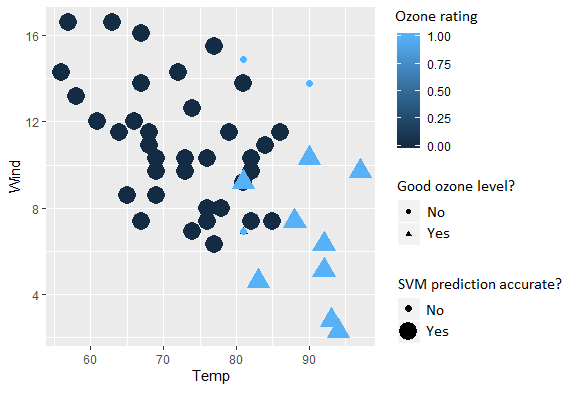
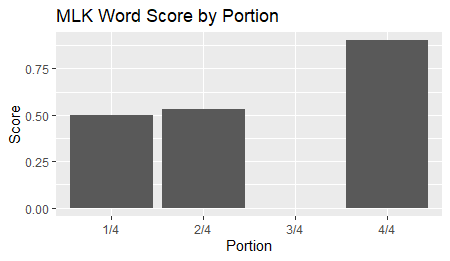
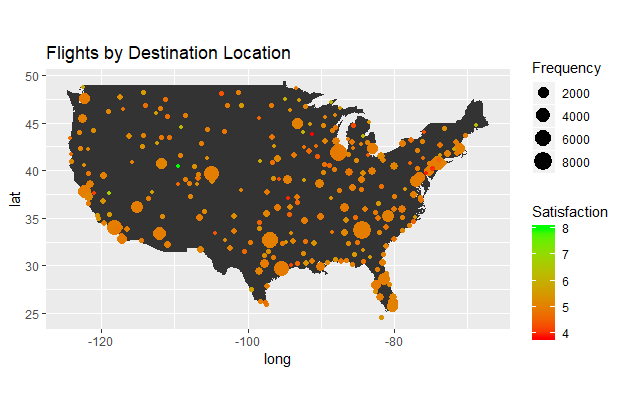
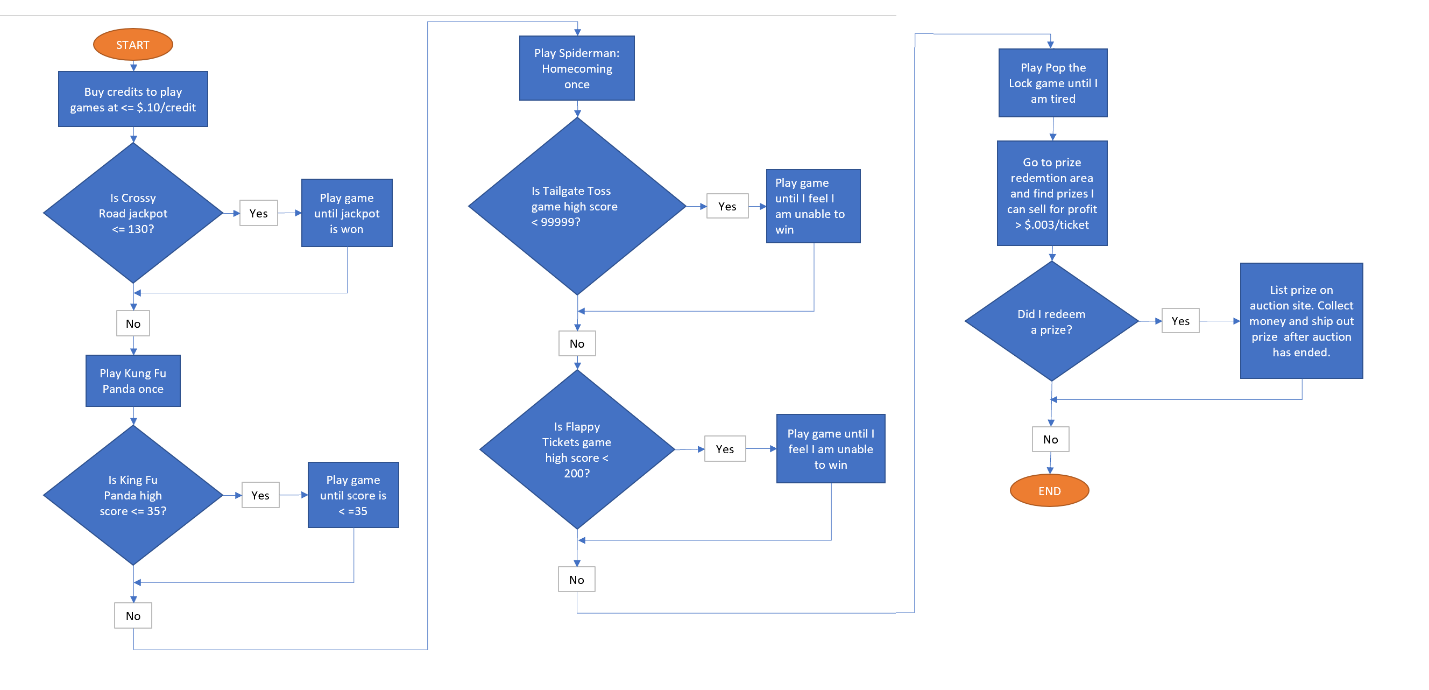
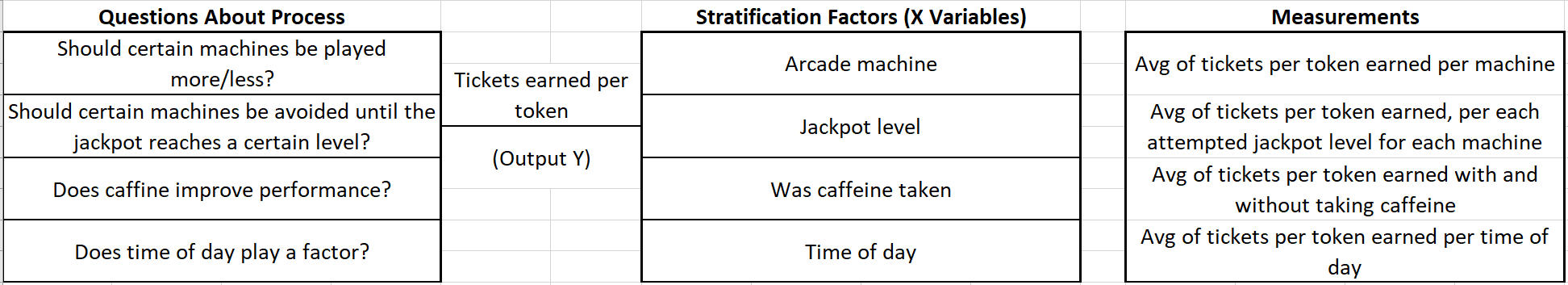
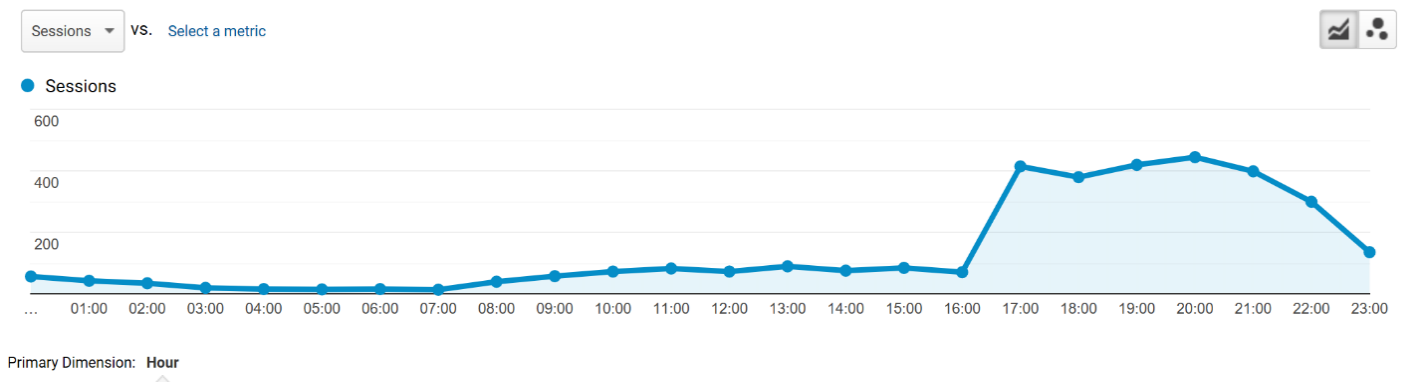
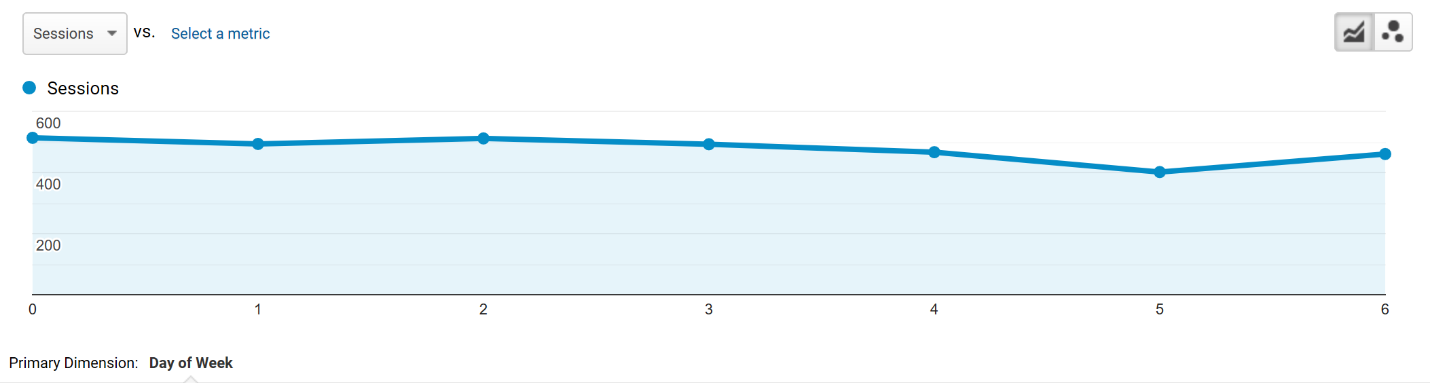
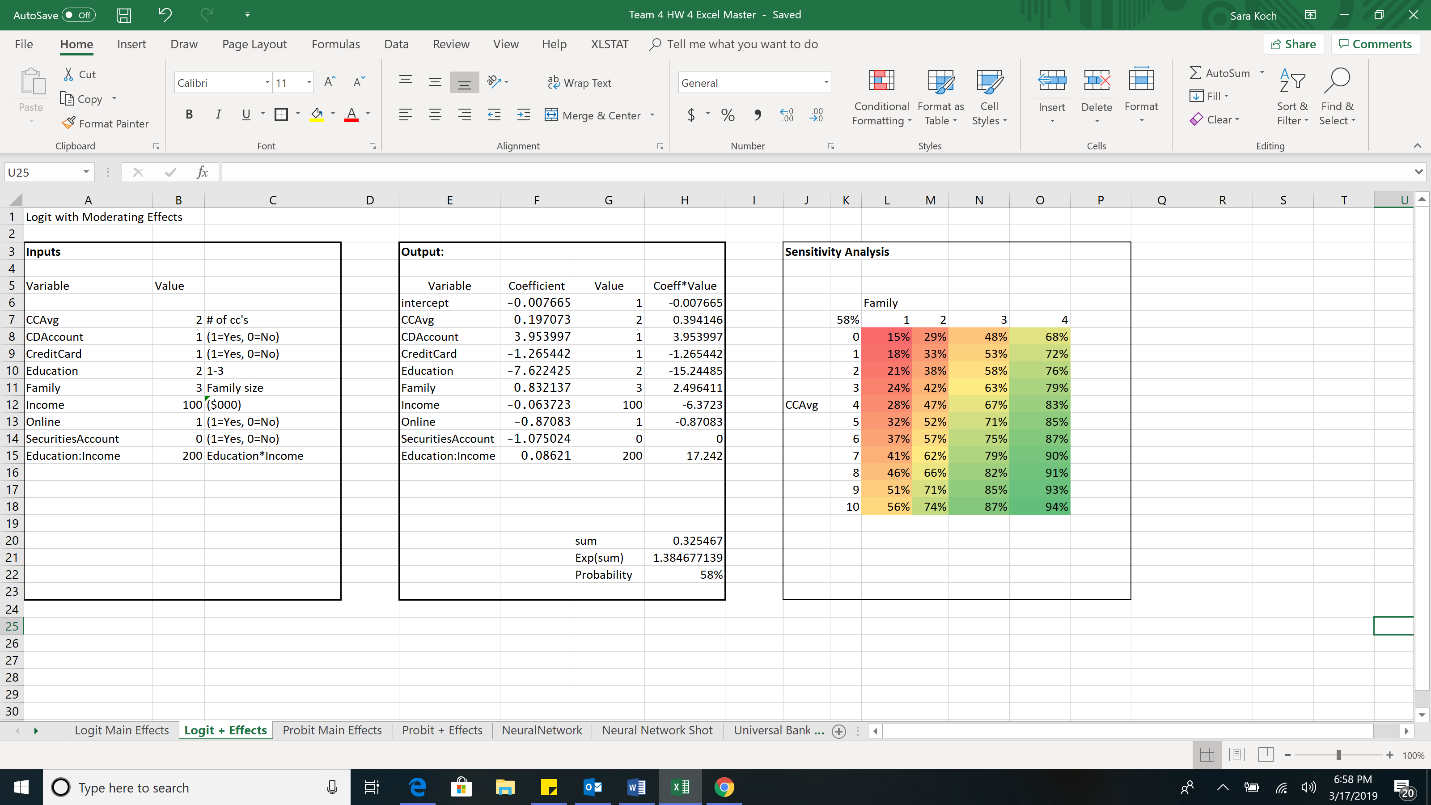
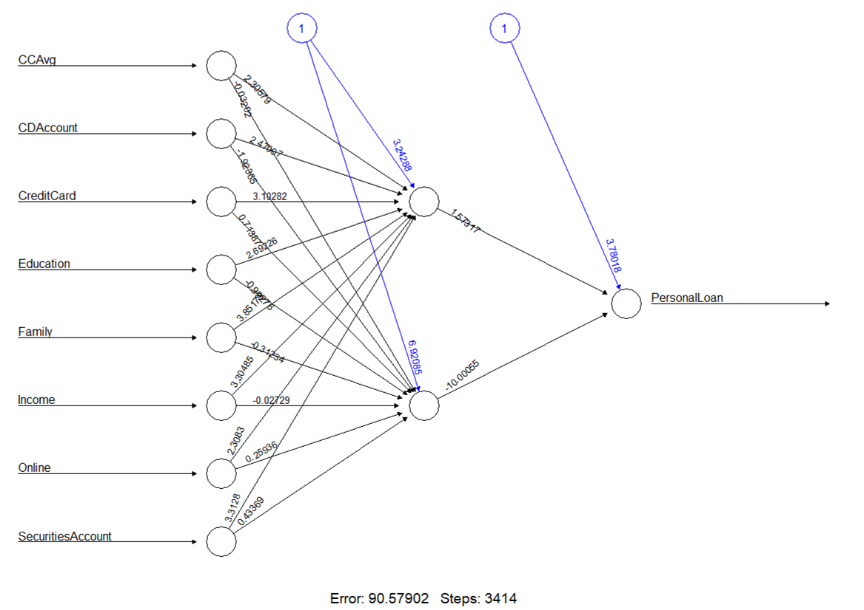
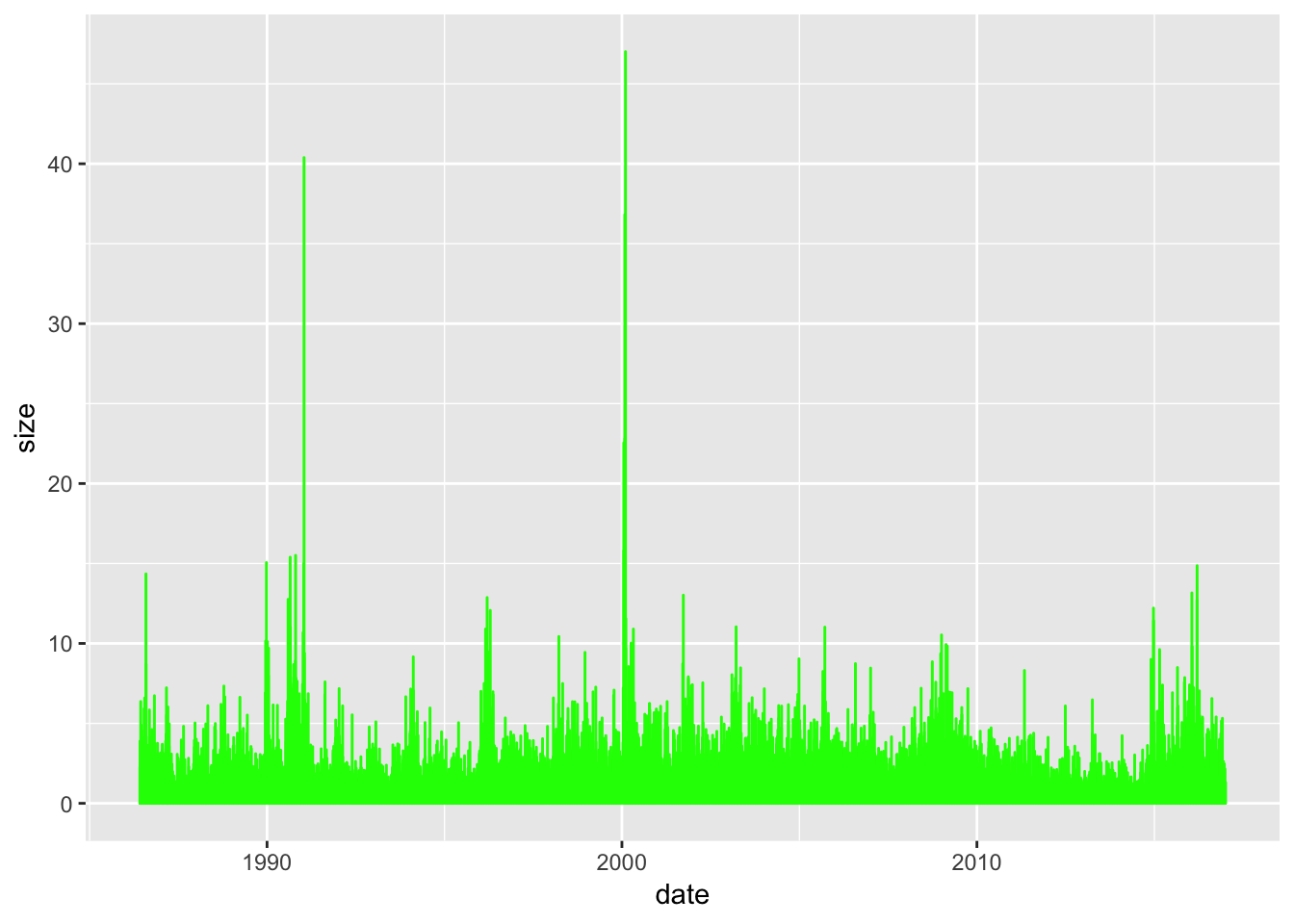
**Portfolio Milestone   
Victor Kaidas  
SUID: 649789315  
Submitted: January 15, 2020  
https://github.com/vksyr/MS\_DS\_Portfolio**  
  
