National Health

CSC 465 Group Project

Group Social Theorist: Kari Palmier, Judy Moran, Harika Rallapalli, Michelle Tang, Rebecca Tung

Table of Contents

I. Introduction

As a group, we are interested in improving the quality of people's life. We would like to find out what demographic and behavioral factors are closely related to people's health.

A recurring theme we looked at even as groups formed was 'obesity'. Body mass index (BMI) is a measure of body fat based on height and weight. We use Body Mass Index (BMI) greater than 30 to identify a respondent as obese.

Using a large dataset of survey data from a 2013-2014 the National Health and Nutrition Examination Survey we did intense analysis and identified Age, Race, Gender and Income factors that appear to be associated with obesity.

We display compelling visuals to support our final statements. These include correlogram, force direct diagram, bar charts, scatterplot, treemap and heat maps. Some of these visuals are familiar from prior courses but most reflect the more advanced work we've learned in CSC 465.

II. Data

Using Kaggle, we found interesting data associated with a 'National Health and Nutrition Examination Survey' to analyze. The following site contains extensive documentation and context around the data.

https://www.kaggle.com/cdc/national-health-and-nutrition-examination-survey

This site contains the following several NHANES datasets from 2013-2014. We derived our final dataset from the Demographics and Questionnaire datasets. Appendix III contains detail around our original list of variables. We eventually settled on exploring the following data.

Main Variables

- Gender
- Age
- Race
- Education Level
- Annual Household Income
- Marital Status
- Height
- Weight
- BMI (calculated from Height and Weight)
- Obesity Indicator (0 for BMI values under 30, 1 for over)

Secondary Variables

- Ratio of Income to Poverty Level
- High Blood Pressure Flag
- Diabetes Flag
- Amount Spent at Grocery Store per Month
- Amount Spent on Non-Food Per Month
- Amount Spent Eating Out Per Month
- Amount Spent Delivery/Carryout Per Month
- Number of Meals Made at Home Per Week
- Number of Fast Food Meals Per Week
- Number of Ready Made Meals Per Week
- Number of Frozen Meals Per Week
- Doctor Said Overweight
- Doctor Said to Lose Weight
- Doctor Said to Exercise
- Number of Sedentary Minutes Per Day
- Has Smoked 100 Cigarettes in Lifetime

The dataset used both Demographic and Questionnaire data from the NHANES site. We combined the files using the SEQN field, ensuring the SEQN values matched in both the Demographic and Questionnaire csv files, copying and pasting the entries into a new xlsx file such that the SEQN numbers align, then creating the final csv. We then filtered out all of the NaN, Refused, and Don't Know values from these attributes to verify that we would still have an adequate dataset. We also filtered out the income categories that covered

ranges (greater than $20,000 and less than $20,000) because these ranges overlap existing income brackets and only contain a small number of data points. A BMI variable was then calculated by using the height and weight (BMI = Weight (kg) / (Height (m))^2. This new BMI variable was then used to create an obesity indicator (value 1 for BMI values greater than or equal to 30, value 0 for BMI less than 30). The final csv file contains 27 variables and 3839 rows of data.

We questioned if the distributions of the main variables changed significantly before and after data cleaning (in other words, did removing NaN rows effect the distribution of the data). To be sure this was not the case, Kari created histograms at the very beginning (directly after the data was read in and before anything was done), then after data cleaning and compared them. Aside from removing the data for people under age 20, there were no other significant changes.

During exploratory analysis phase, it was obvious that an unexpectedly large number of respondents had income levels over $100,000 with low obese levels. We did some digging on the Kaggle website and the CDC links and found that the dataset we are using is for California. This would explain the high salaries and possibly the low obesity (a lot of in shape people in the entertainment field and many people make over $100K in CA).

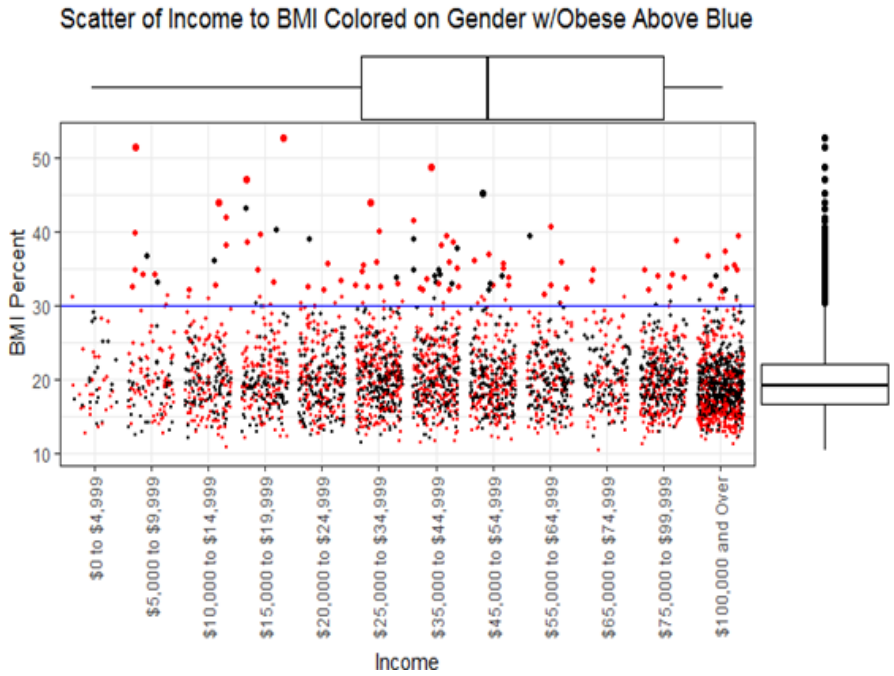
III. Exploratory Analysis

Based on the description of dataset, we decided to compute several exploratory visualizations to understand the broader picture of health and nutrition. There were several main components we used to illustrate these graphs such as demographics like gender, race, age, income and education level of each respondent.

Figure 1 and Figure 2 display the distribution of demographics, income, and education level on BMI percentage and whether the respondent is obese or not obese. Based on Figure 1, respondents with an income level between $5,000 to $9,999 and $35,000 to $44,999 have a higher number of obese respondents in comparison to 10 other income levels. Respondents who have an income level between $0 to $4,999 and $65,000 to $74,999 have the lowest number of obese individuals.

Figure 2 displays education level, marital status, gender, and race counts on whether an individual is obese or not obese. This distribution indicates that those with a high school diploma/GED or some college education have the highest count of obese people compared to others within different educational levels. In addition, respondents who are married have a higher count of obese individuals within their bracket and respondents who are female also exhibit a higher obesity count. Lastly, it shows that non-Hispanic White have a slightly higher count of obese respondents in comparison to non-Hispanic Black.

In reference to the visualizations from Figure 1 and Figure 2, we decided to concentrate more on the different factors that may cause the variations in whether a person is obese or not obese.



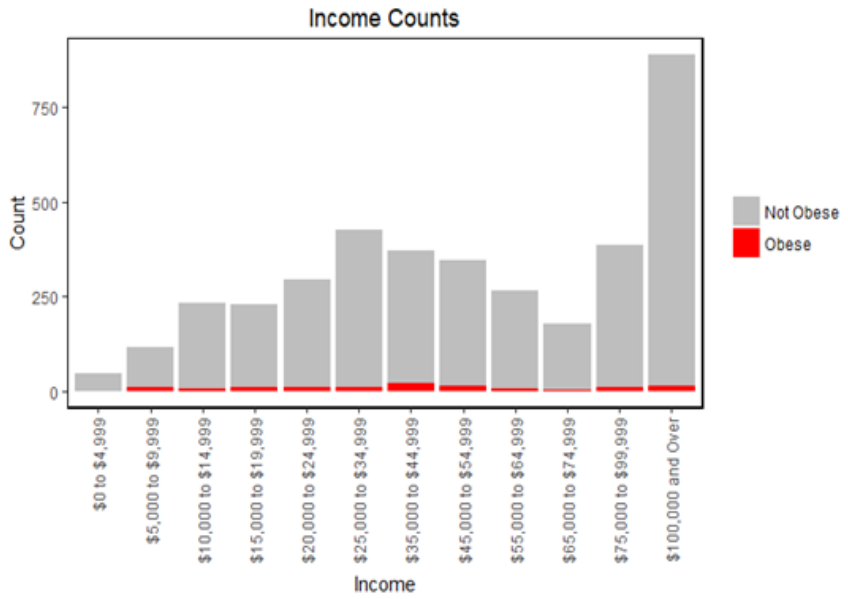


Figure 1: Exploratory 1

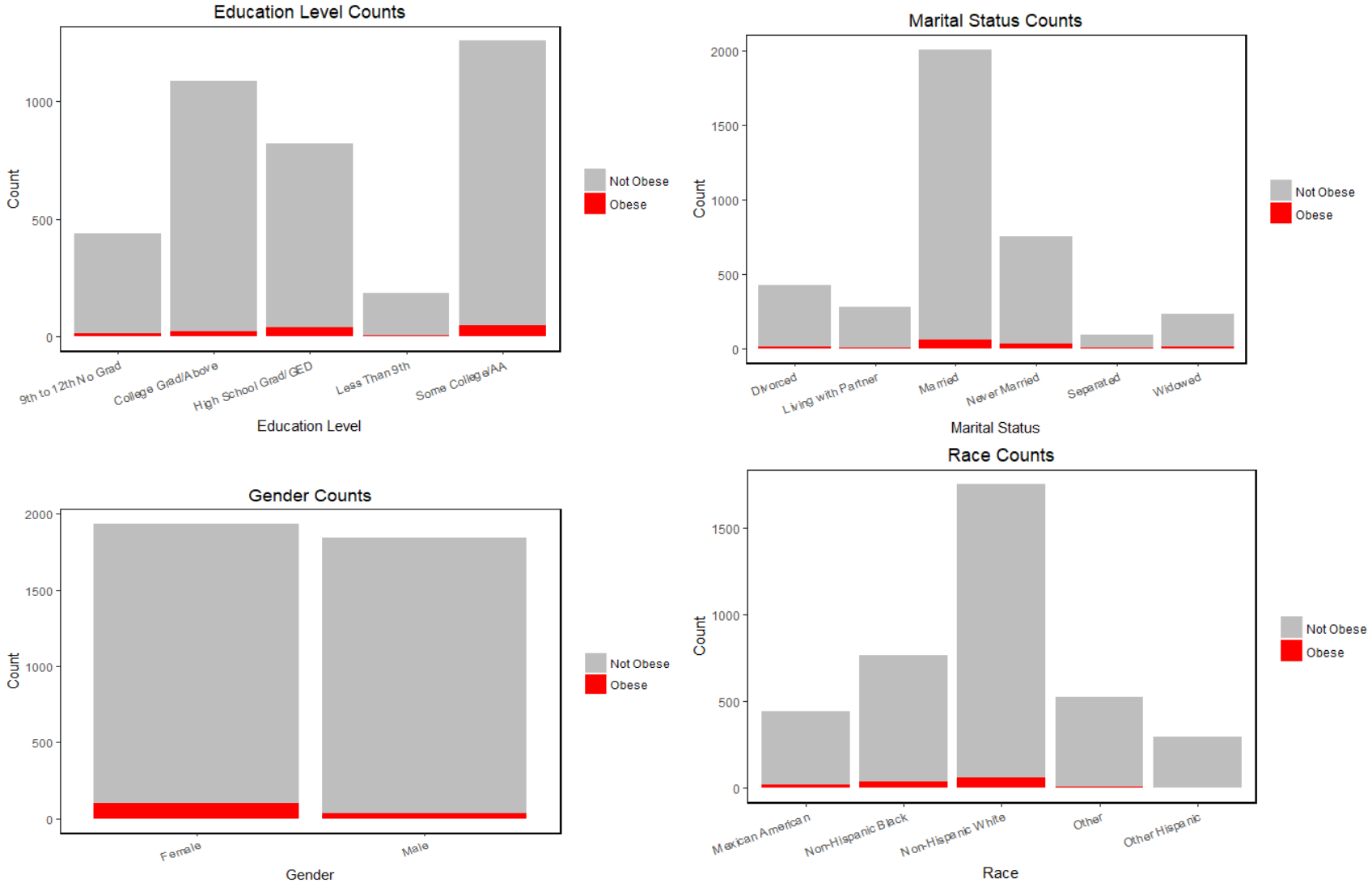


Figure 2: Exploratory 2

IV. Visualizations

We then agreed it also useful to create a plot of the percent of obese people over income level, marital status, education levels and race. We then further partitioned on gender.

We produced the visuals by creating an aggregate dataset with the count of obese people for different attributes, then dividing these counts by the total number of people.

The bar charts with the percent obese are easier to read than the ones of obese and not because there would not be the not obese values overtaking the visual experience.

The visualizations show that the percent obese has a bell-type shape and does taper off as income increases for women. Again, the men have lower percents of obesity across the board. Two of the income brackets did not have any men that were obese (the 0 to $4,000 and the $65,000 to $74,999).

The following force direct and correlogram (Figure 3) shows the correlation among the selected variables we are interested in. The size of the circle indicates the betweenness centrality of each node. *Annual income* appears to have the highest betweenness centrality and follow by *Age* and *Marital Status*. The color and thickness of line indicates the weighed degree of the node. Annual Income and Poverty Ration appears to be strongly correlated. BMI and BMI.Meaning is closely related to Weight, Height, Exercise needed, Weight Lost Needed and Overweight. Note that the BMI.Meaning variable represents four main ranges (underweight for less than 18.5, normal for between 18.5 and 24.9, overweight for between 25 to 29.9, and obese for greater than or equal to 30).

R^2 Forced Direct and Correlation Diagrams for CA Health Survey Data

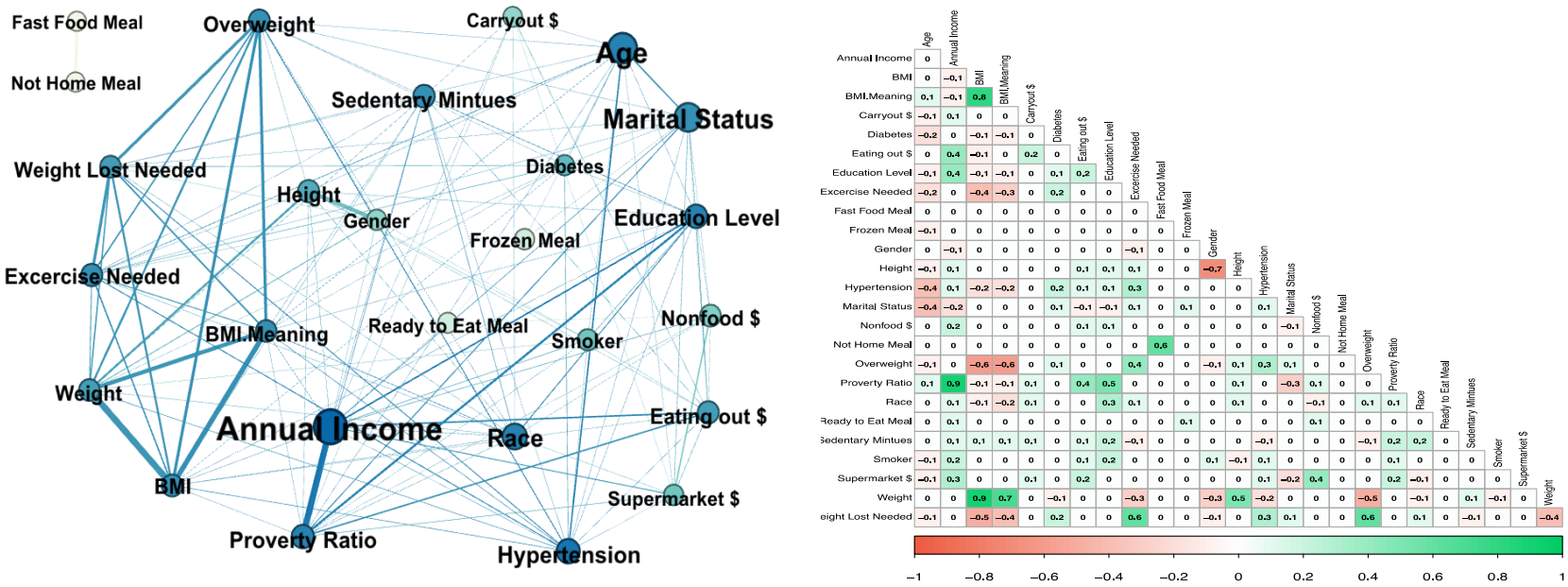


Figure 3: All Variable Correlogram and Force Direct

The hierarchical bar charts (Figure 4) show the percentage of obese respondents for each categorical variable for each gender. The percentage of obese respondents was calculated by finding number of obese respondents and the total number of respondents of each gender for each categorical variable level, then dividing the number of obese by the total number. The percentage of obesity was used in order to eliminate any bias that a straight count would have. For example, non-Hispanic white people have a higher number of obese people, but they also have a much higher total count so although the count of obese people would appear to be the highest, this is only because of the total count (the actual percentage of obese is not as high as the count appears). These bar charts show that women are prone to have a higher percentage of obesity across all categorical variables (race, income, marital status, and education level). The $5,000 to $9,999 income bracket has the highest percentage of obesity for women and the second highest for men (the highest for men is $35,000 to $44, 999). Separated men and women have the highest percentage of obesity out of the marital status variable, followed by married for women and divorced men. Non-Hispanic black people had the highest percentage of obesity, followed by Mexican-American, then non-Hispanic white. People that are high school graduates or have GEDs have the highest percentage of obesity, where people with a college degree or higher had the lowest.

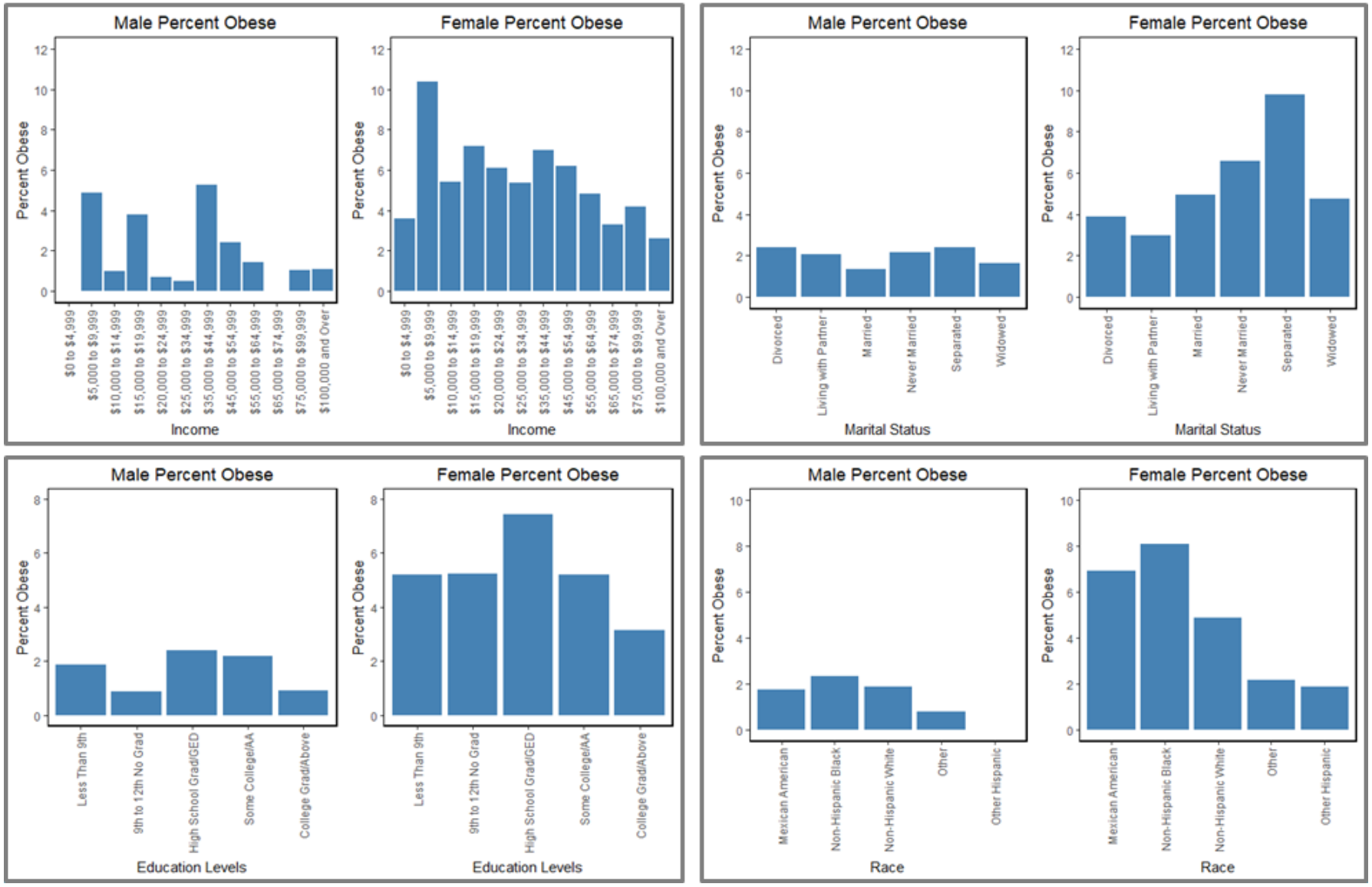


Figure 4: Percentage Obese Hierarchical Bar Charts

The treemap in Figure 4 continues our analysis of BMI Percent, Income and Gender attributes of obese respondents. Using our original dataset, we performed a mean aggregation BMI Percent on Income, Obesity Type and Gender. We then sub setted this dataframe to only look at obese data.

Using the R treemap functionality we built the tree partitioned on Income and Gender and colored on BMI Percent. Initially there were problems with the labels overlapping but we changed the label alignment so that the Income label appears in the center and the Gender labels appear in the upper right.

The original coloring did not clearly display the BMI Percent per Income and Gender so we used the 'RdYlGn' palette and added a mapping so that the BMI Percent ranges from the minimum BMI Percent of 30 to the maximum value of 39 moving from cool greens to hot reds. The visual below clearly shows that among our obese respondents Females tend to have higher BMI percentages.

