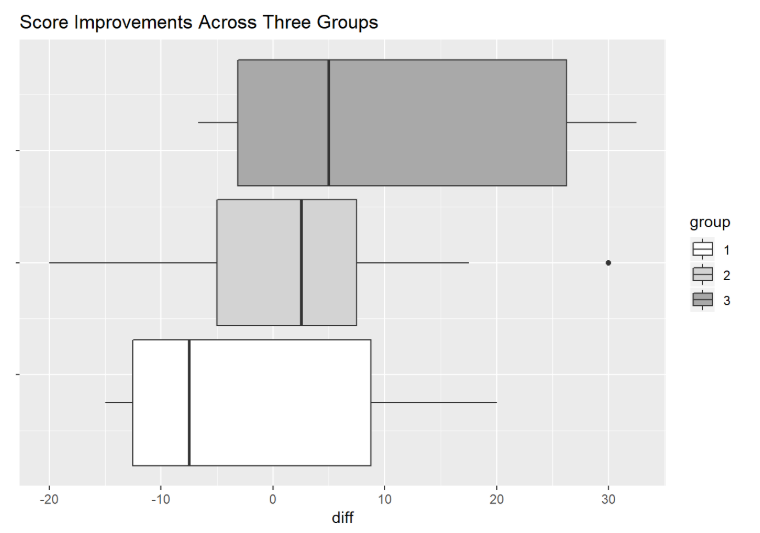
ggplot() + geom\_boxplot(aes(y=diff, group=group, fill=group)) +

scale\_fill\_manual(values=c("white","lightgray","darkgray")) +

theme(axis.text.y=element\_blank()) +

ggtitle("Score Improvements Across Three Groups") +

coord\_flip()



## **Load Built in Data from R**

Some data frames are built in to R, such as mpg. Load the data, then use str and head to look at the data.

{r mpg} loads the data. Alternatively, you can use the command: load(“mpg”)

You will look at the data using the command “str” (gives the structure of the data), “head” (lists the first 6 rows of observations in the dataset), and “describe” from the “psych” package (gives quite detailed summary statistics on the continuous variables).

*# install.packages("tidyverse")*

*# install.packages("psych")*

**library**(tidyverse)

**library**(psych)

##

## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':

##

## %+%, alpha

str(mpg)

## Classes 'tbl\_df', 'tbl' and 'data.frame': 234 obs. of 11 variables:

## $ manufacturer: chr "audi" "audi" "audi" "audi" ...

## $ model : chr "a4" "a4" "a4" "a4" ...

## $ displ : num 1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...

## $ year : int 1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...

## $ cyl : int 4 4 4 4 6 6 6 4 4 4 ...

## $ trans : chr "auto(l5)" "manual(m5)" "manual(m6)" "auto(av)" ...

## $ drv : chr "f" "f" "f" "f" ...

## $ cty : int 18 21 20 21 16 18 18 18 16 20 ...

## $ hwy : int 29 29 31 30 26 26 27 26 25 28 ...

## $ fl : chr "p" "p" "p" "p" ...

## $ class : chr "compact" "compact" "compact" "compact" ...

head(mpg)

## # A tibble: 6 x 11

## manufacturer model displ year cyl trans drv cty hwy fl class

## <chr> <chr> <dbl> <int> <int> <chr> <chr> <int> <int> <chr> <chr>

## 1 audi a4 1.8 1999 4 auto(~ f 18 29 p comp~

## 2 audi a4 1.8 1999 4 manua~ f 21 29 p comp~

## 3 audi a4 2 2008 4 manua~ f 20 31 p comp~

## 4 audi a4 2 2008 4 auto(~ f 21 30 p comp~

## 5 audi a4 2.8 1999 6 auto(~ f 16 26 p comp~

## 6 audi a4 2.8 1999 6 manua~ f 18 26 p comp~

describe(mpg)