**I. Introduction**   
  
The MS of Applied Data Science program at Syracuse University provides students the opportunity to collect, manage, analyze, and develop insights using data from a multitude of domains using various tools and techniques. Data scientists are critical to the success of any organization. According to Glassdoor, data science jobs are ranked No. 1 best job of 2016, 2017, and 2018. As the data science field evolves, the demand for analytics skills continues to grow. Employers are actively seeking candidates with the advanced technical expertise to make data-driven decisions.  
  
Data science is a complex field, incorporating many areas such as mathematics, descriptive statistics, computer science and programming, statistical modeling, database technologies, data modeling, artificial intelligence and learning, text mining, natural language processing, visualization, and predictive analytics and is applicable to many fields including social media, medicine, security, health care, social sciences, biological sciences, engineering, defense, business, economics, finance, marketing, geolocation, and many more.  
  
Through the Applied Data Science program, students learn to collect and organize data, identify patterns in data via visualization, statistical analysis, and data mining. The student learns to develop alternative strategies based on the data and a plan of action to implement the business decisions derived from the analyses. They develop communication skills regarding data and its analysis for managers, IT professionals, programmers, statisticians, and other relevant professionals in their organization. Lastly, the student learns to synthesize the ethical dimensions of data science practice. I started my MS of Applied Data Science program at Syracuse University in September 2018 and anticipate graduation in March, 2020.  
  
**II. Courses  
  
IST 687 – Applied Data Science**  
  
The course introduced applied examples of data collection, processing, transformation, management, and analysis, with hands-on introduction to data science experience. Key concepts explored including applied statistics, information visualization, text mining and machine learning. “R,” an open source statistical analysis and visualization system, was used throughout the course. R is known to be the most popular choice among data analysts worldwide.  
  
In the course, we learned to setup R in R-Studio and install/load packages. We developed skills for manipulating and transforming data, such as accounting for missing values. We manipulated columns, vectors and data frames. We learned to inspect the data with commands such as str() and summary(). In learning to analyze distributions, we learned how to generate random distributions of different shapes using commands, such as rnorm(). We learned to display descriptive statistics, quantiles, and make histograms and boxplots, of the distributions. We learned about sampling in R, with commands such as rep() and sample(), and making inferences about populations from the samples.   
  
We learned about the built-in R datasets and how to import external data and the various formats. We learned to subset data in R. We learned to use the apply functions, such as tapply(). We applied tapply() command when analyzing accident data to answer questions such as “How many accidents had injuries?” and “What are the injuries by day?” We learned to create bar and scatter charts, and heatmaps, and manipulate the plot esthetics. We learned to make visualizations with the “ggplot2” library. We learned how to make visualizations informative and easy to understand. We built boxplots and histograms on air quality data, and built heatmaps and scatter charts to see patterns after exploring the data.  
  
We worked with the ggmap() command to create geographical maps. We customized the map zoom level, used the geocode() function to lookup coordinates. We created maps with points, and depicted densities. Used the tapply() command to analyze and then displayed color-coded median income by zip code. We were introduced to linear predictive modeling in R. We used scatterplots to visualize a correlation, and used the lm() function to make a model and make predictions. We applied this by creating bivariable plots and regression models to predict the number of baby fawns by adult antelope population, precipitation, and severity of the past winter.