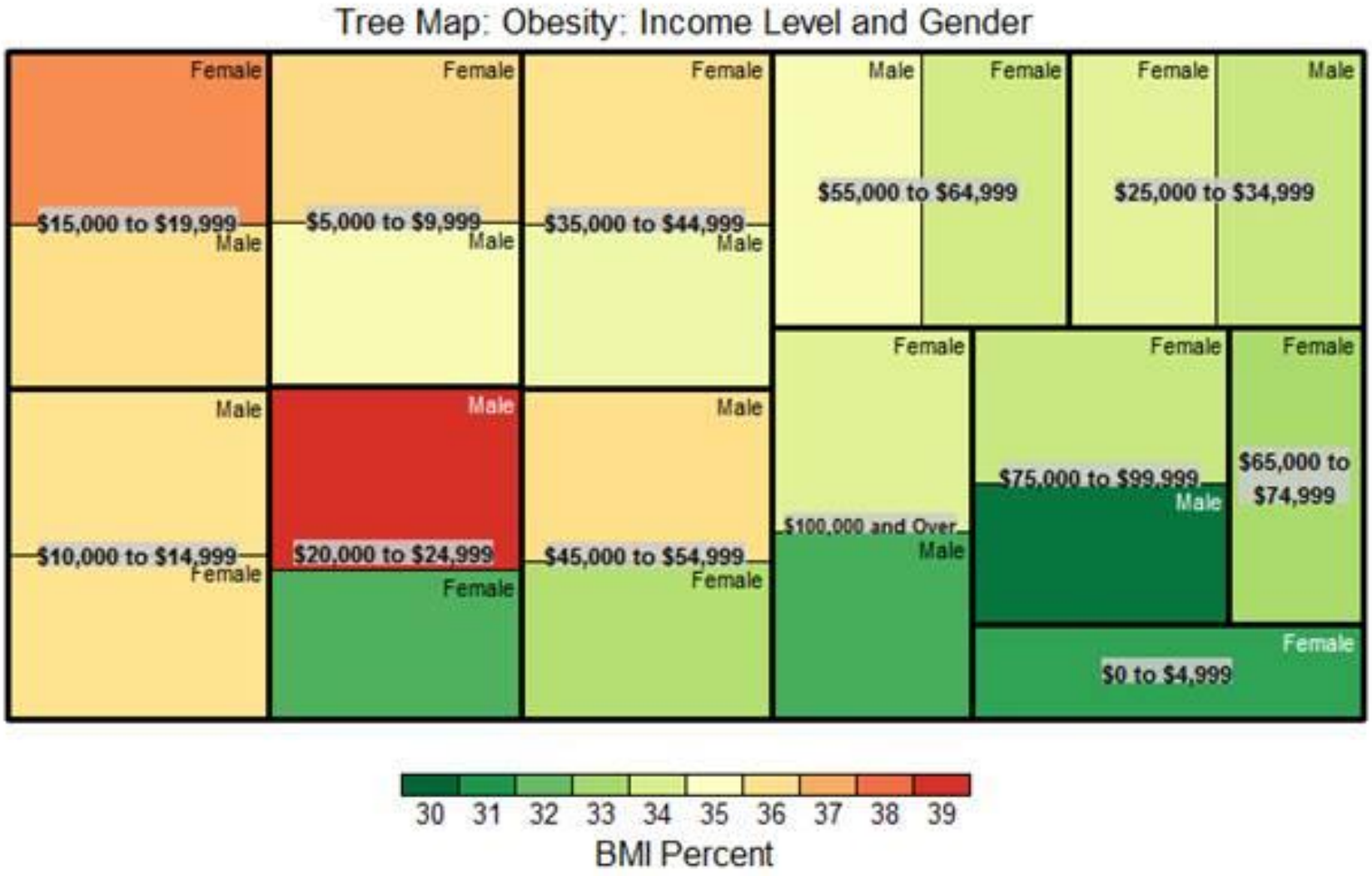


Figure 5: Obese Income Treemap

The heatmaps of mean BMI over age and race and of mean BMI over age and gender were created by first creating a binned age variable (bins are 5 years wide), then aggregating the other variables using this new binned age (shown in Figure 6). The BMI was aggregated using a mean calculation. The mean BMI over age and race shows that the non-Hispanic black and Mexican-American people had the highest BMI values (are the darkest orange and red), followed by the non-Hispanic other people (all dark orange). The highest mean BMI for non-Hispanic black people was between ages 50 and 55. The highest BMI for Mexican-American people was between ages 30 to 35. The mean BMI values for people of other races was the lowest for all ages. The mean BMI over age and gender shows that women ages 50 to 60 had the highest mean BMI. The mean BMI for the rest of the ages and genders were not noticeably different.

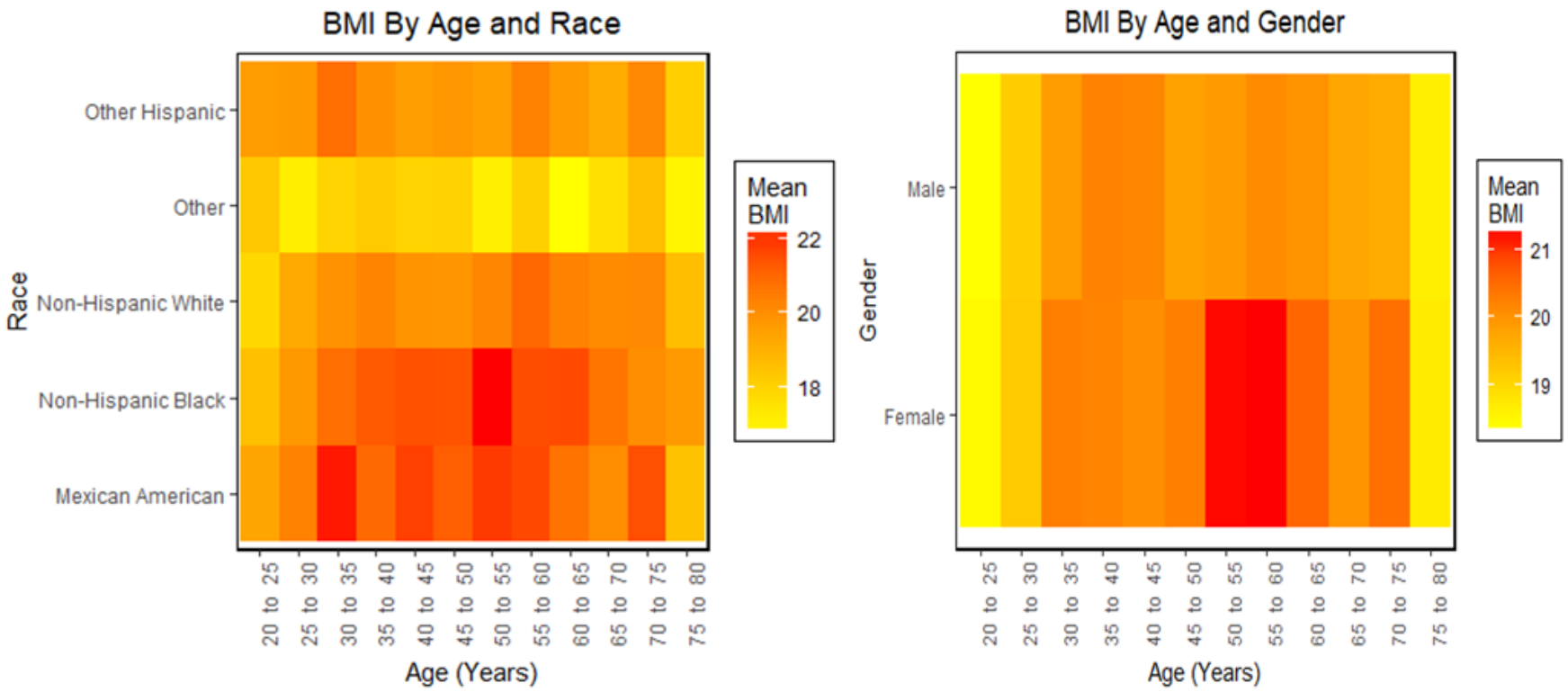


Figure 6: Age, Race, Gender, and Mean BMI Heatmaps

V. Analysis and Discussion

Using visual analysis on the NHANES data, we found the following:

Our visualizations showed that most people in California between 2013 and 2014 who took the survey were not obese. Given that the survey data was taken for a short period of time and only collected data from respondents of one state, the results are not able to be expanded to other states or time periods. California is known as a state where climate is warm, and summers are dry. With that information taken into consideration, it can be inferred the majority of individuals in California may be more physically active due to the effect of the warm climate. The entertainment industry and the technology industries may also contribute to why the obesity is low (people in the entertainment industry must stay thin for their work and the technology industry employs a large number of young people). Variables pertaining to the amount and types of exercise people get per week were in the original NHANES dataset, but were not incorporated into the final dataset due to most people not answering the questions (were almost all NA). If we were able to compare gender and exercise by BMI percentage, it could potentially explain the fluctuations in obesity within California.

Computing multiple visualizations that displayed gender compared to BMI percentage or compare gender by our obese flag variable showed that females have higher BMI and obesity percentages in all categories examined. Figure 6 shows that females within the ages of 50 to 60 seem to have the highest BMI values. As we age, our bodies begin to burn fat slower, which is supported by this trend in women in Figure 6. Lastly, the BMI for men was fairly constant over age and race – it was not as significantly different compared to female.

People of non-Hispanic black decent had the highest percent of obesity followed by Mexican-American and non-Hispanic white. Since females have the highest obesity rate, the approximate order of highest BMI per race for female is American Mexican, Non-Hispanic Black, Other Hispanic, Non-Hispanic Black, then Other. The highest BMI values for women also seemed to

occur between ages 50 and 60 for all races. Note that the BMI for men was fairly constant over age and race.

The $5,000 to $9,999 income bracket has the highest percentage of obesity for women and the second highest for men (the highest for men is $35,000 to $44, 999). Separated men and women have the highest percentage of obesity out of the marital status variable, followed by married for women and divorced men.

Due to the limited time we had to work on this project, we have only scratched the surface of topics that could be studied. For example, Rebecca wanted to study relationships between obesity and specific illnesses such as diabetes or hypertension. Additionally, given the large number of variables in the dataset, Kari and Michelle wanted to do logistical regression in order to find out which variables had the most influence on determining obesity. Overall, there are many factors that should be taken into consideration in order to pin point which variables may be the leading cause for obesity (or any critical health issues).

Appendices

I. Additional Exploratory Analysis

This section includes a brief discussion of the distribution of variables used in our analysis.

The histogram of age shows that the number of people in each age bracket decreases over time. All of the entries for people under 20 were removed during data cleaning because they did not answer most of the survey questions. The histogram of BMI shows is skewed to the right, with the majority of BMI values being less than 25.

The bar chart of gender shows that there was an almost equal number of women and men. The bar chart of education shows that the highest number of people had some college or associates, followed by college graduate and above, then high school graduate or GED. The bar chart of marital status showed that the vast majority of people were married, followed by never married. The bar chart of race shows that the large majority of people were non-Hispanic white, followed by non-Hispanic black, then Mexican-American. The bar chart of the income brackets shows that the majority of people made $100,000 or more, followed by people that made $75,000 to $99,9999. The remaining income brackets appeared to have somewhat normal distribution. The bar chart of the obesity indicator (1 for BMI >= 30, 0 for BMI < 30) shows that the vast majority of people were not obese.

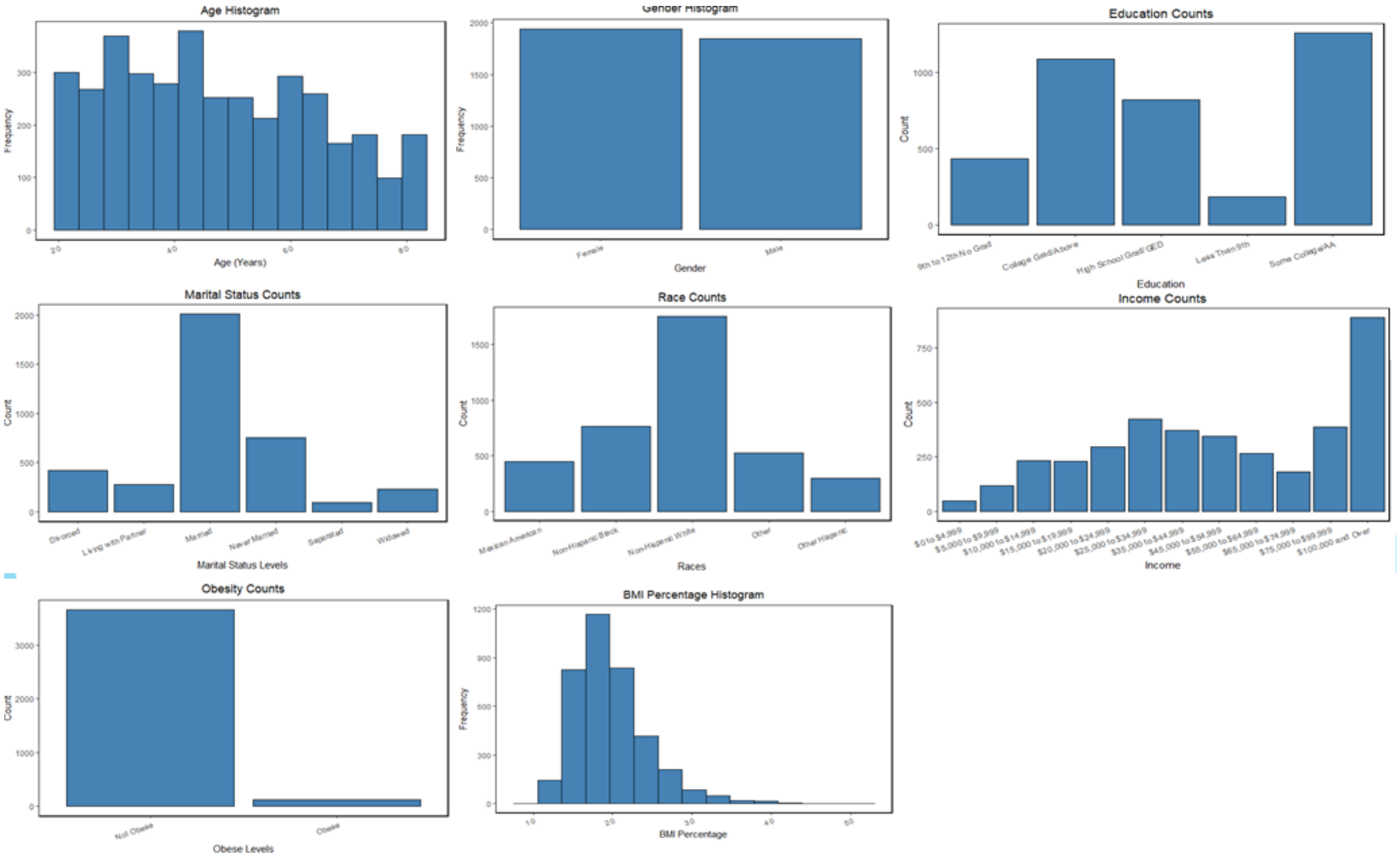


Figure 7: Initial Exploratory Bar Charts and Histograms

II. Individual Reports

This section includes the individual reports of the project members.

1. Rebecca Tung

The following are the list of my contributions:

1. Suggested and got the agreement from our group members on the name of our group
2. Provided purpose and explanation statements why we are interested in the California Health and Nutrition Survey data
3. Attended all team meetings
4. Participated in the project direction discussions and provided feedbacks on through the group email chats
5. Completed on the assigned diagram: treemap, which are not selected as part of the final report
6. Created diagrams showing the correlated variables for hypertension, which are not selected as part of the final report since we determined to focus on BMI as the health indicator
7. Created a correlogram and a force direct network diagram to show the relation among variables we are interested in exploring, which are part of the final report
8. Create a document template in Microsoft Word for our final report

From working on the California Health and Nutrition data set in a group, I have not only learned how to select and visualize the raw data but also how to cooperate with and learn from team members. By using color and size to display degree, betweenness centrality and closeness centrality in Gephi, I can visually interpret R^2 among variables much more naturally than a list of R^2 values with their source and target variables or a

correlation matrix. Data visualization is a powerful tool. When used properly, it can make compilated data patterns into easily consumable formats for human visual interpretation. It was also great to see how each person came with different ways to approach the same raw dataset.

2. Michelle Tang

The visualizations that I created were mainly exploratory visuals in Tableau. These graphs display gender and race compared to BMI percentage and whether an individual was obese or not obese. I was curious to see the difference between BMI percentages between male and female and wanted to analyze that concept in more depth. But unfortunately, we ran out of time to do so for this project. If we potentially had more time with the project, I would have liked to incorporate food types and exercise between genders to see if there was a significant difference between those factors that may determine whether one would be more prone to obesity or not.

In ways I contributed to the team was being actively present when there were team meetings about the project and participating in these meetings. To add on, I participated in the emails so that I would be up-to-date with any changes or concerns within the project dataset and visuals. I also contributed to minor adjustments to a few visuals created by my team members as well. I drafted the final report by writing the exporatory analysis of the visuals and analysis and discussion portion of the report before passing it along to my team members. And lastly, I assisted in drafting the final presentation slides.

By the end of the data visualization project, I was able to truly learn that a picture (or graph) does say a thousand words. Most of my data analysis did not include any advanced visualizations to tell a story about the dataset – I sorely rely on the different algorithmic methods I used to conduct my analysis. After completing this data visualization project, I will now incorporate visuals into my future analysis so that it can support my analysis overall.

3. Harika Rallapalli

My contributions to my group:

1. Participated during initial group discussion using d2l.

2.Participated in brainstorm on what are useful variables and how are they helpful during group meeting.

3. Discussed about what type of visualization to use during group meeting.

4.Created mosaic plot regarding assignment which is not selected as part of group report.

5.Worked on d3.js using html, CSS, Javascript and SVG for developing bar graph visualization using project dataset for final assignment.

6.Created stacked bar graph using tableau for clear visualization about lifestyle activities related to obesity of different gender which is not selected as part of group project report.

7.Created hierarchical bar chart for visualization of obesity effect on chronic diseases such as diabetes and High blood pressure.

As this is my first time working in a group I was very nervous initially but at the end of the project work, I was able to learn a lot on how to work in a group. Thanks to professor who provided this wonderful opportunity as part of the course work. This course helped me to explore various visualization techniques that are discussed clearly during class. I have came to know various visualization tools such as Tableau, R studio, d3.js and put hands on them during assignment and project work. I have used tableau to develop stacked bar chart in this project that i learnt from this course. I have used tableau to develop hierarchical bar chart in this project that i learnt from this course. I have used r studio to develop mosaic plot for assignment which is not selected in final presentation. I also gained basic knowledge on d3.js programming and also learned about js bin where i can run my code of d3.js. This course also

encouraged me to explore various journals papers regarding visualization in d2l.It helped me to know how to create simple and clear visualization example, when to use grid lines for good visualization. I learnt about how to use color constraint for a graph. This course helped me to gain a good knowledge in visualization using which i like to explore more in the future.

4. Judy Moran

I compiled several of our project deliverable documents, handed off for peer review from the group and submitted the final documents to D2L. Like the rest of the group, I also did extensive exploratory analysis and created visuals using the final file that Kari generated.

Early in the process each of us did independent exploratory analysis. I identified income level, gender, race and education as candidate categorical variables to look at along with BMI. These are variables everyone else eventually agreed on. Rebecca had some very interesting work connecting BMI percent to different illnesses but given our timeframes for deliverables we scaled down to demographic factors.

I've had some experience in previous DePaul classes generating basic visualizations for data analysis. This class really took it to a new level.

I had a steep learning curve working with R's ggplot2. Geoms, aesthetics, mappings, scales, guides, themes, etc. each presented it' own challenge. Once I understood how to layer aspects of the visuals though my visuals really became easier to read and understand. Learning to generate treemaps in R was also time consuming but eventually important for me. I used the R help() functionality and Google a lot in this class.

During the exploratory phase I created scores if not over a hundred versions of scatter plots, bar graphs, histograms, etc. I found how to layer boxplots (and histograms) on jittered scatterplots using http://r-statistics.co/Top50-Ggplot2-Visualizations-MasterList-R-Code.html.

The value of this visual, to me, is incorporating several relevant diagrams into a single visual with the intent to communicate important aspects of the data succinctly.

I also created the treemap shown in the final project. I easily created a treemap using Tableau for the first homework assignment but wanted to understand how to create one in R. The R Graphics Cookbook didn't have any information on treemaps, so again I used the R help() functionality and Google. This was a very time consuming and

complex graph for me. My initial graphs were gray scale with horrible labeling. The feedback from Eli and the group was to improve the appearance (obviously). I eventually learned how to work with the labels, aligning them so a user could read them, adding a legend and mapping and finally a color scheme that was user friendly and worked with the other visuals in the project.

Given that we're limited to 5-6 visuals for the final project, I think our visuals are excellent examples of what were presented in this course. Our exploratory visuals show our due diligence in analyzing the data and also why we eventually chose the variables we chose. In the 'Visualizations' section we clearly drill down and identify characteristics of obese respondents. I agree with our summary of data from our NHANES dataset.

Even though our work implies a final product I see it as only first level analysis. Given our conclusions and our intent to help people I want to know why these people are obese, how can we help them, do they want to be helped, etc.

I'm very lucky in this course to work on the 'Social Theorists' team. The hard work, feedback, teamwork and quality of the visuals has made this one of the best team experiences I've had at DePaul. In particular, Kari and Rebecca helped me immensely refine my coding so they're professional quality.

5. Kari Palmier

I found this project to be very challenging, but extremely interesting and beneficial. I do have some experience having to put together visualizations for analysis efforts in the past but these consisted of mostly color and glyph coded scatter plots, bar charts, and histograms, all of which were for scientific audiences. This is the first time I've had to really spend time thinking about what the best way to convey information to all types of people, which I found to be fun and challenging (was hard to switch gears at first from being science focused). I really enjoyed learning about all of the different types of visualizations and how people interpret different aspects of them (how we interpret shapes, lines, colors, etc). This project was a very good way for me to utilize what I learned in the class. Our group was the best group I've worked with so far at DePaul. The input from other group members really helped me to learn how to properly represent the concepts I chose. They also helped me learn about how to create different types of visualization that they chose to pursue. We were able to generate a wide range of visualizations because of how well we all worked together.

