TEAM 12 - ELECTIONS: THE REAL STORY

By: Tina (Tianhui) Diao, Giseob Hyun, Steven Kamen, Iris (Lu) Ni and Wenyu Zhang

CONTENTS

I. Introduction

The United States Presidential campaigns are starting to heat up, and while many of us have seen surface level analysis of past elections, most of the story is unavailable to us. The average citizen thinks the goal of campaign teams is to win the most votes; to win the hearts of the masses. While this should be the ultimate goal, this may not always be optimal.

The United States electoral system is based only indirectly on popular vote. In a presidential campaign, the popular vote of each state is tallied, and then every representative and senator from that state is expected to vote for the candidate that received the most votes. Seldom does a candidate choose to vote differently, however, the representative has this option. Of course, there could be harsh political consequences for voting differently. Thus, ignoring

additional prestige that may come with winning additional votes and focusing only on the purely logical goal of winning electoral votes, a candidate only cares about getting one more vote than his rival from each state. It is theoretically possible for one candidate to get around 70% of the vote, but still lose. This would happen if states tallying 270 electoral votes slightly favor the winner, and every other state votes 100% for the loser (which would never happen).

Thus, there should be reason to believe that campaign teams, paid millions upon millions, are doing more than running ads, making phone calls and hosting events for their candidate. Campaigning not only a game of persona, but also a game of numbers, far more mathematical than the average citizen may expect. The simplest example is as follows: Obama will campaign minimally in Massachusetts because he knows he will get more than half the vote, and once a candidate receives half of the vote, additional support contributes nothing to securing a spot in the White House. On the other hand, both candidates will campaign heavily in Florida and Ohio, swing states where garnering the support of an additional few thousand voters could determine the outcome of the entire national election (we saw this with Bush v. Gore in 2000).

The goal of our project was to both touch on possible strategic considerations as such, while also searching for trends among voters. With a combination of such results and further development of our project, campaign teams could possibly locate more efficient campaign methods that could change our political campaigning and landscape. Again, the basic idea is to expose trends that campaign teams could exploit while adhering to the convoluted framework of America's electoral system. If we assume the only goal is to win the election, which poses philosophical and moral questions outside the scope of our project, then optimizing and exploiting according to the rules of the game should be first and foremost preferred. And as we can see, even the most minor change of rules can justify radically new strategies.

II. What We Looked For

When we started this project, we were looking for some vague concept that would connect known, or possibly stereotypical, trends to correlations we logically thought may exist, but to which we never heard any reference. So, we started by pulling 2008 presidential election, income, population, population density and golf course data. While we had difficulty finding a list of golf courses by county, we eventually came across a site that looked like it had a complete data set. However, because of the way the data was presented, we had no way of knowing how complete the data was until we scraped everything. Additionally, because of the formatting of the website, it took hundreds of lines of code to pull everything from the website and merge it with the data we had. To our disappointment, the data was strikingly incomplete, and in a non-systematic way, so we could get nothing from it. Random counties contained by far the most golf courses and places famous for their golf courses had only a handful, if any. This will be explained in depth toward the end of the report.

So, at this point, we had 2008 election, median income, population and population density by county data. Initially we were going to present everything by state, since only aggregate states have relevance to presidential election, but we decided to do everything by county so our study could be more relevant to more local situations. We would have preferred to do everything by electoral district, but there was a lack of data in this format. More so, we could not find a package in R that would allow us to visualize such data. So, we assumed that by-county data would present a good proxy for by-district data.

At this point, what we had was very standard, and we figured that anyone who knew anything about elections probably looked into this in far more depth than we would, so we had to

add something that could give others a broader focus. We decided to look at university count by county. It is well known that places of higher education are predominantly liberal, but is there a correlation between the universities themselves and the surrounding area? This could potentially be important, because a lot of election studies focus on areas surrounding educational environments. After all, this is a matter of convenience, should students or university faculty be conducting such a study. Then again, this entire idea could be confounded by the fact that universities tend to be located in populated urban centers, which tend to have lower median income and vote in more liberal fashions. Thus, it is important to note that while we looked for concrete trends, the link of causality is up for debate.

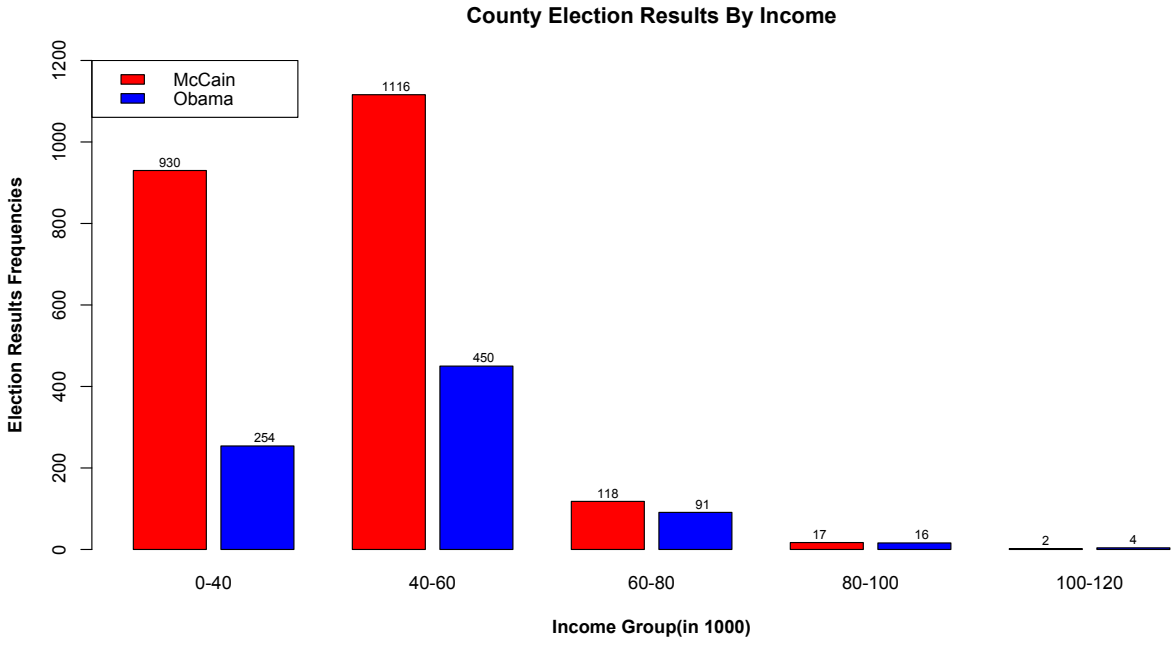
Before setting out to scrape and analyze the data, we hypothesized some of the following ideas: many people higher on the economic scale support Republican candidates over the Democrat candidates. Golf courses tend to be located in more affluent, suburban areas. Here, we adhere to the stereotype that golf is a sport for the rich. More densely populated areas tend to vote for democrats. Educational centers exhibit more liberal trends.

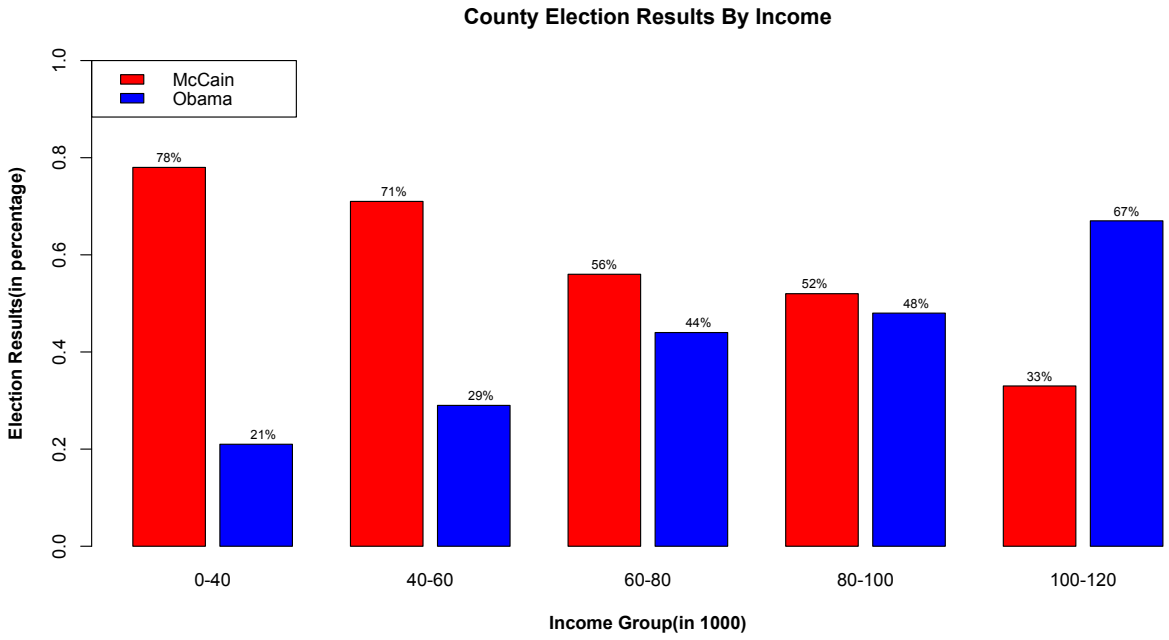
After thinking a bit, we realized that there are major flaws with all of these ideas, with regards to how we would be obtaining our data. Because of the complexity of potential issues and time constraints, we decided to ignore such sources of error, while noting that they exist. While the wealthy may tend to vote for conservatives, median income by county has no way of capturing where these rich people are located. Just because a county's median income is higher does not tell us much; median county income seldom exceeded $100,000. When we say rich people tend to vote for conservatives, we mean two things. First, we imply wealth, which is *essentially* unrelated to income. Second, we mean people who make far more than $100,000 per year. So, what is meant, in the conventional sense, by richer people tend to vote conservative (of

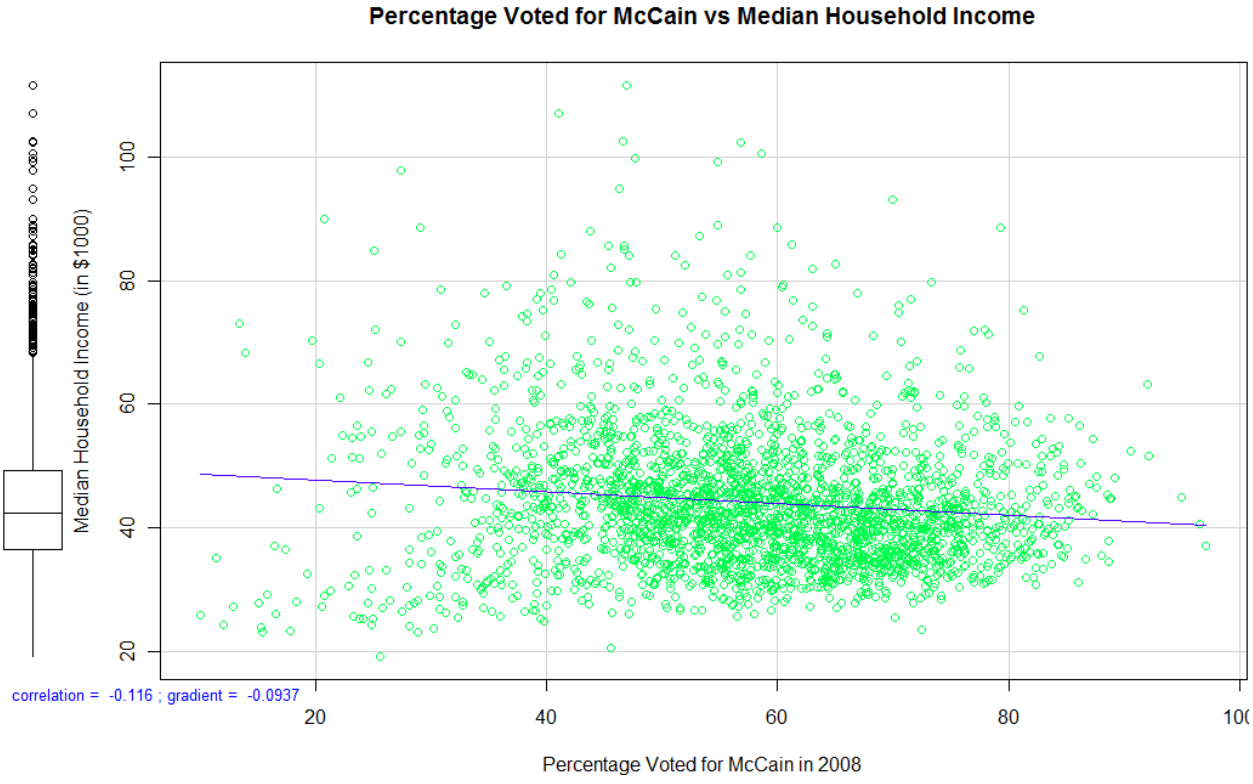
course there are exceptions like Bill Gates) is not entirely captured. Another conceptual issue that arose is that even data by county does not parse our data finely enough. A county may have a suburb with a lot of golf courses, and also an urban center elsewhere with few golf courses, but these two strikingly different areas are averaged into one data point. Also, a county may have the vast majority of its population in an urban center that votes liberal, but because of the size of the county and vast amounts of rural area, the population density is drastically reduced, and we end up with a data point that is semi-sparsely populated but still votes liberal. Regardless, ignoring all such issues our data still gave us a lot to work off of.