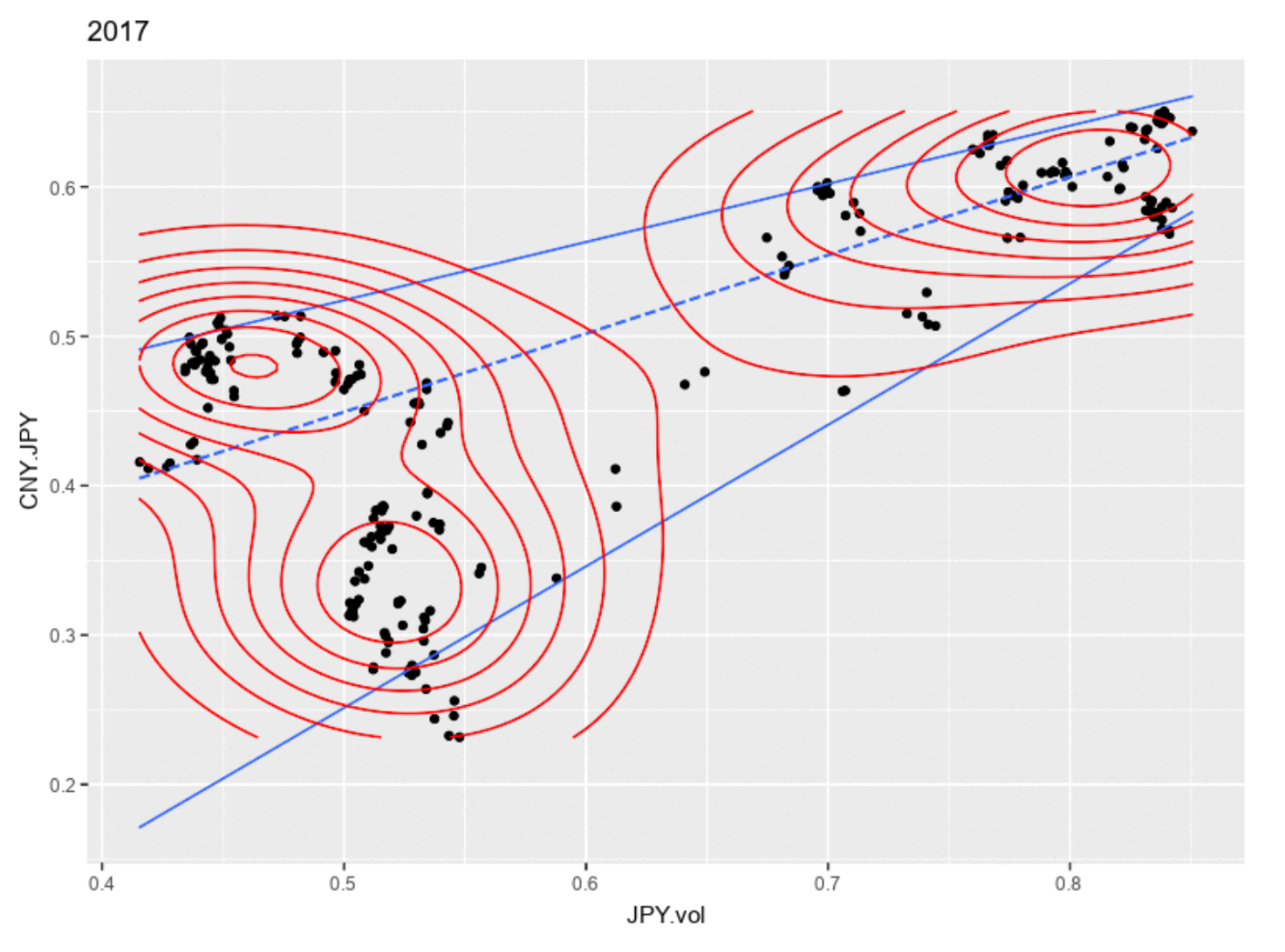
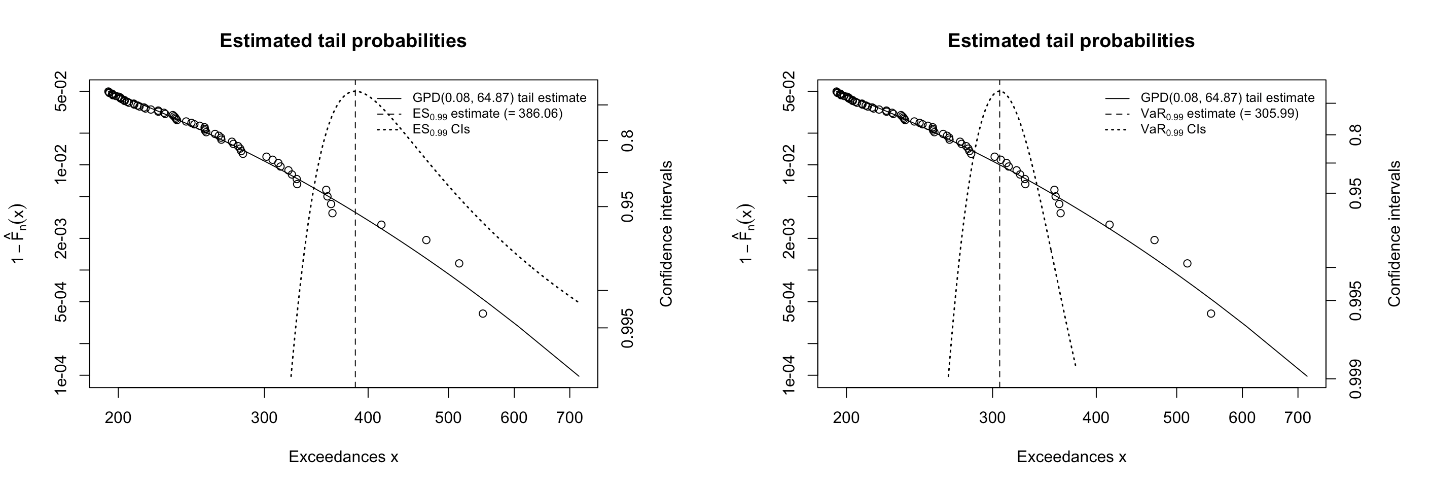
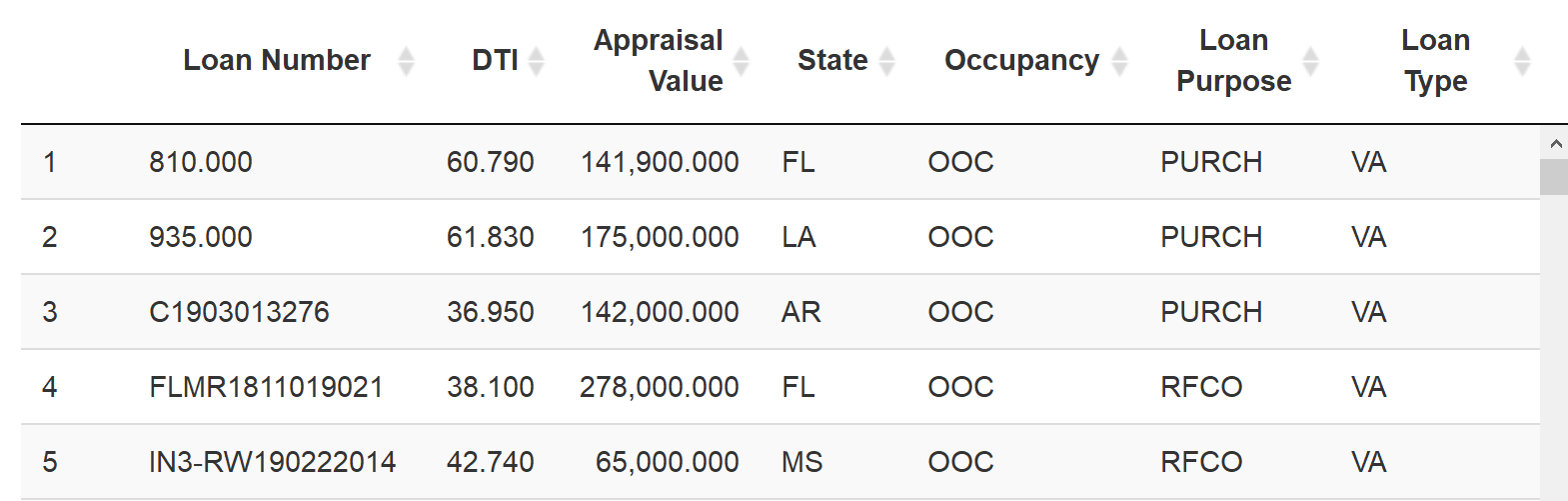
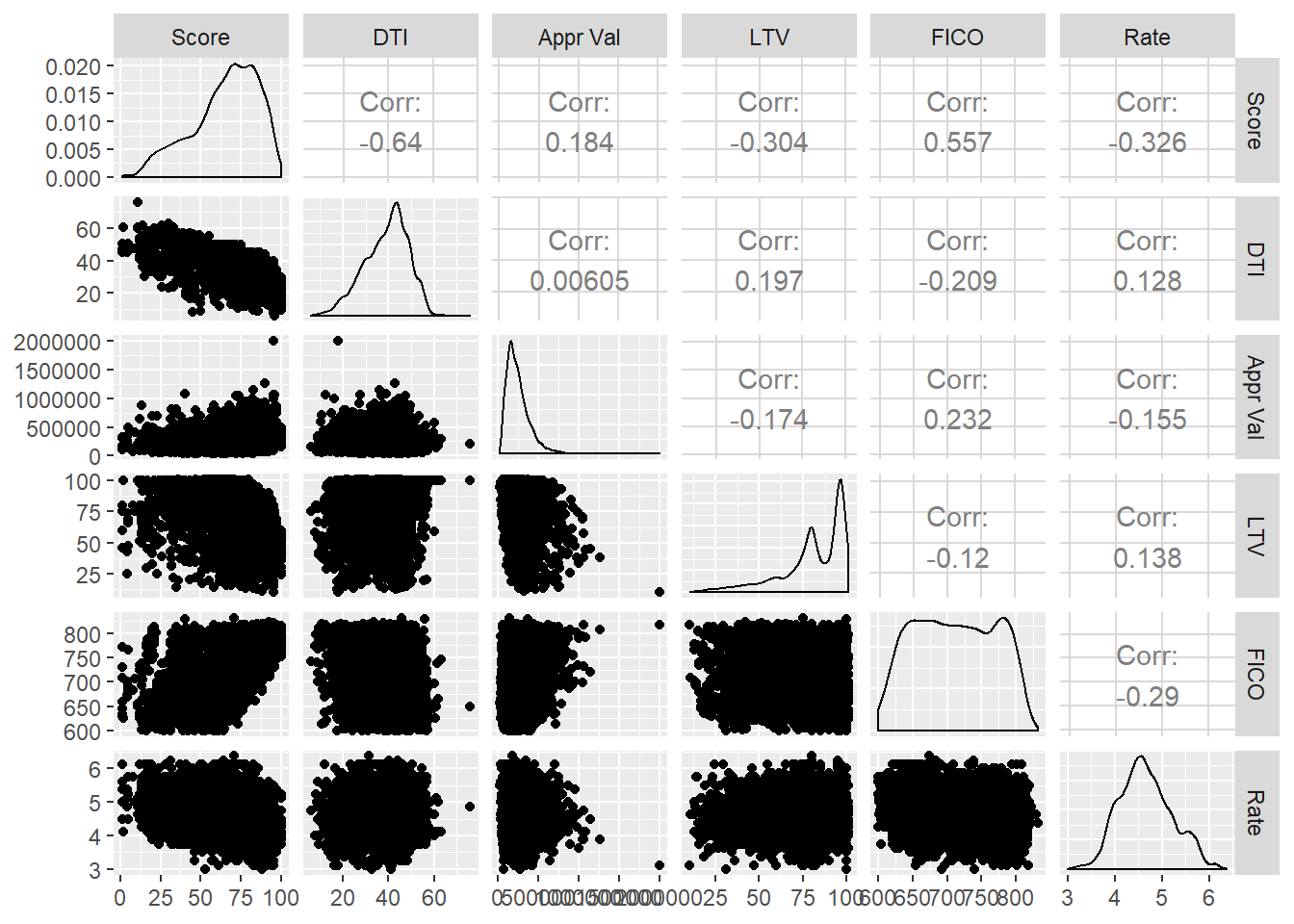
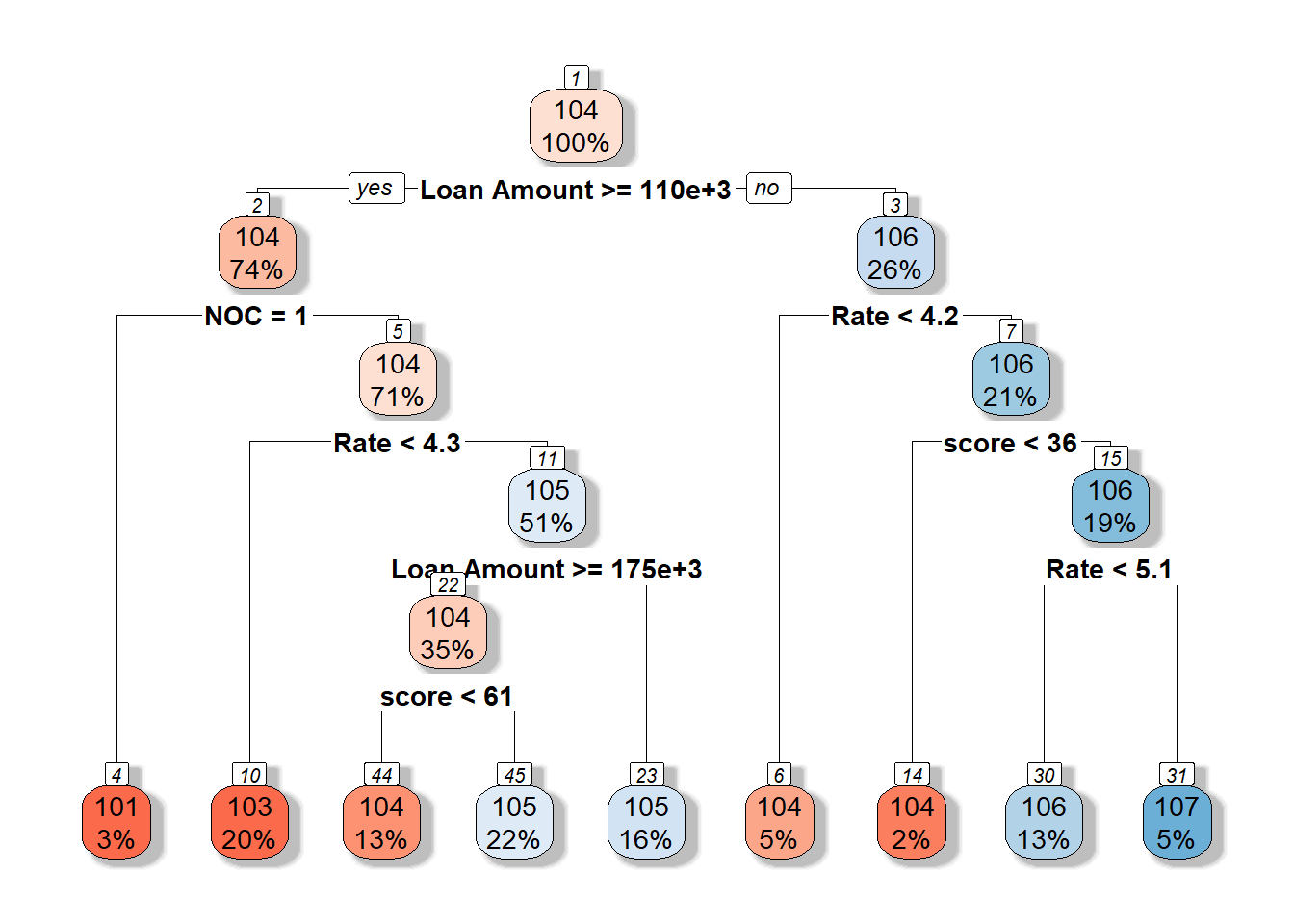
We were introduced to an unsupervised machine learning algorithm, association rule mining (AR). We used the apriori() command from the arules package. For the association rule mining to work, we learned to transpose the data to be categorized. We were introduced to a supervised machine learning algorithm named Support Vector Machine (SVM) and Naïve Bayes (NB). We applied SVM and NB algorithms to predict ozone levels, with the results plotted in figure 1.  
  
We were introduced to text mining and sentiment analysis by analyzing the Martin Luther King Jr speech, I Have a Dream. Overall sentiment value was discovered by cross-referencing sentimental words with a list of scores for each word from -5 to 5. The overall score slightly positive, at 0.45. The speech was split to analyze the sentiment for each quarter, as shown in figure 2.  
  
For a final team project, our mission was to provide insight to a fictitious airline company based on customer satisfaction using customer survey data provided in json format. First, we came up with data questions the company would like answered about the data. Then we inspected and cleaned the data, along with performed descriptive statistics on the data. We built a correlation matrix to study which variables were strongly correlated with satisfaction. We used the tapply() and aggregate() functions to compare means of the various columns by category.   
  
We plotted satisfaction based on destination city on a map, as shown in figure 3 and discovered no correlation to how far north, south, east, and west the destination locations are. This was calculated using the lm() function on the cities’ latitude and then longitude.   
  
We added the most influential variables to create a linear predictive model. While the linear model was statistically significant, we created AR and SVM algorithm models from the subset of data. While, the results from the AR did not appear to be meaningful, the SVM model aligned with our linear model. Along with answering the business questions, we made recommendations to the airline company on the factors deemed most influential to customer satisfaction.  
 **  
Figure 1  
  
  
  
Figure 2****Figure 3  
  
  
MBC 638 – Data Analysis and Decision Making**In the Data Analysis and Decision Making course, we used the six sigma framework, DMAIC, or Define, Measure, Analyze, Improve, and Control, a data-driven improvement cycle for improving, optimizing and stabilizing business processes and designs in order to make better business decisions.   
  
We learned about the fundamentals of statistics and the DMAIC framework. In the “define” phase, we learned to clearly define the business problem, output measure, customer, scope, goals, and resources. We discussed the output formula, descriptive statistics, as well as the different types of data (continuous vs discrete). We learned to measure processes with the Sigma Quality Level (SQL). We about various tools, such as thought process maps, SIPOC, cause and effect diagrams, Paretto charts, histogram/frequency charts, and trend charts and time plots to help define the process.  
  
In the “measure” phase, we learned how to the validate the measurement system and collect baseline data. We discussed mapping a process/value-stream, forms of waste, measurement error, reproducibility, and repeatability. We learned about the importance of operational definitions, and the use of Kappa in measuring agreement in a measurement system. We also talked about using data stratification trees.   
  
In the “analyze” phase, we learned to describe and present the data to discover the root cause(s), identify/prioritize critical inputs, and determine the inputs impact on the output. We discussed common distributions and used hypothesis and Chi-square testing on samples to perform inferential statistics and describe the likelihood of an event occurring. We discussed confidence intervals and how sample size is related. We used regression analysis to identify relationships between the dependent and predictor variables. We performed simple and multiple linear regression, and calculated correlation among variables.   
  
In the “improve” stage, we learned to implement a solution, run a pilot, evaluate the results, and complete a hypothesis test. We used solution selection matrices to assessing the positive impact of each proposed solution. Lastly, in the “control” stage, we learned to implement process changes and controls, verify expected performance was achieved, and monitor performance to sustain new levels. We used Xbar/R and ImR control charts, and time series analysis, to include first order autoregressive (AR1), moving average forecast, and exponential smoothing models.   
  
Throughout the course, I worked on a process improvement project. The project involved following the DMAIC framework to improve a personal process. For the “define” stage, my problem statement read**:**  
*Increase profit made from my hobby profiting at an arcade by improving ticket to money spent ratio. I know that I currently make a profit at a local arcade by redeeming prizes that are valued more than the price I paid to play the games and win the prizes. Improving the process would be helpful so that I can maximize my time to make the most value. I know that I can better maximize my profit because of known deficiencies in my current process. An example of a deficiencies would be attempting to beat a game that already has been beat several times, and now is much harder to beat. I know that my winning chances have decreased, yet the price to play has remained the same.*   
  
The goal was to reach an average profit margin of 50%. I defined the output as yielded profit, and continuous variable. Also under the “define” phase, a process map, shown in figure 1, was created that shows the process to be improved. Under the “measure” stage, I created a data stratification tree, shown in figure 2. The tree maps questions about the process to output and dependent variables, and specifies the measurement systems. I calculated my ideal sample size as 47 observations, using the sample size formula.   
  
My actual observations, consisted of my own performance over 6 days, tracking the following:  
- The name of the game  
- The amount of credits per play (discrete)  
- The time of day (continuous)  
- Whether I had consumed caffeine (discrete, binary)  
- Amount of tickets won (continuous)  
- The high score required to win the jackpot (continuous)  
  
In the “analyze” phase, hypothesis testing and regression analysis was used to discover the root causes, identify/prioritize critical inputs, and determine the inputs impact on the output. With one of the data questions read: “Did caffeine consumption improve performance,” I developed a null hypothesis stating “There is no difference in performance in the population with and without caffeine.” With a p-value > .05, I concluded there was no significant evidence to reject the null hypothesis. Additionally, with significance testing, time-of-day did not appear to be significantly affecting the output. For the data question “Should certain machines be played more/less,”I constructed trend/line charts a horizontal line representing the goal threshold. I was able to visualize and identify which games to play, at when to stop playing.   
  