Below is a list of my contributions to this project:

1. Went through the descriptions for the original Demographics and Questionnaire datasets from Kaggle and selected an initial set of variables (had to go through approximately 1000 variables, but luckily a lot were not applicable).
2. Combined the variables selected from the Demographics and Questionnaire datasets by using the SEQN number of both (essentially performed a manual join on SEQN of the variables from the two datasets – sorted by SEQN first and verified the values and order matched before performing the join).
3. Analyzed the initial dataset to see if the variables had adequate entries to proceed. There were several variables that were mostly NaN (not answered). Removed all of the variables with over half NaN entries (this included all of the information on how often people work out).

4. Removed the variables related to specific diseases after our group decided to focus on more demographic variables.
5. Wrote code to import the dataset, created plots for relevant variable distributions immediately after importing, remove all rows with NaN, refused, or don't know values, remove rows with income entries of the two ranges (greater than $20K and less than $20K) since these overlap brackets present, created plots for same relevant variables as above to verify distributions were not changed after data cleaning, added BMI and obesity indicator variables, generated simple exploratory plots for all variable used to ensure there were no outliers present, and created new variables containing the string descriptions for each categorical value of the variables of interest. I then saved off this final dataset to a csv file so that my group members could use it.
6. Created numerous initial visualizations such as heatmaps versus all interesting categorical variables and age, different types of hierarchical bar charts, and numerous exploratory plots.
7. Created polished exploratory histograms and bar charts used to analyze the different distributions of the variables of interest (some of these are included in Figure 8 in the Exploratory Analysis Appendix section).
8. Created stacked bar chart exploratory visualizations that show the number of obese people versus the total number of people (used in Figures 1 and 2).
9. Created several iterations of heatmaps over age and the variables of interest. After analysis and review by group members and the class, created the final heatmaps in Figure 6.
10. Created hierarchical bar charts of the percentage of obese people over race, income, education, and marital status, all divided by gender (Figure 5).
11. Created the final presentation used in class.
12. Updated the final document to clarify wording of some sections, as well as provide the analysis of the visualizations I created.

13. Attended all group meetings
14. Worked with other group members on how to make their and my own visualizations more clear and polished.

III. Variables

Dataset	Variable Name	Variable Description	Data File Name	Label	Variable Values
Demographic	SEQN	Respondent sequence number.	CDEMO_EH	Respondent sequence number	
Demographic	RIAGENDR	Gender of the participant.	CDEMO_EH	Gender	1 - Male 2 - Female
Demographic	RIDAGEYR	Age in years of the participant at the time of screening. Individuals 80 and over are topcoded at 80 years of age.	CDEMO_EH	Age in Years at Screening	0 to 79 - Value 80 - Anyone 80 and 0lder
Demographic	RIDRETH1	RIDRETH1 Recode of reported race and Hispanic origin information	CDEMO_EH	Race/Hispanic Origin	1 - Mexican American 2 - Other Hispanic 3 - Non-Hispanic White 4 - Non-Hispanic Black 5 - Other Race - Including Multi-Racial
Demographic	DMDEDUC2	What is the highest grade or level of school {you have/SP has} completed or the highest degree {you have/s/he has} received?	CDEMO_EH	Education Level - Adults 20+	1 - Less than 9th grade 2 - 9 to 11th grade (includes 12th with no diploma 3 - High school graduate/GED or equivalent 4 - Some college or AA degree 5 - College graduate or above 7 - Refused 9 - Don't Know
Demographic	DMDMARTL	Marital status	CDEMO_EH	Marital Status	1 - Married 2 - Widowed 3 - Divorced 4 - Separated 5 - Never married 6 - Living with partner 77 - Refused 99 - Don't Know

Demographic	INDHHIN2	Total household income (reported as a range value in dollars)	CDEMO_EH	Annual Household Income	1 - $0 to $4,999 2 - $5,000 to $9,999 3 - $15,000 to $19,999 4 - $20,000 to $24,999 5 - $20,000 to $24,999 6 - $25,000 to $34,999 7 - $35, 000 to $44,999 8 - $45,000 to $54,999 9 - $55,000 to $64,999 10 - $65,000 to $74,999 12 - $20,000 and Over 13 - Under $20,000 14 - $75,000 to $99,999 15 - $100,000 and Over 77 - Refused 99 - Don't Know
Demographic	INDFMPIR	A ratio of family income to poverty guidelines.	CDEMO_EH	Ratio of family income to poverty	0 to 4.99 - Values 5 - 5 and Above
Questionnaire	BPQ020	{Have you/Has SP} ever been told by a doctor or other health professional that {you/s/he} had hypertension, also called high blood pressure?	BPQ_H	Ever told you had high blood pressure	1 - Yes 2 - No 7 - Refused 9 - Don’t Know
Questionnaire	CBD070	The next questions are about how much money {your family spends/you spend} on food. First, I will ask you about money spent at supermarkets or grocery stores. Then we will talk about money spent at other types of stores. During the past 30 days, how much money {did your family/did you} spend at supermarkets or grocery stores? Please include purchases made with food stamps	CBQ_H	Money spent at supermarket/ grocery store	0 to 4285 - Values 777777 - Refused 999999 - Don't Know
Questionnaire	CBD090	About how much money was spent on nonfood items?	CBQ_H	Money spent on nonfood items	0 to 1542 - Values 777777 - Refused 999999 - Don't Know
Questionnaire	CBD120	During the past 30 days, how much money {did your family/did you} spend on eating out? Please include money spent in cafeterias at work or at school or on vending machines, for all family members.	CBQ_H	Money spent on eating out	0 to 2142 - Values 777777 - Refused 999999 - Don't Know

Questionnaire	CBD130	During the past 30 days, how much money {did your family/did you} spend on food carried out or delivered? Please do not include money you have already told me about.	CBQ_H	Money spent on carryout/delivered foods	0 to 1028 - Values 777777 - Refused 999999 - Don't Know
Questionnaire	DIQ010	The next questions are about specific medical conditions. {Other than during pregnancy, {have you/has SP}/ {Have you/Has SP}} ever been told by a doctor or health professional that {you have/ {he/she/SP} has} diabetes or sugar diabetes?	DIQ_H	Doctor told you have diabetes	1 - Yes 2 - No 3 - Borderline 7 - Refused 9 - Don't Know
Questionnaire	DBD895	Next, I am going to ask you about meals. By meal, I mean breakfast, lunch and dinner. During the past 7 days, how many meals {did you/did SP} get that were prepared away from home in places such as restaurants, fast food places, food stands, grocery stores, or from vending machines? {Please do not include meals provided as part of the school lunch or school breakfast./Please do not include meals provided as part of the community programs you reported earlier.}	DBQ_H	# of meals not home prepared	1 to 21 - Values 0 - None 5555 - More than 21 7777 - Refused 9999 - Don't Know
Questionnaire	DBD900	How many of those meals {did you/did SP} get from a fast food or pizza place?	DBQ_H	# of meals from fast food or pizza place	1 to 21 - Values 0 - None 5555 - More than 21 7777 - Refused 9999 - Don't Know

Questionnaire	DBD905	Some grocery stores sell "ready to eat" foods such as salads, soups, chicken, sandwiches and cooked vegetables in their salad bars and deli counters. During the past 30 days, how often did {you/SP} eat "ready to eat" foods from the grocery store? Please do not include sliced meat or cheese you buy for sandwiches and frozen or canned foods.	DBQ_H	# of ready-to-eat foods in past 30 days	1 to 180 - Values 0 - None 5555 - More than 21 7777 - Refused 9999 - Don't Know
Questionnaire	DBD910	During the past 30 days, how often did you {SP} eat frozen meals or frozen pizzas? Here are some examples of frozen meals and frozen pizzas.	DBQ_H	# of frozen meals/pizza in past 30 days	1 to 180 - Values 0 - None 5555 - More than 21 7777 - Refused 9999 - Don't Know
Questionnaire	INQ244	Do {you/NAMES OF OTHER FAMILY/you and NAMES OF FAMILY MEMBERS} have more than $5,000 in savings at this time? Please include money in your checking accounts.	INQ_H	Family has savings more than $5000	1 - Yes 2 - No 7 - Refused 9 - Don’t Know
Questionnaire	IND247	Total savings or cash assets at this time for {you/NAMES OF OTHER FAMILY/your family}.	INQ_H	Total savings/cash assets for the family	1 - Less than $500 2 - $501 to $1,000 3 - $1,001 to $2000 4 - $2,001 to $3,000 5 - $3,001 to $4,000 6 - $4,001 to $5,000 77 - Refused 99 - Don't Know
Questionnaire	MCQ080	Has a doctor or other health professional ever told {you/SP} that {you were/s/he/SP was} overweight?	MCQ_H	Doctor ever said you were overweight	1 - Yes 2 - No 7 - Refused 9 - Don’t Know
Questionnaire	MCQ365a	To lower {your/SP's} risk for certain diseases, during the past 12 months {have you/has s/he} ever been told by a doctor or health professional to: control {your/his/her} weight or lose weight?	MCQ_H	Doctor told you to lose weight	1 - Yes 2 - No 7 - Refused 9 - Don’t Know

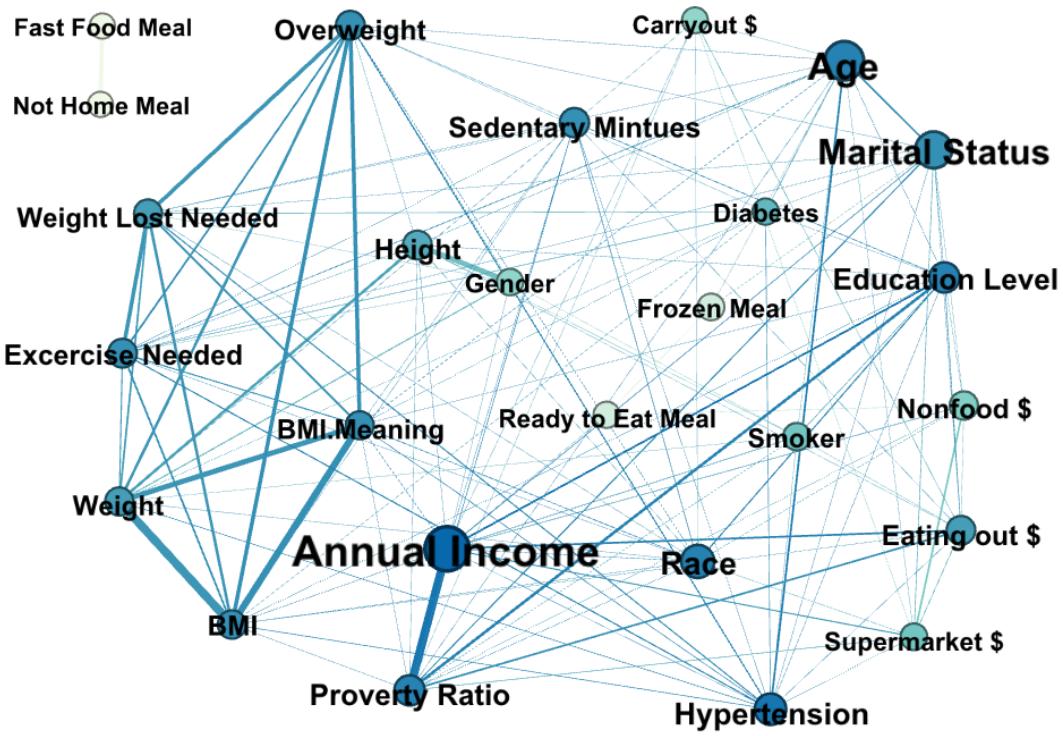
Questionnaire	MCQ365b	To lower {your/SP's} risk for certain diseases, during the past 12 months {have you/has s/he} ever been told by a doctor or health professional to: increase {your/his/her} physical activity or exercise?	MCQ_H	Doctor told you to exercise	1 - Yes 2 - No 7 - Refused 9 - Don't Know
Questionnaire	PAD680	The following question is about sitting at school, at home, getting to and from places, or with friends including time spent sitting at a desk, traveling in a car or bus, reading, playing cards, watching television, or using a computer. Do not include time spent sleeping. How much time {do you/does SP} usually spend sitting on a typical day?	PAQ_H	Minutes sedentary activity	0 to 1200 - Values 7777 - Refused 9999 - Don't Know
Questionnaire	SMQ020	These next questions are about cigarette smoking and other tobacco use. {Have you/Has SP} smoked at least 100 cigarettes in {your/his/her} entire life?	SMQ_H	Smoked at least 100 cigarettes in life	1 - Yes 2 - No 7 - Refused 9 - Don't Know
Questionnaire	WHD010	These next questions ask about {your/SP's} height and weight at different times in {your/his/her} life. How tall {are you/is SP} without shoes?	WHQ_H	Current self-reported height (inches)	48 to 81 - Values 7777 - Refused 9999 - Don't Know
Questionnaire	WHD020	How much {do you/does SP} weigh without clothes or shoes?	WHQ_H	Current self-reported weight (pounds)	75 to 493 - Values 7777 - Refused 9999 - Don't Know

IV. Corresponding Code for Visuals

1. Rebecca Tung

1) Force Direct Network Diagram for CA Health and Nutrition Survey Variables

This diagram is created by using Gephi. Please see the attached RebeccaTung_CSC465_Proejct.gephi file for the generated source code



2) R code for the correlogram and the creation of the two cvs files for Gephi to create the force direct diagram

```
#####################################################################
#############################################################
#
```

```
#  Rebecca Tung CSC 465 Final Project
#
##########################################################################
##############################################################

library(ggplot2)
library(lubridate)
library(plyr)
#library(mosaic)
library(scales)
library(psych)
library(chron)
library(ggcorrplot)
library(lattice)
library(treemap)
library(xtable)
library(devtools)
library(easyGgplot2)
library(zoo)
library("igraph")
library("plyr")
library(xlsx)

##########################################################################
##############################################################
#
#  Clean Data
#
##########################################################################
##############################################################
# Clear workspace
rm(list = ls())

setwd('/Users/rebeccatung/Desktop/School/Depaul/CSC 465/Project/findings')
nhanesData = read.table("NHANES_ConbinedProjectDataset.csv", sep=",", header=T)
#nhanesData$SEQN = nhanesData$SEQN
nhanesData$SEQN = NULL

# Print summary of original data
nrow(nhanesData)
summary(nhanesData)
describe(nhanesData)
colSums(is.na(nhanesData))
head(nhanesData)
```

```
str(nhanesData)


refusedEntries = c(0, 0, 0, 7, 77, 77, 0, 7, 777777, 777777, 777777, 777777, 7, 7777,
7777, 7777, 7777, 7, 7, 7, 7777, 7, 7777, 7777, 0)
dkEntries = c(0, 0, 0, 9, 99, 99, 0, 9, 999999, 999999, 999999, 999999, 9, 9999, 9999,
9999, 9999, 9, 9, 9, 9999, 9, 9999, 9999, 0)

# Remove NaN values
attrs = colnames(nhanesData)
numAttrs = length(attrs)
for (i in 1:numAttrs){
  current_attr = attrs[i]
  notNaNdx = !is.na(nhanesData[current_attr])
  nhanesData = nhanesData[notNaNdx,]

  print(i)
  print(current_attr)
  print('After NaN Removal')
  print(nrow(nhanesData))

}

# Remove refused values
for (i in 1:numAttrs){
  current_attr = attrs[i]

  refNdx = nhanesData[current_attr] == refusedEntries[i]
  if (any(refNdx)){
    nhanesData = nhanesData[!refNdx,]
  }

  print(i)
  print(current_attr)
  print(refusedEntries[i])
  print('After Refused Removal')
  print(nrow(nhanesData))

}

# Remove don't know values
for (i in 1:numAttrs){
  current_attr = attrs[i]
```

```
  dkNdx = nhanesData[current_attr] == dkEntries[i]
  if (any(dkNdx)){
    nhanesData = nhanesData[!dkNdx,]
  }

  print(i)
  print(current_attr)
  print(dkEntries[i])
  print('After DK Removal')
  print(nrow(nhanesData))

}

# Print summary of filtered data
nrow(nhanesData)
summary(nhanesData)
describe(nhanesData)
colSums(is.na(nhanesData))
head(nhanesData)
str(nhanesData)

##############################################################################
#############################################################
#
# Find Correlation among variables
#
##############################################################################
#############################################################


# Create BMI attribute
lb_to_kg_const =  0.45359237
inch_to_m_const = 0.025
nhanesData$BMI = ((nhanesData$WHD020 * lb_to_kg_const) / ((nhanesData$WHD010
* inch_to_m_const)^2))
#Create BMI Classification
nhanesData$BMIMeaning <-ifelse(nhanesData$BMI<19,1,
                       ifelse(nhanesData$BMI>=19 & nhanesData$BMI<25,2,
                            ifelse(nhanesData$BMI>=25 & nhanesData$BMI<30,3,4
                            )))
nhanesData=nhanesData[ which(as.numeric(nhanesData$BMI) <= 100),]

#Make column name more meaniful
names(nhanesData)
```

```
colnames(nhanesData) <- c("Gender", "Age", "Race", "Education Level", "Marital
Status",
                          "Annual Income", "Proverty Ratio", "Hypertension", "Supermarket $",
                          "Nonfood $", "Eating out $", "Carryout $", "Diabetes", "Not Home Meal",
                          "Fast Food Meal", "Ready to Eat Meal", "Frozen Meal", "Overweight",
"Weight Lost Needed","Excercise Needed",
                          "Sedentary Mintues", "Smoker", "Height", "Weight", "BMI",
"BMI.Meaning")
#relation <- cor(nhanesData[,unlist(lapply(nhanesData, is.numeric))])
relation <- cor(nhanesData)

# Correlation matrix
corr <- round(relation, 1)


dev.off()
par(xpd=TRUE)
corrplot(corr,
       type ="lower",
       method = "color",
       addgrid.col = "darkgray",
       tl.offset = 0.1,
       diag = FALSE,
       order = "alphabet",
       tl.cex = 0.6, tl.col = 'black',
       cl.cex = 0.75,
       addCoef.col = "black", number.digits = 2, number.cex = 0.5,
       col = colorRampPalette(c("tomato2", "white", "springgreen3"))(100),
       mar = c(0,0,1,0))


#########################################################
#home work 4 1.a
#########################################################

# Plotting networks in R
# An example how to plot networks and customize their appearance in Cytoscape
directly from R, using RCytoscape package

#########################################################################
####################

# Load libraries
library("igraph")
```

```
library("plyr")
library("reshape2")

# Read a data set.
# Data format: dataframe with 3 variables; variables 1 & 2 correspond to interactions;
variable 3 corresponds to the weight of interaction
#dataSet <- read.table("lesmis.txt", header = FALSE, sep = "\t")

upperTriangle<-upper.tri(corr, diag=F) #turn into a upper triangle
correlations.upperTriangle<-corr #take a copy of the original cor-mat
correlations.upperTriangle[!upperTriangle]<-NA#set everything not in upper triangle o
NA
correlations_melted<-na.omit(melt(correlations.upperTriangle, value.name
="correlationCoef")) #use melt to reshape the matrix into triplets, na.omit to get rid of
the NA rows
colnames(correlations_melted)<-c("V1", "V2", "V3")
dataSet <- correlations_melted[ which(correlations_melted$V3 != 0), ]
dataSet$V3 <- dataSet$V3^2
rownames(dataSet) <- 1:nrow(dataSet)
Cahealth <- dataSet
colnames(Cahealth)<-c("Source", "Target", "Weight")
CAName <- data.frame(unique(correlations_melted$V2),
unique(correlations_melted$V2))
colnames(CAName)<-c("id", "label")

write.csv(Cahealth, file = "CAHealthData.csv", row.names=FALSE)
write.csv(CAName, file = "CAHealthName.csv", row.names=FALSE)
```

3) R code for miscellaneous diagrams

```
#############################################################
#############################################################
##############
#
#  Rebecca Tung CSC 465 Final Project
#
#############################################################
#############################################################
##############

library(ggplot2)
library(lubridate)
library(plyr)
#library(mosaic)
library(scales)
library(psych)
library(chron)
library(ggcorrplot)
library(lattice)
library(treemap)
library(xtable)
library(devtools)
library(easyGgplot2)
library(zoo)


#############################################################
#############################################################
##############
#
#  Clean Data
#
#############################################################
#############################################################
##############
setwd('/Users/rebeccatung/Desktop/School/Depaul/CSC 465/Project/findings')
nhanesData = read.table("NHANES_ConbinedProjectDataset.csv", sep=",", header=T)
#nhanesData$SEQN = nhanesData$SEQN
nhanesData$SEQN = NULL

# Print summary of original data
nrow(nhanesData)
summary(nhanesData)
describe(nhanesData)
```

```
colSums(is.na(nhanesData))
head(nhanesData)
str(nhanesData)


refusedEntries = c(0, 0, 0, 7, 77, 77, 0, 7, 777777, 777777, 777777, 777777, 7, 7777,
7777, 7777, 7777, 7, 7, 7, 7777, 7, 7777, 7777, 0)
dkEntries = c(0, 0, 0, 9, 99, 99, 0, 9, 999999, 999999, 999999, 999999, 9, 9999, 9999,
9999, 9999, 9, 9, 9, 9999, 9, 9999, 9999, 0)

# Remove NaN values
attrs = colnames(nhanesData)
numAttrs = length(attrs)
for (i in 1:numAttrs){
  current_attr = attrs[i]
  notNaNdx = !is.na(nhanesData[current_attr])
  nhanesData = nhanesData[notNaNdx,]

  print(i)
  print(current_attr)
  print('After NaN Removal')
  print(nrow(nhanesData))

}

# Remove refused values
for (i in 1:numAttrs){
  current_attr = attrs[i]

  refNdx = nhanesData[current_attr] == refusedEntries[i]
  if (any(refNdx)){
    nhanesData = nhanesData[!refNdx,]
  }

  print(i)
  print(current_attr)
  print(refusedEntries[i])
  print('After Refused Removal')
  print(nrow(nhanesData))

}

# Remove don't know values
for (i in 1:numAttrs){
  current_attr = attrs[i]
```

```
  dkNdx = nhanesData[current_attr] == dkEntries[i]
  if (any(dkNdx)){
    nhanesData = nhanesData[!dkNdx,]
  }

  print(i)
  print(current_attr)
  print(dkEntries[i])
  print('After DK Removal')
  print(nrow(nhanesData))

}

# Print summary of filtered data
nrow(nhanesData)
summary(nhanesData)
describe(nhanesData)
colSums(is.na(nhanesData))
head(nhanesData)
str(nhanesData)

##############################################################
##############################################################
##############
#
# Find Correlation among variables
#
##############################################################
##############################################################
##############


# Create BMI attribute
lb_to_kg_const =  0.45359237
inch_to_m_const = 0.025
nhanesData$BMI = ((nhanesData$WHD020 * lb_to_kg_const) /
((nhanesData$WHD010 * inch_to_m_const)^2))
#Create BMI Classification
nhanesData$BMIMeaning <-ifelse(nhanesData$BMI<19,1,
                ifelse(nhanesData$BMI>=19 & nhanesData$BMI<25,2,
                    ifelse(nhanesData$BMI>=25 & nhanesData$BMI<30,3,4
                        )))
nhanesData=nhanesData[ which(as.numeric(nhanesData$BMI) <= 100),]

#Make column name more meaniful
names(nhanesData)
```