III. Trends We Saw and What it Means

We created a combination of scatter plots, histograms and maps. The maps geographically displayed our data, but from our more conventional plots, a couple of other important trends emerged.

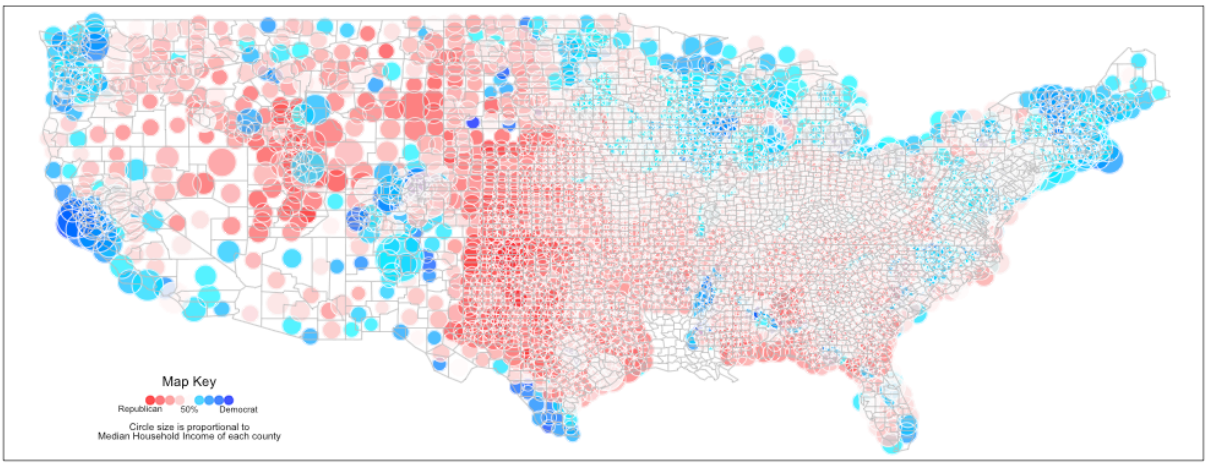



County Election Results By Income
McCain
Obama
Election Results(in percentage)
0.0
0.2
0.4
0.6
0.8
1.0
78%
21%
71%
29%
56%
44%
52%
48%
33%
67%
0-40
40-60
60-80
80-100
100-120
Income Group(in 1000)


Percentage Voted for McCain vs Median Household Income
Median Household Income (in $1000)
20
40
60
80
100
correlation = -0.116 ; gradient = -0.0937
20
40
60
80
Percentage Voted for McCain in 2008

We started with a set of histograms that show median county income and the frequency a county of that sort voted for a specific candidate. We supplemented this with a scatter plot, so show the overall trend. What the histograms show, contrary to initial intuition, is that poorer counties tend to vote for McCain, and only in the wealthiest of counties do we see Obama getting more of the counties. But, there a couple of things we need to do to properly interpret this data and apply it to our initial intuition. When we think poverty, we think about inner city districts, and not about rural poverty. The more conservative base that lives far from the coasts sees a lower cost of living, so less income for them may mean a higher standard of living. Also, just because a lot of the lower income counties vote for McCain does not mean that the wealthiest in American society do not; it need not be such a linear trend. Also, as previously mentioned, when we say the rich tend to vote conservative, we mean the majority of the top 5%, but in every county, this small subset of people hardly influences the median income, and thus our data really does not identify this set of people. There is also the issue of wealth; old money tends to be more conservative than new money, but median income only looks at new money and not the accumulation of wealth. So, in order to provide a more geographical perspective, we put this data onto a map.

Median Household Income and 2008 U.S. Presidential Election



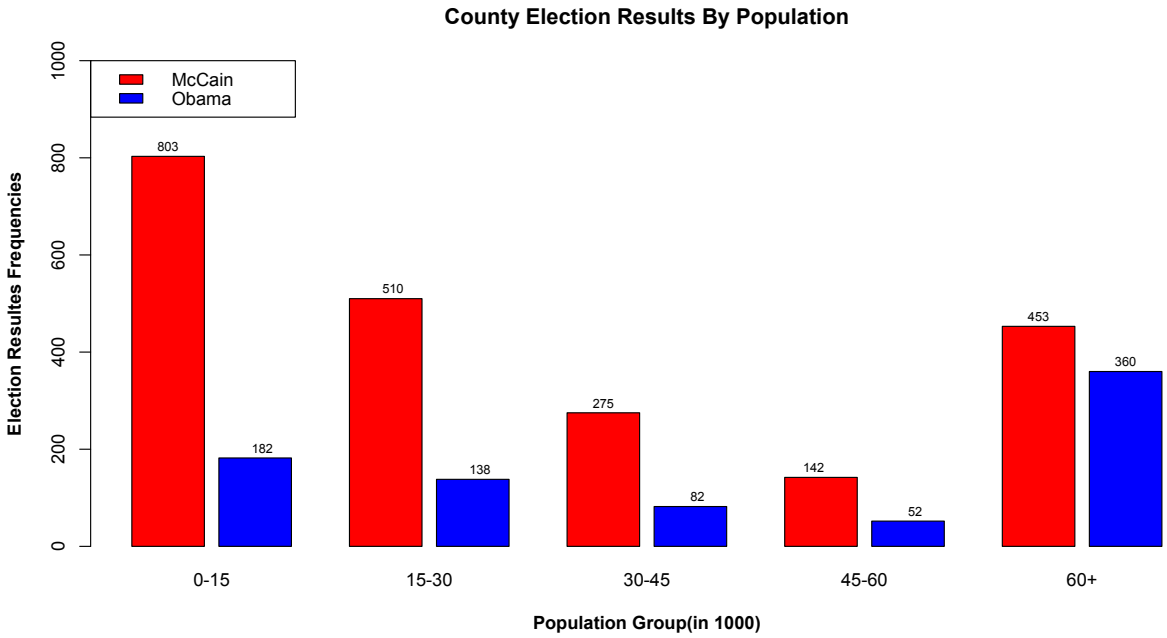
This graph shows median income by county. The color of the dots represents the % who voted for McCain or Obama. Dark blue means very Obama and dark red means very McCain. The circle sizes are proportional to the median household income in each county. Because median income is so similar across counties, the map shows us little in that regard, but it does give us a couple of standards by which we can assume later. Some of the trends include:

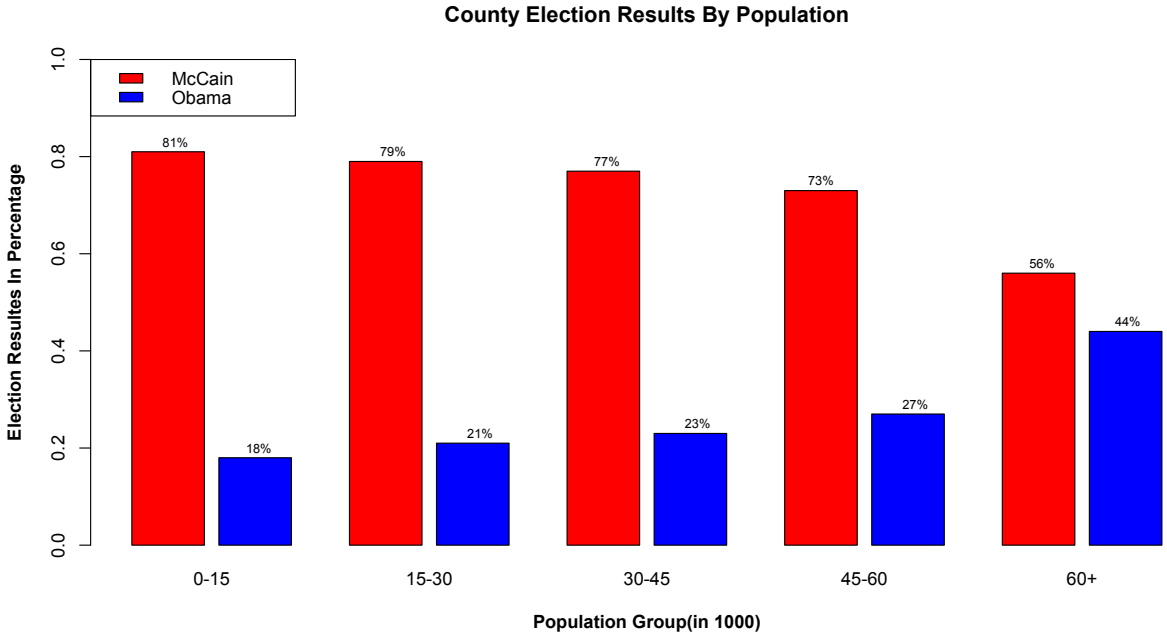
1. Coastal counties tend to have higher median income
2. Coastal counties tend to vote more for Obama than does the average county
3. Median income is similar across the United States, at least in terms of order of magnitude
4. Most of the counties tended to vote for McCain, even though Obama crushed him in the election

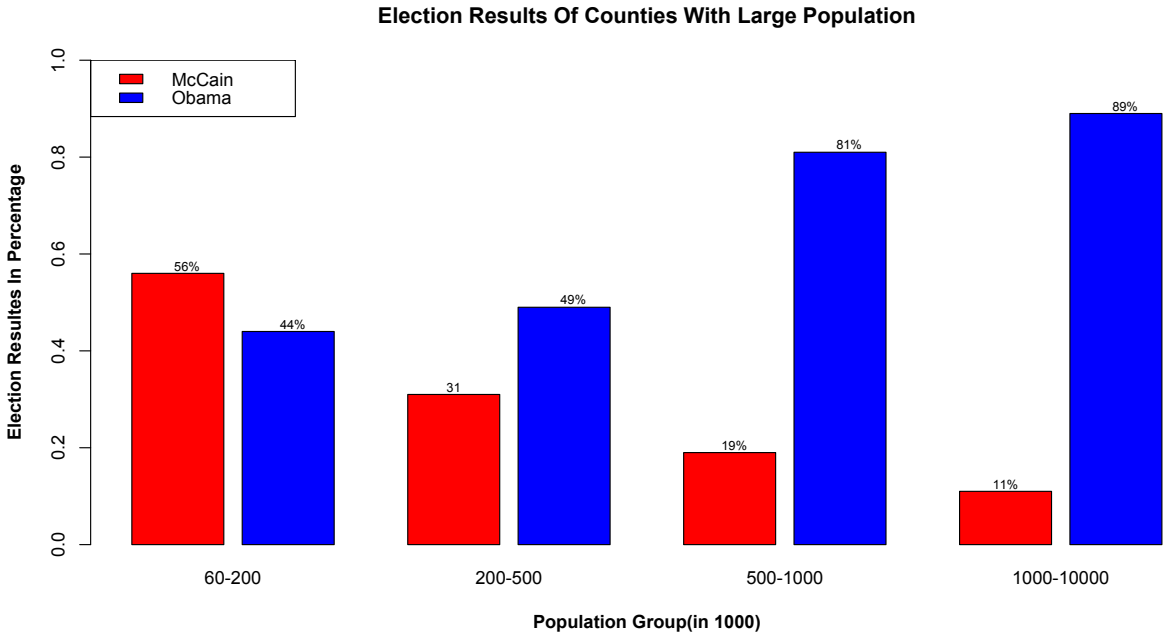
To offer a possible quick explanation for each:

1. The standard of living is higher on the coasts and there is more demand to live here (better climate, convenience, etc.)
2. Vast urban centers tend to be located near coasts, and urban centers vote liberal

3. While there may be rich communities throughout the United States, counties are generally large enough that these relatively small communities only marginally affect median incomes.
4. This is the curious trend.