In the “improve” process, a new process map was drawn based on the analysis from the line charts. In the last phase, “control,” a time-series control chart was created and performance was verified as the new ticket-to-token ratio jumped from 36.26 to 55.16, exceeding the goal and increasing the Sigma Quality Level (SQL) from 1.24σ to 1.86σ. Additionally in the control phase, a moving average model was produced to forecast results from the improved process. **Figure 1**

**  
Figure 2  
  
  
SCM 651 – Business Analytics**Business analytics are analytical techniques that facilitate achievement of business objectives. These techniques are achieved by reporting data to analyze trends, creating predictive models for forecasting, and optimizing business processes for enhanced performance. Business analytics is important in business today, from the banking industry, handling risk, credit analysis, and fraud detection, to retail, handling marketing, pricing, and retail sales, to big business, handling supply chain, transportation, and optimization.  
  
We analyzed a data set of house prices and performed categorization of data. We developed two pivot tables of average price and average square feet by type of construction (brick) and neighborhood, as well as a graphical representation with pivot charts. Next, we performed correlation and regression analysis of all quantitative variables in the data. We discovered that the largest factor on house prices is square footage. As square footage increase, price also increases. Additionally, we found a surprising correlation that the number of offers increases, the price of the home decreases. We also performed multivariate regression to quantify how multiple variables impact the house prices. Since it was discovered that square footage increases a home’s price, we decided to run a sensitivity analysis on how the number of bedrooms and bathrooms impacts a home’s price**.**  
  
We used Google Analytics to analyze the effectiveness of ads from the Whitman School of Management. We looked at the number of clicks, bounce rate, sessions, average number of pages visited, and total cost to identity the most effect campaigns. We analyze specific geographic regions to discover the best regions to focus on, along with best keywords, and times generating the most visitors. Based on the reports in figure 1, we discovered it is best to spend our advertising dollars daily, between the hours of 5:00pm and midnight and not focus on any specific days.  
  
We analyzed a sales data set on Harry Potter book #7 to determine the best price on the next sequel to the Harry Potter series. Using power regression, we created an equation to predict the percent purchased based on the price. We calculated price points for different scenarios of price and books sold, factoring in the cost to publish each book. In deciding which cost point to accept from the publisher we considered two things. First, we looked at which option yielded the highest profit. Second, we assessed which option appears to be the most realistic.   
  
We performed logit and probit regression analysis on universal bank data to determine significant factors which influence whether a customer takes out a loan. Moderating effects were added to these models to pick the best model. We chose the logit model with moderating effects because we wanted to get a more accurate picture of the extreme values between our data. A sensitivity analysis of this model is shown in figure 2. A neural network was created in R, as shown in figure 3 and aligned with the logit model in its direction, however the probability values were much lower in the neural network.   
  
  
  


 **Figure 1**

**Figure 2** **Figure 3**  
 **FIN 654 – Financial Analytics**

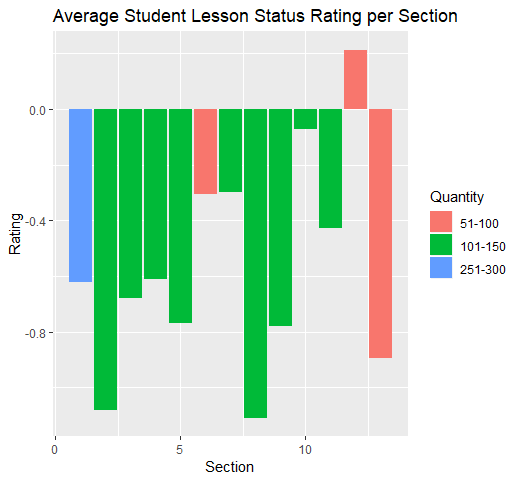
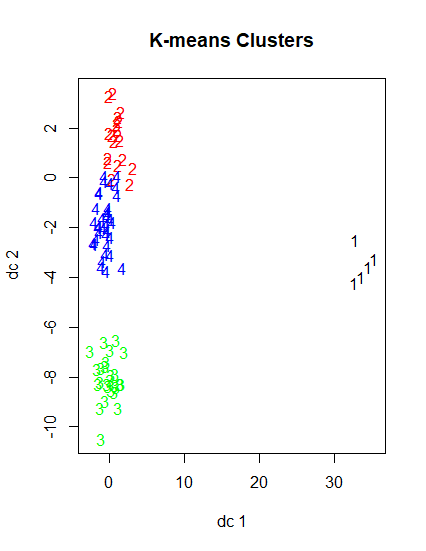
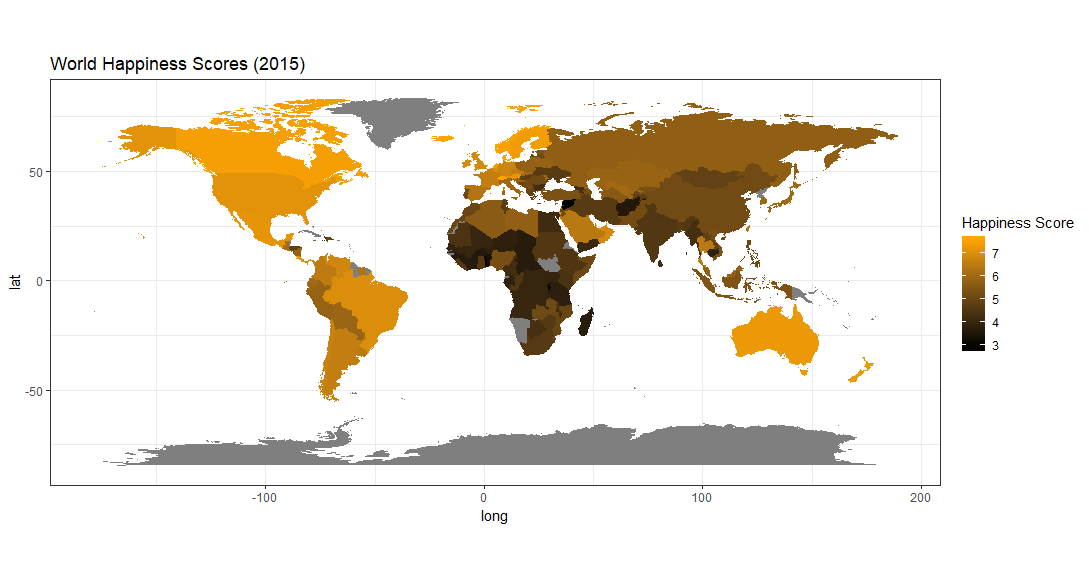
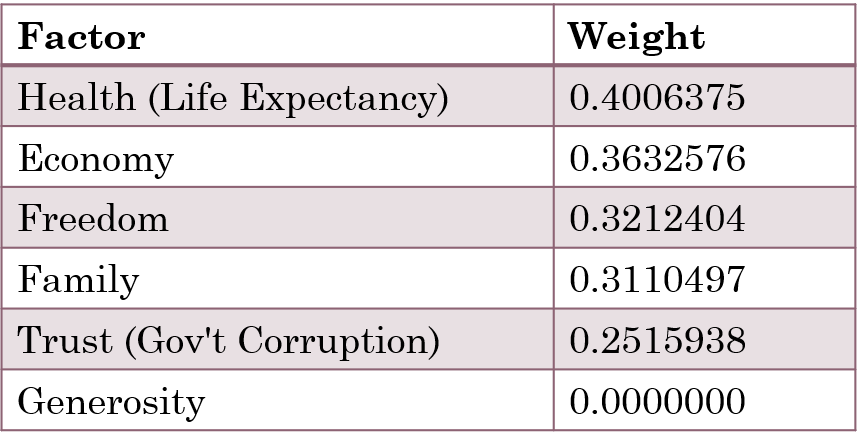
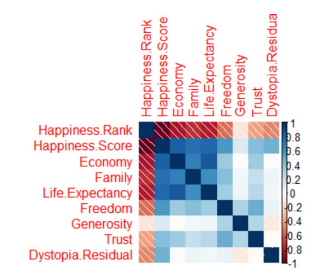
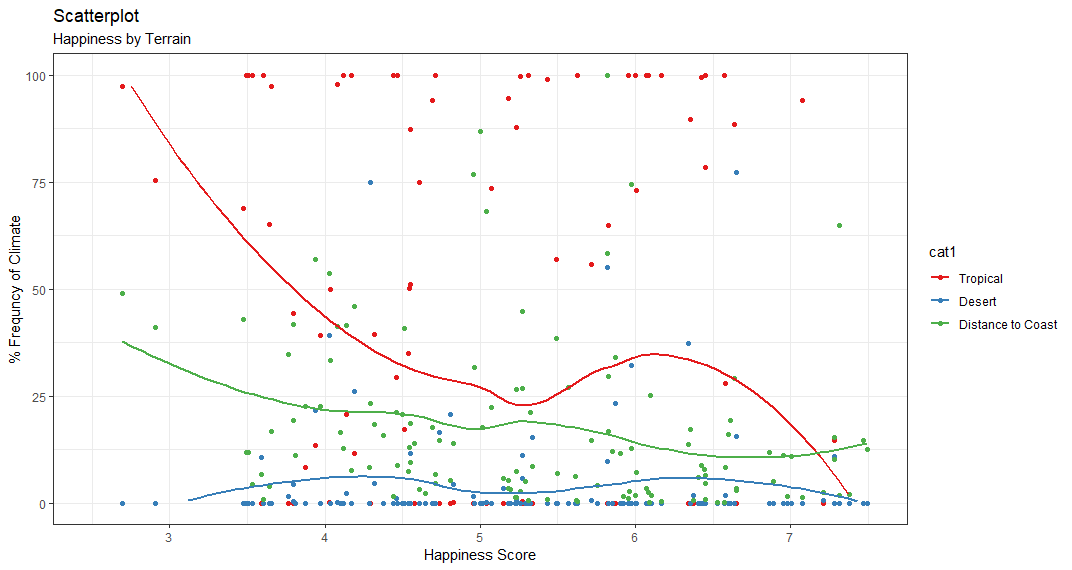
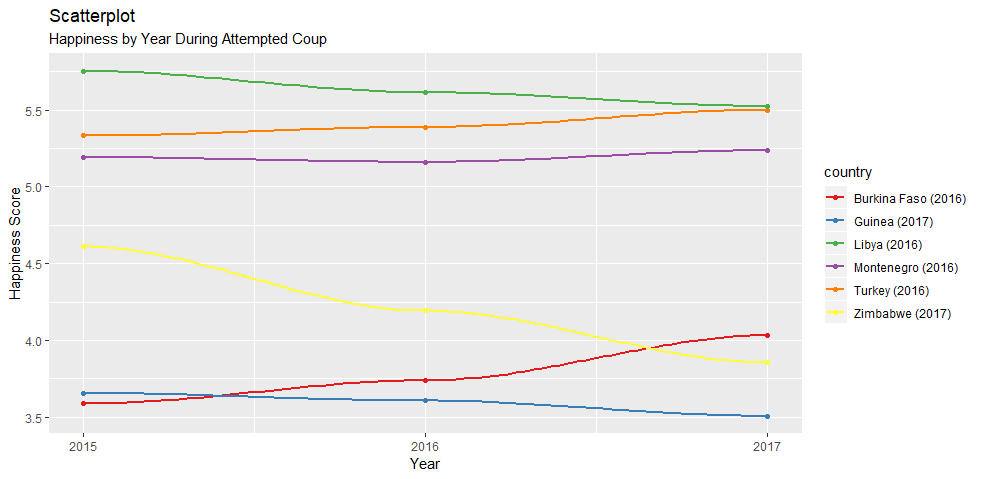
Financial analytics combines statistics, probability, operations research, data science, and computational technologies to analyze various types of financial data sets, and to make meaningful decisions based on analyses obtained from the data. The topics covered various areas in the financial industry, ranging from analyzing transactional data (credit card receivables) to studying global relations between macroeconomic events to managing risk and return in multi-asset portfolios, and the calculation of risk-based capital. Analyses deployed a wide range of techniques including quantile estimation, portfolio analytics, risk measurement, extreme value analysis, forecasting and predictive techniques, and financial modeling. We used R in the class to perform the data analytics.  
  
We analyzed heating oil price (HO2) data. After cleaning the data, we discovered the volatility of HO2 returns by visualizing changes calculated with the Euler method. We plotted the magnitude of changes, as shown in figure 1. We performed descriptive statistics on the returns, to include skewness and kurtosis of the data. we calculated a tolerance level of risk using the cumulative relative frequency function to show the majority of the H02 market is within the tolerable limit for the company. To simulate future movements in HO2 returns, we chose the Hessian distribution to use for fitting in the MASS package.  
  
We performed Macro Financial Analysis in a foreign exchange markets dataset. After cleaning the data, we observed the currency appreciation and depreciation by plotting the exchange rates percent changes. Along with description statistics, autocorrelations and partial autocorrelations were produced on returns and sizes as a timeseries to discover if the returns are dependent on the past. Next, we visualized the shape of our exposure to euros by plotting the cumulative relative frequency, with an intercept at the given tolerance level. We performed a cross-correlation with time to visualize the correlations over a specified lag period. To determine if we needed to be concerned with inter-market transactions, we determined how related the correlations and volatilities are with a fit-test. We used the log()-log() transformation to interpret the regression coefficients as elasticities and then plotted the results. Figure 2 shows the regression analysis of CNY-JPY exchange risk relationships, with x-axis blue lines at the 5% and 95% quantiles. The density is shown with red lines.  
  
We applied market risk analysis on a metals market data to help a fictitious freight-forwarder optimize their portfolio. We looked at the tails of the returns distributions and defined risk measures. We graphed the distribution of returns on nickel, copper, and aluminum, respectively, with added expected shortfall and value at risk metrics. We viewed the expected shortfall (ES) and value at risk (VaR) estimations based on a Generalized Pareto Distribution. We then plotted this distribution with corresponding confidence intervals, as shown in figure 3, to determine we were comfortable with the amount of capital the company has allocated to the venture.  
  
We performed portfolio analytics on the metals market data by calculating the optimal metal risk-return allocation for the freight-forwarder. Using the given tolerance risk, we graphed a tangency portfolio to find the optimal combination of risky assets. We built an efficiency frontier, in which we observed that being too conservative with less risky metals were not profitable. We produced a Sharpe ratio to evaluate our portfolio assets on the efficient frontier. We then performed a tangency portfolio sampled mean simulation to evaluate the Sharpe ratio with 1000 simulations.  
  