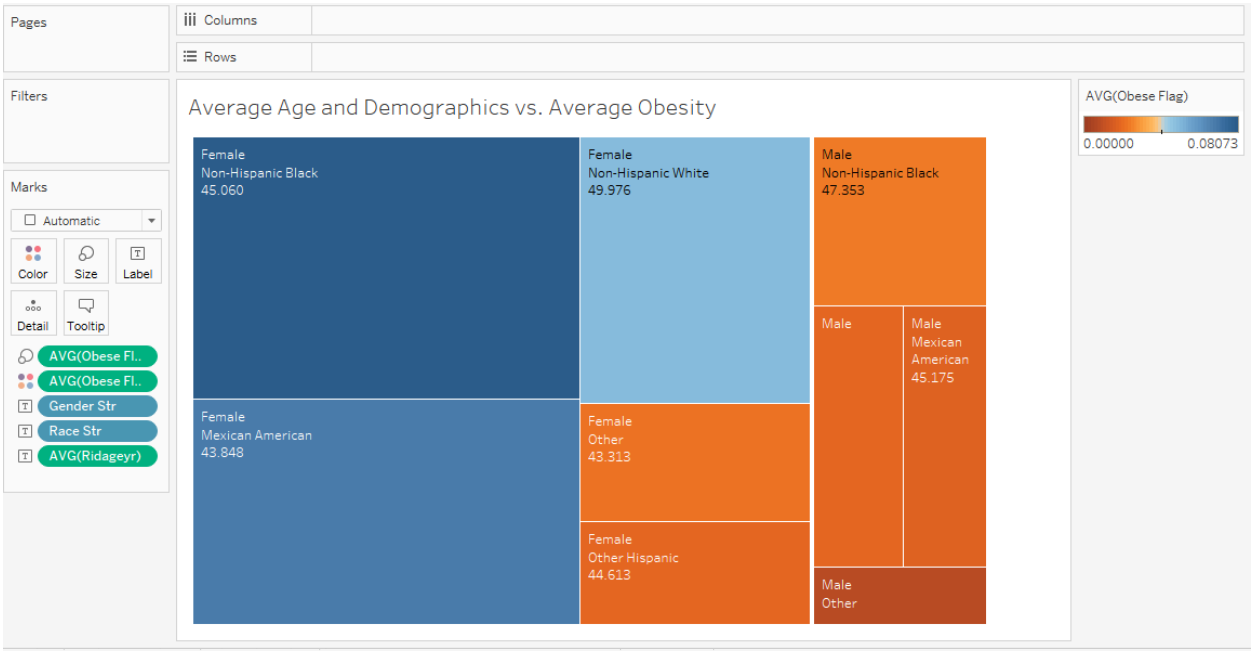
```
colnames(nhanesData) <- c("Gender", "Age", "Race", "Education Level", "Marital
Status",
                          "Annual Income", "Proverty Ratio", "Hypertension", "Supermarket $",
                          "Nonfood $", "Eating out $", "Carryout $", "Diabetes", "Not Home Meal",
                          "Fast Food Meal", "Ready to Eat Meal", "Frozen Meal", "Overweight",
"Weight Lost Needed","Excercise Needed",
                          "Sedentary Mintues", "Smoker", "Height", "Weight", "BMI",
"BMI.Meaning")
#relation <- cor(nhanesData[,unlist(lapply(nhanesData, is.numeric))])
relation <- cor(nhanesData)

# Correlation matrix
corr <- round(relation, 1)

# Plot Correlation
ggcorrplot(corr,
 #         hc.order = TRUE,
          type = "lower",
          lab = TRUE,
          lab_size = 2,
#           method="circle",
           show.diag = TRUE,
          colors = c("tomato2", "white", "springgreen3"),
          title="Correlation of National Health and Nutrition Examination Survey data",
          ggtheme=theme_bw) +
  theme(plot.title = element_text(hjust = 0.5))


##############################################################
##############################################################
##############
#
# Create Heap map to show relations between Hypertesnion and Age/BMI
#
##############################################################
##############################################################
##############

# Create Age Range
nhanesData$ageRange <-ifelse(nhanesData$Age<30,"20-29",
                       ifelse(nhanesData$Age < 40 & nhanesData$Age > 30 ,"30-39",
                       ifelse(nhanesData$Age < 50 & nhanesData$Age > 40 ,"40-49",
                       ifelse(nhanesData$Age < 60 & nhanesData$Age > 50 ,"50-59",
                       ifelse(nhanesData$Age < 70 & nhanesData$Age > 60 ,"60-69",
                      ifelse(nhanesData$Age < 80 & nhanesData$Age > 70 ,"70-79",
"80+"))))))
```

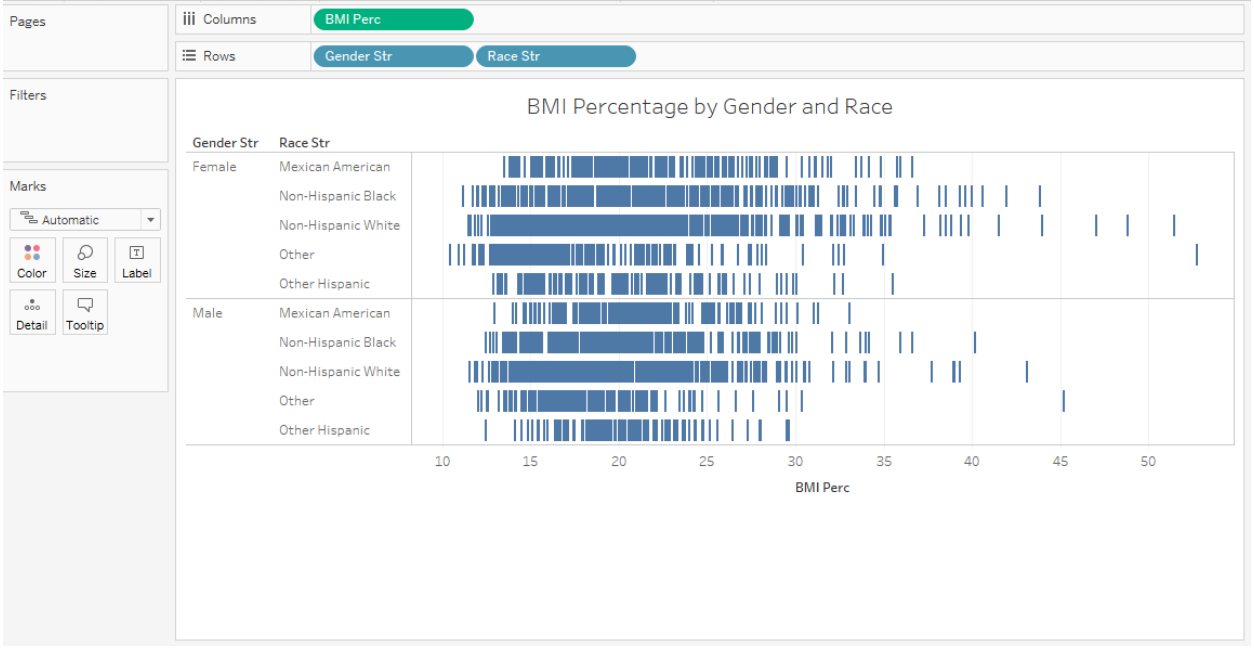
```
#Classify BMI
nhanesData$BMI.Classification <-ifelse(nhanesData$BMI<19,"Under Weight",
                    ifelse(nhanesData$BMI>=19 & nhanesData$BMI<25,"Normal
Weight",
                           ifelse(nhanesData$BMI>=25 &
nhanesData$BMI<30,"Overweight","Obesity"
                           )))

#Make Hypertension into readable format
nhanesData$Hypertension <-ifelse(nhanesData$Hypertension == 1,"Hypertension",
                           ifelse(nhanesData$Hypertension == 2,"Non-
Hypertension","Unknown"
                                  ))
#Classify BMI
nhanesData$BMI.Classification <-ifelse(nhanesData$BMI<19,"Under Weight",
                           ifelse(nhanesData$BMI>=19 & nhanesData$BMI<25,"Normal
Weight",
                                  ifelse(nhanesData$BMI>=25 &
nhanesData$BMI<30,"Overweight","Obesity"
                                  )))
#Create heatmap
ggplot(nhanesData,aes(x = Age, y=round(as.numeric(BMI),0), fill=Hypertension)) +
 geom_tile() +
 # scale_x_continuous(breaks = seq(20, 80, by = 5)) +
 theme_bw() +
 scale_fill_brewer(palette = "Pastel1") +
 # scale_fill_discrete( h =c(220,260)) +
 labs(title = "Hyptertension Distribution",
     x = "Age", y = "BMI ") +
 theme(plot.title = element_text(hjust = 0.5))

##############################################################
##############################################################
##############
#
# Create panel plat to show correlation between Hypertesnion and Age/BMI
#
##############################################################
##############################################################
##############

nhanesData$MaritalStatus.Meaning <-NULL
nhanesData$MaritalStatus.Meaning <-ifelse(nhanesData$`Marital Status` ==
1,"Married",
                           ifelse(nhanesData$`Marital Status` == 2,"Widowed",
                           ifelse(nhanesData$`Marital Status` == 3,"Divoced",
```

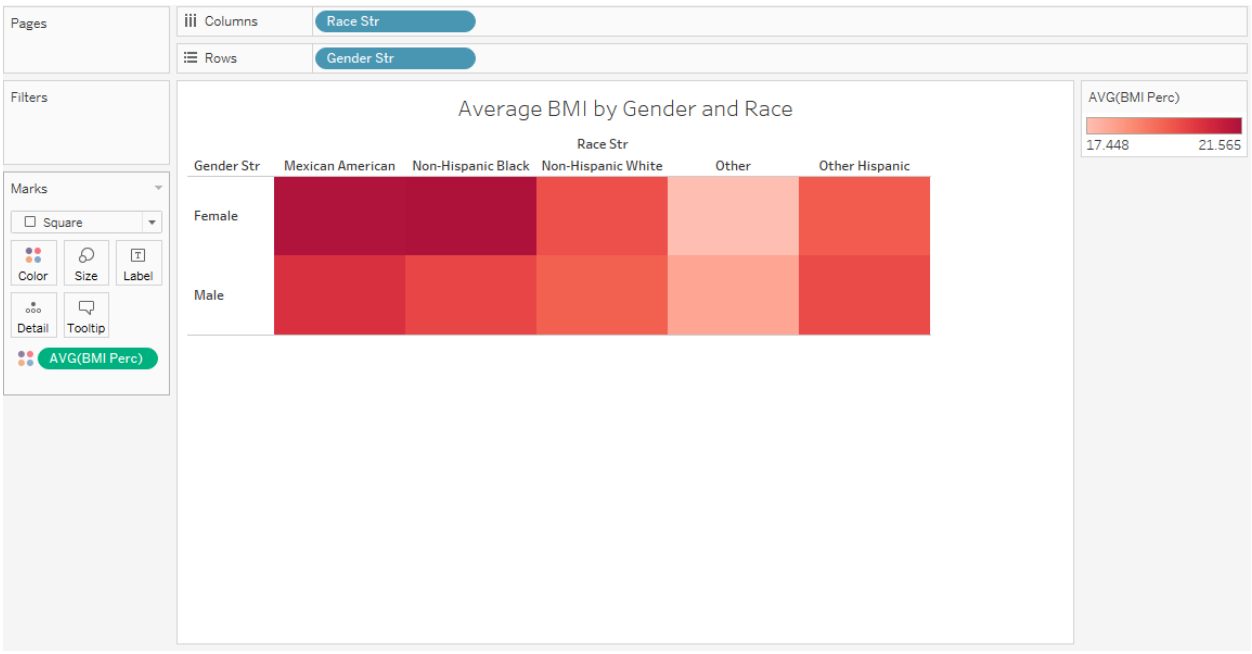
```
                        ifelse(nhanesData$`Marital Status` == 4,"Separated",
                        ifelse(nhanesData$`Marital Status` == 5,"Never Married","Living
w Partner")))))
#Create panel Plots
ggplot(nhanesData,aes(x = Age, y=round(as.numeric(BMI),0))) +
  geom_point(aes(color=Hypertension), shape=21)+
  theme_bw() +
  labs(title = "Hyptertension Distribution",
       x = "Age", y = "BMI ") +
  theme(plot.title = element_text(hjust = 0.5)) +
  facet_grid(MaritalStatus.Meaning ~ .)
```

2. Michelle Tang

Used Tableau

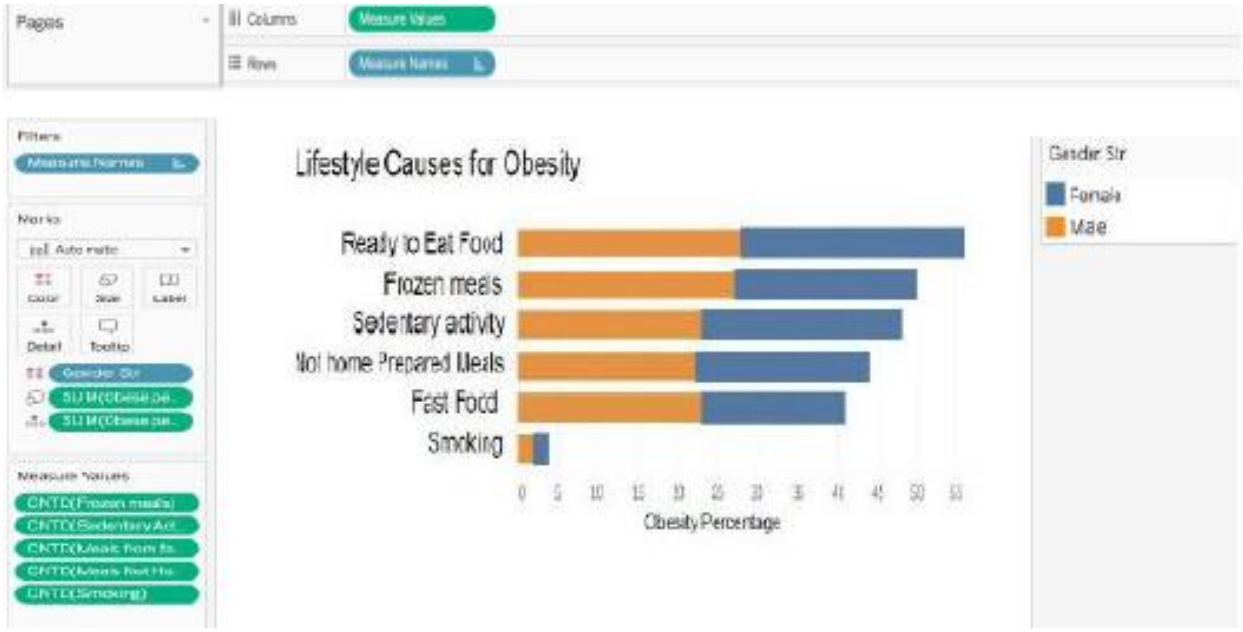





Pages
Columns
Race Str
Rows
Gender Str
Filters
Marks
Square
Color
Size
Label
Detail
Tooltip
AVG(BMI Perc)
Average BMI by Gender and Race
Race Str
Gender Str
Mexican American
Non-Hispanic Black
Non-Hispanic White
Other
Other Hispanic
Female
Male
AVG(BMI Perc)
17.448
21.565

3. Harika Rallapalli

Used Tableau



//Code in d3.js to create bar chart of bmi for various ages(it is not selected for presentation)

```
<!DOCTYPE html>
<meta charset="utf-8">

<head>
        <style>

        .axis {
          font: 10px sans-serif;
        }

        .axis path,
        .axis line {
          fill: none;
          stroke: #000;
          shape-rendering: crispEdges;
        }

        </style>
</head>

<body>
```

```
<script src="http://d3js.org/d3.v3.min.js"></script>

<script>

var margin = {top: 20, right: 20, bottom: 70, left: 40},
    width = 600 - margin.left - margin.right,
    height = 300 - margin.top - margin.bottom;
var x = d3.scale.ordinal().rangeRoundBands([0, width], .05);

var y = d3.scale.linear().range([height, 0]);

var xAxis = d3.svg.axis()
    .scale(x)
    .orient("bottom")
    .tickFormat(d3.format("%0"));

var yAxis = d3.svg.axis()
    .scale(y)
    .orient("left")
    .ticks(10);

var svg = d3.select("body").append("svg")
    .attr("width", width + margin.left + margin.right)
    .attr("height", height + margin.top + margin.bottom)
  .append("g")
    .attr("transform",
          "translate(" + margin.left + "," + margin.top + ")");

d3.csv("bar-data.csv", function(error, data) {

    data.forEach(function(d) {
        d.bmi = parsebmi(d.bmi);
        d.value = +d.value;
    });

  x.domain(data.map(function(d) { return d.bmi; }));
  y.domain([0, d3.max(data, function(d) { return d.value; })]);
```

```
  svg.append("g")
     .attr("class", "x axis")
     .attr("transform", "translate(0," + height + ")")
     .call(xAxis)
   .selectAll("text")
     .style("text-anchor", "end")
     .attr("dx", "-.8em")
     .attr("dy", "-.55em")
     .attr("transform", "rotate(-90)" );

  svg.append("g")
     .attr("class", "y axis")
     .call(yAxis)
   .append("text")
     .attr("transform", "rotate(-90)")
     .attr("y", 6)
     .attr("dy", ".71em")
     .style("text-anchor", "end")
     .text("Value ($)");

  svg.selectAll("bar")
     .data(data)
   .enter().append("rect")
     .style("fill", "steelblue")
     .attr("x", function(d) { return x(d.bmi); })
     .attr("width", x.rangeBand())
     .attr("y", function(d) { return y(d.value); })
     .attr("height", function(d) { return height - y(d.value); });

});

</script>

</body>
```

Used R Studio

//Mosaic plot to know Obesity among different Races by means of Gender

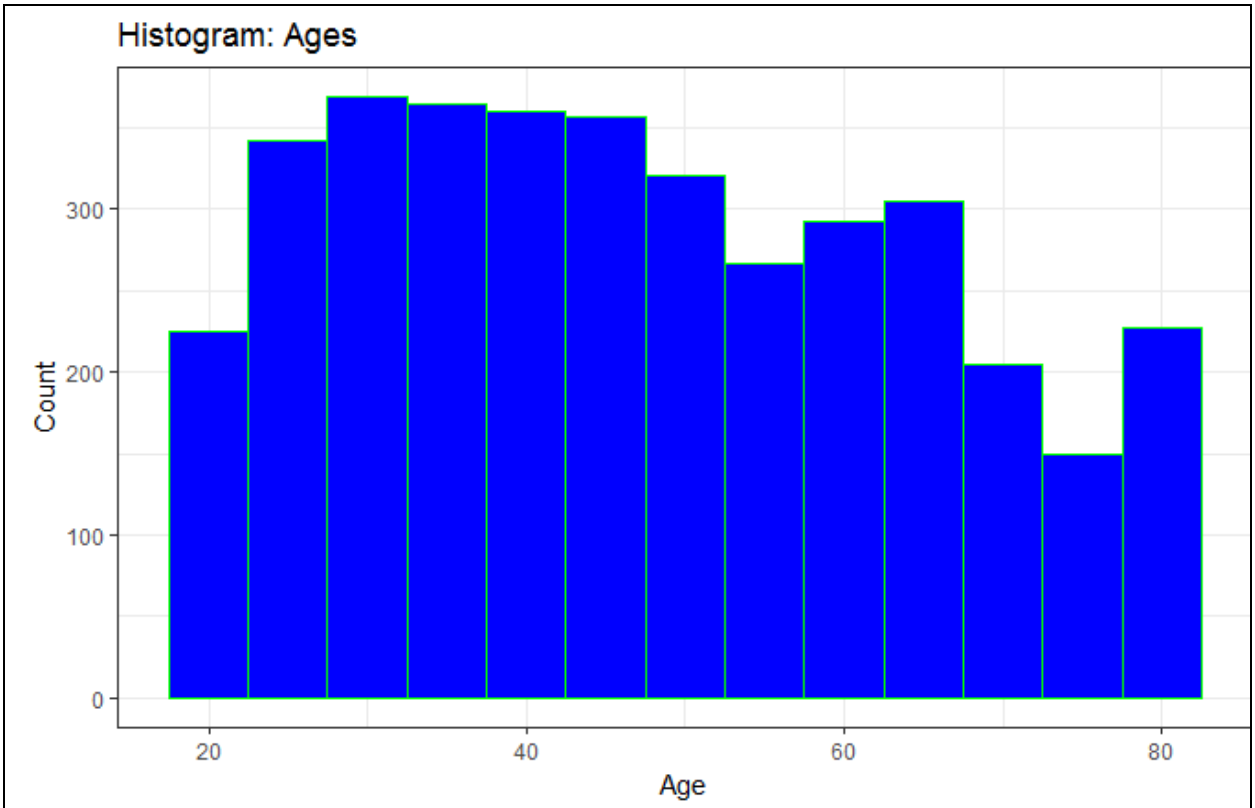
```
> library(readr)
>
NHANES_Filtered_ConbinedProjectDataset_csv_NHANES_Filtered_ConbinedProjectDataset_csv
<- read_csv("~/Desktop/data visualization assignment -
3/NHANES_Filtered_ConbinedProjectDataset.csv -
NHANES_Filtered_ConbinedProjectDataset.csv.csv")
>
View(NHANES_Filtered_ConbinedProjectDataset_csv_NHANES_Filtered_ConbinedProjectDatas
et_csv)
>
dimnames(NHANES_Filtered_ConbinedProjectDataset_csv_NHANES_Filtered_ConbinedProject
Dataset_csv)
> install.packages("Grid")
> library(grid)
> library(vcd)
> mosaic( ~ ObeseFlag + GenderStr + RaceStr,
data=NHANES_Filtered_ConbinedProjectDataset_csv_NHANES_Filtered_ConbinedProjectDatas
et_csv)
> mosaic( ~ ObeseFlag + GenderStr + RaceStr,
data=NHANES_Filtered_ConbinedProjectDataset_csv_NHANES_Filtered_ConbinedProjectDatas
et_csv, highlighting="ObeseFlag", highlighting_fill=c("lightblue", "pink"),
```