## vars n mean sd median trimmed mad min max range

## manufacturer\* 1 234 NaN NA NA NaN NA Inf -Inf -Inf

## model\* 2 234 NaN NA NA NaN NA Inf -Inf -Inf

## displ 3 234 3.47 1.29 3.3 3.39 1.33 1.6 7 5.4

## year 4 234 2003.50 4.51 2003.5 2003.50 6.67 1999.0 2008 9.0

## cyl 5 234 5.89 1.61 6.0 5.86 2.97 4.0 8 4.0

## trans\* 6 234 NaN NA NA NaN NA Inf -Inf -Inf

## drv\* 7 234 4.00 0.00 4.0 4.00 0.00 4.0 4 0.0

## cty 8 234 16.86 4.26 17.0 16.61 4.45 9.0 35 26.0

## hwy 9 234 23.44 5.95 24.0 23.23 7.41 12.0 44 32.0

## fl\* 10 234 NaN NA NA NaN NA Inf -Inf -Inf

## class\* 11 234 NaN NA NA NaN NA Inf -Inf -Inf

## skew kurtosis se

## manufacturer\* NA NA NA

## model\* NA NA NA

## displ 0.44 -0.91 0.08

## year 0.00 -2.01 0.29

## cyl 0.11 -1.46 0.11

## trans\* NA NA NA

## drv\* NaN NaN 0.00

## cty 0.79 1.43 0.28

## hwy 0.36 0.14 0.39

## fl\* NA NA NA

## class\* NA NA NA

It is essential to recognise that variables may be: int (integer), num (numeric), or double vs char (character) and factor (for categories)

Typically, chr or factor are used for discrete variables and int, dbl, or num for continuous variables.

### Introducing ggplot2 and the grammar of graphics

The “gg” in **ggplot2** stands for “[grammar of graphics](http://www.amazon.com/The-Grammar-Graphics-Statistics-Computing/dp/0387245448),” an approach to drawing charts devised by the statistician Leland Wilkinson. Rather than thinking in terms of finished charts like a scatter plot or a column chart, it starts by defining the coordinate system (usually the X and Y axes of a cartesian system), maps data onto those coordinates, and then adds layers such as points, bars and so on. This is the logic behind ggplot2 code.

Some key things to understand about **ggplot2**:

* ggplot This is the master function that creates a **ggplot2** chart.
* aes This function, named for “aesthetic mapping,” is used whenever data values are mapped onto a chart. So it is used when you define which variables are plotted onto the X and Y axes, and also if you want to change the size or color of parts of the chart according to values for a variable.
* geom All of the functions that add layers to a chart start with geom, followed by an underscore, for example geom\_point() or geom\_bar(). The code in the brackets for any geom layer styles the items in that layer, and can include aes mappings of values from data.
* theme This function modifies the appearance of elements of a plot, used, for example, to set size and font face for text, the position of a legend, and so on.
* scale Functions that begin with scale, followed by an underscore, are used to modify the way an aes mapping of data appears on a chart. They can change the axis range, for example, or specify a color palette to be used to encode values in the data.
* + is used each time you add a layer, a scale, a theme, or elements like axis labels and a title After a + you can continue on the same line of code or move the next line. I usually write a new line after each +, which makes the code easier to follow.

## **Now make a scatterplot using ggplot2**

Make a scatterplot of city vs highway miles per gallon, but sort/color points by either 4-wheel, front-wheel, or rear-wheel drive

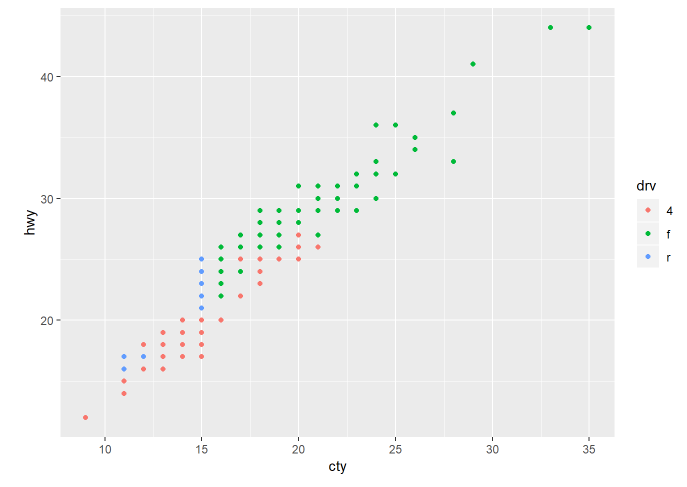
Here is how we will code:

1. name the plot: “plot1” <-
2. call back the name of the dataset “mpg” and “pipe it” (more on that later)
3. call “ggplot” to make a set of axes, with the aesthetics (aes) for city and highway mpg, but color points by the factors for drv
4. add geom\_point to see the points
5. call plot1 to see the entire plot

plot1 <- mpg %>% ggplot(aes(cty, hwy, color = drv))+

geom\_point()

plot1



Notice that the blue points for rear-wheel drive are only at the lower left side of the plot (i.e., not great mpg). Red points for 4-wheel drive have a wider spread of points, but they are also mainly at the lower left corner of the plot. The green points for front-wheel drive are mostly at the upper right, for the higher mpg.

## **Add a title and labels**

Although there are already axes labels, we can do better. We should also add a title

plot1 <- mpg %>% ggplot(aes(cty, hwy, color = drv))+

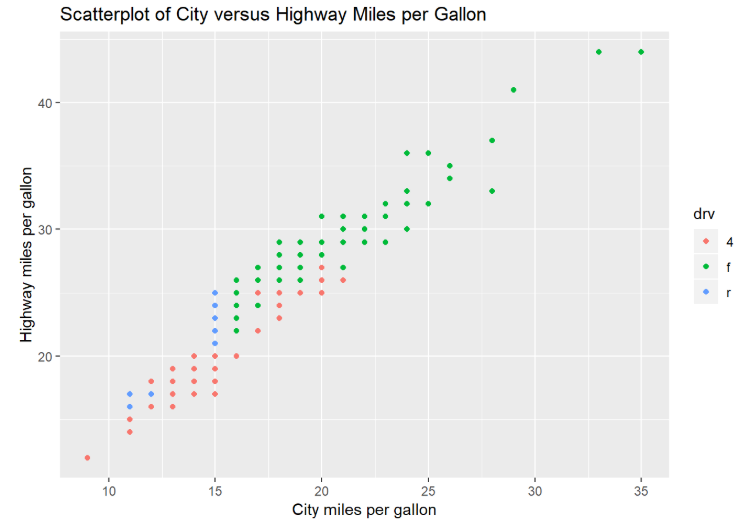
geom\_point()+

xlab("City miles per gallon") +

ylab("Highway miles per gallon") +

ggtitle("Scatterplot of City versus Highway Miles per Gallon")

plot1



**Homework Unit 1**

1. (Ungraded) Install R and R-Studio on your computer using the documents and videos provided. Sign up for a free Rpubs account. Start exploring R Studio by copying the code from the Markdown page into your own Markdown document.
2. (Worth up to 10 points) Listen to the podcast: [Stats and Stories](https://statsandstories.net/media1/2018/11/15/understanding-data-in-the-digital-age-stats-and-stories-episode-70) with Statistician and Journalist Mark Hanson  ([here is the abstract](https://bb-montgomerycollege.blackboard.com/bbcswebdav/pid-4278450-dt-content-rid-35530298_1/xid-35530298_1)). Write a one-paragraph reaction to this podcast. Be sure to edit for grammar/spelling/sentence structure.
3. (Worth up to 10 points)  Explore the “reputable sites” listed in the notes above (Vox, NY Times, FiveThiryEight, etc.).
   1. Select one site and then select one relatively current visualization.
   2. On a Word document, paste the image (must have color).
   3. Write a summary about this image.
   4. Write what has been done well.
   5. Write what could be improved; might show selection bias or any other type of distortion.
   6. Be sure to include a link/url to your data visualization (and be sure to put your name at the top)

Submit the two assignments via the course Assignment Dropbox Unit 1 by **11:59 pm next Monday.**We will present/discuss your submissions during the next class.