For a final project, my partner and I worked to increase Gateway First Bank in Tulsa, OK’s correspondent loan volume in the near future. The major problem beginning this initiative was a general lack of awareness and analytics surrounding the correspondent loan process at Gateway. It had been calculated that Gateway won about 2% of the loans it has bid on the previous year, while there did not appear to be a notable difference in the few loans won and the thousands of losses. In addition to the loan and customer information, we developed a basic loan scoring system to ensure that Gateway’s model is not prone to win the riskiest loans. We next trained a machine learning algorithm to act as the “new pricing model,” ideally recommending bids such that Gateway wins high quality loans at slim margins and bids comparatively in the basement for low quality loans.  
  
For the project, we used a directory of historical correspondent bid tapes that Gateway participated in. The initial problem with this dataset was that an individual bid tape may contain 2-50 mortgages, and no two banks’ bid tapes are alike. We needed to script a very robust (and tedious) R program that would read through the directory and append a master dataset, matching columns that describe the same data, but are all named differently. Figure 4 shows a subset of the raw data after being joined by the first script.   
  
Next, we developed a basic loan scoring function inspired by FNMA’s loan-level price adjustment matrix (available online). A column was created to help make the data narrative more succinct. We performed descriptive statistics on the data, comparing the range, mean, median, IQR, standard deviation, kurtosis, and skewness for the wins and losses. We decided to map out percent differences to visualize with bar plots in an effort to understand the changes between won and lost mortgages. At this point, we noticed that the loans won are usually slightly better than the lost ones, but the issue of having so few wins and no immediately distinct difference in the quality between won and loss loans remains. The differences in key risk factors were not significant enough to have any level of confidence in the legacy pricing model. We also attempted to discover if the current model made significant price adjustments based on the state of the property address, we created bar plots comparing wins and losses by state.  
  
We let the k-means algorithm take a shot at separating the loans purely based on a loans characteristics and without any historical bidding information, thinking it would be nice to see the two clusters being separated on the varying risk levels. By visually comparing the descriptive statistics on the two clusters, we observed the results were mediocre, separating somewhat based on loan quality. It’s notable at least that a simple, computationally inexpensive algorithm like k-means could probably perform better than our current pricing model though.  
  
We knew that the score column would be linearly dependent on some of the bigger risk factors, e.g. DTI ratio. We needed to visualize the predictor-wise correlation to determine if we should choose an algorithm that is more robust to collinearity. This correlation matrix showed us that there wasn’t as much collinearity as we suspected, but we still decided to look into robust algorithms. We plotted a correlation matrix using the tidyverse package in Figure 5.  
  
Next, we implemented a supervised machine learning model, using a random forest algorithm to predict prices on a test set, check the difference in the mean score between W/L, and finally produced the feature importance. Since the prices were not strongly correlated with loan quality (i.e. risk level), we decided to alter the winning bid depending on the loan score. The model was trained on the pattern that good loans are won at 25 basis points higher than the actual winning bid, and very bad loans are won at 2.5% lower than actual value. This produced favorable results that were able to recommend an increased win percentage of 39%, up from originally 2%. We discovered score was just important enough to give the model an edge on winning good loans by slim margins and losing bad loans almost every time because we bid relatively low on those. We detected noticeable changes from win to loss in several of the metrics in the summary statistics.  
  
We generated a tree visualization of the model, shown in figure 6, to help us understand the inner workings of the model. The tree would also be important when presenting to management and colleagues without statistical backgrounds. Great effort should be put into making an algorithm seem as far away from a black box as possible if for buy-in.   
  
  
**Figure 1**

**  
Figure 2  
  
  
Figure 3  
  
  
Figure 4  
  
  
Figure 5  
  
  
**

**Figure 6**  
**IST 707 – Data Analytics**The Data Analytics course provided an introduction to data mining techniques, familiarity with particular real-world applications, challenges involved in these applications, and future directions of the field. We dove into the various approaches and algorithms of machine learning and received hands-on experience with R and open-source software packages.   
We learned about classification vs clustering. We discussed data types and data quality issues, such as outliers, missing values, duplicate data. We worked in R perform data exploration and transformation techniques on student data that measures student lesson status, i.e. very behind, middling, very ahead. By subsetting the data, and converting the status to a numeric value, I was able to compare student status section, as shown in figure 1. A regression model was created to test if the quantity of students is correlated to section, but did not appear as correlated due to three section outliers. After these sections were removed from the model, the R-squared value increases from 0.14 to 0.67. It appears these sections were causing students to fall behind.  
  
We talked about business application for association rule (AR) mining, and metrics to evaluate its strength. We applied the apriori method in R to banking data to determine customers who will want to obtain a new Personal Equity Plan. We identified rules with high lift and confidence which at the same time have relatively good support, and made recommendations to the bank, based on the discovered rules.  
  
We learned about partitional and hierarchical clustering analysis, and the various methods used today such as k-means. We learned to evaluate the results with some criterion, e.g., minimizing the sum of square errors. We learned about Euclidean and Manhattan distance measures and the cosine similarity. We learned how to prepare the data to be clustered. We used clustering methods to solve a mystery in history, who wrote the anonymous essays in the Federalist Papers. We used both the Hierarchical Agglomerative Clustering (HAC) and K-means to separate the essays by author. Figure 2 shows the plot of the clusters generated from the K-means model, where the numbers are indicative of the authors.  
  
We learned to classify using decision trees, specifically the C4.5 algorithm. We learned about the various splitting arguments. We learned to measure how well a given attribute separates the training examples with information gain and gain ratio, and to measure impurity with entropy. We learned about the dangers of overfitting, and avoid it by pruning. As an assignment, I used the J48 algorithm in R on the Federalist Papers dataset. The model was tested as both a pruned and unpruned tree, changing the minimum number of instances and pruning confidence for the best fit model. I also created an rpart tree that uses the CART algorithm.  
  
We learned overfitting can be caused by noise and insufficient samples in the data and to reduce decision tree model complexity if possible. We learned to evaluate models based on hold-out and cross-validation tests. We learned to detect errors with a confusion matrix and measure precision, recall, and F-measure. We learned that having not enough data can affect prediction accuracy, so semi-supervised learning, active learning, or crowdsourcing can be implemented to account for this. We learned to evaluate reliability of human annotation with Kappa.   
  
We used the Naïve Bayes (NB) classifier and about the Bayes’ theorem of calculating probabilities based on posterior observations. As an assignment, I compared decision tree and NB models from a handwriting dataset. The goal was to recognize digits 0 to 9 in handwritten images. A three-fold evaluation was used to test the accuracy of each model and produce a confusion matrix. Additionally, the time was captured before and after each model was produced. The NB algorithm produced the best fit model, due to both speed and accuracy. Due to the complexity of the tree, overfitting seemed to occur.  
  
We learned about the K-Nearest Neighbor (K-NN) algorithm, and it, unlike decision trees, can be used when the decision function to be learned is very complex. Since it does not use a linear boundary, it differs from NB in that its boundary has no defined shape. The disadvantages is that is sensitive to noisy data, and computationally expensive. We learned the SVM is an algorithm that can solve both linearly separable and inseparable problems with kernel functions. We learned about random forests and why ensemble works.   
  
As an assignment, we used SVMs, k-NN, and Random Forest algorithms in R to analyze the handwritten digits dataset. To find the most accurate model, different values of K, or the number of instances taken into account for determination of affinity with the class were specified for the k-NN models. Different cost values, or the parameter for the soft margin, were entered in the SVM models, and set to use the polynomial kernel. Different values for ntree, or number of trees, were entered in the Random Forest model.   
  
For the final project, my teammate and I analyzed world happiness data from the Happiness Report, originating from the Gallup World Poll. The report ranks 155 countries by their happiness levels and was released in the United Nations. It spans 10 regions and shows that governments, organizations and civil societies increasingly use happiness indicators as resources of information for business utilization and policy-making decisions.   
  
In addition to each country’s happiness rating, the country’s rating in economy, family, health, freedom, trust, and generosity were also included in the dataset. Through descriptive statistics and visualizations such as bar chart and box plots, we were able to visualize the distribution of happiness. We also created a geographic map, shading the countries based on happiness score, as shown in figure 3. We observed happiness scores gathered in 2015 by country, with Switzerland having the highest score of 7.587 and Togo having the lowest score of 2.839.   
  
The information gain ratio table, shown in figure 4, shows the factors of health, economy, freedom, family, and trust all affect the happiness score, with health having the largest weight. The generosity factor has no weight on the happiness score. A correlation heat map, shown in figure 5, revealed the strongest correlation in the dataset is between happiness and health. The other significant variables are economy and family. Freedom, Generosity and Trust all had correlations less than 60%- which for this analysis was deemed insignificant. Although there is a strong positive correlation with the happiness score, there are exceptions. There are some Asian countries in which the life expectancy (health) is long, but the happiness scores are lower. That signifies underlying variables effecting the relationship between happiness and health. These outliers, however, are beyond the scope of this analysis.  
  
In order to predict the happiness score from the factors deemed significant, models were run on the data, with the happiness rating discretized into five categories “low,” “medium-low,” “medium,” “medium-high,” and “high.” Then, the predictor variables were weighted in relation to the lowest and maximum value for the factor for all countries. The weight for these variables was in percent and remained unfactored. All other variables were removed. Seventy percent of the discretized dataset was randomly broken down into a training set for the supervised models, while the remaining thirty percent was reserved for the training set. The classification models utilized in this particular analysis include: J48 Decision Tree, SVM, Naïve Bayes (NB), k-nearest neighbor (k-NN), Random Forest, and clustering (HAC).

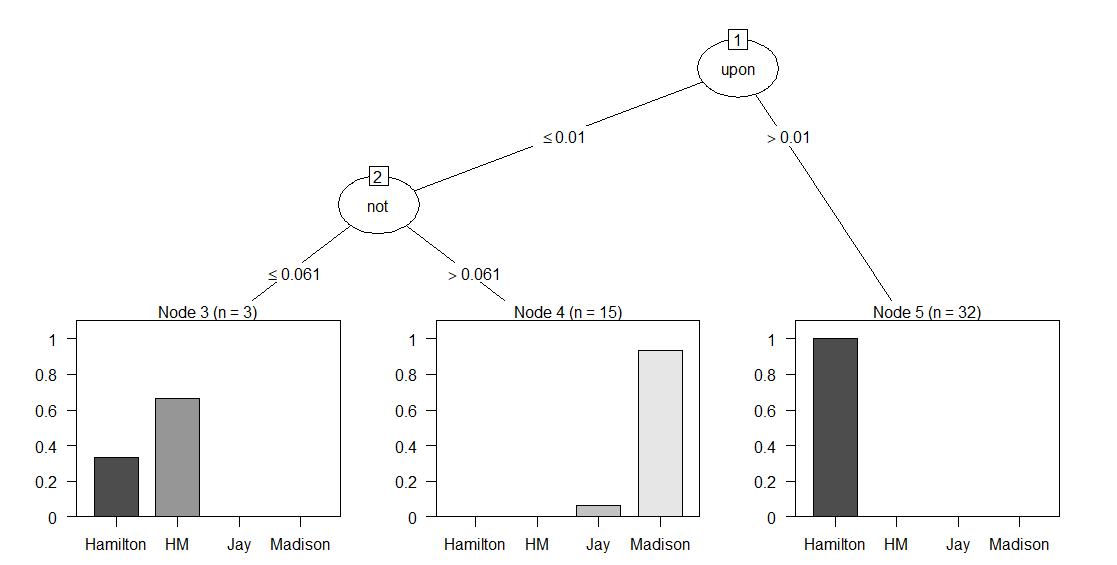
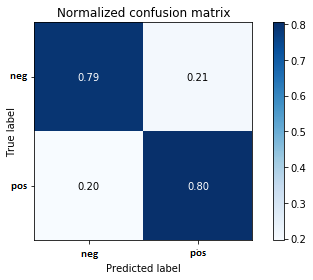
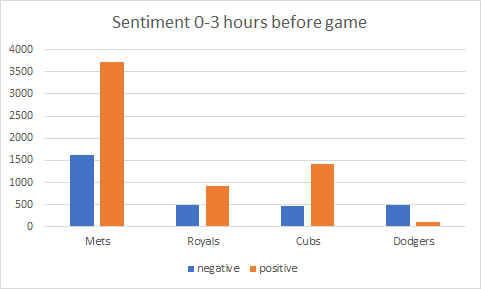
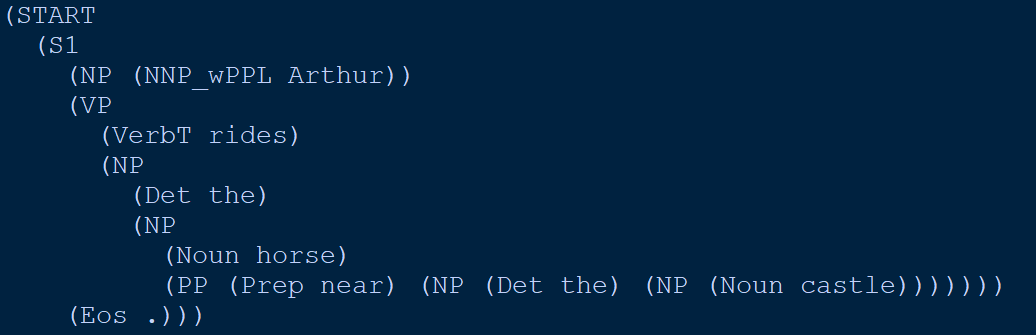
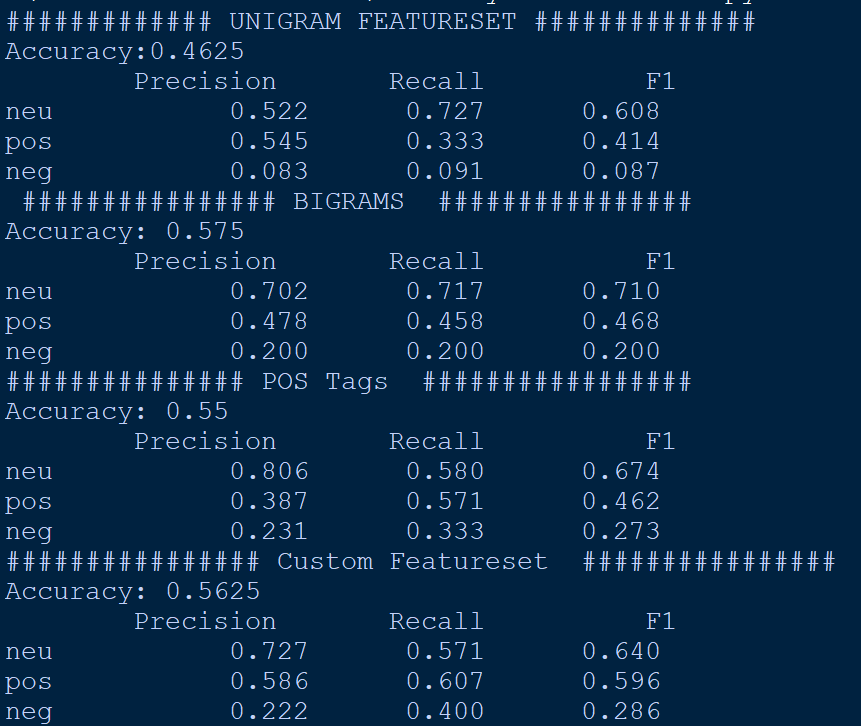
J48- Decision Tree Analysis was conducted next. All the variables were used in this model. Decision Trees are a type of classification technique. A series of questions are applied to the attributes of the data and the results are broken down into smaller and smaller groups. The series of questions and the possible answers are broken out which results in a tiered structure that resembles a tree. Often times, the algorithms used in J48 need to be optimized to display the most relevant links in a given dataset. The best yielding decision tree specified a minimum number of nodes, or “M” parameter, of 16.