4. Judy Moran

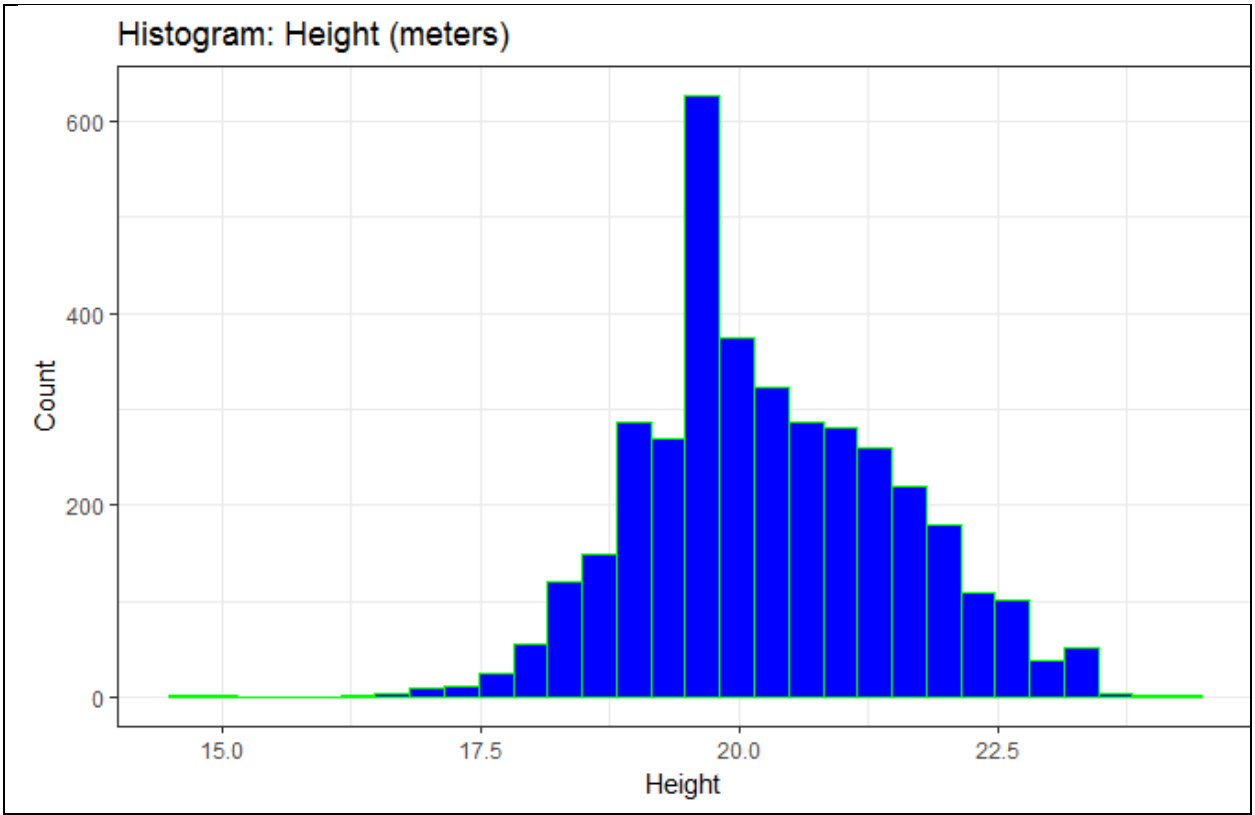
EXPLORATORY WORK: Kari created one of our final datasets (nhanesData.csv) from the original demographic and questionnaire datasets. There are 3786 rows of data in the nhanesData.csv dataset.
My initial exploration concentrated on BMI_Percent along with age, height, weight,]gender (), income (), race () and marital status ().

Variable	Observation
RIDAGEYR	Ages relatively consistent. Some skewing to the right which is expected with human mortality. Any respondent age 80+ is defaulted to age 80.
Height_m	Height in meters is relatively normally distributed as expected. There is a slight left skew.
Weight_kg	Weight is approximately normally distributed but with obvious skewing to the right. This is the data we're interested in, i.e. weights
RIAGENDR/ GenderStr	It appears that there are approximately the same number of respondents that are Male as Female

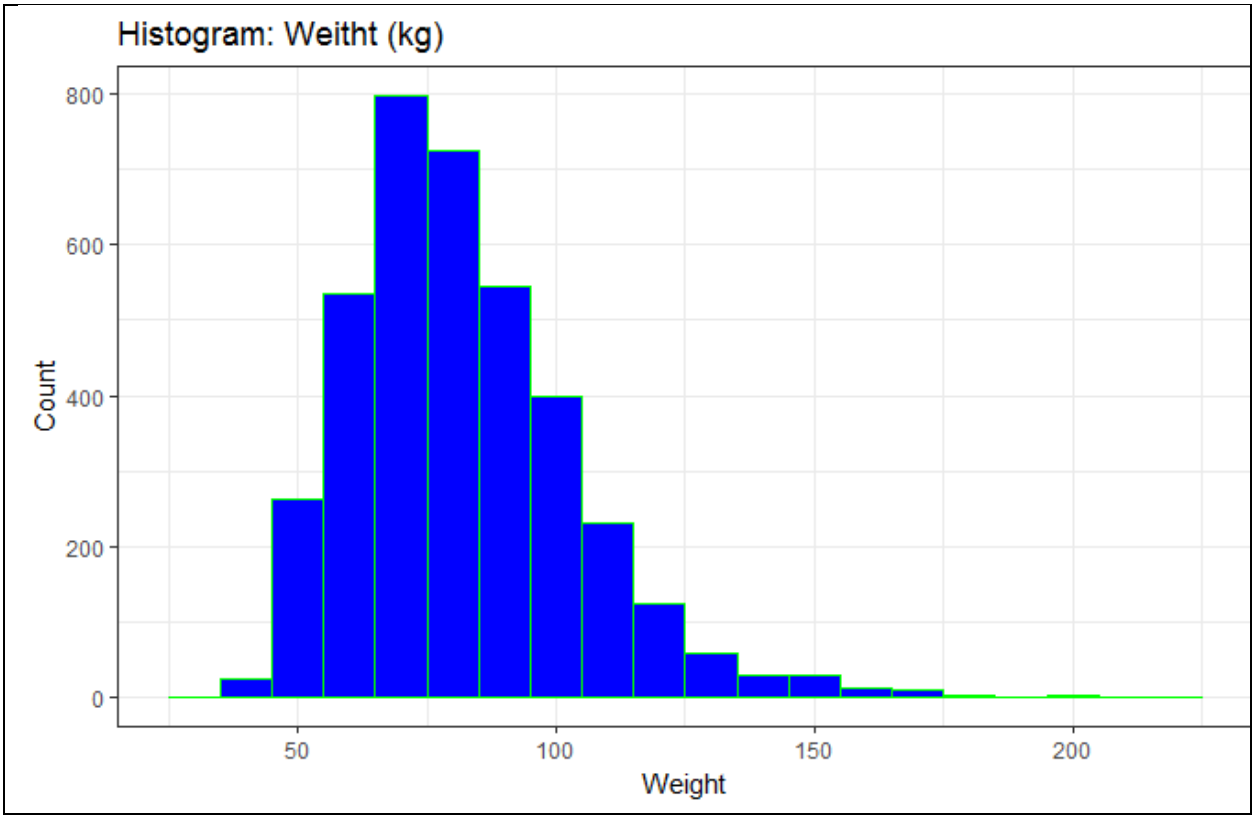
It looks like age is relatively consistent. The age 80 captures all respondents age 80+. There's a slight right skew but this is consistent with human mortality.



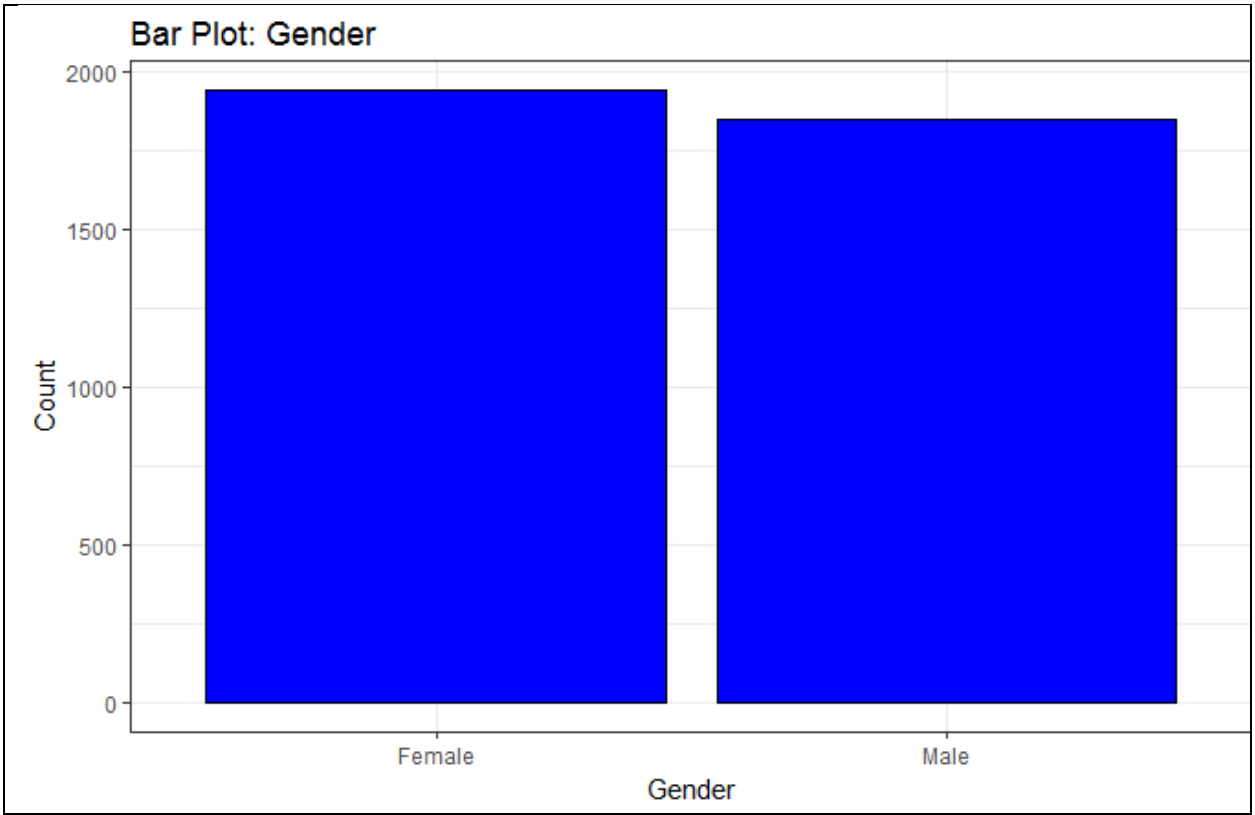

```
# Age relatively consistent - slight right skewed - all 80+ ages captured with 80
ggplot(nhanesData, aes(x=RIDAGEYR)) +
  geom_histogram(binwidth=5, colour="green", fill="blue") +
  labs(x="Age", y="Count", title="Histogram: Ages")
```



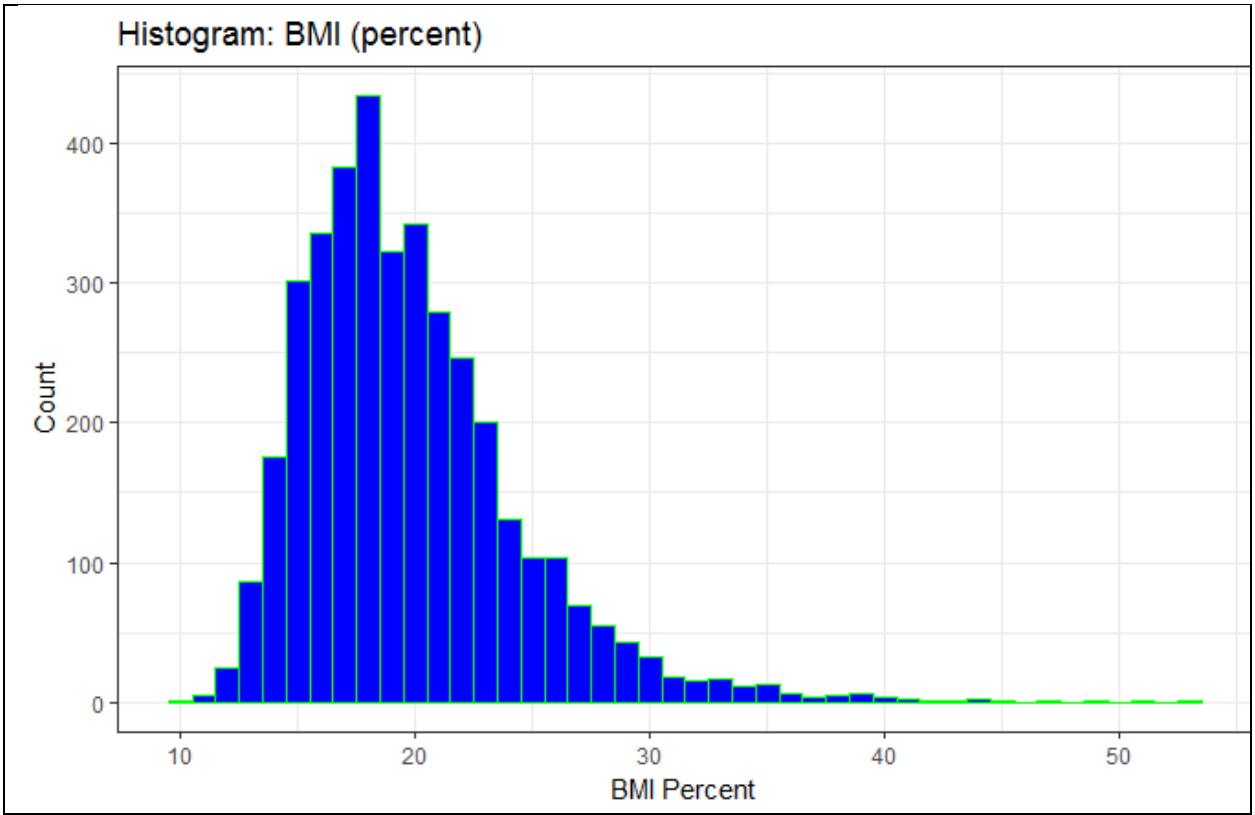

```
# Height approximately normally distributed as expected
ggplot(nhanesData, aes(x=Height_m)) +
  geom_histogram(binwidth=.333, colour="green", fill="blue") +
  labs(x="Height", y="Count", title="Histogram: Height (meters)")
```



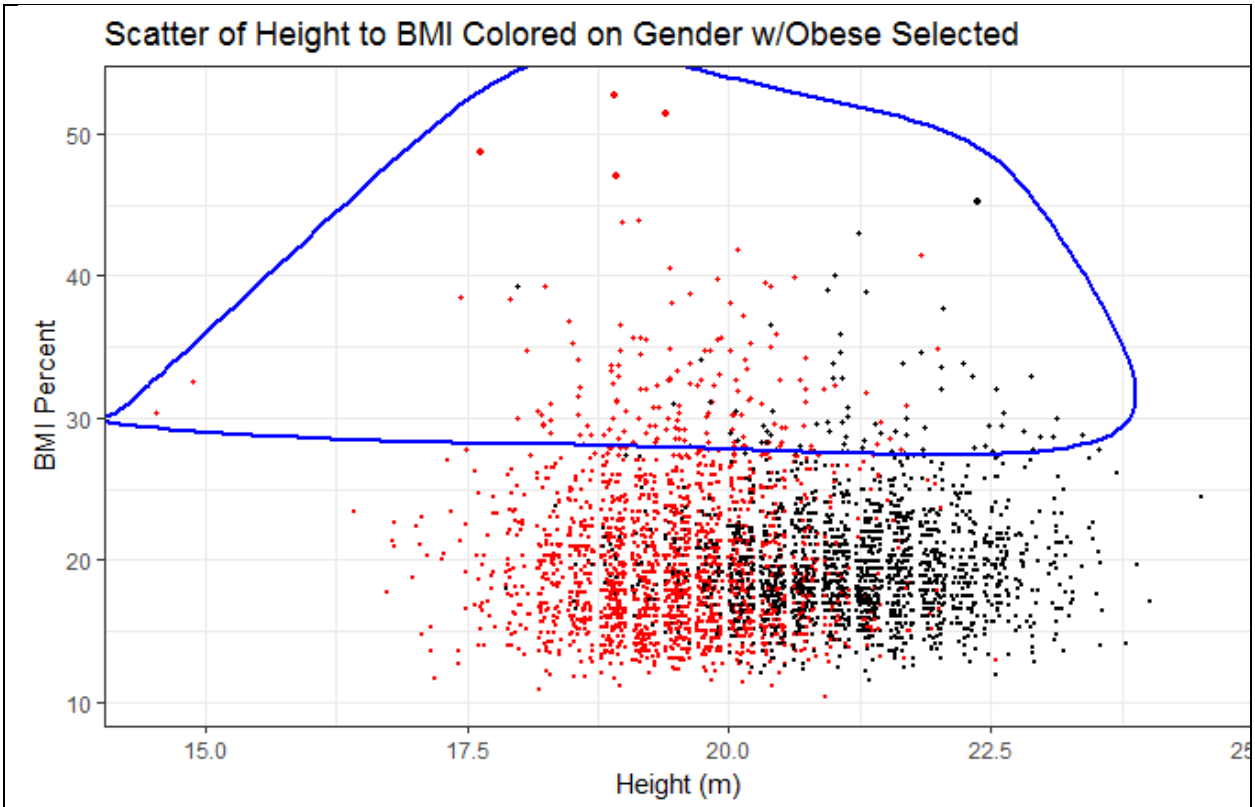

```
# Weight skewed to the right
ggplot(nhanesData, aes(x=Weight_kg)) +
  geom_histogram(binwidth=10, colour="green", fill="blue") +
  labs(x="Weight", y="Count", title="Histogram: Weight (kg)")
```



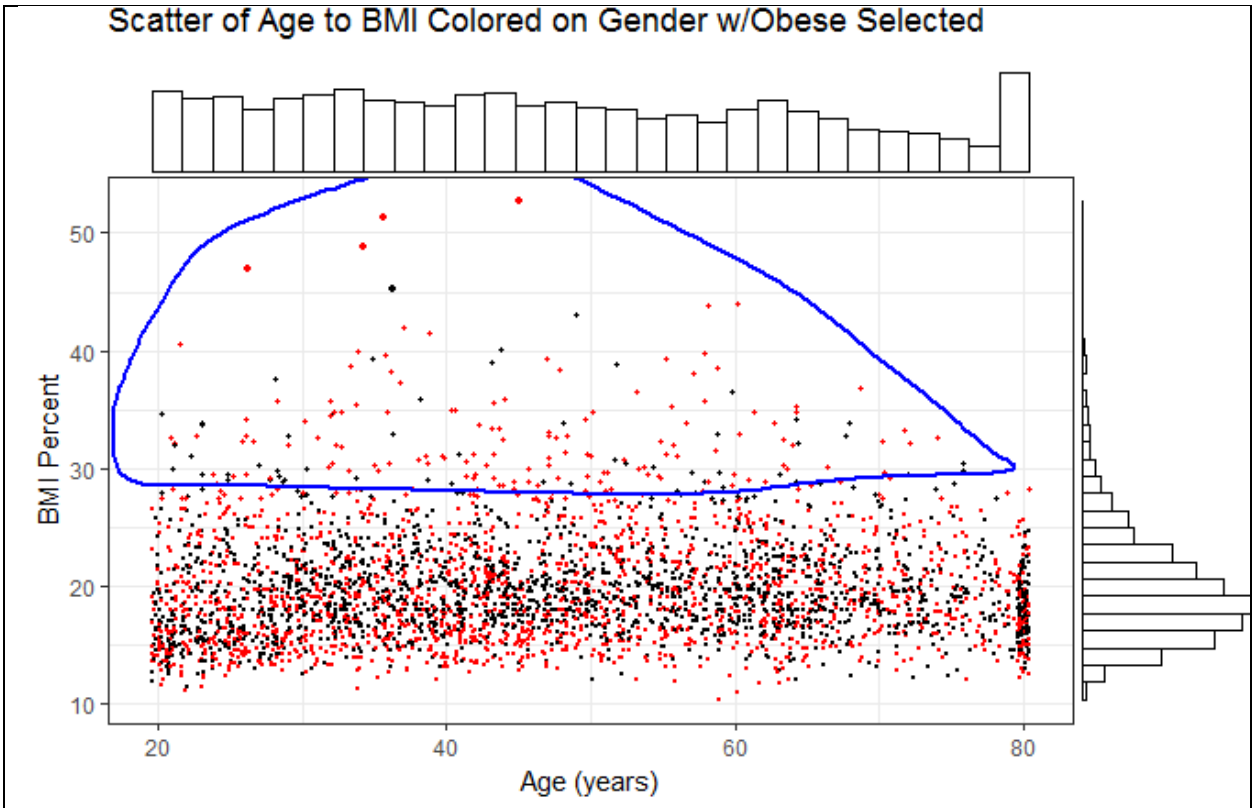

```
# 1939 females and 1847 males

# Create gender Male and Female attribute (instead of 1 and 2)
nhanesData$GenderStr = rep(1, nrow(nhanesData))
maleNdx = nhanesData$RIAGENDR == 1
femaleNdx = nhanesData$RIAGENDR == 2
nhanesData$GenderStr[maleNdx] = 'Male'
nhanesData$GenderStr[femaleNdx] = 'Female'

ggplot(nhanesData, aes(x=GenderStr)) +
  geom_bar(color="black", fill="blue") +
  labs(x="Gender", y="Count", title="Bar Plot: Gender")
```