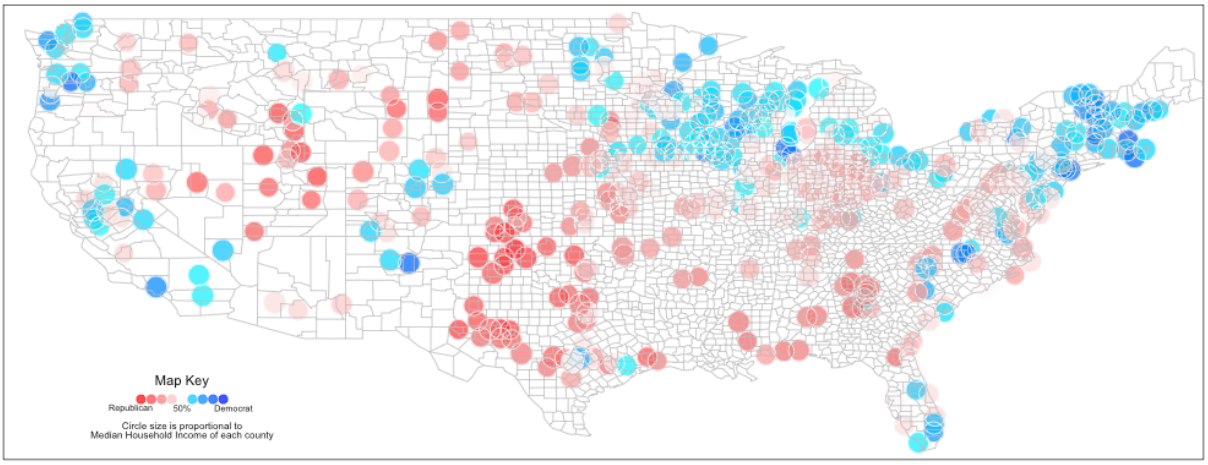




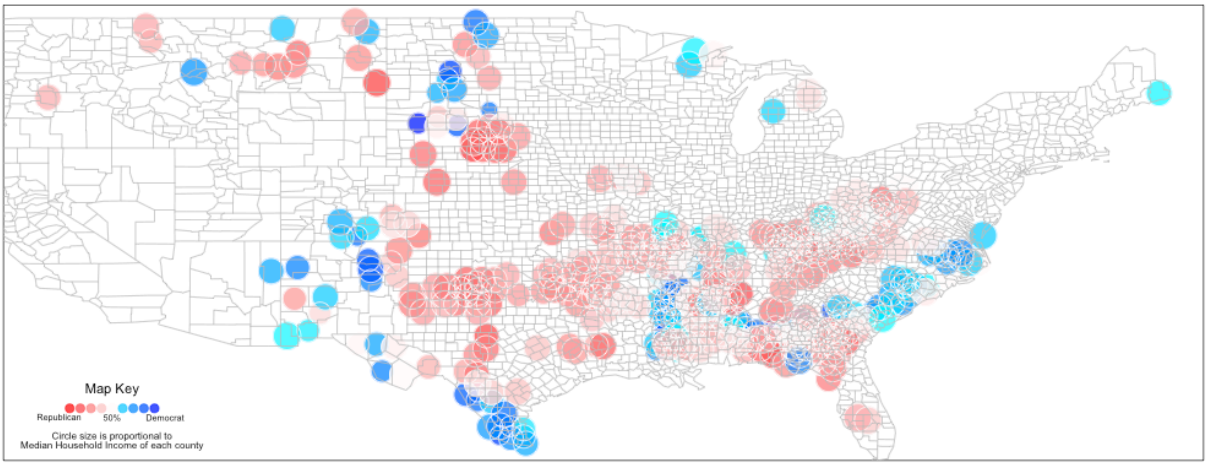
We decided to explore the fourth trend further with the above bar graphs. How can most of the counties vote for McCain, and yet Obama got way more electoral votes? The reason is that on this income map, each county, no matter the population, "looks" identical. What happened is the most populated counties voted overwhelmingly for Obama, causing the popular vote within each state to be dominated by Obama supporters. How can we draw that from this bar graph? Looking at the first of the three above bar graphs, if we look at the most sparsely populated counties, we see McCain won more than 4 times as many as Obama. As we move to the most populated counties, we see that Obama won almost as many as McCain. We figured that a better way to see a trend in this data would be to say, of all counties in this population range, what % went for McCain and what % went for Obama. There is a very clear trend that as population rises, Obama wins a larger % of them. Because of this strong trend, we decided to keep going, and saw in the third bar graph that the hugely populated counties, those with hundreds of thousands to millions of citizens, tended to generally vote for Obama. So, while

everything is one data point, these are weighted much higher in the general election. Clearly, from the map and from this histogram, McCain won most of the counties, but we know from the results that Obama won the popular and electoral votes. Before exploring the population trends, we decided to look at the richest and poorest 500 counties.

Richest 500 counties and 2008 U.S. Presidential Election

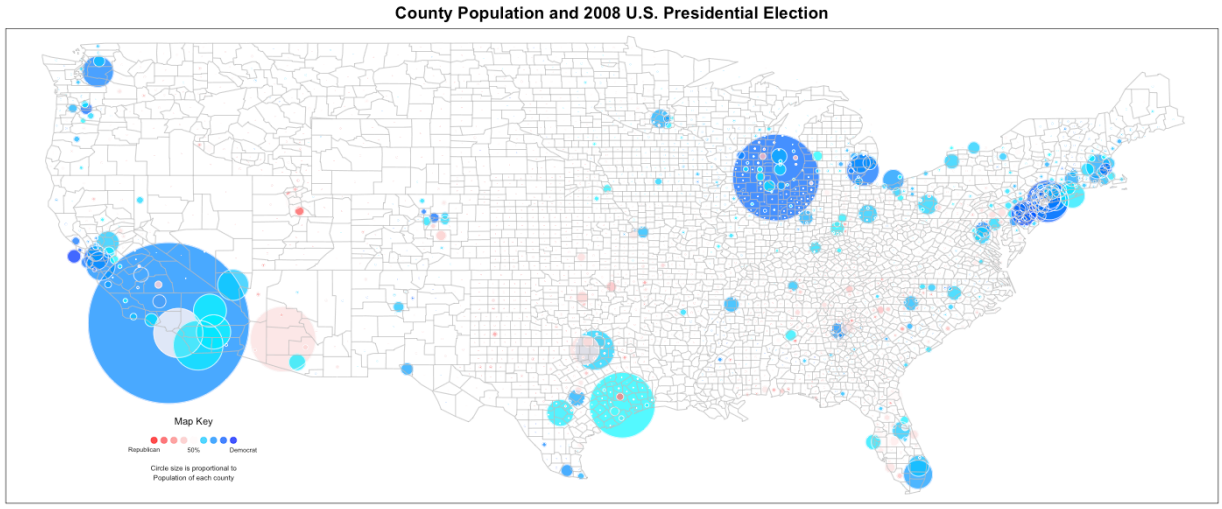


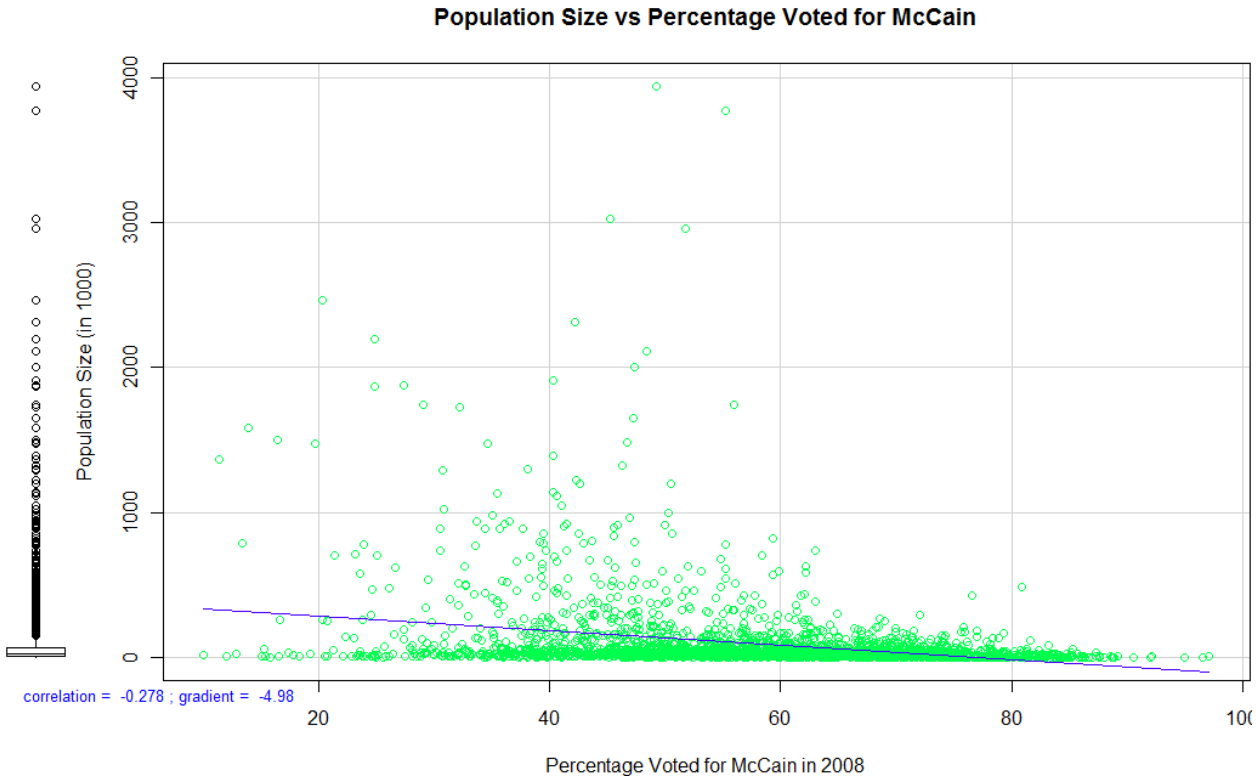
Poorest 500 counties and 2008 U.S. Presidential Election



In these two graphs, we see a couple of trends, and a little bit in terms of how rich and poor counties tend to vote. Since McCain won way more counties, it is expected that both of these graphs would be dominated by red. However, it is very clear that there is far more blue in the rich 500 counties than there is in the poor 500 counties. Some of the trends that we see,

however, are more geographical. This confirms the idea that coastal counties tend to have higher median income. Also, this shows that the poorest part of our nation is in the "south." The poverty seems a lot more concentrated; a lot of states have only a few, or even no, poor counties. However, the distribution of states with rich counties is a lot more even. As a caveat, we can see that the wealth is really concentrated in three areas: New England, California and near the Great Lakes. Ironically, the 3 financial centers of the US are New York, San Francisco and Chicago, which are surrounded by wealthy counties. Regardless, knowing that giving equal weight to each county skews our data, we decided to look at population trends.