The NB Classifier is based on the Bayes Theorem. This prediction model estimates the probability by assuming the attributes in a given dataset are independent from one another. This means that with the World Happiness Dataset, each variable was assumed to be independent of each other during NB analysis. An NB model was applied to the dataset, with a laplace of zero. The kNN algorithm is a lazy learning classification method, where the distance between the nearest neighbor(s) of a given variable is found and grouped together. It is assumed that the variables grouped together by this algorithm all share similar characteristics. The kNN model was applied to the dataset, with a k value of 19.   
  
A random forest is an ensemble learning method for classification that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy. A random forest model was run on the dataset, with an nTree parameter value of 55. The next model used was Hierarchical Clustering. Hierarchical clustering is an unsupervised classification method which seeks to build a hierarchy of clusters. Complete, average, and minimum hierarchical clustering models were run on the dataset. Finally, an SVM model was used to create algorithms to classify variables by finding the hyperplane that maximizes the edge between any two variables. The SVM model was applied to the dataset, using a polynomial kernel and cost parameter of four.  
  
The model with the most accurate results was performed through the SVM algorithm, predicting happiness scores with an accuracy of 76.6%. This model was run 1000 times, randomizing the training/testing data each time. The countries incorrectly were plotted, revealing most of the countries in South America were predicted incorrectly.   
  
In search of other influential factors on happiness, terrain data was extracted from the source: 'Ruggedness: The blessing of bad geography in Africa', published in the Review of Economics and Statistics​. Figure 6 shows that tropical and desert terrain both have a negative correlation with happiness. We attempted to graph the happiness levels of countries where a coup d'états had occurred during the years covered in the dataset. According to the line graph in figure 7, Zimbabwe, Libya, and Guinea’s happiness score steadily declined up until the year of the coup. Turkey and Burkina Faso, however, seemed to increase in happiness the year before the coup occurred. Overall, we were able to conclude regions that were healthier and wealthier, such as Australia/New Zealand and North America, hold the highest happiness scores, while regions such as Sub-Sahara Africa, who have low health and economy ratings, rank the lowest. My teammate and I presented our findings to the class.

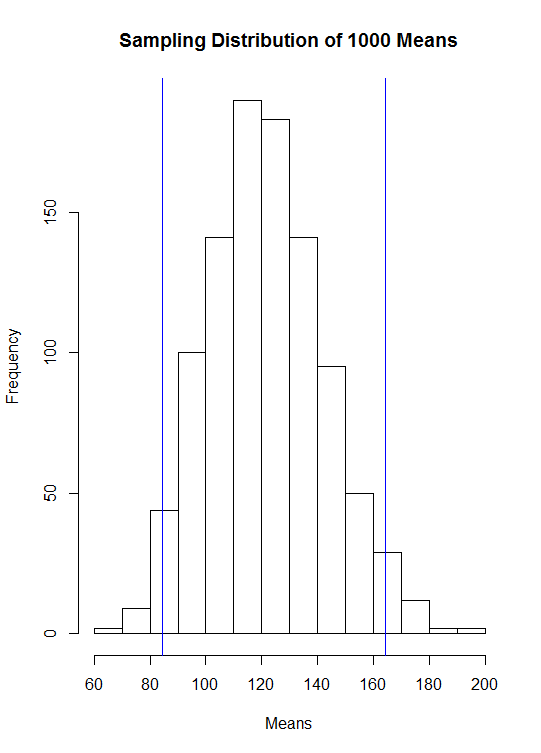
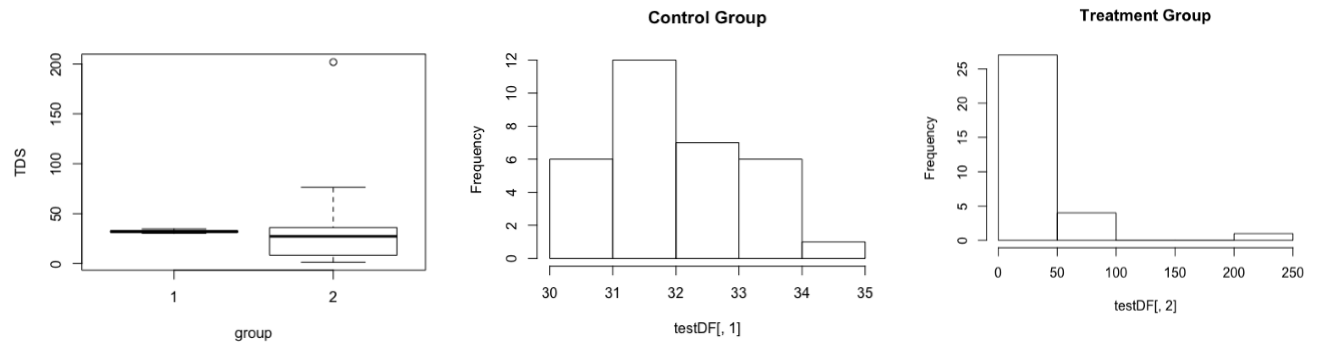
  
**Figure 1** **Figure 2** **Figure 3** **Figure 4** **Figure 5** **Figure 6** **Figure 7**  
  
  
**IST 687 – Text Mining**The main goal of this course was to increase awareness of the power of large amounts of text data and computational methods to find patterns in large text corpora. This course was designed as a general introductory level course for all students who are interested in text mining. This course introduced the concepts and methods of text mining technologies rooted from machine learning, natural language processing, and statistics. This course also showcased the applications of text mining technologies in information organization and access, business intelligence, social behavior analysis, and digital humanities.  
  
We looked at the vectorization process, regarding both what to count and how to count that would be best for the goal. This includes the stemming, and whether we want to eliminate important linguistic information. For an assignment, I vectorized tweet data for sentiment value. Stemming did not provide the best results, but the elimination of stop- words from the NLTK-English library did yield better results.   
  
We looked further into vectorizing options, for an assignment analyzing the writings of the Federalist Papers to discover who authored the anonymous essays. In order to vectorize the documents, line breaks and punctuation were removed and all words were converted to lowercase. A J48 decision tree model, using a pruning confidence of .5 and minimum number of instances of two performed the best, achieving 83.3% accuracy of the testing data. Disabling pruning and changing the pruning confidence did not affect the accuracy. The J48 model plotted in figure 1. There are two words “upon” and “not” that were calculated as most influential.  
  
We looked into the application of the Multinomial Naïve Bayes (MNB) machine learning algorithm for text mining. For an assignment, I applied NLTK’s NB model to customer reviews for sentiment analysis and for lie detection. I used a 10-fold test for evaluation. While stemming did not provide better results, the filtering of stop-words did. An additional step of allowing just POS adjectives words as features did not yield better results. An information gain test showed four words that most influence sentiment value in the dataset are “amazing,” “terrible,” “took,” and “best.” We used the Amazon Mechanical Turk website to collect manual sentiment classifications on restaurant reviews. To determine each worker’s quality of work, Kappa values were calculated and compared to the ground truth.   
  
We compared the Benoulli (BNB) and Multinomial Naïve Bayes algorithm. In an assignment, I used both to perform sentiment and lie detection analysis on customer reviews. I used the words were tokenized as binary for BNB and both normal and term frequency-inverse document frequency (TF-IDF) for MNB. Unigrams and bigrams were tested under each algorithm, along with stop-words, minimum frequency, and conversion to lowercase.   
  
We used evaluation methods to evaluate machine learning prediction accuracies. In an assignment, I applied the MNB and SVM algorithms, with different options, to movie reviews and compared models in a confusion matrix and calculated precision and recall. Word gain was also used to report the most informative words.   
  
We learned about topic modeling and used Latent Dirichlet Allocation (LDA) to summarize the main topics of the floor debate of the 110th Congress. I used MALLET with different topic sizes and compared the relevancy of the results until an optimal size was achieved. After reducing the topic size to 45, since there usually are about 40-50 topics per floor, and removing stop-words, the best results were achieved.   
  
As a final team project, we worked to perform sentiment analysis on social media before a game to see if there is a relationship to the outcome of the game. To observe the social media sentiment prior to a game, four MLB teams were chosen: New York Mets, Kansas City Royals, Chicago Cubs, and Los Angeles Dodgers. The Twitter API was used to export tweets from related to these teams from August 17th to 31st, 2019.   
  
In order to capture tweets related to these teams, the Twitter API request query specified one of the following tags exist for the tweets: #mets , #nymets, #dodgers, #ladodgers, #cubs, #chicubs, #royals, or #kcroyals. Python code was used to make the Twitter request through the Tweepy library, and one CSV file per day was exported with the tweets for that day. The exported CSVs contain approximately 4000 tweets, or rows, per day. The tweets were cleaned of stop-words, emoticons, and punctuations.   
  
In order to perform sentiment analysis on the exported MLB team tweets, a subset of 50,000 positive, and 50,000 negative tweets from Kaggle (<https://www.kaggle.com/kazanova/sentiment140>) were used to train the computer. A custom set of stop-words were utilized, prior to vectorization.   
  
To perform sentiment analysis on the tweets, supervised machine learning was applied to the labeled tweet dataset. Both MNB and SVM algorithms were used to predict sentiment value on the labeled data. The tweets were vectorized as word count, binary, and TF-IDF to compare models. In addition to frequency, the minimum count needed to be recorded as a vector was configured to find the optimal value. Tokenization with and without all lowercase letters, and the collection of unigrams, bigrams, and both unigrams and bigrams together were collected.  
  
Once an instance of the vectorizer class was created, the fit() function was called in order to learn a vocabulary from the documents. Next, the transform() function was applied to the documents to encode each as a vector. An encoded vector was returned with a length of the entire vocabulary and an integer count for the number of times each word appeared in the document.

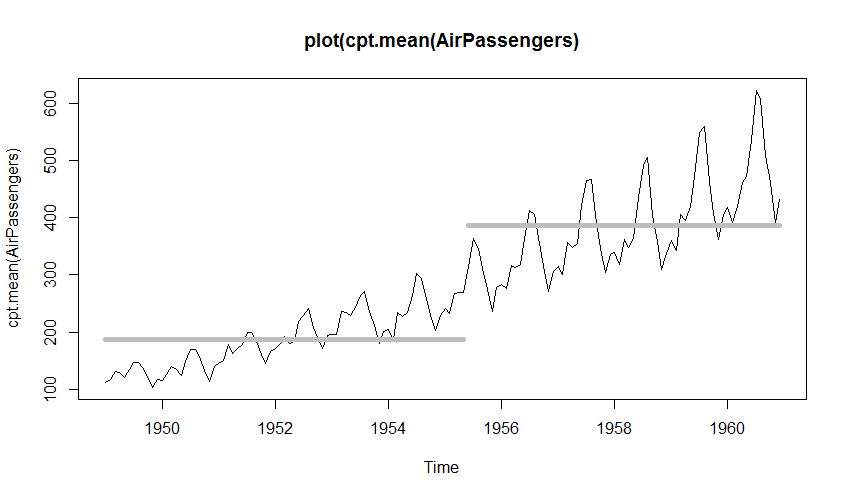
After the vectorization process was complete, the MNB and SVM algorithms were applied to the feature sets. While only the default parameters were accepted with the MNB algorithm, the SVM algorithm cost, or c parameter, was set from 1 to 10 to test model accuracy. To test the best model, the 100,000 rows of labeled data was divided into 67% test and 33% training sets. Once the best model was discovered, the entire labeled set was applied to the model for training, and used to predict sentiment on the unlabeled MLB tweet data.  
  
The SVM model with unigrams and bigrams performed best when for sentiment analysis on the MLB tweet data. The filtering of hashtags decreased accuracy on both MNB models and SVM models. The filtering of numbers helped increase the MNB model accuracy, but decreased the SVM model accuracy. Additionally, count frequency performed best on both algorithms. Table 1 shows the change in accuracy from the original baseline with the inclusion of unigrams only. The SVM model performed best with the cost value of one. As shown in figure two, the accuracy decreased as the cost value increased. The best model with SVM and bigram inclusion was used to detect sentiment analysis on the MLB tweet data. According to the confusion matrix shown in figure 2, the model predicts positive and negative sentiment values with approximately equal accuracy.  
  
Once sentiment values were predicted for the MLB team data, the tweets were split by team and by day. Figure 3 shows the sentiment distribution among the teams, once sentiment analysis was performed on the MLB tweet data, and the data was filtered by team, date, and by the last three hours before a game. The Mets had the most tweets attributed to a team. While the Mets, Royals, and Cubs had more positive than negative values, the Dodgers had more negative than positive values. When correlating wins and losses to team tweet sentiment value, no correlation was discovered. The mean accuracy of win to positive value and loss to negative value was 48%, which does not indicate a positive or negative correlation. When separated by team, the Mets mean accuracy was 53.8%, Royals was 28.6%, Cubs was 69.2%, and Dodgers was 42.9%. Since such as small subset of games and tweets were correlated, it may be worthwhile to look into teams individually, such as the Royals and Cubs who had means much higher and lower than 50%.

To analyze the tweets further, topic Modeling was implemented against the tweet data. Specifically, the data was vectorized both as unigram and bigrams. The data was then used against the LDA and Non-Negative Matrix Factorization (NMF) algorithms. The two different approaches were utilized to provide comparative results of a probabilistic approach (LDA) against a deterministic algorithm (NMF). The topics were then cross referenced with real world events. We were able to cross reference these events to a sentimental value. My teammate and I presented our findings to the class.