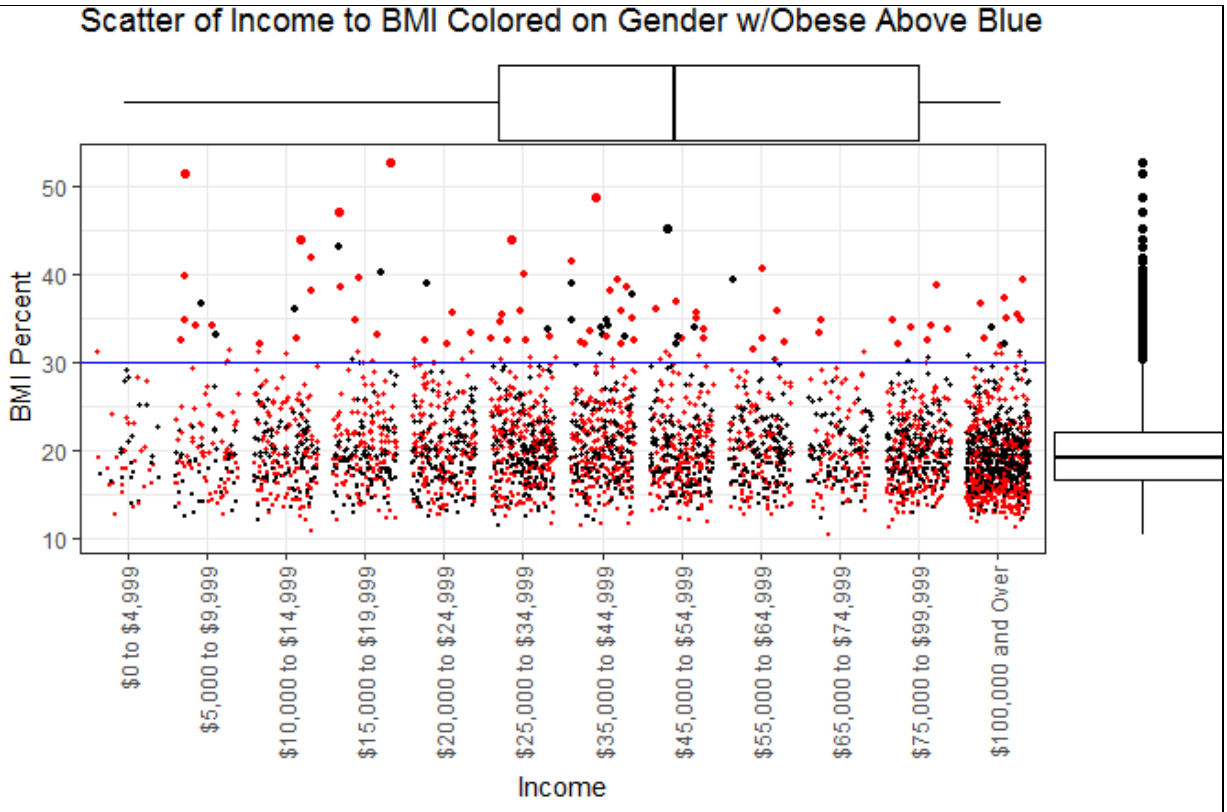

```
# BMI Percent is skewed to the right.  Percent > 30 implies Obese
ggplot(nhanesData, aes(x=BMI_Perc)) +
  geom_histogram(binwidth=1, colour="green", fill="blue") +
  labs(x="BMI Percent", y="Count", title="Histogram: BMI (percent)")
```




```
# Start looking at scatter of BMI to continuous variables.
# Create gender Male and Female attribute (instead of 1 and 2)
library(ggalt)
nhanesData$GenderStr = rep(1, nrow(nhanesData))
maleNdx = nhanesData$RIAGENDR == 1
femaleNdx = nhanesData$RIAGENDR == 2
nhanesData$GenderStr[maleNdx] = 'Male'
nhanesData$GenderStr[femaleNdx] = 'Female'

BMI_select <- nhanesData[nhanesData$BMI_Perc > 30, ]

# Is BMI associated with particularly short people?  Doesn't look like it.
ggplot(nhanesData, aes(x=Height_m, y=BMI_Perc), color=GenderStr) +
  geom_point(col=nhanesData$RIAGENDR, size=nhanesData$BMI_Perc/50,
position="jitter") +
  labs(x="Height (m)", y="BMI Percent", title="Scatter of Height to BMI Colored on
Gender w/Obese Selected") +
  geom_encircle(aes(x=Height_m, y=BMI_Perc), data=BMI_select, color='blue', size=2,
expand=0.08)
```

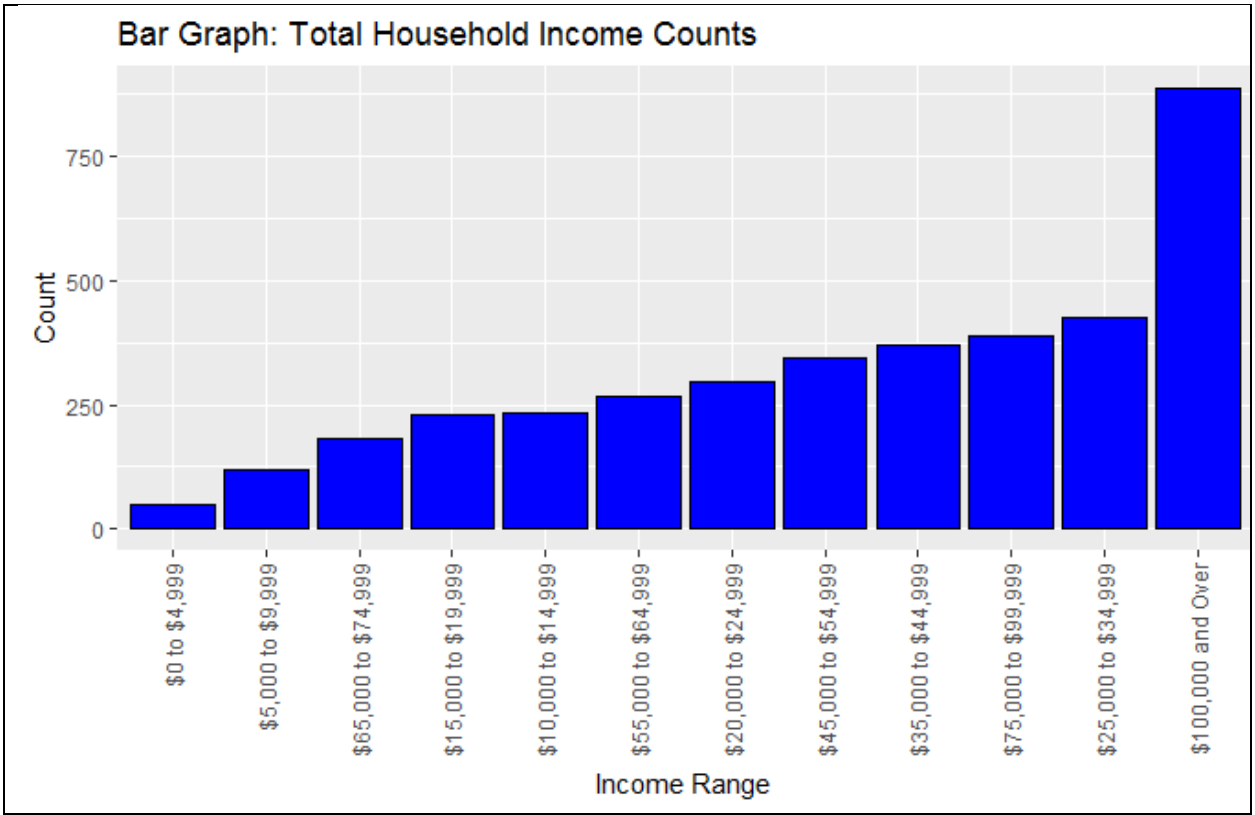



```
#histogram for BMI skewed to obese.  Note histogram for age 80 – because anyone
#greater than or over 80 defaulted to age 80
#use marginal histogram
ggMarginal(g, type="histogram", fill="transparent")

# Doesn't appear age influences BMI
g <- ggplot(nhanesData, aes(x=RIDAGEYR, y=BMI_Perc), color=GenderStr) +
  geom_point(col=nhanesData$RIAGENDR, size=nhanesData$BMI_Perc/50,
position="jitter") +
  labs(x="Age (years)", y="BMI Percent", title="Scatter of Age to BMI Colored on Gender
w/Obese Selected") +
  geom_encircle(aes(x=RIDAGEYR, y=BMI_Perc), data=BMI_select, color='blue', size=2,
expand=0.08)

ggMarginal(g, type="histogram", fill="transparent")
ggMarginal(g, type="boxplot", fill="transparent")
```



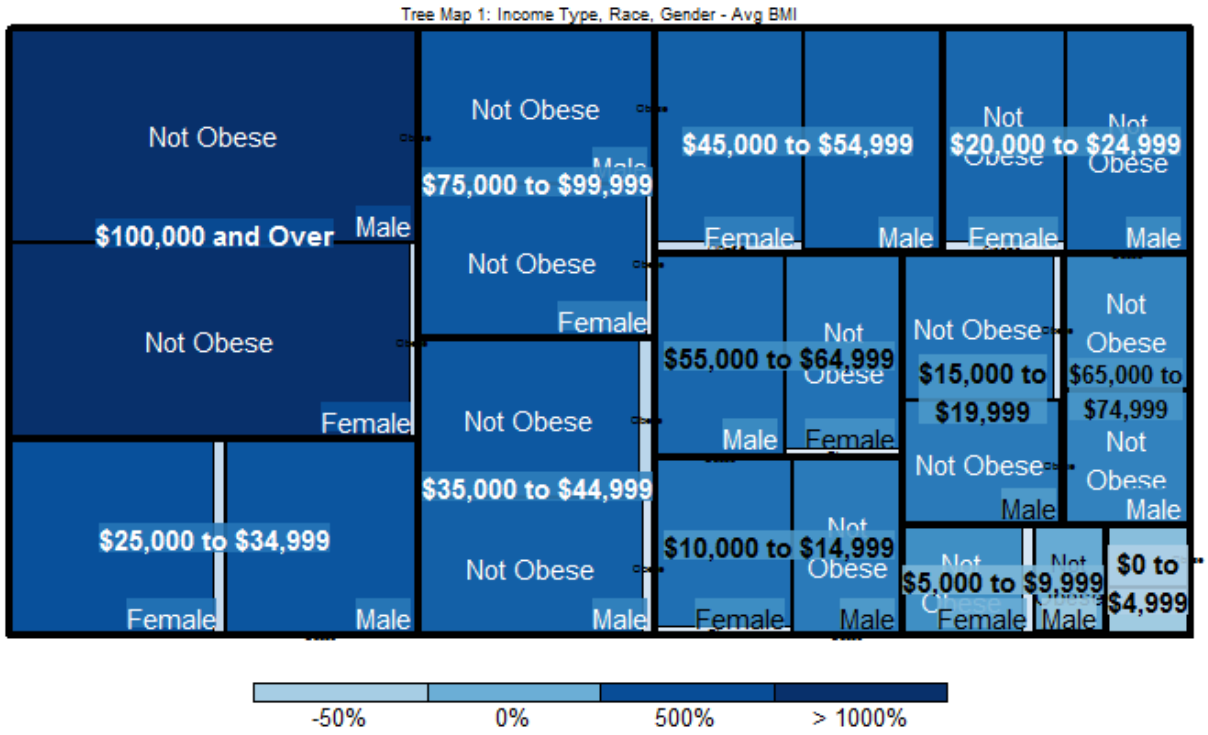

```
# Presentation
# Doesn't appear age influences BMI – Good graphic for identifying obese as outliers
positions <- c("$0 to $4,999",
               "$5,000 to $9,999",
               "$10,000 to $14,999",
               "$15,000 to $19,999",
               "$20,000 to $24,999",
               "$25,000 to $34,999",
               "$35,000 to $44,999",
               "$45,000 to $54,999",
               "$55,000 to $64,999",
               "$65,000 to $74,999",
               "$75,000 to $99,999",
               "$100,000 and Over")

g <- ggplot(nhanesData, aes(x=IncomeStr, y=BMI_Perc), color=GenderStr) +
  geom_point(col=nhanesData$RIAGENDR, size=nhanesData$BMI_Perc/35,
position="jitter") +
  scale_x_discrete(limits=positions) +
  geom_hline(yintercept = 30, color = 'blue') +
  labs(x="Income", y="BMI Percent", title="Scatter of Income to BMI Colored on Gender
w/Obese Above Blue Line") +
  theme(axis.text.x = element_text(angle=90, hjust=1, vjust=.5))

ggMarginal(g, type="boxplot", fill="transparent")
```

Interesting notes are that a substantial number of respondents had income of $100,00 and over.



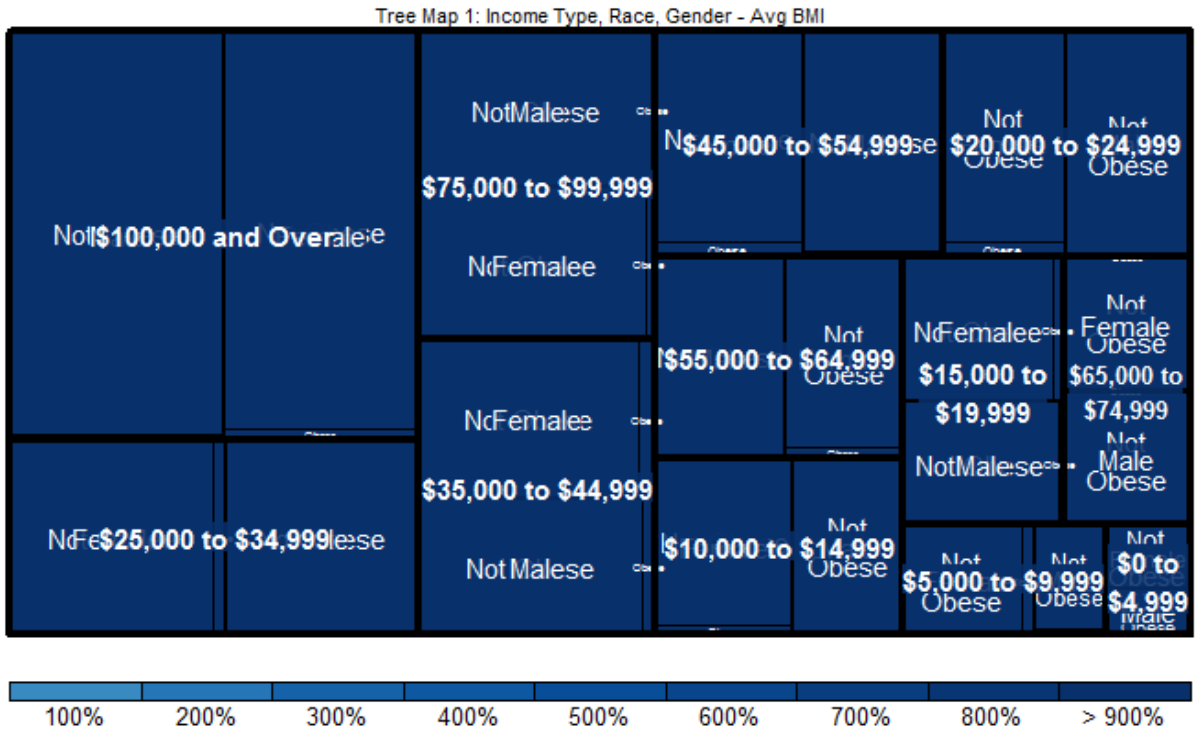

```
# Create Income attibutes (instead of 1-5, 77, 99)
nhanesData$IncomeStr = rep(1, nrow(nhanesData))
aNdx        = nhanesData$INDHHIN2 == 1
bNdx        = nhanesData$INDHHIN2 == 2
cNdx        = nhanesData$INDHHIN2 == 3
dNdx        = nhanesData$INDHHIN2 == 4
eNdx        = nhanesData$INDHHIN2 == 5
fNdx        = nhanesData$INDHHIN2 == 6
gNdx        = nhanesData$INDHHIN2 == 7
hNdx        = nhanesData$INDHHIN2 == 8
iNdx        = nhanesData$INDHHIN2 == 9
jNdx        = nhanesData$INDHHIN2 == 10
kNdx        = nhanesData$INDHHIN2 == 11
lNdx        = nhanesData$INDHHIN2 == 12
mNdx        = nhanesData$INDHHIN2 == 13
nNdx        = nhanesData$INDHHIN2 == 14
oNdx        = nhanesData$INDHHIN2 == 15
refusedNdx   = nhanesData$INDHHIN2 == 77
dontKnowNdx  = nhanesData$INDHHIN2 == 99

nhanesData$IncomeStr[aNdx]        = '$0 to $4,999'
nhanesData$IncomeStr[bNdx]        = '$5,000 to $9,999'
nhanesData$IncomeStr[cNdx]        = '$10,000 to $14,999'
nhanesData$IncomeStr[dNdx]        = '$15,000 to $19,999'
nhanesData$IncomeStr[eNdx]        = '$20,000 to $24,999'
nhanesData$IncomeStr[fNdx]        = '$25,000 to $34,999'
```

```
nhanesData$IncomeStr[gNdx]       = '$35,000 to $44,999'
nhanesData$IncomeStr[hNdx]       = '$45,000 to $54,999'
nhanesData$IncomeStr[iNdx]       = '$55,000 to $64,999'
nhanesData$IncomeStr[jNdx]       = '$65,000 to $74,999'
nhanesData$IncomeStr[kNdx]       = ' '
nhanesData$IncomeStr[lNdx]       = '$20,000 and Over'
nhanesData$IncomeStr[mNdx]       = 'Under $20,000'
nhanesData$IncomeStr[nNdx]       = '$75,000 to $99,999'
nhanesData$IncomeStr[oNdx]       = '$100,000 and Over'
nhanesData$IncomeStr[refusedNdx]  = 'Refused'
nhanesData$IncomeStr[dontKnowNdx] = 'Do Not Know'

# First level counts: Income - Most participants have income over $100K
positions <- c("$0 to $4,999",
               "$5,000 to $9,999",
               "$65,000 to $74,999",
               "$15,000 to $19,999",
               "$10,000 to $14,999",
               "$55,000 to $64,999",
               "$20,000 to $24,999",
               "$45,000 to $54,999",
               "$35,000 to $44,999",
               "$75,000 to $99,999",
               "$25,000 to $34,999", "$100,000 and Over")

ggplot(nhanesData, aes(x=IncomeStr)) +
  geom_bar(color="black", fill="blue") +
  scale_x_discrete(limits=positions) +
  labs(x="Marital Status", y="Count", title="Bar Graph: Total Household Income Counts")
+
  theme(axis.text.x = element_text(angle=90, hjust=1, vjust=.5))
```



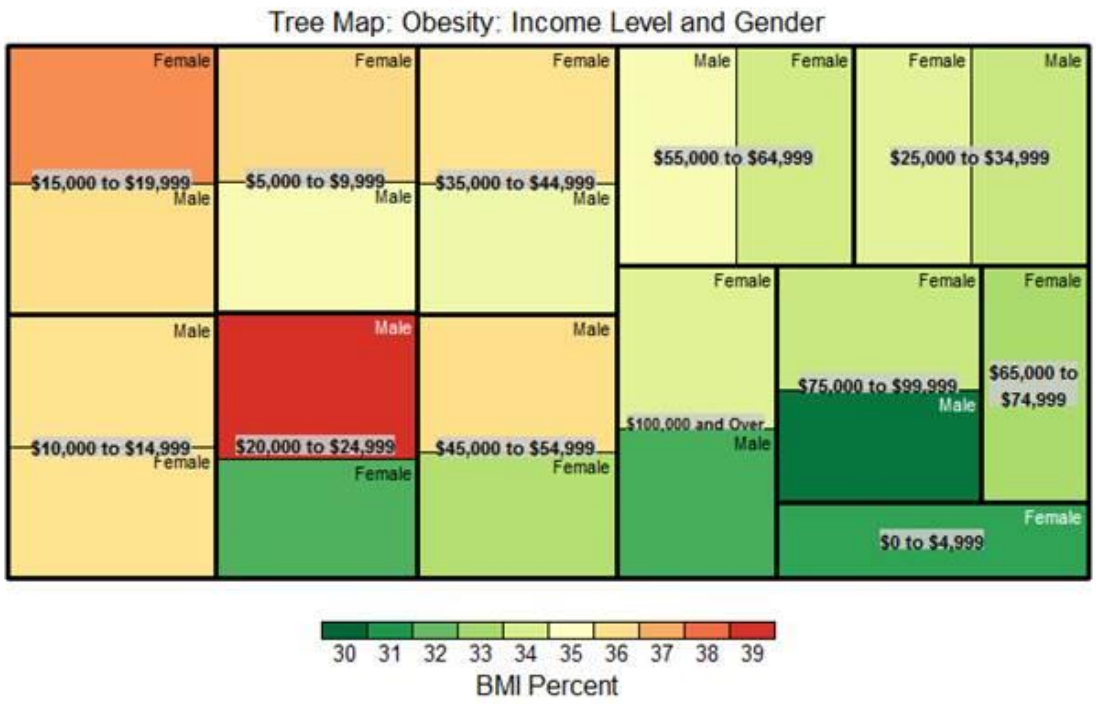

```
#Tree map on Income Level, Race, Gender

library(gcookbook)
library(ggplot2)
library(lubridate)
library(dplyr)
library(mosaic)
library(scales)
library(psych)
library(chron)
library(treemap)
library(ggcorrplot)

nhanesData$obeseType <- ifelse(nhanesData$BMI_Perc < 30, "Not Obese", "Obese")

#Aggregate on field #28 which is BMI_Perc
d <- as.data.frame(aggregate(nhanesData[, 28], list(nhanesData$IncomeStr,
nhanesData$obeseType,nhanesData$GenderStr), mean))

d <- as.data.frame(aggregate(nhanesData[, 29], list(nhanesData$IncomeStr,
nhanesData$obeseType,nhanesData$GenderStr), count))

totalCount <- sum((nhanesData$GenderStr == "Male") | (nhanesData$GenderStr ==
"Female"))

d$pct <- d$x/totalCount
d$pct <- d$pct*100
```

```
head(d)

d  #Output below

d <- d[c(1,3,2)]

treemap(d, #Your data frame object
        index=c("Group.1","Group.3", "Group.2"),  #A list of your categorical variables
        vSize = "pct",  #This is your quantitative variable
        type="comp", #Type sets the organization and color scheme of your treemap
        palette = "Blues",  #Select your color palette from the RColorBrewer )
        title="Tree Map 1: Income Type, Race, Gender - Avg BMI", #Customize your title
        fontsize.title = 7,
        force.print.labels = TRUE,
        align.labels=list(c("center","center"), c("right", "bottom"), c("center", "center")),
)
```




```
# Work on color and scaling above not good.

treemap(d, #Your data frame object
        index=c("Group.1","Group.3", "Group.2"),  #A list of your categorical variables
        vSize = "pct",  #This is your quantitative variable
        #type="index", #Type sets the organization and color scheme of your treemap
        type="comp",
        palette = "Blues",  #Select your color palette from the RColorBrewer )
        title="Tree Map 1: Income Type, Race, Gender - Avg BMI", #Customize your title
        fontsize.title = 8 ,
        #fontsize.labels = 5,
        force.print.labels = TRUE,
```

```
        overlap.labels=1
)
```




```
#Final Presentation

nhanesData$obeseType <- ifelse(nhanesData$BMI_Perc < 30, "Not Obese", "Obese")

head(nhanesData)

###d <- as.data.frame(aggregate(nhanesData[, 29], list(nhanesData$IncomeStr,
nhanesData$obeseType,nhanesData$GenderStr), mean))

d_new <- as.data.frame(aggregate(nhanesData$BMI_Perc, list(nhanesData$IncomeStr,
nhanesData$obeseType,nhanesData$GenderStr), mean))
dSubset_new <- subset(d_new, Group.2 == "Obese", c("Group.1", "Group.2", "Group.3",
"x"))

treemap(dSubset_new, #Your data frame object
        index=c("Group.1","Group.3"),  #A list of your categorical variables
        vSize = "x",  #This is your quantitative variable
        vColor="x",
        type="value", #Type sets the organization and color scheme of your treemap
        palette = "RdYlGn",  #Select your color palette from the RColorBrewer
        title="Tree Map: Obesity: Income Level and Gender", #Customize your title,
        fontsize.title = 13,
        fontsize.labels = 8,
        force.print.labels = TRUE,
        align.labels=list(c("center","center"), c("right", "top")),
        title.legend = 'BMI Percent',
```

```
        mapping=c(40, 35, 30)
)
```