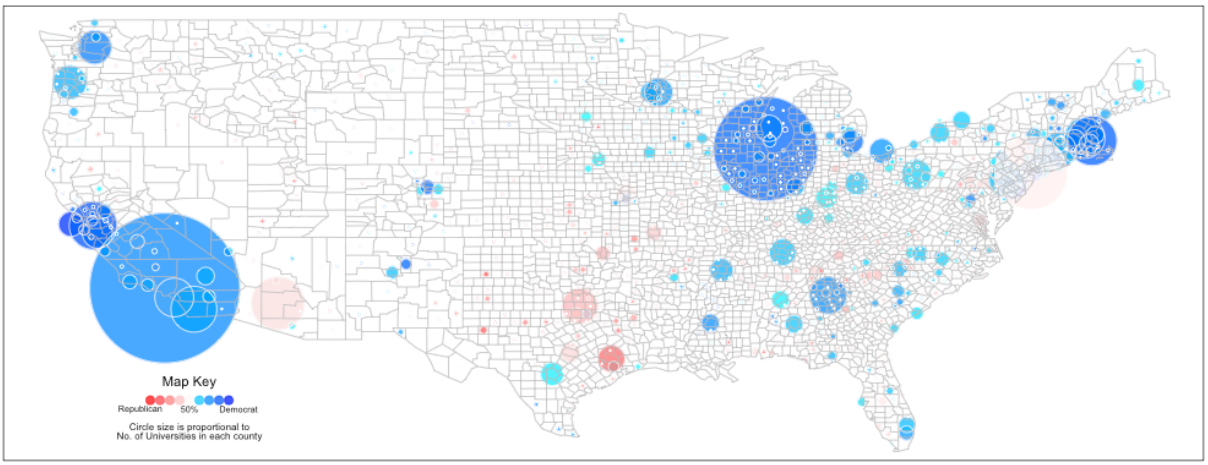




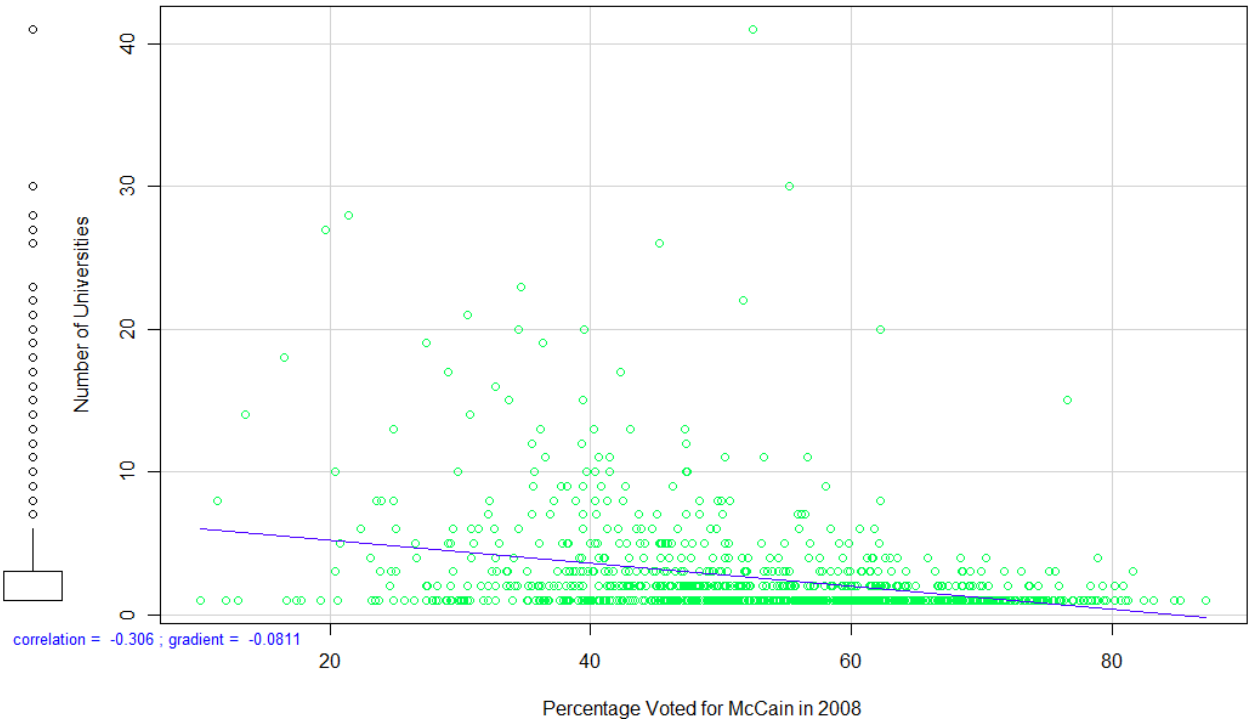
And with this it becomes painfully clear why, even though McCain won most of the land area of the United States, Obama could dominate in the overall election. This map is dominated by blue, indicating a couple of things. First off, if you sum the area of the circles in each state, you get their electoral votes contributed to electing the president. Of course, there is the issue of senators, so small states get a slightly higher representation, which actually allows conservatives

to win more often than the popular vote says they should, but we will ignore this fact. So, any state with a large dot, namely California, Illinois, Texas (which actually votes conservative), New York, Arizona, Florida and Michigan is worth a lot of points. If we look at this map to get a rough sense of their popular vote, we see blue domination, indicating that not only should Obama have won most of these states, but also that these states contribute the most to a candidates electability. Of course, there are forces pulling against this visualization. For example, Harris County, TX is very light blue, meaning Obama barely won it. This means that a series of less-populated counties that are predominantly conservative could reverse the trend in TX, which they did. However, for Los Angeles, it is a much darker blue, so it be essentially impossible for smaller counties to reverse its trend (and Oakland, San Francisco, etc. certainly did not help McCain's cause). The scatter plot shows a general trend; as population increases, % votes for McCain decrease. What is interesting about this scatter plot is that from it, it is possible to figure out exactly how many people voted for McCain and for Obama. Just multiply the percentage by each value on the y-axis and sum them to get % for McCain; do (1 – %) and sum for Obama.

Number of Universities and 2008 U.S. Presidential Election



Number of Universities vs Percentage Voted for McCain



Finally, we looked at number of universities per county and analyzed such trends. Again, we found what we expected, though the trend was slightly less powerful than we thought. Again, this map is dominated by blue, but arguably less so than the population map. Also, a lot of the same counties with high population had a lot of universities, which makes sense. Regardless, it seems like catering a campaign to the few but densely populated urban centers may be a good

strategy to win an election, something that democrats typically appeal far more to than do republicans. Yet again, the scatter plot shows this negative trend. While we admit that both of this slope and correlation is coming from outliers, these outliers are extremely important, because it is the few "outlier" cities that determine the outcome of the entire national election.

In creating scatter plots for most of this data, we found that the histograms offered a better form of visualization. We considered removing the outliers from the scatter plots, but we realized that these data points were some of the most critical of our entire data set. It is the outliers that caused Obama to win the election. Generally, we think of an outlier as something "weird," but there are concrete social explanations that justify these trends for this context, and thus we could not throw them out. Only in the case of median income were there no outliers, because median income is a measure of central tendency, rather than a sum of variables.

IV. Explanation of Methods

Walking through the code:

Most of the issues we encountered had to do with making sure the county and state names exactly matched those in the R package, which we used to plot all of our data on the map. The issue was that every source formatted the names differently, making spaces, dashes, capitalizing certain letters, abbreviating some counties, including additional cities, leaving out counties, etc. It became a mess and caused us to lose a couple hundred data points, but we still salvaged the vast majority of the data.

1. Getting the county names as stored in R
    a. Download the maps package and load it
    b. county.names1 pulls a vector of state,county in all lowercase

c. state.county splits this vector into a data frame with two columns

2. Getting and formatting the university data
    a. Look at the http://www.univsource.com/county-wise-list.htm and notice how they format state and county names; get the names in R into this form
    b. Generate all ~3000 urls
    c. Look through all of these urls and see if they exist or not (county.universityTF); this is what takes an hour. If the link does not exist, the county has no university, so it will get a final value of 0.
    d. Looking at the source code for links that do exist, each time regexp.university is matched, there is 1 university, so every working link has at least 1 match
    e. Count how many times it appears in each working link and fill that vector element with this number; if the link does not exist, put 0
    f. Aggregate all of the data and write it to "university.csv".
3. Setting directory and obtaining the files
    a. First create a directory and begin saving everything in that directory.
    b. Download all files and title them identically to how they are in the first section of the code
    c. Because the university data takes an hour to pull from the website with code commented out, we pulled it and saved it in "university.csv".
4. Getting and formatting income data
    a. Create an index for the lines where the data exists
    b. Merge it with the county

c. Look at the two letter state abbreviation and match it to a state, then count how many times each element of the state vector must be replicated to match the state with the county. We need to do this because multiple states have counties with the same name, so the state is needed to make the point unique and unambiguous

d. Match everything together to get a data frame with state,county and median income

5. Getting and formatting the election data
    a. Edit the formatting of the counties to make them lower case
    b. Match these new state names, which match those in R, to the voting data
6. Getting and formatting the population data
    a. Again, make the state names match those in R
    b. Add the population data vector into the state names in the form of a data frame
7. Getting and formatting the coordinates
    a. As explained previously, again, match the county names, format the names properly, and match the coordinate data
8. Getting and formatting the golf course data
    a. Ultimately, we realized that this data set was horribly incomplete, so none of this code was used in our final project.
    b. There are 16 pages of data; the first is: http://www.courseiq.net/golf-course-list.aspx?
    c. The other 15 are formatting differently, so we generate the other 15 names using a for loop
    d. Generate two blank lists to store the data in

e. Go over all 16 pages via a for loop and extract the data
    i. Read the lines of the page
    ii. Figure out which lines of the source code a specific pattern occurs on
    iii. Pull out the county names and state names where the counties exist
    iv. Since some lines have multiple counties, split it by the "/" delineation
    v. Figure out how many counties are on each line and replicate the state that was on the line of the site that many times, so when the counties are unlisted, the length of the state vector will be the same length, and the state names will still match the proper counties by position
    vi. Make a lot of edits to the county names via regular expressions to match those in R
    vii. Finally, store everything in a list
f. Bind all of the lists together, figure out how many unique courses there are and see how many times each of these unique entries appears in the list
g. Find what position the county linked with the number of golf courses in that county is in the R maps vector
h. Using a composition of function, make a data frame using the following information: vector of all counties in R, vector of the position of the county linked with the number of golf courses in the R vector, the number of golf courses per unique county. If no value is assigned, the value is 0

9. Putting it all into one
    a. Merge everything, after having done our best to make sure every data point we have has a county name that will match

b. After merging everything we still had about 90% or more of our data; not every data set was complete, so we did lose some points

10. Creating Map Plots

a. Write code for the color representation in rgb format of each county according to its political standing. More percentage votes for McCain will show more intense red and more percentage votes for Obama will show more intense blue color.

b. To visualize the Median Household Income with Election results for each county, order the income from lowest to highest. Using the plot() function, plot the longitude and latitude of as the horizontal and vertical axes respectively, using the color gradient coded above. The size of the circles was scaled to be proportional to their relative sizes in income. Circles of white borders were added to help distinguish overlapping circles. Then add county map on top to give geographic sense to the data.

c. Because of the large data sets, the amount of overlapping of circles was too large. Hence, the 500 counties with the highest and lowest income were respectively plotted subsequently for further analysis.

d. Similarly, to visualize the number of universities and population with Election results, we plotted the points with the relative size of circles to be proportional to the relative sizes in number of universities and county population. The county map was layered on top.

11. Creating Scatter Plots

a. Extract election, population, university, university density and income data from the complete data set. The population and income data are divided by 1000 so that they are scaled appropriately for plotting purposes.
b. Scatter plot using the data extracted. Remove the outliers by looking at these plots and setting a data range to remove the extreme points.
c. Compute correlation values between each pair of variables.
d. Compute the regression line gradients by the formula covariance(X,Y)/variance(X).
e. Using the car package and the scatterplot function, we create scatter plots for each pair of variables. We also include the correlation and gradient information at the bottom margin of the plots.
f. Then, to further investigate the trends, exclude counties with no universities and repeat the entire process.

12. Creating the Bar Plots
    a. Explanation within code

A large majority of our time for creating the code went into scraping the golf course data. Because it was so disjoint on the website, we could not tell that it was incomplete until we scraped it. We encountered all sorts of problems. To name some:

1. Urls were not named systematically
2. Counties were named not only differently from that of R, but they were also named differently in non-systematic ways
3. Sometimes the names used as locations were not even counties; they could be cities

4. A variable number of counties would be on each line of code; generally it was one, but sometimes we found more
5. It turned out to be selected golf courses, based on essentially nothing of value

Although election results, population and median household income data were readily available, we had to wrestle through many sleepless nights to web scrape the golf course data. First, we had difficulty finding a comprehensive data-set that contained golf courses by county. We had multiple sources with golf course directories by county or by city, but the comprehensive data that contained nationwide information was rare. After modifying and scrolling through many search engine pages, we found a website that supposedly contained all golf courses by county. The name of the website is "CourseIQ" and it specializes in golf game strategies by providing previews of the courses. Its mission is to "produce virtual course flyovers with integrated community message boards that together provide the ideal platform for learning and sharing information about how to play a course" (http://www.courseiq.net/Default.aspx). Under the list of golf courses, we found sixteen pages worth of data that included the course name and its location. Deceived by the sheer volume of the golf course data, we granted complete credibility to the website. Contrary to our belief, the website only offers data on golf courses that it has fly over video data for. However, we failed to recognize the incompleteness of data and we continued with the web scraping process.