  
**Figure 1**  
  
**Figure 2** **Figure 3**  
  
**IST 664 – Natural Language Processing**In this course, we developed an understanding of how natural language processing (NLP) can process written text and produce a linguistic analysis that can be used in other applications. The course primarily covered the techniques of NLP in the levels of linguistic analysis, going through tokenization, word­level semantics, part­of­speech tagging, syntax, semantics, and on up to the discourse level. It also included the use of the NLP techniques, such as information retrieval, question answering, sentiment analysis, summarization, and dialogue systems, in applications. We learned to process text through the language levels using the resources of the Natural Language Toolkit (NLTK) and some rudimentary use of the programming language Python.  
  
We learned about the levels of language analysis. We used NLTK Frequency Distributions to Count Unigrams, Bigrams, and calculate a mutual information score. We learned to filter our results by filtering stop-words and applying minimum frequency conditions. We learned the importance of tokenization, and the removal of punctuation. For an assignment, I analyzed two resumes to compare work experience. For the tokenization, words were converted to lowercase, stripped of default stop words and numbers, and lemmatized. Ranked by Mutual Information Score, the top unigrams and bigrams were observed, indicating subjects of expertise for each candidate.  
  
We learned to use regular expression with filtering and tokenization. We used word-stemming to match words that have the same base. For an assignment, I identified obfuscated email addresses and phone numbers from html documents with regular expressions. For example, email address “ada&#x40;graphics.stanford.edu” was identified with the regular expression  
“([A-Za-z]+)&#x40;([A-Za-z.]+)\.(edu)”.  
  
We were introduced to part-of-speech (POS) tagging and named-entity recognition**.** We learned to use context-free grammars (CFG) and how to add CFG rules to the NLTK PCFG parser. For an assignment, we wrote grammar rules for part-of-speech tagging specific to a corpus. Figure 1 shows the rules added for the sentence, “Arthur rides the horse near the castle.” The noun phrase, “Arthur” is followed by the verb phrase “rides the horse near the castle.” The The verb phrase consists of the prepositional phrase “near the castle.” The word “rides” is a third person singular present verb (VerbT).  
  
For the final project, I performed sentiment analysis on a sample Twitter tweet set. Before the tweets were imported from the external file “downloaded-tweeti-b-dist,” the tweet was stripped of the ending punctuation, and regular expressions are used to strip the text of numbers and social media tags that start with a hashtag or “@” sign. The NLTK tweet tokenizer was used to tokenize each tweet.   
  
I imported an external stop-word list, “stopwords\_twitter,” provided in the course materials, and checked each tokenized word against the stop-word list before adding to a list. I then added a white list that replaced the stop-word list. The white list was created by combining positive and negative sentiment words obtained from Github. Both whitelists were imported from the external files “pos\_words.txt” and “neg\_words.txt.” The token must be contained in the white list in order to remain. The words from Github were obtained here:   
Negative words: <https://gist.github.com/mkulakowski2/4289441>  
Positive words: <https://gist.github.com/mkulakowski2/4289437>  
  
I added an additional condition before the token is added, to ensure the character length is larger than two. The tokens, that passed the requirements, are added to the list “filtered\_tokens” in a stemmed format, using the PorterStemmer from NLTK. Separately from the tweet tokens, the tweet is the analyzed for the sentiment value. The value can by of several values, but is simplified into three “pos,” “neg,” and “neu.” The “filtered\_tokens” list and simplified sentiment value is then appended to another list, “tweetdocs.”  
  
In order to make predictions on the tweets, a total of four definitions were added to return a featureset. Since all definitions require a list of the most common tokens or bigrams found within the entire tweetset, the NLTK “FreqDist” function was used, and then “most\_common” to get a list of the top 1500 words used. The “BigramCollocationFinder” was used to find the most common bigrams. The top 500 were used, calculated using Chi Squared.  
  
I created several definitions to test the most efficient for sentiment analysis. The first definition, was obtained from the recorded sessions and returns a featureset consisting of a list of all the common words, and a boolean value for each. The boolean value is determined if the common word is found within the tweet tokens. Both the tweet tokens and list of common words are sent as a parameter.  
  
The second definition was also obtained from the recorded sessions. It returns a featureset like the first definition, except with the most common bigrams. The third definition, like the other two, was obtained from the recorded sessions. It returns a featureset containing a count of the nouns, verbs, adjectives, and adverbs in the tokenized tweet. I created the last definition to return a featureset like the “document\_feautures,” but also counts the amount of words in the positive and negative word lists, imported earlier. Two features were added to the returned featureset, one containing a boolean value if more positive words were found, and the other if more negative words were found.  
  
Once the tweets have been filtered and tokenized, and the featuresets are created, the featuresets can be run through a Naïve Bayes model in order to predict the label, or the sentiment. A preliminary model was created for each featureset from the four featureset definition generators. To train and test each model, the featureset was randomized and divided into 80% for training and 20% for testing. With no stop-words applied, the unigram featureset   
  
To achieve more predictable results, a 10-fold test definition was created, called “cross\_validation\_accuracy.” In the 10-fold test, the featureset is randomly divided into ten portions, with each being the test set. The accuracy of each test is averaged to provide a more predictable accuracy percentage.  
  
At first, the four featuresets were tested with the filters and sizes specified above, along with the stopword list. Figure one shows the output of accuracy, precision, recall, and F-measure score of each featureset. Precision is the fraction of test values that appear in the reference set. Recall is fraction of reference values that appear in the test set. F-measure is the harmonic mean of the precision and recall, weighted by alpha. It also appears negative precision may be the most difficult to predict since it has the lowest precision, recall, and F-measure. It appears neutral precision may be the easiest to predict since it has the highest overall precision, recall, and F-measure. According to the accuracy, the custom featureset predicted sentiment most accurately. During the experiments, the accuracy was measure on the 10-fold test for a more reliable accuracy.  
  
  
**Figure 1** **Figure 2**  
  
  
**IST 772 – Quantitative Reasoning in Data Science**Statistical inference is the process by which we make sense of uncertainty in data, and this course focused on stabling a thorough understanding of statistical inference. The course helped us understand contemporary methods of statistical inference regardless of the specific type of analysis being applied. The primary goal of this course was to learn techniques and concepts that facilitate drawing sensible conclusions from samples of quantitative data.   
  
We learned to make sensible choices about how a data collection and the applicable analysis processes relate to the kinds of inferences that can be drawn. We developed the skill of planning data collection and measurement to facilitate appropriate analysis. We practiced effective data science analytics by preparing data for analysis, including screening data, dealing with missing data, and performing data transformations. We tested assumptions that data must meet for analyses and inferences to be reasonable and interpreted the results. We then learned to communicate them to others using language that accurately describes the uncertainty. We learned to leave a documentation/provenance trail for other analysts to follow and reproduce the work. We demonstrated competence of the skills needed with R, to conduct sound and reproducible analyses  
  
We started the course learning about descriptive statistics, to include the mean, median, mode. We learned measures of dispersion, such as range, deviations from the mean, sum of squares, variance, and standard deviation. We learned about distributions, such as normal, Poisson, and binomial, to generate a randomized distribution in R and view as a histogram. We produced outcome tables to list a linked set of linked set of outcomes, and produced contingency tables to calculate probability based on a posterior set of probabilities.   
  
We learned about sampling of a population and learning to generate random sampling with and without replacement in R. We learned to use the replicate() R function to make replications. We learned the central limit theorem and the law of large numbers to describe a distribution. We learned to output quantiles and use the summary() function for a summary of a dataset’s quartile, mean, and range. In an assignment shown in figure 1, we constructed a sampling distribution of 1000 means of chick weights using the built-in R ChickWeight dataset. We displayed a histogram and added vertical lines at the 2.5% and 97.5% quantiles.

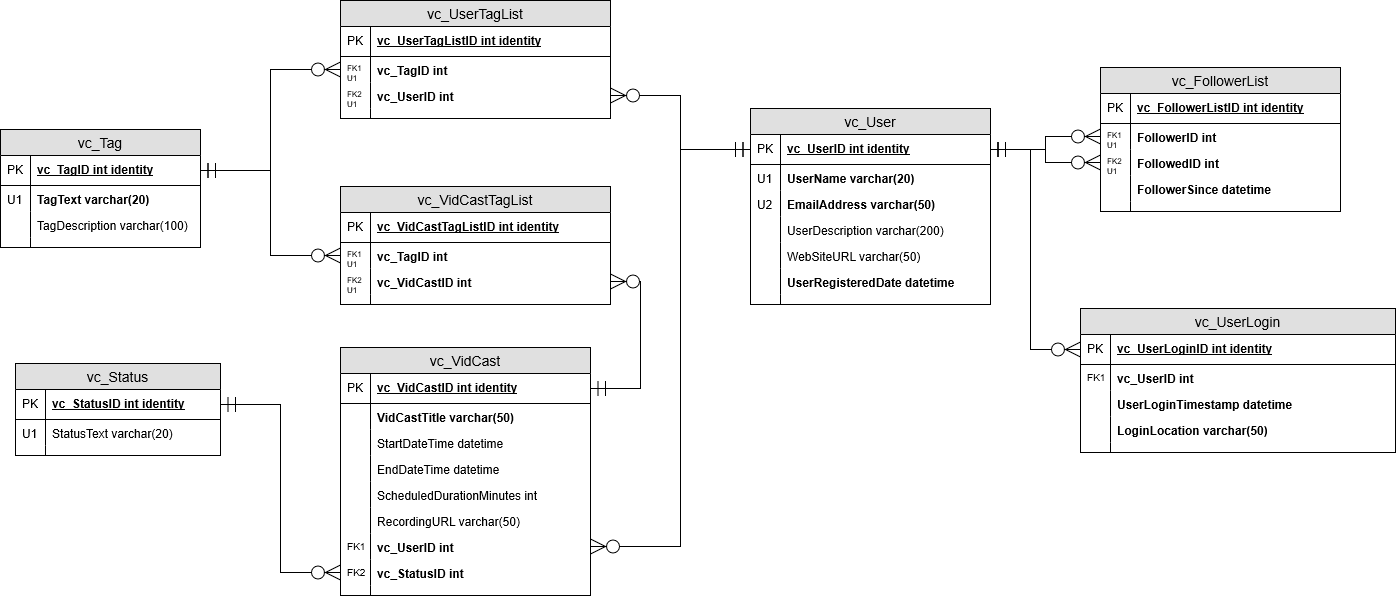
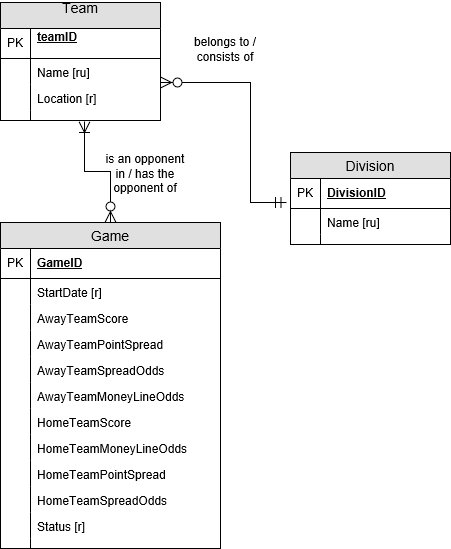
We learned the importance of inference, in that we cannot prove anything from samples about the population, but instead make inferences. We learned about independent (predictors) vs dependent variables. We built boxplots to compare distributions. We explored the variability of sample means with repetitive sampling. We learned about significance testing confidence testing to obtain a confidence interval that can strengthen our statistical evidence that there is a difference in population mean. We ran a t-test to compare the means of a control group and treatment group from the PlantGrowth dataset in R. From a p-value < 0.05, I concluded there was evidence to reject the null hypothesis that the population mean growth between the two groups was equal. The 95% confidence interval suggested that the population mean difference lies somewhere between -0.29 and 1.02.  
  
We were introduced to the Bayesian theorem and the likelihood of an occurrence given a posterior occurrence. We learned to run Markov chain Monte Carlo (MCMC) simulations in R to obtain a common 95% highest density interval (HDI). We learned this represents 95% of the likely values of the population mean difference lie between the HDI. We learned to use this, in conjunction with the convention significance testing as supporting evidence. We learned to observe the HDI overlap of zero as supporting evidence to fail to reject the null hypothesis from a significance test.  
  
As an assignment, I analyzed a built-in R dataset to compare “total dissolved solids” in water between a control and treatment group. The water in the treatment group was applied a biofilm to remove dissolved solids. Conventional significance testing, confidence interval, and MCMC simulations were run to test the effectiveness in removing solids. Figure 2 shows the descriptive statistics performed comparing the distributions. The excerpt below is the technical report detailing information that would be important for other statisticians to know about the data.  
  
*A sample of water was collected, measuring total dissolved solids, or TDS, after a biofilm was applied. The effectiveness of the biofilm was tested by performing three statistical tests on sample observations to determine whether there is a difference between the control group and the treatment group. In the first test, a null hypothesis significance test with an alpha threshold of .05, we fail to reject the null hypothesis and conclude that not enough evidence is available to suggest a difference in the mean TDS between the treatment and control group. A 95% confidence interval test was then performed and confirms our previous lack of evidence.   
  
The third test, performed with a Bayesian BEST method, shows that the HDI of the posterior distribution of mean differences overlaps with zero, providing evidence suggesting support for an alternative hypothesis of credible differences between the control and treatment groups. In fact, according to the Bayesian method, the treatment group is approximately 8.2PPM lower than the control group. Even though this result contradicts with the previous frequentist tests, it suggests, as a whole, that biofilm should be looked into further as a promising alternative to traditional filtering techniques*We learned about using analysis of variance (ANOVA) to examine combinations of different factors. We learned about the difference of between-groups and within-groups to evaluate whether samples might have come from the same underlying population. We learned the reasoning behind degrees-of-freedom (df) and interpret the F-test significance from the ANOVA output. We can support our evidence from ANOVA by using MCMC simulations to show the range of values for the grand mean. We learned to calculate the Bayes Factor to produce meaningful odds from our Bayesian analysis.   
  
We looked at associations between variables and performing a null hypothesis testing on correlation among variables. We learned to support our significance testing with Bayesian testing on the correlation coefficient. We learned to perform categorical association. We learned to calculate the Chi-squared value of contingency tables in R for significance testing and a confidence interval. We learned to support our evidence with MCMC simulations and report a Bayes Factor.  
  