5. Kari Palmier

The code below is for all of the distribution analysis, data cleaning, filtered final csv creation, polished exploratory, the exploratory stacked bar charts, the heat maps, and the percentage obese hierarchical bar charts. Note that I did create other exploratory plots and different versions of heatmaps (against education, marital status, and income), and differnet bar charts that I decided were not interesting or helpful. The code below does not contain all of these versions (I deleted a lot of code that was not interesting to me during the process to declutter).

```
#############################################################################################################
#########################
#
#  Kari Palmier CSC 465 Final Project
#
#############################################################################################################
#########################

library(ggplot2)
library(lubridate)
library(dplyr)
library(mosaic)
library(scales)
library(psych)
library(gridExtra)

#############################################################################################################
#########################
#
#  Initial Exploratory Analysis
#
#############################################################################################################
#########################

nhanesData = read.table("C:\\DePaul Coursework\\Fall 2017 CSC 465\\Final Project\\NHANES_ConbinedProjectDataset.csv",
sep=",", header=T)
nhanesData$SEQN = nhanesData$ï..SEQN
nhanesData$ï..SEQN = NULL

# Print summary of original data
nrow(nhanesData)
summary(nhanesData)
describe(nhanesData)
colSums(is.na(nhanesData))
head(nhanesData)
str(nhanesData)


#############################################################################################################
#########################
#
```

```
#  Initial Exploratory Histograms
#
#############################################################################################################
#########################
```

```
ggplot(data = nhanesData, aes(x = RIAGENDR)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(0, 3)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
       panel.background = element_rect(fill = "white", color = "white",size = 0.5),
       panel.border = element_rect(colour = "black", fill=NA, size=1),
       plot.title = element_text(hjust = 0.5),
       legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Initial Gender Counts") +
 labs(x = "Gender", y = "Count")

ggplot(data = nhanesData, aes(x = RIDAGEYR)) + geom_histogram(bins = 15, fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(0, 81)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
       panel.background = element_rect(fill = "white", color = "white",size = 0.5),
       panel.border = element_rect(colour = "black", fill=NA, size=1),
       plot.title = element_text(hjust = 0.5),
       legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Initial Age Histogram") +
 labs(x = "Age (Years)", y = "Frequency")

ggplot(data = nhanesData, aes(x = DMDEDUC2)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(0, 6)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
       panel.background = element_rect(fill = "white", color = "white",size = 0.5),
       panel.border = element_rect(colour = "black", fill=NA, size=1),
       plot.title = element_text(hjust = 0.5),
       legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Initial Education Counts") +
 labs(x = "Education", y = "Count")

ggplot(data = nhanesData, aes(x = DMDMARTL)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(0, 7)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
       panel.background = element_rect(fill = "white", color = "white",size = 0.5),
       panel.border = element_rect(colour = "black", fill=NA, size=1),
       plot.title = element_text(hjust = 0.5),
       legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Initial Marital Status Counts") +
 labs(x = "Marital Status Levels", y = "Count")

ggplot(data = nhanesData, aes(x = RIDRETH1)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(0, 6)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
       panel.background = element_rect(fill = "white", color = "white",size = 0.5),
       panel.border = element_rect(colour = "black", fill=NA, size=1),
       plot.title = element_text(hjust = 0.5),
       legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Initial Race Counts") +
 labs(x = "Races", y = "Count")

ggplot(data = nhanesData, aes(x = INDHHIN2)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(0, 16)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
```

```
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Initial Income Counts") +
 labs(x = "Income", y = "Count")

ggplot(data = nhanesData, aes(x = BPQ020)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(0, 3)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Initial Hypertension Counts") +
 labs(x = "Hypertension Levels", y = "Count")

ggplot(data = nhanesData, aes(x = WHD010)) + geom_histogram(bins = 15, fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(47, 82)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Initial Height Histogram") +
 labs(x = "Height (in)", y = "Frequency")

ggplot(data = nhanesData, aes(x = WHD020)) + geom_histogram(bins = 15, fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(74, 494)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Initial Weight Histogram") +
 labs(x = "Weight (lbs)", y = "Frequency")

ggplot(data = nhanesData, aes(x = BPQ020)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(0, 4)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Initial Diabetes Counts") +
 labs(x = "Diabetes Levels", y = "Count")


#############################################################################################################
#########################
#
#  Data cleaning
#
#############################################################################################################
#########################

attrs = colnames(nhanesData)
numAttrs = length(attrs)
```

```
refusedEntries = c(0, 0, 0, 7, 77, 77, 0, 7, 777777, 777777, 777777, 777777, 7, 7777, 7777, 7777, 7777, 7, 7, 7, 7777, 7, 7777, 7777,
0)
dkEntries = c(0, 0, 0, 9, 99, 99, 0, 9, 999999, 999999, 999999, 999999, 9, 9999, 9999, 9999, 9999, 9, 9, 9, 9999, 9, 9999, 9999, 0)


# Remove NaN values
for (i in 1:numAttrs){
  current_attr = attrs[i]
  notNaNdx = !is.na(nhanesData[current_attr])
  nhanesData = nhanesData[notNaNdx,]

  print(i)
  print(current_attr)
  print('After NaN Removal')
  print(nrow(nhanesData))

}

# Remove refused values
for (i in 1:numAttrs){
  current_attr = attrs[i]

  refNdx = nhanesData[current_attr] == refusedEntries[i]
  if (any(refNdx)){
    nhanesData = nhanesData[!refNdx,]
  }

}

# Remove don't know values
for (i in 1:numAttrs){
  current_attr = attrs[i]

  dkNdx = nhanesData[current_attr] == dkEntries[i]
  if (any(dkNdx)){
    nhanesData = nhanesData[!dkNdx,]
  }

}

# Remove income values for $20,000 and over (12) and under $20,000 because the majority of data was not in these bins
# Also because having a range like this and bins that overlap the range will lead to issues using the variable
twelveNdx = nhanesData$INDHHIN2 == 12
if (any(twelveNdx)){
  nhanesData = nhanesData[!twelveNdx,]
}


thirteenNdx = nhanesData$INDHHIN2 == 13
if (any(thirteenNdx)){
  nhanesData = nhanesData[!thirteenNdx,]
}


#############################################################################################################
#########################
#
```

```
# Exploratory Histograms After Data Cleaning (To Ensure Distributions are still maintained)
#
#########################################################################################################################
#########################

ggplot(data = nhanesData, aes(x = RIAGENDR)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
  scale_x_continuous(limits = c(0, 3)) +
  theme( axis.text.x = element_text(angle = 20, hjust = 1),
         panel.background = element_rect(fill = "white", color = "white",size = 0.5),
         panel.border = element_rect(colour = "black", fill=NA, size=1),
         plot.title = element_text(hjust = 0.5),
         legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
  ggtitle("After Cleaning Gender Counts") +
  labs(x = "Gender", y = "Count")

ggplot(data = nhanesData, aes(x = RIDAGEYR)) + geom_histogram(bins = 15, fill = "steelblue", color = "black") +
  scale_x_continuous(limits = c(0, 81)) +
  theme( axis.text.x = element_text(angle = 20, hjust = 1),
         panel.background = element_rect(fill = "white", color = "white",size = 0.5),
         panel.border = element_rect(colour = "black", fill=NA, size=1),
         plot.title = element_text(hjust = 0.5),
         legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
  ggtitle("After Cleaning Age Histogram") +
  labs(x = "Age (Years)", y = "Frequency")

ggplot(data = nhanesData, aes(x = DMDEDUC2)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
  scale_x_continuous(limits = c(0, 6)) +
  theme( axis.text.x = element_text(angle = 20, hjust = 1),
         panel.background = element_rect(fill = "white", color = "white",size = 0.5),
         panel.border = element_rect(colour = "black", fill=NA, size=1),
         plot.title = element_text(hjust = 0.5),
         legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
  ggtitle("After Cleaning Education Counts") +
  labs(x = "Education", y = "Count")

ggplot(data = nhanesData, aes(x = DMDMARTL)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
  scale_x_continuous(limits = c(0, 7)) +
  theme( axis.text.x = element_text(angle = 20, hjust = 1),
         panel.background = element_rect(fill = "white", color = "white",size = 0.5),
         panel.border = element_rect(colour = "black", fill=NA, size=1),
         plot.title = element_text(hjust = 0.5),
         legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
  ggtitle("After Cleaning Marital Status Counts") +
  labs(x = "Marital Status Levels", y = "Count")

ggplot(data = nhanesData, aes(x = RIDRETH1)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
  scale_x_continuous(limits = c(0, 6)) +
  theme( axis.text.x = element_text(angle = 20, hjust = 1),
         panel.background = element_rect(fill = "white", color = "white",size = 0.5),
         panel.border = element_rect(colour = "black", fill=NA, size=1),
         plot.title = element_text(hjust = 0.5),
         legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
  ggtitle("After Cleaning Race Counts") +
  labs(x = "Races", y = "Count")

ggplot(data = nhanesData, aes(x = INDHHIN2)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
  scale_x_continuous(limits = c(0, 16)) +
  theme( axis.text.x = element_text(angle = 20, hjust = 1),
```

```
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("After Cleaning Income Counts") +
 labs(x = "Income", y = "Count")

ggplot(data = nhanesData, aes(x = BPQ020)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(0, 3)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("After Cleaning Hypertension Counts") +
 labs(x = "Hypertension Levels", y = "Count")

ggplot(data = nhanesData, aes(x = WHD010)) + geom_histogram(bins = 15, fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(47, 82)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("After Cleaning Height Histogram") +
 labs(x = "Height (in)", y = "Frequency")

ggplot(data = nhanesData, aes(x = WHD020)) + geom_histogram(bins = 15, fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(74, 494)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("After Cleaning Weight Histogram") +
 labs(x = "Weight (lbs)", y = "Frequency")

ggplot(data = nhanesData, aes(x = BPQ020)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
 scale_x_continuous(limits = c(0, 4)) +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("After Cleaning Diabetes Counts") +
 labs(x = "Diabetes Levels", y = "Count")


#############################################################################################################
#########################
#
# Create New Variables
#
#############################################################################################################
#########################

lb_to_kg_const =  0.45359237
ft_to_m_const = 0.3048
```

```
nhanesData$Weight_kg = nhanesData$WHD020 * lb_to_kg_const
nhanesData$Height_m = nhanesData$WHD010 * ft_to_m_const

nhanesData$BMI_Perc = (nhanesData$Weight_kg / (nhanesData$Height_m^2)) * 100

# Create ObeseFlag attribute for if person is obese (obesity is BMI >= 30)
nhanesData$ObeseFlag = rep(1, nrow(nhanesData))
notObeseNdx = nhanesData$BMI_Perc < 30
nhanesData$ObeseFlag[notObeseNdx] = 0

#############################################################################################################
#########################
#
# Create Basic Exploratory Plots And Text File For All Vars
#
#############################################################################################################
#########################

# Create path to save exploratory plots
explorePath = "C:\\DePaul Coursework\\Fall 2017 CSC 465\\Final Project\\Exploratory\\"

# Create output file
dir.create(explorePath, showWarnings = FALSE)
outFileName = paste(explorePath, "Explore_R_Output.txt", sep="")
outFile = file(outFileName, open="wt")

sumData = summary(nhanesData)
descData = describe(nhanesData)

# Print descriptive statistics
sink(file=outFile, append=TRUE)
print("Data Set Info:")
str(nhanesData)
print("", quote=FALSE)
print("Data Set Descriptive Statistics Summary:", quote=FALSE)
print("", quote=FALSE)
print(sumData, quote=FALSE)
print("", quote=FALSE)
print("Data Set Descriptive Statistics Describe Values:", quote=FALSE)
print("", quote=FALSE)
print(descData, quote=FALSE)
print("", quote=FALSE)
sink()

attrNames = colnames(nhanesData)
dir.create(file.path(explorePath, "Histograms"), showWarnings = FALSE)
newPath = paste(explorePath, "Histograms\\", sep="")
for (j in 1:length(attrNames)){
  # Plot Y Histogram
  fileName = paste(newPath, attrNames[j], "Hist.jpeg", sep="")
  plotTitle = paste(attrNames[j], "Histogram")
  jpeg(file=fileName)
  hData = c(t(nhanesData[attrNames[j]]))
  hist(hData, freq=TRUE, main=plotTitle, xlab=attrNames[j])
  dev.off()
}
```

```
dir.create(file.path(explorePath, "Box Plots"), showWarnings = FALSE)
newPath = paste(explorePath, "Box Plots\\", sep="")
for (i in 1:length(attrNames)){
 var = attrNames[i]
 fileName = paste(newPath, attrNames[i], "_BoxPlot.jpeg", sep="")
 plotTitle = paste(attrNames[i], " Box Plot")
 jpeg(file=fileName)
 plotData = c(t(nhanesData[var]))
 boxplot(plotData, main=plotTitle)
 dev.off()
}

# Create Correlation Values for all data
numAllVars = length(attrNames)
sink(file=outFile, append=TRUE)
print("", quote=FALSE)

for(i in 1:numAllVars){
 currentCor = cor(nhanesData[attrNames[i]], nhanesData[attrNames])

 assign("last.warning", NULL, envir = baseenv())

 print("", quote=FALSE)
 print(currentCor, quote=FALSE)
 print("", quote=FALSE)
 print("", quote=FALSE)

 assign("last.warning", NULL, envir = baseenv())

}

sink()

close(outFile)
closeAllConnections()


#############################################################################################################
#########################
#
# Convert nominal values to appropriate data types based on variable values from codebook
#
#############################################################################################################
#########################
nhanesData$RIAGENDR = as.factor(nhanesData$RIAGENDR)
nhanesData$RIDRETH1 = as.factor(nhanesData$RIDRETH1)
nhanesData$DMDEDUC2 = as.factor(nhanesData$DMDEDUC2)
nhanesData$DMDMARTL = as.factor(nhanesData$DMDMARTL)
nhanesData$INDHHIN2 = as.factor(nhanesData$INDHHIN2)
nhanesData$BPQ020 = as.factor(nhanesData$BPQ020)
nhanesData$DIQ010 = as.factor(nhanesData$DIQ010)
nhanesData$MCQ080 = as.factor(nhanesData$MCQ080)
nhanesData$MCQ365A = as.factor(nhanesData$MCQ365A)
nhanesData$MCQ365B = as.factor(nhanesData$MCQ365B)
nhanesData$SMQ020 = as.factor(nhanesData$SMQ020)
nhanesData$ObeseFlag = as.factor(nhanesData$ObeseFlag)
```

```
#############################################################################################################
#########################
#
#  Create factor attributes with descriptive strings for their levels for plotting
#
#############################################################################################################
#########################

# Create Gender attribute with male and female for plotting
nhanesData$GenderStr = rep(1, nrow(nhanesData))
maleNdx = nhanesData$RIAGENDR == 1
femaleNdx = nhanesData$RIAGENDR == 2
nhanesData$GenderStr[maleNdx] = 'Male'
nhanesData$GenderStr[femaleNdx] = 'Female'

# Create Race attribute with race strings for plotting
nhanesData$RaceStr = rep(1, nrow(nhanesData))
mexAmNdx = nhanesData$RIDRETH1 == 1
othHispNdx = nhanesData$RIDRETH1 == 2
nonHispWNdx = nhanesData$RIDRETH1 == 3
nonHispBNdx = nhanesData$RIDRETH1 == 4
otherNdx = nhanesData$RIDRETH1 == 5
nhanesData$RaceStr[mexAmNdx] = 'Mexican American'
nhanesData$RaceStr[othHispNdx] = 'Other Hispanic'
nhanesData$RaceStr[nonHispWNdx] = 'Non-Hispanic White'
nhanesData$RaceStr[nonHispBNdx] = 'Non-Hispanic Black'
nhanesData$RaceStr[otherNdx] = 'Other'

# Create Income attibutes description strings for plotting
nhanesData$IncomeStr = rep(1, nrow(nhanesData))
ndx1 = nhanesData$INDHHIN2 == 1
ndx2 = nhanesData$INDHHIN2 == 2
ndx3 = nhanesData$INDHHIN2 == 3
ndx4 = nhanesData$INDHHIN2 == 4
ndx5 = nhanesData$INDHHIN2 == 5
ndx6 = nhanesData$INDHHIN2 == 6
ndx7 = nhanesData$INDHHIN2 == 7
ndx8 = nhanesData$INDHHIN2 == 8
ndx9 = nhanesData$INDHHIN2 == 9
ndx10 = nhanesData$INDHHIN2 == 10
ndx14 = nhanesData$INDHHIN2 == 14
ndx15 = nhanesData$INDHHIN2 == 15

nhanesData$IncomeStr[ndx1] = '$0 to $4,999'
nhanesData$IncomeStr[ndx2] = '$5,000 to $9,999'
nhanesData$IncomeStr[ndx3] = '$10,000 to $14,999'
nhanesData$IncomeStr[ndx4] = '$15,000 to $19,999'
nhanesData$IncomeStr[ndx5] = '$20,000 to $24,999'
nhanesData$IncomeStr[ndx6] = '$25,000 to $34,999'
nhanesData$IncomeStr[ndx7] = '$35,000 to $44,999'
nhanesData$IncomeStr[ndx8] = '$45,000 to $54,999'
nhanesData$IncomeStr[ndx9] = '$55,000 to $64,999'
nhanesData$IncomeStr[ndx10] = '$65,000 to $74,999'
nhanesData$IncomeStr[ndx14] = '$75,000 to $99,999'
nhanesData$IncomeStr[ndx15] = '$100,000 and Over'

# Create Marital Status strings for plotting
nhanesData$MaritalStr = rep(1, nrow(nhanesData))
```

```
marriedNdx = nhanesData$DMDMARTL == 1
widowedNdx  = nhanesData$DMDMARTL == 2
divorcedNdx = nhanesData$DMDMARTL == 3
separatedNdx = nhanesData$DMDMARTL == 4
neverMarriedNdx = nhanesData$DMDMARTL == 5
partnerNdx = nhanesData$DMDMARTL == 6

nhanesData$MaritalStr[marriedNdx]       = 'Married'
nhanesData$MaritalStr[widowedNdx]      = 'Widowed'
nhanesData$MaritalStr[divorcedNdx]      = 'Divorced'
nhanesData$MaritalStr[separatedNdx]    = 'Separated'
nhanesData$MaritalStr[neverMarriedNdx] = 'Never Married'
nhanesData$MaritalStr[partnerNdx]       = 'Living with Partner'

# Create Race attribute with race strings for plotting
nhanesData$EdStr = rep(1, nrow(nhanesData))
ndx1 = nhanesData$DMDEDUC2 == 1
ndx2 = nhanesData$DMDEDUC2 == 2
ndx3 = nhanesData$DMDEDUC2 == 3
ndx4 = nhanesData$DMDEDUC2 == 4
ndx5 = nhanesData$DMDEDUC2 == 5
nhanesData$EdStr[ndx1] = 'Less Than 9th'
nhanesData$EdStr[ndx2] = '9th to 12th No Grad'
nhanesData$EdStr[ndx3] = 'High School Grad/GED'
nhanesData$EdStr[ndx4] = 'Some College/AA'
nhanesData$EdStr[ndx5] = 'College Grad/Above'

# Create hypertension attribute with strings for plotting
nhanesData$BPStr = rep(1, nrow(nhanesData))
ndx1 = nhanesData$BPQ020 == 1
ndx2 = nhanesData$BPQ020 == 2
nhanesData$BPStr[ndx1] = 'Hypertension'
nhanesData$BPStr[ndx2] = 'No Hypertension'

# Create diabetes attribute with strings for plotting
nhanesData$DiabetesStr = rep(1, nrow(nhanesData))
ndx1 = nhanesData$DIQ010 == 1
ndx2 = nhanesData$DIQ010 == 2
ndx3 = nhanesData$DIQ010 == 3
nhanesData$DiabetesStr[ndx1] = 'Diabetes'
nhanesData$DiabetesStr[ndx2] = 'No Diabetes'
nhanesData$DiabetesStr[ndx3] = 'Borderline Diabetes'

nhanesData$obeseType = rep(1, nrow(nhanesData))
nhanesData$obeseType <- ifelse(nhanesData$BMI_Perc < 30, "Not Obese", "Obese")

#############################################################################################################
#########################
#
#  Write CSV file for other group members
#
#############################################################################################################
#########################

write.csv(nhanesData, "C:\\DePaul Coursework\\Fall 2017 CSC 465\\Final
Project\\NHANES_Filtered_ConbinedProjectDataset.csv")
```

```
#########################################################################################################################
#########################
#
#  Exploratory Histograms With Proper Data Labels
#
#########################################################################################################################
#########################

ggplot(data = nhanesData, aes(x = GenderStr)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
  theme( axis.text.x = element_text(angle = 20, hjust = 1),
         panel.background = element_rect(fill = "white", color = "white",size = 0.5),
         panel.border = element_rect(colour = "black", fill=NA, size=1),
         plot.title = element_text(hjust = 0.5),
         legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
  ggtitle("Gender Counts") +
  labs(x = "Gender", y = "Count")

ggplot(data = nhanesData, aes(x = RIDAGEYR)) + geom_histogram(bins = 15, fill = "steelblue", color = "black") +
  theme( axis.text.x = element_text(angle = 20, hjust = 1),