The process of transforming sixteen web pages to workable data in R was difficult and complex. Each website page, totaling up to 16, has a list of n counties and each list of n counties has one state attached to it. Initially, in order to read it into R, we first created a list that holds all 16 pages of source codes. Once we have read the source pages, we had to use regular expressions to find and filter out the necessary lines that had information on county and state. Simultaneously,

we matched these counties with the county map in the maps package. Afterwards, we looped through all 16 pages to pull out state and county names of each code. Once we got the necessary data from all 16 pages, we bounded them together. We then had to format so that it's formatting matches the names given in the R maps. For example, we had to get rid of all quotations and capital letters. But the most difficult part was to separate chain golf courses that existed in multiple counties. Once the data was all set, we looked for the frequency of the county to look for number of golf courses. With this unique set of data, we matched it with unique county position in R list so that in the end we had number of golf courses by county.

However, the problem rose to the surface when we plotted the data. We created scatter plots using data on income and number of golf courses. However, we found no relevant correlation or trend, because we were missing a large part of the golf course data. In addition, it was impossible for us to find any correlation between voting trends and golf courses. In fact, when we plotted the final data on the map of the U.S, the missing golf courses became very apparent. To our dismay, we had to throw away the golf course data which we spent most of our time crafting.

V. Conclusion

To start, our group learned that we should pay much more attention to our data sources before trusting them. For us, the major issue was dealing with how to get trusted data, complete data and how to compile it in a systematic way that would allow us to visualize the large data sets.

This research also teaches us that we have to be very careful that we understand what data we are looking at. For example, while we still believe the wealthiest Americans tend to vote more conservatively, our data maintains that counties with the highest income vote for liberally more frequently. If we think a bit, we realize that these two things are entirely different. Wealth is completely different from income. Also, when we say wealthiest, we mean a small subset at the top of the economic ladder, data which is eliminated in measures of central tendency (we used median). Also, we have the subtlety of the weighting factor by county. What we are interested is electoral votes. Because electoral votes are determined solely by population, something not reflected on a geographically realistic map, looking at land area is a poor measure. When we converted to a more population-based map from the income by county map, it became obvious that Obama should have won 2008's election. This was also reflected in the university map, which has a strong correlation with population (more people means more universities). Finally, our arguably most interesting trend came out of the bar graph showing population of counties and frequency. At a glance, all of the red bars are higher, so one may conclude that McCain won the election. However, looking at it more closely, we notice that the proportion of counties that went for Obama increases as the population gets higher. When we look at bars in the hundreds of thousands of population, we see that Obama begins to dominate. Combining this with the fact that not only should these be weighted far more, but also that each state is an all or nothing deal for each candidate, we reverse our conclusion and say that it is obvious Obama should have won, according to such a graph. So with that, we have a little more insight into the republican and democratic base, and on what they must focus in order to tilt the popular vote in their favor.

VII. Appendix - codes

```
## Change to different directory
setwd("")

#####################################################################
#################### County Names  ##################################
#####################################################################
library(maps)

county.names1 = map('county')$names

data(countyMapEnv)
all.names <- map('county',namesonly=T)
all.names1 <- data.frame(mapm=all.names, col=NA,ordin=1:length(all.names))
all.names2 = all.names1$mapm

regexp.county = "^.*,(.*$)"
county.names = gsub(regexp.county, "\\1", all.names2)

regexp.state = "(^.*),.*$"
state.names = gsub(regexp.state, "\\1", all.names2)

state.county = data.frame(state.names, county.names)

#####################################################################
######## Create and Save file for University Data  ##############
#####################################################################

##state2 = gsub(pattern = "[[:blank:]]", replacement = "-", x = state.county[,1])
##county2 = gsub(pattern = "[[:blank:]]", replacement = "-", x = state.county[,2])

##url.university = rep(NA,nrow(state.county))

##for(i in 1:nrow(state.county))
##{
##url.university[i]           =           paste("http://www.univsource.com/county-wise-
#list/",state2[i],"/",county2[i],"-county-colleges-universities.htm",sep = "")
##}

##regexp.university                                                                    =
"[[:blank:]][[:blank:]][[:blank:]][[:blank:]]<td[[:blank:]]width=.50%.><a[[:blank:]]hr
ef=.\\.\\./\\.\\./"

##county.universityTF = rep(NA,length(url.university))

##for (i in 1:length(url.university))
##{
##county.universityTF[i]           =           class(try(readLines(url.university[i]),
```

```
silent=TRUE))=="try-error"
##}

##county.university = rep(NA,length(url.university))

##for (i in 1:length(url.university))
##{
##        if    (county.universityTF[i]    ==    FALSE)    county.university[i]    =
length(grep(regexp.university,readLines(url.university[i]))) else county.university[i]
= 0
##}

##state.county2 = paste(state.county[,1], state.county[,2], sep =",")
##county.university = data.frame("Location" = state.county2, "University Count" =
county.university)

##write.table(county.university,                          file                          =
"~/Documents/Berkeley.S5/Classes/Stat.133/Homework/Final.Project/university2500.csv",
sep = ",", row.names = FALSE)

#####################################################################
#################### All Files ######################################
#####################################################################
election08 = read.csv(file = "election08.csv")
election = election08[c("State", "County", "McCain..", "Obama..")]
##       from       http://www-958.ibm.com/software/data/cognos/manyeyes/datasets/2008-
presidential-election-results/versions/1.txt

income <- read.csv(file= "est08ALL.csv")
##                  from                  top                  link                  at
http://www.census.gov/did/www/saipe/data/statecounty/data/2008.html

population08 <- read.csv(file = "population.csv")
##               from               internet:               population08               <-
read.csv("http://www.census.gov/popest/data/intercensal/county/files/CO-EST00INT-
TOT.csv")
population <- population08[c("STNAME","CTYNAME","POPESTIMATE2008")]

coord.table = read.csv("countycoord.csv")
## from http://www.census.gov/geo/www/gazetteer/files/Gaz_places_national.txt

university = read.csv("university.csv")
## created above, but the code takes an hour to run

#####################################################################
#################### Income Data  ###################################
#####################################################################

index.inc = grep(pattern = ".*[[:blank:]]County|Boroush|Area|Municipality|Parish", x =
income[,2])
income1 = income[index.inc,]
county.inc = gsub(pattern = "(.*)[[:blank:]]County|Boroush|Area|Municipality|Parish",
replacement = "\\1", x = income1[,2])
```

```
StateAbr = read.csv(file = "StateAbr.csv", header = F) #########MUST EDIT PATH FOR
YOUR COMPUTER
abr.factor = lapply(income1[,1], factor, levels = StateAbr[,2])
abr.count = sapply(abr.factor, tabulate, nbins = length(StateAbr[,2]))
total.count = apply(abr.count, 1, sum)
county.count = rep(StateAbr[,1], total.count)

income2.state.county = paste(sapply(county.count, tolower), sapply(county.inc,
tolower), sep =",")
income2 = data.frame("Location" = income2.state.county, "Median Household Income" =
income1[,3])

#####################################################################
#################### Election Data  ##############################
#####################################################################

election2.state.county = paste(sapply(election[,1], tolower), sapply(election[,2],
tolower), sep =",")
election2 = data.frame("Location" = election2.state.county, "McCain.." = election[,3],
"Obama.." = election[,4])

#####################################################################
#################### Population Data  ##############################
#####################################################################

index.pop = grep(pattern = ".*[[:blank:]]County|Boroush|Area|Municipality|Parish", x =
population[,2])
population1 = population[index.pop,]
county.pop = gsub(pattern = "(.*)[[:blank:]]County|Boroush|Area|Municipality|Parish",
replacement = "\\1", x = population1[,2])

population2.state.county = paste(sapply(population1[,1], tolower), sapply(county.pop,
tolower), sep =",")
population2 = data.frame("Location" = population2.state.county, "Pop Estimate" =
population1[,3])

#####################################################################
#############  County Coordinates Data  #############################
#####################################################################

index.dc = coord.table[,1] == "DC"
index.pr = coord.table[,1] == "PR"
index.dc.pr = index.dc + index.pr
coord = coord.table[!index.dc.pr,c(1, 4, 9, 10)]

county.coord = gsub(pattern = "(.*)[[:blank:]]County|Boroush|Area|Municipality|Parish",
replacement = "\\1", x = coord[,2])
abr.factor.coord = lapply(coord[,1], factor, levels = StateAbr[,2])
abr.count.coord = sapply(abr.factor.coord, tabulate, nbins = length(StateAbr[,2]))
total.count.coord = apply(abr.count.coord, 1, sum)
county.count.coord = rep(StateAbr[,1], total.count.coord)
```

```
coord.state.county = paste(sapply(county.count.coord, tolower), sapply(county.coord,
tolower), sep =",")
coord2 = data.frame("Location" = coord.state.county, "Lat" = coord[,3], "Long" =
coord[,4])

#####################################################################
#################### Golf Course Data ###############################
#####################################################################

h = 1:16

urls = rep(NA,length(h))

urls[1] = "http://www.courseiq.net/golf-course-list.aspx?"
for (i in 2:length(h))
{
  urls[i]   =   paste("http://www.courseiq.net/golf-course-list.aspx?st=date&pg=",h[i-
1],sep = "")
}
urls

l = replicate(length(h), list())
g = replicate(length(h), list())

for (i in 1:length(h))
{

pg = readLines(urls[i])

l[[i]] = pg

index = grep("list[Alt]*BodyTextMedBold", pg)
index2 = index+2
index3 = index+3
golf1 = pg[index2]
golf2 = pg[index3]

golf.c = gsub(pattern = "<.*>(.*)</td>", replacement = "\\1", x = golf1)
golf.s = gsub(pattern = "<.*>(.*)</td>", replacement = "\\1", x = golf2)
golf.county = gsub(pattern = "^[[:blank:]]+(.*)$", replacement = "\\1", x = golf.c)
golf.s1 = gsub(pattern = "^[[:blank:]]+(.*)$", replacement = "\\1", x = golf.s)
golf.state = unlist(lapply(golf.s1, tolower))

golf.county1 = strsplit(golf.county,"\\/")

x = rep(NA,length(golf.county1))

for (j in 1:length(golf.county1))
{
  x[j] = length(golf.county1[[j]])
}

x = as.numeric(x)
```

```
y = rep(NA,length(x))

for (j in 1:length(golf.state))
{
  y[j] = list(rep(golf.state[j], x[j]))
}

golf.county1.5 = unlist(golf.county1)

state.county1.5 = unlist(y)

regexp.names = "^ ?([[:upper:]].*)"
golf.county2 = gsub(regexp.names, "\\1", golf.county1.5)

regexp.names1 = "(^[[:upper:]].*) $"
golf.county3 = gsub(regexp.names1, "\\1", golf.county2)

regexp.names2 = "(.*) County"
golf.county4 = gsub(regexp.names2, "\\1", golf.county3)

regexp.names3 = "(.*) Counties"
golf.county5 = gsub(regexp.names3, "\\1", golf.county4)

regexp.names3 = "(.*) Counties"
golf.county5 = gsub(regexp.names3, "\\1", golf.county4)

golf.county6 = lapply(golf.county5, tolower)

golf.county7 = unlist(golf.county6)

golf.county8 = gsub('[[:punct:]]', "", golf.county7)

g[[i]] = data.frame(state.county1.5, golf.county8)

}

golf.courses                                                                    =
rbind(g[[1]],g[[2]],g[[3]],g[[4]],g[[5]],g[[6]],g[[7]],g[[8]],g[[9]],g[[10]],g[[11]],g
[[12]],g[[13]],g[[14]],g[[15]],g[[16]])

golf.courses1 = paste(golf.courses$state.county1.5,golf.courses$golf.county8,sep = ",")

unique.golf.courses = unique(golf.courses1)
length(unique(golf.courses1))

freq.golf.courses = rep(NA, length(unique(golf.courses1)))

golf.course.table = table(golf.courses1)

for (i in 1:length(unique(golf.courses1)))
{
  freq.golf.courses[i] = golf.course.table[[i]]
```

```
}

unique.golf.courses.TF = is.element(unique.golf.courses,county.names1)

unique.golf.courses.TF1 = grep("TRUE",unique.golf.courses.TF)

unique.golf.courses1 = unique.golf.courses[unique.golf.courses.TF1]

golf.course.position = rep(NA,length(unique.golf.courses1))

for (i in 1:length(unique.golf.courses1))
{
     golf.course.position[i] = grep(unique.golf.courses1[i],county.names1)
}

freq.golf.courses1 = freq.golf.courses[unique.golf.courses.TF1]

v = rep(0,length(county.names1))
v[golf.course.position] = freq.golf.courses1

w = data.frame("Location" = county.names1,"Golf" = v)

######################################################################
#########  Matching POP, INC, ELEC, COORD DATA  ######################
######################################################################
income.election = merge(election2, income2)
population.income.election = merge(population2, income.election)
lat.long.population.income.election = merge(coord2, population.income.election)
head(lat.long.population.income.election)
all.data <- merge(lat.long.population.income.election, university)

# Next we clean the data of Median Income and Percentage Votes
McCain.votes1 <- all.data[,"McCain.."]
McCain.votes2 <- gsub("%","",McCain.votes1)
McCain.votes <- as.numeric(gsub("\\.","",McCain.votes2))/10000

Obama.votes1 <- all.data[,"Obama.."]
Obama.votes2 <- gsub("%","",Obama.votes1)
Obama.votes <- as.numeric(gsub("\\.","",Obama.votes2))/10000

all.data$McCainVotes <- McCain.votes
all.data$ObamaVotes <- Obama.votes

incomeonly1 <- (all.data[,"Median.Household.Income"])
incomeonly <- as.numeric(gsub(",","",incomeonly1))
all.data$MedianIncome <- incomeonly
## all.data now has numeric MedianIncome, and Percentage Votes

######################################################################
#################### Codes for colors  ###############################
######################################################################
all.data$M.more <- as.numeric(all.data[, "McCainVotes"] >= 0.5)
all.data$M.color <- ((all.data[,"McCainVotes"] - 0.5)*2*all.data[,"M.more"])
```

```
all.data$O.more <- as.numeric(all.data[, "McCainVotes"] < 0.5)
all.data$O.color <- ((1-2*(all.data[,"McCainVotes"]))*all.data[,"O.more"])

a <- all.data$M.more
c <- all.data$McCainVotes
b <- all.data$ObamaVotes

g2 <- a*(c-0.5)*2
g <- rep(NA, 2671)
for (i in 1:length(g2)){
  if (g2[i] == 0) {
    g[i] <- 1- (b[i]-0.5)*2
  } else {
    g[i] <- 1- g2[i]
  }
}

b1 <- rep(NA, 2671)
for (i in 1:length(a)) {
  if (a[i] == 1) {
  b1[i] <- 1- a[i]*(c[i]-0.5)*2
} else {
  b1[i] <- 1
}

}

par(mar=c(0, 0, 0, 0))
map("county", fill = T, col = rgb(a, g, b1))

# a is all.data$M.more
all.data$g <- g
all.data$b1 <- b1

#########################################################
############# Map - Income/ Political ##################
#########################################################
ordered.income <- all.data[order(all.data$MedianIncome),]

## ordered data
size1 <- exp(ordered.income$MedianIncome/60000)
par(mar=c(1, 1, 2, 1))
plot(ordered.income$Long, ordered.income$Lat,
     xlab = "", ylab = "",
     xaxt = "n", yaxt = "n",
     col=rgb(ordered.income$M.more,ordered.income$g,ordered.income$b1, alpha = 0.8),
     pch = 19, cex = size1,
     main = "Median Household Income and 2008 U.S. Presidential Election")
points(ordered.income$Long, ordered.income$Lat,
       pch = 1, col ="white", cex = size1)
map("county", add = TRUE, col = "gray")
```

```
## for legend
max(all.data$McCainVotes)
#[1] 0.9712
colorgrad <- all.data[all.data$McCainVotes ==0.9712,]
colorgrad[2,] <- all.data[22,]
colorgrad[3,] <- all.data[23,]
colorgrad[4,] <- all.data[40,]
colorgrad[5,] <- all.data[18,]
colorgrad[6,] <- all.data[50,]
colorgrad[7,] <- all.data[172,]
colorgrad[8,] <- all.data[154,]
min(all.data$McCainVotes)
#[1] 0.1001
colorgrad[9,] <- all.data[all.data$McCainVotes ==0.1001,]
colorgrad$Lat <- rep(28,9)
colorgrad$Long <- seq(-119,-115, length=9)

text(x = -117, y = 29,
     "Map Key",
     cex=0.7)
points(colorgrad$Long, colorgrad$Lat,
       col=rgb(colorgrad$M.more,colorgrad$g,colorgrad$b1, alpha = 0.8),
       pch = 19, cex = 1)
text(x=-119.5, y = 27.5, "Republican", cex = 0.45)
text(x=-117, y = 27.5, "50%", cex = 0.45)
text(x=-114.5, y = 27.5, "Democrat", cex = 0.45)
text(x = -117, y = 26.5,
     "Circle size is proportional to",
     cex=0.5)
text(x = -117, y = 26,
     "Median Household Income of each county",
     cex=0.5)


## not ordered
par(mar=c(0, 0, 0, 0))
plot(all.data$Long, all.data$Lat,
     xlab = "", ylab = "",
     xaxt = "n", yaxt = "n",
     col=rgb(all.data$M.more,all.data$g,all.data$b1, alpha = 0.8),
     pch = 19, cex = size1)
points(all.data$Long, all.data$Lat,
       pch = 1, col ="white", cex= size1)
map("county", add = TRUE, col = "gray")

##The 'Rich 500'
rich.500 <- ordered.income[2172:2671,]
size2 <- exp(rich.500$MedianIncome/62000)
par(mar=c(1, 1, 2, 1))
plot(rich.500$Long, rich.500$Lat,
     xlab = "", ylab = "",
     xaxt = "n", yaxt = "n",
     col=rgb(rich.500$M.more,rich.500$g,rich.500$b1, alpha = 0.8),
```

```
     pch = 19, cex = size2,
     main = "Richest 500 counties and 2008 U.S. Presidential Election")
points(rich.500$Long, rich.500$Lat,
       pch = 1, col ="white", cex= size2)
map("county", add = TRUE, col = "gray")
text(x = -117, y = 29,
     "Map Key",
     cex=0.7)
points(colorgrad$Long, colorgrad$Lat,
       col=rgb(colorgrad$M.more,colorgrad$g,colorgrad$b1, alpha = 0.8),
       pch = 19, cex = 1)
text(x=-119.5, y = 27.5, "Republican", cex = 0.45)
text(x=-117, y = 27.5, "50%", cex = 0.45)
text(x=-114.5, y = 27.5, "Democrat", cex = 0.45)
text(x = -117, y = 26.5,
     "Circle size is proportional to",
     cex=0.5)
text(x = -117, y = 26,
     "Median Household Income of each county",
     cex=0.5)
### Can't exactly tell any trend regarding income and political standing
### at most we can say where rich democratic supporters are geographcially (on East
and West coasts) and Republican supportors are

##The 'Poor 500'
poor.500 <- ordered.income[1:500,]
size3 <- exp(poor.500$MedianIncome/30000)
par(mar=c(1, 1, 2, 1))
plot(poor.500$Long, poor.500$Lat,
     xlab = "", ylab = "",
     xaxt = "n", yaxt = "n",
     col=rgb(poor.500$M.more,poor.500$g,poor.500$b1, alpha = 0.8),
     pch = 19, cex = size3,
     main = "Poorest 500 counties and 2008 U.S. Presidential Election")
points(poor.500$Long, poor.500$Lat,
       pch = 1, col ="white", cex= size3)
map("county", add = TRUE, col = "gray")
text(x = -117, y = 29,
     "Map Key",
     cex=0.7)
points(colorgrad$Long, colorgrad$Lat,
       col=rgb(colorgrad$M.more,colorgrad$g,colorgrad$b1, alpha = 0.8),
       pch = 19, cex = 1)
text(x=-119.5, y = 27.5, "Republican", cex = 0.45)
text(x=-117, y = 27.5, "50%", cex = 0.45)
text(x=-114.5, y = 27.5, "Democrat", cex = 0.45)
text(x = -117, y = 26.5,
     "Circle size is proportional to",
     cex=0.5)
text(x = -117, y = 26,
     "Median Household Income of each county",
     cex=0.5)
## Maybe we can tell where the poor democratic/ republican supporters are
```

```
##############################################
### Map - University/ Political ##############
##############################################
## ordered data
ordered.uni <- all.data[order(all.data$University.Count),]
par(mar=c(1, 1, 2, 1))
plot(ordered.uni$Long, ordered.uni$Lat,
     xlab = "", ylab = "",
     xaxt = "n", yaxt = "n",
     col=rgb(ordered.uni$M.more,ordered.uni$g,ordered.uni$b1, alpha = 0.8),
     pch = 19, cex = ordered.uni$University.Count/5,
     main = "Number of Universities and 2008 U.S. Presidential Election")
points(ordered.uni$Long, ordered.uni$Lat,
       pch = 1, col ="white", cex= ordered.uni$University.Count/5)
map("county", add = TRUE, col = "gray")
text(x = -117, y = 29,
     "Map Key",
     cex=0.7)
points(colorgrad$Long, colorgrad$Lat,
       col=rgb(colorgrad$M.more,colorgrad$g,colorgrad$b1, alpha = 0.8),
       pch = 19, cex = 1)
text(x=-119.5, y = 27.5, "Republican", cex = 0.45)
text(x=-117, y = 27.5, "50%", cex = 0.45)
text(x=-114.5, y = 27.5, "Democrat", cex = 0.45)
text(x = -117, y = 26.5,
     "Circle size is proportional to",
     cex=0.5)
text(x = -117, y = 26,
     "No. of Universities in each county",
     cex=0.5)
## good! more universities ~ democratic

## 'sanity check'
map.text('county', 'massachusetts', add=T)
map.text('state', 'new york', add=T)

## not ordered version does not have white borders of the circle, see which one you
guys like
par(mar=c(1, 1, 2, 1))
plot(all.data$Long, all.data$Lat,
     xlab = "", ylab = "",
     xaxt = "n", yaxt = "n",
     col=rgb(all.data$M.more,all.data$g,all.data$b1, alpha = 0.8),
     pch = 19, cex = all.data$University.Count/5,
     main = "Number of Universities and 2008 U.S. Presidential Elections")
points(all.data$Long, all.data$Lat,
       pch = 1, col ="white", cex= oall.data$University.Count/5)
map("county", add = TRUE, col = "gray")
text(x = -117, y = 29,
     "Map Key",
     cex=0.7)
points(colorgrad$Long, colorgrad$Lat,
```

```
       col=rgb(colorgrad$M.more,colorgrad$g,colorgrad$b1, alpha = 0.8),
       pch = 19, cex = 1)
text(x=-119.5, y = 27.5, "Republican", cex = 0.45)
text(x=-117, y = 27.5, "50%", cex = 0.45)
text(x=-114.5, y = 27.5, "Democrat", cex = 0.45)
text(x = -117, y = 26.5,
     "Circle size is proportional to",
     cex=0.5)
text(x = -117, y = 26,
     "Median Household Income of each county",
     cex=0.5)

################################################
### Map - Population/ Political ##############
################################################
size4 <- all.data$Pop.Estimate
par(mar=c(1, 1, 2, 1))
plot(all.data$Long, all.data$Lat,
     xlab = "", ylab = "",
     xaxt = "n", yaxt = "n",
     col=rgb(all.data$M.more,all.data$g,all.data$b1, alpha = 0.8),
     pch = 19, cex = size4/mean(size4)/4,
     main = "County Population and 2008 U.S. Presidential Election")
points(all.data$Long, all.data$Lat,
       pch = 1, col ="white", cex= size4/mean(size4)/4)
map("county", add = TRUE, col = "gray")
text(x = -117, y = 29,
     "Map Key",
     cex=0.7)
points(colorgrad$Long, colorgrad$Lat,
       col=rgb(colorgrad$M.more,colorgrad$g,colorgrad$b1, alpha = 0.8),
       pch = 19, cex = 1)
text(x=-119.5, y = 27.5, "Republican", cex = 0.45)
text(x=-117, y = 27.5, "50%", cex = 0.45)
text(x=-114.5, y = 27.5, "Democrat", cex = 0.45)
text(x = -117, y = 26.5,
     "Circle size is proportional to",
     cex=0.5)
text(x = -117, y = 26,
     "Population of each county",
     cex=0.5)
## pretty good as well, more people ~ democratic



################################################
###     Scatter Plots    ######################
################################################
library(car)

plot.McCain = as.numeric(gsub("%", "", all.data$McCain))
plot.population = all.data$Pop.Estimate/1000
plot.university = all.data$University.Count
```

```
plot.universitydensity = plot.university/plot.population
plot.income = as.numeric(gsub(",", "", all.data$Median.Household.Income))/1000

# removing outliers

pop.range = boxplot.stats(plot.population, coef = 100)$stats[c(1,5)]
pop.index = (plot.population > pop.range[1]) & (plot.population < pop.range[2])
uni.range = boxplot.stats(plot.university, coef = 60)$stats[c(1,5)]
uni.index = (plot.university > uni.range[1]) & (plot.university < uni.range[2])
uniden.range = boxplot.stats(plot.universitydensity, coef = 75)$stats[c(1,5)]
uniden.index = (plot.universitydensity > uniden.range[1]) & (plot.universitydensity <
uniden.range[2])

# correlation values

cor.pop = cor(plot.McCain[pop.index], plot.population[pop.index])
cor.uni = cor(plot.McCain[uni.index], plot.university[uni.index])
cor.uniden = cor(plot.McCain[uniden.index], plot.universitydensity[uniden.index])
cor.inc = cor(plot.McCain, plot.income)

# regression line gradients

grad.pop                          =                          cov(plot.McCain[pop.index],
plot.population[pop.index])/var(plot.McCain[pop.index])
grad.uni                          =                          cov(plot.McCain[uni.index],
plot.university[uni.index])/var(plot.McCain[uni.index])
grad.uniden                       =                       cov(plot.McCain[uniden.index],
plot.universitydensity[uniden.index])/var(plot.McCain[uniden.index])
grad.inc = cov(plot.McCain, plot.income)/var(plot.McCain)

# scatter plots

scatterplot(x = plot.McCain[pop.index], y = plot.population[pop.index],
            main = "Population Size vs Percentage Voted for McCain",
            xlab = "Percentage Voted for McCain in 2008",
            ylab = "Population Size (in 1000)",
            boxplots = "y", smooth = FALSE,
            col = palette(topo.colors(6)))
mtext(paste("correlation = ", formatC(cor.pop, 3), "; gradient = ", formatC(grad.pop,
3), collapse = ""),
      side = 1, at = 11, col = "blue", cex = 0.8)
scatterplot(x = plot.McCain[uni.index], y = plot.university[uni.index],
            main = "Number of Universities vs Percentage Voted for McCain",
            xlab = "Percentage Voted for McCain in 2008",
            ylab = "Number of Universities",
            boxplots = "y", smooth = FALSE,
            col = palette(topo.colors(6)))
mtext(paste("correlation = ", formatC(cor.uni, 3), "; gradient = ", formatC(grad.uni,
3), collapse = ""),
      side = 1, at = 11, col = "blue", cex = 0.8)
scatterplot(x = plot.McCain[uniden.index], y = plot.universitydensity[uniden.index],
            main = "Number of Universities per 1000 people vs Percentage Voted for
McCain",
```

```
            xlab = "Percentage Voted for McCain in 2008",
            ylab = "Number of Universities per 1000 people",
            boxplots = "y", smooth = FALSE,
            col = palette(topo.colors(6)))
mtext(paste("correlation = ", formatC(cor.uniden, 3), "; gradient = ",
formatC(grad.uniden, 3), collapse = ""),
      side = 1, at = 11, col = "blue", cex = 0.8)
scatterplot(x = plot.McCain, y = plot.income,
            main = "Percentage Voted for McCain vs Median Household Income",
            xlab = "Percentage Voted for McCain in 2008",
            ylab = "Median Household Income (in $1000)",
            boxplots = "y", smooth = FALSE,
            col = palette(topo.colors(6)))
mtext(paste("correlation = ", formatC(cor.inc, 3), "; gradient = ", formatC(grad.inc,
3), collapse = ""),
      side = 1, at = 11, col = "blue", cex = 0.8)

### excludes counties with 0 universities

all.data.nozero = all.data[all.data$University.Count>0,]
plot.McCain.nozero = as.numeric(gsub("%", "", all.data.nozero$McCain))
plot.population.nozero = all.data.nozero$Pop.Estimate/1000
plot.university.nozero = all.data.nozero$University.Count
plot.universitydensity.nozero = plot.university.nozero/plot.population.nozero
plot.income.nozero = as.numeric(gsub(",", "",
all.data.nozero$Median.Household.Income))/1000

# remove outliers

pop.range0 = boxplot.stats(plot.population.nozero, coef = 50)$stats[c(1,5)]
pop.index0 = (plot.population.nozero > pop.range0[1]) & (plot.population.nozero <
pop.range0[2])
uni.range0 = boxplot.stats(plot.university.nozero, coef = 40)$stats[c(1,5)]
uni.index0 = (plot.university.nozero > uni.range0[1]) & (plot.university.nozero <
uni.range0[2])
uniden.range0 = boxplot.stats(plot.universitydensity.nozero, coef = 50)$stats[c(1,5)]
uniden.index0 = (plot.universitydensity.nozero > uniden.range0[1]) &
(plot.universitydensity.nozero < uniden.range0[2])

# correlation values

cor.pop0 = cor(plot.McCain.nozero[pop.index0], plot.population.nozero[pop.index0])
cor.uni0 = cor(plot.McCain.nozero[uni.index0], plot.university.nozero[uni.index0])
cor.uniden0 = cor(plot.McCain.nozero[uniden.index0],
plot.universitydensity.nozero[uniden.index0])
cor.inc0 = cor(plot.McCain.nozero, plot.income.nozero)

# regression line gradients

grad.pop0 = cov(plot.McCain.nozero[pop.index0],
plot.population.nozero[pop.index0])/var(plot.McCain.nozero[pop.index0])
grad.uni0 = cov(plot.McCain.nozero[uni.index0],
plot.university.nozero[uni.index0])/var(plot.McCain.nozero[uni.index0])
```

```
grad.uniden0                    =                    cov(plot.McCain.nozero[uniden.index0],
plot.universitydensity.nozero[uniden.index0])/var(plot.McCain.nozero[uniden.index0])
grad.inc0 = cov(plot.McCain.nozero, plot.income.nozero)/var(plot.McCain.nozero)

# scatter plots

scatterplot(x = plot.McCain.nozero[pop.index0], y = plot.population.nozero[pop.index0],
            main = "Population Size vs Percentage Voted for McCain [2]",
            xlab = "Percentage Voted for McCain in 2008",
            ylab = "Population Size (in 1000)",
            boxplots = "y", smooth = FALSE,
            col = palette(topo.colors(6)))
mtext(paste("correlation = ", formatC(cor.pop0, 3), "; gradient = ", formatC(grad.pop0,
3), collapse = ""),
      side = 1, at = 11, col = "blue", cex = 0.8)
scatterplot(x = plot.McCain.nozero[uni.index0], y = plot.university.nozero[uni.index0],
            main = "Number of Universities vs Percentage Voted for McCain [2]",
            xlab = "Percentage Voted for McCain in 2008",
            ylab = "Number of Universities",
            boxplots = "y", smooth = FALSE,
            col = palette(topo.colors(6)))
mtext(paste("correlation = ", formatC(cor.uni0, 3), "; gradient = ", formatC(grad.uni0,
3), collapse = ""),
      side = 1, at = 11, col = "blue", cex = 0.8)
scatterplot(x         =         plot.McCain.nozero[uniden.index0],         y         =
plot.universitydensity.nozero[uniden.index0],
            main = "Number of Universities per 1000 people vs Percentage Voted for
McCain [2]",
            xlab = "Percentage Voted for McCain in 2008",
            ylab = "Number of Universities per 1000 people",
            boxplots = "y", smooth = FALSE,
            col = palette(topo.colors(6)))
mtext(paste("correlation   =   ",   formatC(cor.uniden0,   3),   ";   gradient   =   ",
formatC(grad.uniden0, 3), collapse = ""),
      side = 1, at = 11, col = "blue", cex = 0.8)
scatterplot(x = plot.McCain.nozero, y = plot.income.nozero,
            main = "Percentage Voted for McCain vs Median Household Income [2]",
            xlab = "Percentage Voted for McCain in 2008",
            ylab = "Median Household Income (in $1000)",
            boxplots = "y", smooth = FALSE,
            col = palette(topo.colors(6)))
mtext(paste("correlation = ", formatC(cor.inc0, 3), "; gradient = ", formatC(grad.inc0,
3), collapse = ""),
      side = 1, at = 11, col = "blue", cex = 0.8)


#####################################################################
####################     Bar Plots    ###############################
#####################################################################
```