         panel.background = element_rect(fill = "white", color = "white",size = 0.5),
         panel.border = element_rect(colour = "black", fill=NA, size=1),
         plot.title = element_text(hjust = 0.5),
         legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
  ggtitle("Age Histogram") +
  labs(x = "Age (Years)", y = "Frequency")

ggplot(data = nhanesData, aes(x = EdStr)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
  theme( axis.text.x = element_text(angle = 20, hjust = 1),
         panel.background = element_rect(fill = "white", color = "white",size = 0.5),
         panel.border = element_rect(colour = "black", fill=NA, size=1),
         plot.title = element_text(hjust = 0.5),
         legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
  ggtitle("Education Counts") +
  labs(x = "Education", y = "Count")

ggplot(data = nhanesData, aes(x = MaritalStr)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
  theme( axis.text.x = element_text(angle = 20, hjust = 1),
         panel.background = element_rect(fill = "white", color = "white",size = 0.5),
         panel.border = element_rect(colour = "black", fill=NA, size=1),
         plot.title = element_text(hjust = 0.5),
         legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
  ggtitle("Marital Status Counts") +
  labs(x = "Marital Status Levels", y = "Count")

ggplot(data = nhanesData, aes(x = RaceStr)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
  theme( axis.text.x = element_text(angle = 20, hjust = 1),
         panel.background = element_rect(fill = "white", color = "white",size = 0.5),
         panel.border = element_rect(colour = "black", fill=NA, size=1),
         plot.title = element_text(hjust = 0.5),
         legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
  ggtitle("Race Counts") +
  labs(x = "Races", y = "Count")

ggplot(data = nhanesData, aes(x = IncomeStr)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
  scale_x_discrete(limits = c('$0 to $4,999', '$5,000 to $9,999', '$10,000 to $14,999', '$15,000 to $19,999', '$20,000 to $24,999',
                              '$25,000 to $34,999', '$35,000 to $44,999', '$45,000 to $54,999', '$55,000 to $64,999',
                              '$65,000 to $74,999', '$75,000 to $99,999', '$100,000 and Over')) +
  theme( axis.text.x = element_text(angle = 20, hjust = 1),
```

```
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Income Counts") +
 labs(x = "Income", y = "Count")

ggplot(data = nhanesData, aes(x = WHD010)) + geom_histogram(bins = 15, fill = "steelblue", color = "black") +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Height Histogram") +
 labs(x = "Height (in)", y = "Frequency")

ggplot(data = nhanesData, aes(x = WHD020)) + geom_histogram(bins = 15, fill = "steelblue", color = "black") +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Weight Histogram") +
 labs(x = "Weight (lb)", y = "Frequency")

ggplot(data = nhanesData, aes(x = obeseType)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Obesity Counts") +
 labs(x = "Obese Levels", y = "Count")

ggplot(data = nhanesData, aes(x = BMI_Perc)) + geom_histogram(bins = 15, fill = "steelblue", color = "black") +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("BMI Percentage Histogram") +
 labs(x = "BMI Percentage", y = "Frequency")

ggplot(data = nhanesData, aes(x = BPStr)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("Hypertension Counts") +
 labs(x = "Hypertension Levels", y = "Count")

ggplot(data = nhanesData, aes(x = DiabetesStr)) + geom_bar(stat = "count", fill = "steelblue", color = "black") +
 theme( axis.text.x = element_text(angle = 20, hjust = 1),
      panel.background = element_rect(fill = "white", color = "white",size = 0.5),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
```

```
 ggtitle("Diabetes Counts") +
 labs(x = "Diabetes Levels", y = "Count")

#########################################################################################################################
#########################
#
#  Creation of Heatmap with Age Versus Race and Color = BMI
#
#########################################################################################################################
#########################

ageMins = c(20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75)
ageMaxs = c(25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80)
nhanesData$ageGroups = rep(1, length(nhanesData$RIDAGEYR))
for (i in 1:length(ageMaxs)){
 ageStr = paste(ageMins[i], ' to ', ageMaxs[i])
 if (ageMaxs[i] == 80){
  ageNdx = (nhanesData$RIDAGEYR >= ageMins[i]) & (nhanesData$RIDAGEYR <= ageMaxs[i])
 }
 else{
  ageNdx = (nhanesData$RIDAGEYR >= ageMins[i]) & (nhanesData$RIDAGEYR < ageMaxs[i])
 }
 nhanesData$ageGroups[ageNdx] = ageStr
}


aggData = as.data.frame(aggregate(nhanesData$BMI_Perc, list(nhanesData$ageGroups, nhanesData$RaceStr), mean))

ggplot(data = aggData, aes(x = Group.1, y = Group.2)) +
 geom_tile(aes(fill = aggData$x)) +
 scale_fill_gradient(name = "Mean\nBMI", low = "yellow",high = "red") +
 theme( axis.text.x = element_text(angle = 90, hjust = 1),
       panel.background = element_rect(fill = "white", color = "white",size = 0.5),
       panel.border = element_rect(colour = "black", fill=NA, size=1),
       plot.title = element_text(hjust = 0.5),
       legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("BMI By Age and Race") +
 labs(x = "Age (Years)", y = "Race")


aggData2 = as.data.frame(aggregate(nhanesData$BMI_Perc, list(nhanesData$ageGroups, nhanesData$GenderStr), mean))

ggplot(data = aggData2, aes(x = Group.1, y = Group.2)) +
 geom_tile(aes(fill = aggData2$x)) +
 scale_fill_gradient(name = "Mean\nBMI", low = "yellow",high = "red") +
 theme( axis.text.x = element_text(angle = 90, hjust = 1),
       panel.background = element_rect(fill = "white", color = "white",size = 0.5),
       panel.border = element_rect(colour = "black", fill=NA, size=1),
       plot.title = element_text(hjust = 0.5),
       legend.background = element_rect(color = "black",  fill = "white", size = 0.5, linetype = "solid")) +
 ggtitle("BMI By Age and Gender") +
 labs(x = "Age (Years)", y = "Gender")



#########################################################################################################################
#########################
#
```

```
#  Creation of Stacked Bar Chart Of Count of Obese Versus Income (Income sorted by range values)
#
#############################################################################################################
#########################

ggplot(nhanesData, aes(x=IncomeStr)) +
 geom_bar(aes(fill=nhanesData$obeseType)) +
 scale_x_discrete(limits = c('$0 to $4,999', '$5,000 to $9,999', '$10,000 to $14,999', '$15,000 to $19,999', '$20,000 to $24,999',
                             '$25,000 to $34,999', '$35,000 to $44,999', '$45,000 to $54,999', '$55,000 to $64,999',
                             '$65,000 to $74,999', '$75,000 to $99,999', '$100,000 and Over')) +
 scale_fill_manual(values = c("gray", "red")) +
 labs(x="Income", y="Count", title="Income Counts") +
 theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.title = element_blank(),
      axis.text.x = element_text(angle=90, hjust=1, vjust=.5))


#############################################################################################################
#########################
#
#  Creation of Stacked Bar Chart Of Count of Obese Versus Marital Status
#
#############################################################################################################
#########################

ggplot(nhanesData, aes(x=MaritalStr)) +
 geom_bar(aes(fill=nhanesData$obeseType)) +
  scale_fill_manual(values = c("gray", "red")) +
 labs(x="Marital Status", y="Count", title="Marital Status Counts") +
 theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.title = element_blank(),
      axis.text.x = element_text(angle = 20, hjust = 1))


#############################################################################################################
#########################
#
#  Creation of Stacked Bar Chart Of Count of Obese Versus Education Level
#
#############################################################################################################
#########################

ggplot(nhanesData, aes(x=EdStr)) +
 geom_bar(aes(fill=nhanesData$obeseType)) +
 scale_fill_manual(values = c("gray", "red")) +
 labs(x="Education Level", y="Count", title="Education Level Counts") +
 theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
      panel.border = element_rect(colour = "black", fill=NA, size=1),
      plot.title = element_text(hjust = 0.5),
      legend.title = element_blank(),
      axis.text.x = element_text(angle = 20, hjust = 1))
```

```
#############################################################################################################
#########################
#
#  Creation of Stacked Bar Chart Of Count of Obese Versus Gender
#
#############################################################################################################
#########################

ggplot(nhanesData, aes(x=GenderStr)) +
 geom_bar(aes(fill=nhanesData$obeseType)) +
 scale_fill_manual(values = c("gray", "red")) +
 labs(x="Gender", y="Count", title="Gender Counts") +
 theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
       panel.border = element_rect(colour = "black", fill=NA, size=1),
       plot.title = element_text(hjust = 0.5),
       legend.title = element_blank(),
       axis.text.x = element_text(angle = 20, hjust = 1))


#############################################################################################################
#########################
#
#  Creation of Stacked Bar Chart Of Count of Obese Versus Race
#
#############################################################################################################
#########################

ggplot(nhanesData, aes(x=RaceStr)) +
 geom_bar(aes(fill=nhanesData$obeseType)) +
 scale_fill_manual(values = c("gray", "red")) +
 labs(x="Race", y="Count", title="Race Counts") +
 theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
       panel.border = element_rect(colour = "black", fill=NA, size=1),
       plot.title = element_text(hjust = 0.5),
       legend.title = element_blank(),
       axis.text.x = element_text(angle = 20, hjust = 1))


#############################################################################################################
#########################
#
#  Percent obese by income bar chart
#
#############################################################################################################
#########################

# create data aggregated by year, with mean of adjusted close (before transformation)
uniqIncomes = unique(nhanesData$IncomeStr)

IncomeTotCount = rep(1, length(uniqIncomes))
ObeseIncCount = rep(1, length(uniqIncomes))
femaleObeseCount = rep(1, length(uniqIncomes))
maleObeseCount = rep(1, length(uniqIncomes))
femaleTotCount = rep(1, length(uniqIncomes))
maleTotCount = rep(1, length(uniqIncomes))
for (i in 1:length(uniqIncomes)){
 entryNdx = uniqIncomes[i]
 currentIncomeNdx = nhanesData$IncomeStr == entryNdx
```

```
  IncomeTotCount[i] = sum(currentIncomeNdx)

  currentIncomeData = nhanesData[currentIncomeNdx,]
  ObeseIncCount[i] = count(currentIncomeData$ObeseFlag)
  femaleTotCount[i] = sum(currentIncomeData$GenderStr == "Female")
  maleTotCount[i] = sum(currentIncomeData$GenderStr == "Male")

  obeseNdx = currentIncomeData$ObeseFlag == 1
  femaleObeseCount[i] = sum(currentIncomeData$GenderStr[obeseNdx] == "Female")
    maleObeseCount[i] = sum(currentIncomeData$GenderStr[obeseNdx] == "Male")

}

percObese = (ObeseIncCount / IncomeTotCount) * 100
femalePercObese = (femaleObeseCount / femaleTotCount) * 100
MalePercObese = (maleObeseCount / maleTotCount) * 100

aggData = data.frame(uniqIncomes, ObeseIncCount, IncomeTotCount, percObese, femalePercObese, MalePercObese)


femalePlot = ggplot(aggData, aes(x=uniqIncomes, femalePercObese)) + geom_bar(stat = "identity", fill = "steelblue") +
  scale_x_discrete(limits = c('$0 to $4,999', '$5,000 to $9,999', '$10,000 to $14,999', '$15,000 to $19,999', '$20,000 to $24,999',
                              '$25,000 to $34,999', '$35,000 to $44,999', '$45,000 to $54,999', '$55,000 to $64,999',
                              '$65,000 to $74,999', '$75,000 to $99,999', '$100,000 and Over')) +
  labs(x="Income", y="Percent Obese", title="Female Percent Obese") +
  theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
        panel.border = element_rect(colour = "black", fill=NA, size=1),
        plot.title = element_text(hjust = 0.5),
        axis.text.x = element_text(angle=90, hjust=1, vjust=.5)) +
  scale_y_continuous(breaks = c(0, 2, 4, 6, 8, 10, 12), limits = c(0,12))

malePlot = ggplot(aggData, aes(x=uniqIncomes, MalePercObese)) + geom_bar(stat = "identity", fill = "steelblue") +
  scale_x_discrete(limits = c('$0 to $4,999', '$5,000 to $9,999', '$10,000 to $14,999', '$15,000 to $19,999', '$20,000 to $24,999',
                              '$25,000 to $34,999', '$35,000 to $44,999', '$45,000 to $54,999', '$55,000 to $64,999',
                              '$65,000 to $74,999', '$75,000 to $99,999', '$100,000 and Over')) +
  labs(x="Income", y="Percent Obese", title="Male Percent Obese") +
  theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
        panel.border = element_rect(colour = "black", fill=NA, size=1),
        plot.title = element_text(hjust = 0.5),
        axis.text.x = element_text(angle=90, hjust=1, vjust=.5)) +
  scale_y_continuous(breaks = c(0, 2, 4, 6, 8, 10, 12), limits = c(0,12))


grid.arrange(malePlot, femalePlot, nrow=1)


#############################################################################################################
#########################
#
#  Percent obese by education level bar chart
#
#############################################################################################################
#########################

# create data aggregated by year, with mean of adjusted close (before transformation)
uniqEds = unique(nhanesData$EdStr)

EdTotCount = rep(1, length(uniqEds))
```

```
ObeseEdCount = rep(1, length(uniqEds))
femaleObeseCount = rep(1, length(uniqEds))
maleObeseCount = rep(1, length(uniqEds))
femaleTotCount = rep(1, length(uniqEds))
maleTotCount = rep(1, length(uniqEds))
for (i in 1:length(uniqEds)){
  entryNdx = uniqEds[i]
  currentEdNdx = nhanesData$EdStr == entryNdx
  EdTotCount[i] = sum(currentEdNdx)

  currentEdData = nhanesData[currentEdNdx,]
  ObeseEdCount[i] = count(currentEdData$ObeseFlag)
  femaleTotCount[i] = sum(currentEdData$GenderStr == "Female")
  maleTotCount[i] = sum(currentEdData$GenderStr == "Male")

  obeseNdx = currentEdData$ObeseFlag == 1
  femaleObeseCount[i] = sum(currentEdData$GenderStr[obeseNdx] == "Female")
  maleObeseCount[i] = sum(currentEdData$GenderStr[obeseNdx] == "Male")

}

percObese = (ObeseEdCount / EdTotCount) * 100
femalePercObese = (femaleObeseCount / femaleTotCount) * 100
MalePercObese = (maleObeseCount / maleTotCount) * 100

aggData = data.frame(uniqEds, ObeseEdCount, EdTotCount, percObese, femalePercObese, MalePercObese)


femalePlot = ggplot(aggData, aes(x=uniqEds, femalePercObese)) + geom_bar(stat = "identity", fill = "steelblue") +
  labs(x="Education Levels", y="Percent Obese", title="Female Percent Obese") +
  scale_x_discrete(limits = c('Less Than 9th', '9th to 12th No Grad', 'High School Grad/GED', 'Some College/AA', 'College
Grad/Above')) +
  theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
        panel.border = element_rect(colour = "black", fill=NA, size=1),
        plot.title = element_text(hjust = 0.5),
        axis.text.x = element_text(angle=90, hjust=1, vjust=.5)) +
  scale_y_continuous(breaks = c(0, 2, 4, 6, 8), limits = c(0,8))

malePlot = ggplot(aggData, aes(x=uniqEds, MalePercObese)) + geom_bar(stat = "identity", fill = "steelblue") +
  labs(x="Education Levels", y="Percent Obese", title="Male Percent Obese") +
  scale_x_discrete(limits = c('Less Than 9th', '9th to 12th No Grad', 'High School Grad/GED', 'Some College/AA', 'College
Grad/Above')) +
  theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
        panel.border = element_rect(colour = "black", fill=NA, size=1),
        plot.title = element_text(hjust = 0.5),
        axis.text.x = element_text(angle=90, hjust=1, vjust=.5)) +
  scale_y_continuous(breaks = c(0, 2, 4, 6, 8), limits = c(0,8))


grid.arrange(malePlot, femalePlot, nrow=1)


#############################################################################################################
#########################
#
#  Percent obese by marital status bar chart
#
```

```
#############################################################################################################
#########################

# create data aggregated by year, with mean of adjusted close (before transformation)
uniqMarital = unique(nhanesData$MaritalStr)

MaritalTotCount = rep(1, length(uniqMarital))
ObeseMaritalCount = rep(1, length(uniqMarital))
femaleObeseCount = rep(1, length(uniqMarital))
maleObeseCount = rep(1, length(uniqMarital))
femaleTotCount = rep(1, length(uniqMarital))
maleTotCount = rep(1, length(uniqMarital))
for (i in 1:length(uniqMarital)){
  entryNdx = uniqMarital[i]
  currentMaritalNdx = nhanesData$MaritalStr == entryNdx
  MaritalTotCount[i] = sum(currentMaritalNdx)

  currentMaritalData = nhanesData[currentMaritalNdx,]
  ObeseMaritalCount[i] = count(currentMaritalData$ObeseFlag)
  femaleTotCount[i] = sum(currentMaritalData$GenderStr == "Female")
  maleTotCount[i] = sum(currentMaritalData$GenderStr == "Male")

  obeseNdx = currentMaritalData$ObeseFlag == 1
  femaleObeseCount[i] = sum(currentMaritalData$GenderStr[obeseNdx] == "Female")
  maleObeseCount[i] = sum(currentMaritalData$GenderStr[obeseNdx] == "Male")

}

percObese = (ObeseMaritalCount / MaritalTotCount) * 100
femalePercObese = (femaleObeseCount / femaleTotCount) * 100
MalePercObese = (maleObeseCount / maleTotCount) * 100

aggData = data.frame(uniqMarital, ObeseMaritalCount, MaritalTotCount, percObese, femalePercObese, MalePercObese)


femalePlot = ggplot(aggData, aes(x=uniqMarital, femalePercObese)) + geom_bar(stat = "identity", fill = "steelblue") +
  labs(x="Marital Status", y="Percent Obese", title="Female Percent Obese") +
  theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
        panel.border = element_rect(colour = "black", fill=NA, size=1),
        plot.title = element_text(hjust = 0.5),
        axis.text.x = element_text(angle=90, hjust=1, vjust=.5)) +
  scale_y_continuous(breaks = c(0, 2, 4, 6, 8, 10, 12), limits = c(0,12))

malePlot = ggplot(aggData, aes(x=uniqMarital, MalePercObese)) + geom_bar(stat = "identity", fill = "steelblue") +
  labs(x="Marital Status", y="Percent Obese", title="Male Percent Obese") +
  theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
        panel.border = element_rect(colour = "black", fill=NA, size=1),
        plot.title = element_text(hjust = 0.5),
        axis.text.x = element_text(angle=90, hjust=1, vjust=.5)) +
  scale_y_continuous(breaks = c(0, 2, 4, 6, 8, 10, 12), limits = c(0,12))


grid.arrange(malePlot, femalePlot, nrow=1)


#############################################################################################################
#########################
#
```

```
#  Percent obese by race bar chart
#
#############################################################################################################
#########################

# create data aggregated by year, with mean of adjusted close (before transformation)
uniqRaces = unique(nhanesData$RaceStr)

RaceTotCount = rep(1, length(uniqRaces))
ObeseRaceCount = rep(1, length(uniqRaces))
femaleObeseCount = rep(1, length(uniqRaces))
maleObeseCount = rep(1, length(uniqRaces))
femaleTotCount = rep(1, length(uniqRaces))
maleTotCount = rep(1, length(uniqRaces))
for (i in 1:length(uniqRaces)){
 entryNdx = uniqRaces[i]
 currentRaceNdx = nhanesData$RaceStr == entryNdx
 RaceTotCount[i] = sum(currentRaceNdx)

 currentRaceData = nhanesData[currentRaceNdx,]
 ObeseRaceCount[i] = count(currentRaceData$ObeseFlag)
 femaleTotCount[i] = sum(currentRaceData$GenderStr == "Female")
 maleTotCount[i] = sum(currentRaceData$GenderStr == "Male")

 obeseNdx = currentRaceData$ObeseFlag == 1
 femaleObeseCount[i] = sum(currentRaceData$GenderStr[obeseNdx] == "Female")
 maleObeseCount[i] = sum(currentRaceData$GenderStr[obeseNdx] == "Male")

}

percObese = (ObeseRaceCount / RaceTotCount) * 100
femalePercObese = (femaleObeseCount / femaleTotCount) * 100
MalePercObese = (maleObeseCount / maleTotCount) * 100

aggData = data.frame(uniqRaces, ObeseRaceCount, RaceTotCount, percObese, femalePercObese, MalePercObese)


femalePlot = ggplot(aggData, aes(x=uniqRaces, femalePercObese)) + geom_bar(stat = "identity", fill = "steelblue") +
 labs(x="Race", y="Percent Obese", title="Female Percent Obese") +
 theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
       panel.border = element_rect(colour = "black", fill=NA, size=1),
       plot.title = element_text(hjust = 0.5),
       axis.text.x = element_text(angle=90, hjust=1, vjust=.5)) +
 scale_y_continuous(breaks = c(0, 2, 4, 6, 8, 10), limits = c(0,10))

malePlot = ggplot(aggData, aes(x=uniqRaces, MalePercObese)) + geom_bar(stat = "identity", fill = "steelblue") +
 labs(x="Race", y="Percent Obese", title="Male Percent Obese") +
 theme(panel.background = element_rect(fill = "white", color = "white",size = 0.5, linetype = "solid"),
       panel.border = element_rect(colour = "black", fill=NA, size=1),
       plot.title = element_text(hjust = 0.5),
       axis.text.x = element_text(angle=90, hjust=1, vjust=.5)) +
 scale_y_continuous(breaks = c(0, 2, 4, 6, 8, 10), limits = c(0,10))


grid.arrange(malePlot, femalePlot, nrow=1)
```