transfer the factors to numeric values in the data frame

```
bp<-population.income.election
McCain.votes1 <- bp[,"McCain.."]
```

```
McCain.votes2 <- gsub("%","",McCain.votes1)
McCain.votes <- as.numeric(gsub("\\.","",McCain.votes2))/10000
bp$McCainVotes <- McCain.votes
Obama.votes1 <- bp[,"Obama.."]
 Obama.votes2 <- gsub("%","",Obama.votes1)
 Obama.votes <- as.numeric(gsub("\\.","",Obama.votes2))/10000
 bp$ObamaVotes <- Obama.votes
 incomeonly1 <- (bp[,"Median.Household.Income"])
incomeonly <- as.numeric(gsub(",","",incomeonly1))
bp$MedianIncome <- incomeonly

#####election results in all income groups

#subset out different income groups

income40000<-subset(bp,bp$MedianIncome<40000)
dim(income40000) #1185 8
income60000<-subset(bp,bp$MedianIncome<60000&bp$MedianIncome>=40000)
dim(income60000)   #1568 8
income80000<-subset(bp,bp$MedianIncome<80000&bp$MedianIncome>=60000)
dim(income80000)   #209 8
income100000<-subset(bp,bp$MedianIncome<100000&bp$MedianIncome>=80000)
dim(income100000) #33 8
income120000<-subset(bp,bp$MedianIncome<120000&bp$MedianIncome>=100000)
dim(income120000)   #6 8
totp<-c(1185,1568,209,33,6)

 #count the election results of Mc and Ob in each group

 income40000M<-subset(income40000,income40000$McCainVotes>income40000$ObamaVotes)
 dim(income40000M)  #930 8
 income40000o<-subset(income40000,income40000$McCainVotes<income40000$ObamaVotes)
dim(income40000o)  #254 11

income60000M<-subset(income60000,income60000$McCainVotes>income60000$ObamaVotes)
dim(income60000M) #1116 8
income60000o<-subset(income60000,income60000$McCainVotes<income60000$ObamaVotes)
dim(income60000o)  450 8

income80000M<-subset(income80000,income80000$McCainVotes>income80000$ObamaVotes)
 dim(income80000M)  #118
 income80000o<-subset(income80000,income80000$McCainVotes<income80000$ObamaVotes)
 dim(income80000o)  #91

 income100000M<-subset(income100000,income100000$McCainVotes>income100000$ObamaVotes)
 dim(income100000M)   #17
 income100000o<-subset(income100000,income100000$McCainVotes<income100000$ObamaVotes)
 dim(income100000o)  #16

 income120000M<-subset(income120000,income120000$McCainVotes>income120000$ObamaVotes)
 dim(income120000M) #2
 income120000o<-subset(income120000,income120000$McCainVotes<income120000$ObamaVotes)
dim(income120000o) #4
```

#make a matrix for barplot

```
incomex=c(40000,60000,80000,100000,120000)
Mc<-c(930,1116,118,17,2)
Ob<-c(254,450,91,16,4)
incomeVote<-                                                    matrix(c(Mc,Ob),
nrow=length(incomex),ncol=2,dimnames=list(incomex,c("Mc","Ob")))
incomeVote2<-t(incomeVote)
```

#barplot code

```
barplot(incomeVote2,beside=TRUE,space=c(0.2,0.8),names.arg=c("0-40","40-60","60-
80","80-100","100-120"),col=c("red","blue"),border="black",main=c("County      Election
Results    By    Income"),xlab="Income    Group(in    1000)",ylab="Election    Results
Frequencies",font.lab=2,ylim=c(0,1200))

legend("topleft",legend=c("McCain","Obama"),fill=c("red","blue"))
text(x=c(1.3,2.6,4.4,5.6,7.3,8.6,10.2,11.6,13.3,14.5),y=c(950,274,1136,470,138,111,37,
36,22,24),label=c("930","254","1116","450","118","91","17","16","2","4"),cex=0.7)
```

```
####election results by population
```

#subset out different population groups

```
pop15<-subset(bp,bp$Pop.Estimate<15000)
dim(pop15) #986
pop30<-subset(bp,bp$Pop.Estimate<30000&bp$Pop.Estimate>15000)
dim(pop30) #649
pop45<-subset(bp,bp$Pop.Estimate<45000&bp$Pop.Estimate>30000)
dim(pop45) #357
pop60<-subset(bp,bp$Pop.Estimate<60000&bp$Pop.Estimate>45000)
dim(pop60) #195
pop60a<-subset(bp,bp$Pop.Estimate>60000)
dim(pop60a) #814
```

#count the election results of Mc and Ob in different groups

```
pop15M<-subset(pop15,pop15$McCainVotes>pop15$ObamaVotes)
 dim(pop15M) #803
 pop15o<-subset(pop15,pop15$McCainVotes<pop15$ObamaVotes)
dim(pop15o) #182
pop30M<-subset(pop30,pop30$McCainVotes>pop30$ObamaVotes)
 dim(pop30M) #510
 pop30o<-subset(pop30,pop30$McCainVotes<pop30$ObamaVotes)
dim(pop30o) #138
pop45m<-subset(pop45,pop45$McCainVotes>pop45$ObamaVotes)
 dim(pop45m) #275
 pop45o<-subset(pop45,pop45$McCainVotes<pop45$ObamaVotes)
 dim(pop45o) #82
 pop60m<-subset(pop60,pop60$McCainVotes>pop60$ObamaVotes)
  dim(pop60m) #142
pop60o<-subset(pop60,pop60$McCainVotes<pop60$ObamaVotes)
   dim(pop60o)#52
```

```
 pop60am<-subset(pop60a,pop60a$McCainVotes>pop60a$ObamaVotes)
  dim(pop60am)#453
  pop60ao<-subset(pop60a,pop60a$McCainVotes<pop60a$ObamaVotes)
 dim(pop60ao) #360
```

#make a matrix for the plot

```
popx<-c(15000,30000,45000,60000,60001)
mc<-c(803,510,275,142,453)
ob<-c(182,138,82,52,360)
popv<-matrix(c(mc,ob),nrow=length(popx),ncol=2,dimnames=list(popx,c("mc","ob")))
popv2<-t(popv)
```

#barplot code

```
barplot(popv2,beside=TRUE,space=c(0.2,0.8),names.arg=c("0-15","15-30","30-45","45-
60","60+"),col=c("red","blue"),border="black",main="County Election Results By
Population",xlab="Population Group(in 1000)", ylab="Election Resultes Frequencies",
font.lab=2, ylim=c(0,1000))
legend("topleft",legend=c("McCain","Obama"),fill=c("red","blue"))
text(x=c(1.3,2.6,4.4,5.6,7.3,8.6,10.2,11.6,13.3,14.5),y=c(823,202,530,158,295,102,162,
72,473,380),label=c("803","182","510","138","275","82","142","52","453","360"),cex=0.7)
```

```
###election results by income in percentage
```

#calculate the election results in percentage

```
mc4<-Mc/totp
ob4<-Ob/totp
mc4 # 0.7848101 0.7117347 0.5645933 0.5151515 0.3333333
ob4 # 0.2143460 0.2869898 0.4354067 0.4848485 0.6666667
mc4<-c(0.78,0.71,0.56,0.52,0.33)
ob4<-c(0.21,0.29,0.44,0.48,0.67)
```

#make a matrix for plot

```
incomeVote2<-                                                    matrix(c(mc4,ob4),
nrow=length(incomex),ncol=2,dimnames=list(incomex,c("mc4","ob4")))
incomeVote3<-t(incomeVote2)
```

#barplot code

```
barplot(incomeVote3,beside=TRUE,space=c(0.2,0.8),names.arg=c("0-40","40-60","60-
80","80-100","100-120"),col=c("red","blue"),border="black",main=c("County Election
Results By Income"),xlab="Income Group(in 1000)",ylab="Election Results(in
percentage)",font.lab=2,ylim=c(0,1))
text(x=c(1.3,2.6,4.4,5.6,7.3,8.6,10.2,11.6,13.3,14.5),y=c(0.8,0.23,0.73,0.31,0.58,0.46
,0.54,0.5,0.35,0.69),label=c("78%","21%","71%","29%","56%","44%","52%","48%","33%","67
%"),cex=0.7)
legend("topleft",legend=c("McCain","Obama"),fill=c("red","blue"))
```

```
##election results by population in per
```

#calculate the election result in percentage

```
totalpop<-c(986,649,357,195,814)
mc5<-mc/totalpop
mc5 #0.8144016 0.7858243 0.7703081 0.7282051 0.5565111
ob5<-ob/totalpop
ob5 #0.1845842 0.2126348 0.2296919 0.2666667 0.4422604
```

#make a matrix for barplot

```
mc6<-c(0.81,0.79,0.77,0.73,0.56)
 ob6<-c(0.18,0.21,0.23,0.27,0.44)
 popvp<-matrix(c(mc6,ob6),nrow=length(popx),ncol=2,dimnames=list(popx,c("mc6","ob6")))
 popvp1<-t(popvp)
```

#barplot code

```
 barplot(popvp1,beside=TRUE,space=c(0.2,0.8),names.arg=c("0-15","15-30","30-45","45-
60","60+"),col=c("red","blue"),border="black",main="County    Election    Results    By
Population",xlab="Population Group(in 1000)", ylab="Election Resultes In Percentage",
font.lab=2, ylim=c(0,1))
 legend("topleft",legend=c("McCain","Obama"),fill=c("red","blue"))

text(x=c(1.4,2.55,4.4,5.6,7.35,8.6,10.3,11.6,13.35,14.53),y=c(0.83,0.2,0.81,0.23,0.79,
0.25,0.75,0.29,0.58,0.46),label=c("81%","18%","79%","21%","77%","23%","73%","27%","56%
","44%"),cex=0.7)


 ######for big counties with pop 500000+##############################
```

#subset the counties with large population

```
 pop80<-subset(bp,bp$Pop.Estimate>500000&bp$Pop.Estimate<800000)
 dim(pop80) #68
 pop110<-subset(bp,bp$Pop.Estimate>800000&bp$Pop.Estimate<1100000)
 dim(pop110) #28 8
 pop140<-subset(bp,bp$Pop.Estimate>1100000&bp$Pop.Estimate<1400000)
 dim(pop140) #11 8
 pop140a<-subset(bp,bp$Pop.Estimate>1400000)
 dim(pop140a) #24 8
```

#calculate the election results for mc and ob in those groups

```
pop80m<-subset(pop80,pop80$McCainVotes>pop80$ObamaVotes)
 dim(pop80m) #16 8
  pop80o<-subset(pop80,pop80$McCainVotes<pop80$ObamaVotes)
 dim(pop80o) #45 8
 pop110m<-subset(pop110,pop110$McCainVotes>pop110$ObamaVotes)
 dim(pop110m) #4 8
 pop110o<-subset(pop110,pop110$McCainVotes<pop110$ObamaVotes)
dim(pop110o) #24 8
pop140m<-subset(pop140,pop140$McCainVotes>pop140$ObamaVotes)
dim(pop140m) #1 8
 pop140o<-subset(pop140,pop140$McCainVotes<pop140$ObamaVotes)
 dim(pop140o) #10 8
 pop140am<-subset(pop140a,pop140a$McCainVotes>pop140a$ObamaVotes)
 dim(pop140am) #3 8
```

```
 pop140ao<-subset(pop140a,pop140a$McCainVotes<pop140a$ObamaVotes)
 dim(pop140ao) #21 8
```
#make a matrix for the plot

```
mcb<-c(16,4,1,3)
obb<-c(45,24,10,21)
popbx<-c(500000,800000,1100000,1400000)
popb<-matrix(c(mcb,obb),nrow=length(popbx),ncol=2,dimnames=list(popbx,c("mcb","obb")))
popbt<-t(popb)
```

#barplot code

```
barplot(popbt,beside=TRUE,space=c(0.2,0.8),names.arg=c("500-800","800-1100","1100-
1400","1400+"),col=c("red","blue"),border="black",main="Election Results Of Counties
With Large Population",xlab="Population Group(in 1000)", ylab="Election Resultes
Frequencies", font.lab=2, ylim=c(0,50))
 legend("topright",legend=c("McCain","Obama"),fill=c("red","blue"))

text(x=c(1.3,2.55,4.4,5.6,7.35,8.6,10.3,11.6),y=c(17,46,5,25,2,11,4,22),label=c(16,45,
4,24,1,10,3,21),cex=0.7)

####percentage of results in big pop counties#####
```

#calculate the percentage results of ob and mc

```
totalcount<-c(61,28,11,24)
mcbp<-mcb/totalcount
mcbp #0.26229508 0.14285714 0.09090909 0.12500000
obbp<-obb/totalcount
obbp # 0.7377049 0.8571429 0.9090909 0.8750000
```

#make a matrix for plot

```
mcbp1<-c(0.26,0.14,0.09,0.13)
obbp1<-c(0.74,0.86,0.91,0.88)
popbp<-
matrix(c(mcbp1,obbp1),nrow=length(popbx),ncol=2,dimnames=list(popbx,c("mcbp1","obbp1")
))
popbpt<-t(popbp)
```

#barplot code

```
barplot(popbpt,beside=TRUE,space=c(0.2,0.8),names.arg=c("500-800","800-1100","1100-
1400","1400+"),col=c("red","blue"),border="black",main="Election Results Of Counties
With Large Population",xlab="Population Group(in 1000)", ylab="Election Resultes In
Percentage", font.lab=2, ylim=c(0,1))
 legend("topleft",legend=c("McCain","Obama"),fill=c("red","blue"))

text(x=c(1.4,2.55,4.3,5.6,7.35,8.6,10.3,11.6),y=c(0.28,0.76,0.16,0.88,0.11,0.93,0.15,0
.9),label=c("26%","74%","14%","86%","9%","91%","13%","88%"),cex=0.7)

#######large population county with uneven cate#####
```

#subset different population groups

```
pop60<-subset(bp,bp$Pop.Estimate>60000&bp$Pop.Estimate<2000000)
dim(pop60) #802   8
pop200<-subset(bp,bp$Pop.Estimate>200000&bp$Pop.Estimate<5000000)
dim(pop200) #286   8
pop500<-subset(bp,bp$Pop.Estimate>500000&bp$Pop.Estimate<10000000)
dim(pop500) #124   8
pop1000<-subset(bp,bp$Pop.Estimate>1000000&bp$Pop.Estimate<10000000)
dim(pop1000) #37  8
totalun<-c(802,286,124,37)
```

#calculate results of ob and mc

```
pop60m<-subset(pop60,pop60$McCainVotes>pop60$ObamaVotes)
pop60o<-subset(pop60,pop60$McCainVotes<pop60$ObamaVotes)
dim(pop60m) #451  8
dim(pop60o)  #350   8
pop200m<-subset(pop200,pop200$McCainVotes>pop200$ObamaVotes)
pop200o<-subset(pop200,pop200$McCainVotes<pop200$ObamaVotes)
dim(pop200m) # 88
dim(pop200o) #198
pop500m<-subset(pop500,pop500$McCainVotes>pop500$ObamaVotes)
pop500o<-subset(pop500,pop500$McCainVotes<pop500$ObamaVotes)
dim(pop500m) # 24 8
dim(pop500o) # 100 8
pop1000m<-subset(pop1000,pop1000$McCainVotes>pop1000$ObamaVotes)
pop1000o<-subset(pop1000,pop1000$McCainVotes<pop1000$ObamaVotes)
dim(pop1000m) # 4 8
dim(pop1000o)#33 8
```

#make a matrix for plot

```
mcun<-c(451,88,24,4)
obun<-c(350,198,100,33)
unx<-c(60,200,500,1000)
mcun1<-mcun/totalun
obun1<-obun/totalun
mcun1
obun1
mcun2<-c(0.56,0.31,0.19,0.11)
obun2<-c(0.44,0.49,0.81,0.89)
unbp<-
matrix(c(mcun2,obun2),nrow=length(unx),ncol=2,dimnames=list(unx,c("mcun2","obun2")))
unbp2<-t(unbp)
```

#barplot codes

```
barplot(unbp2,beside=TRUE,space=c(0.2,0.8),names.arg=c("60-200","200-500","500-
1000","1000-10000"),col=c("red","blue"),border="black",main="Election    Results    Of
Counties  With  Large  Population",xlab="Population  Group(in  1000)",  ylab="Election
Resultes In Percentage", font.lab=2, ylim=c(0,1))
 legend("topleft",legend=c("McCain","Obama"),fill=c("red","blue"))
```

```
text(x=c(1.4,2.55,4.3,5.6,7.35,8.6,10.3,11.6),y=c(0.575,0.455,0.325,0.505,0.205,0.825,
0.125,0.905),label=c("56%","44%","31","49%","19%","81%","11%","89%"),cex=0.7)
```