We looked into linear multiple regression and the importance of the sum of squared errors of prediction, or the error that results from our best- fitting line. We learned about the least-squares criterion technique to find the best-fitting line. We tested for multicollinearity to detect predictor variable dependence with one another. We used MCMC simulations to provide Bayesian estimates of R-squared in the 95% HDI. For an assignment, we tested Chilean votes in favor for a president in 1988 for correlation among the age and scale of support for the status-quo factors of the voters. We imported the Chile R dataset from the “car” package and converted the vote to a dichotomous variable. We performed a general linear model with the glm() function. With only one variable, which indicated whether the voter was ok with the status-quo, being significant in the model, we were able to lend support for the alternative hypothesis that the log-odds of statusquo is 0 in the population. We convert the log-odds to regular odds to reveal that the testing suggested that for every additional year of age, a person is about 1.01% more likely to vote “Yes.” We calculated confidence intervals and a Bayesian 95% HDI to further support the evidence from the generalized linear model.   
  
Lastly, we learned how to analyze change over time. We learned about time-series analysis and change point analysis. For an assignment, I was able to perform change point analysis by comparing the change point variance and change point mean on the built-in R time-series dataset, AirPassengers. Figure 3 shows the plot from the change point mean. In the plot, there are two grey lines that represent the mean level of the air passengers during the whole period of time covered by the line. The point in the middle of these two lines represents the change point mean. I was able to pinpoint the passenger frequency increased substantially in 1955. I was able to reference this year to the advancement of jets in commercial airlines.

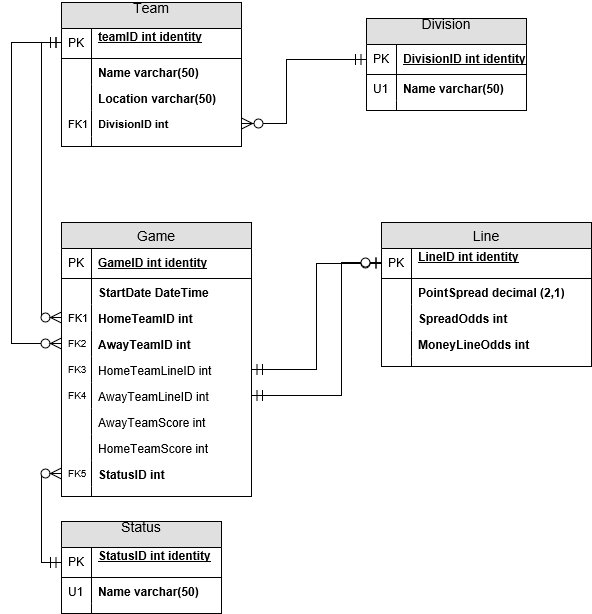
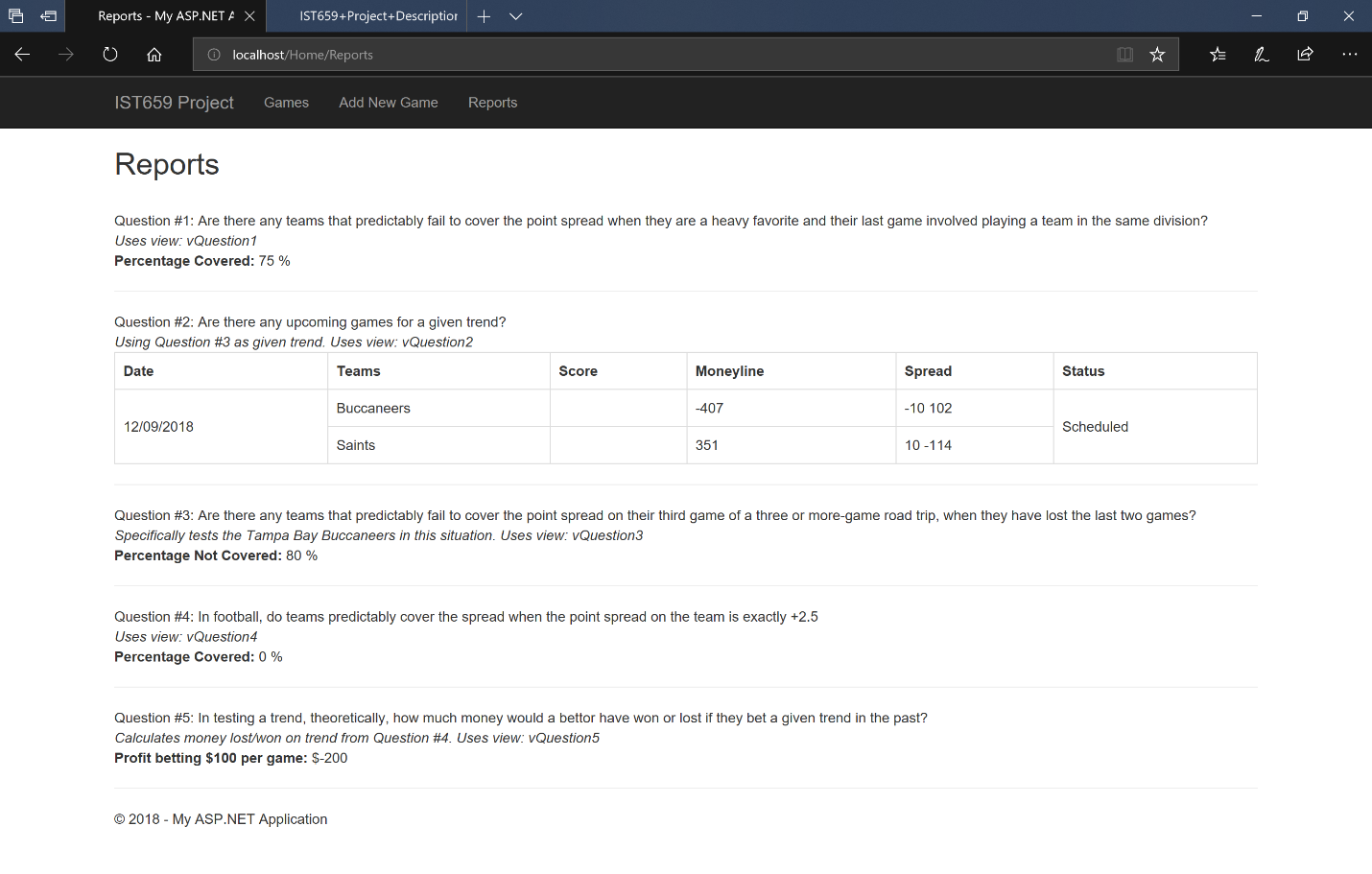
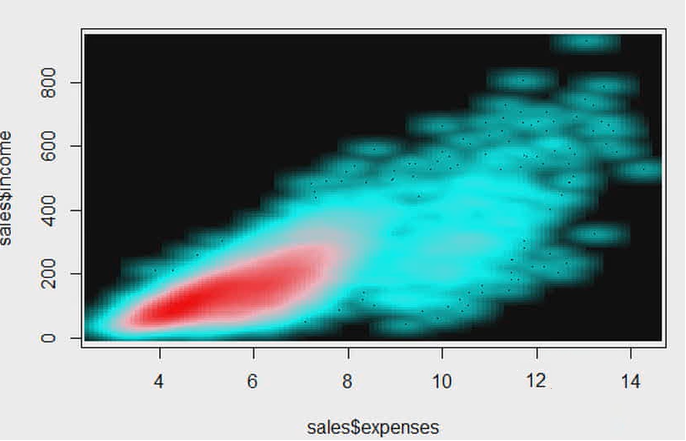
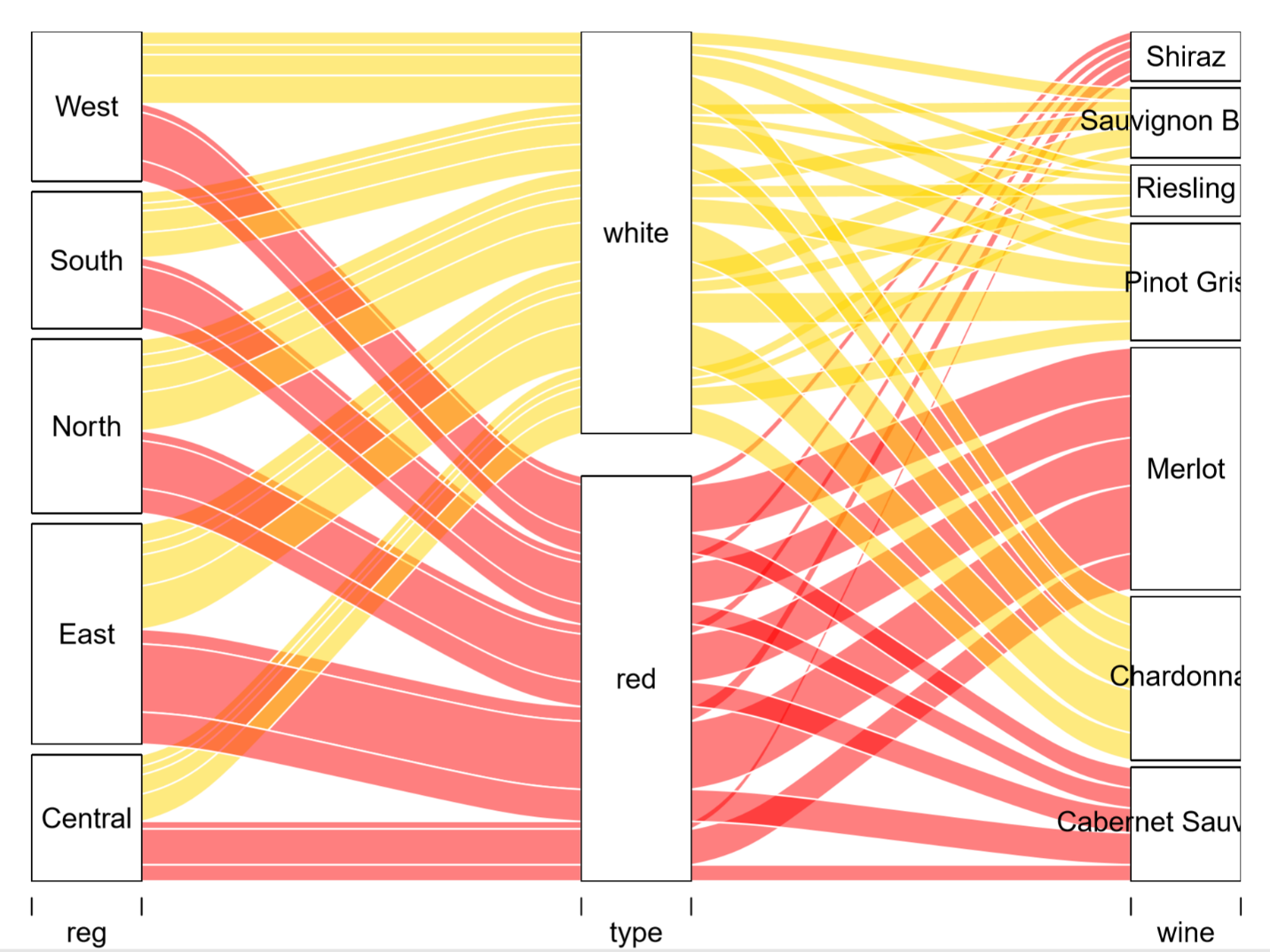
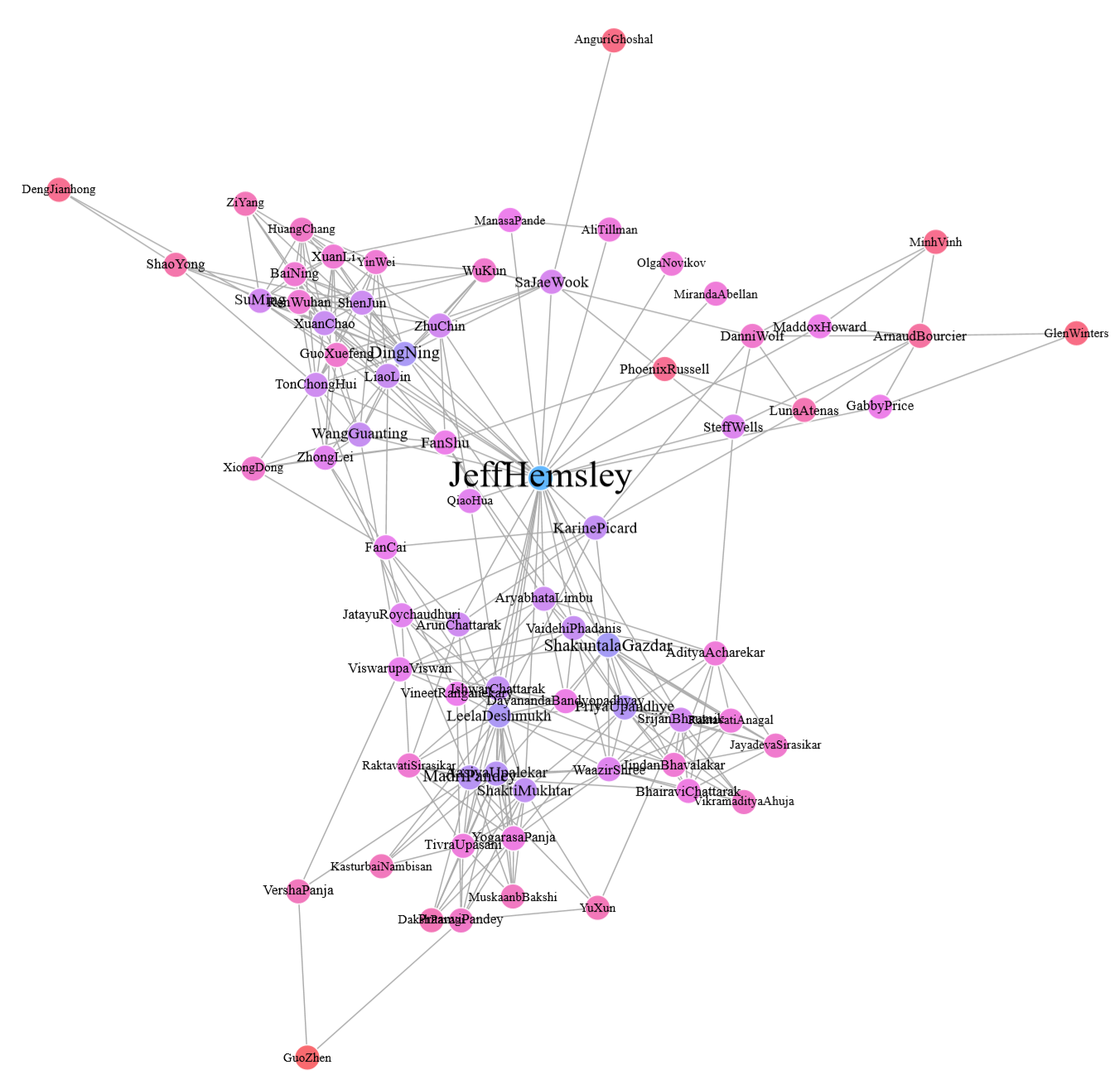
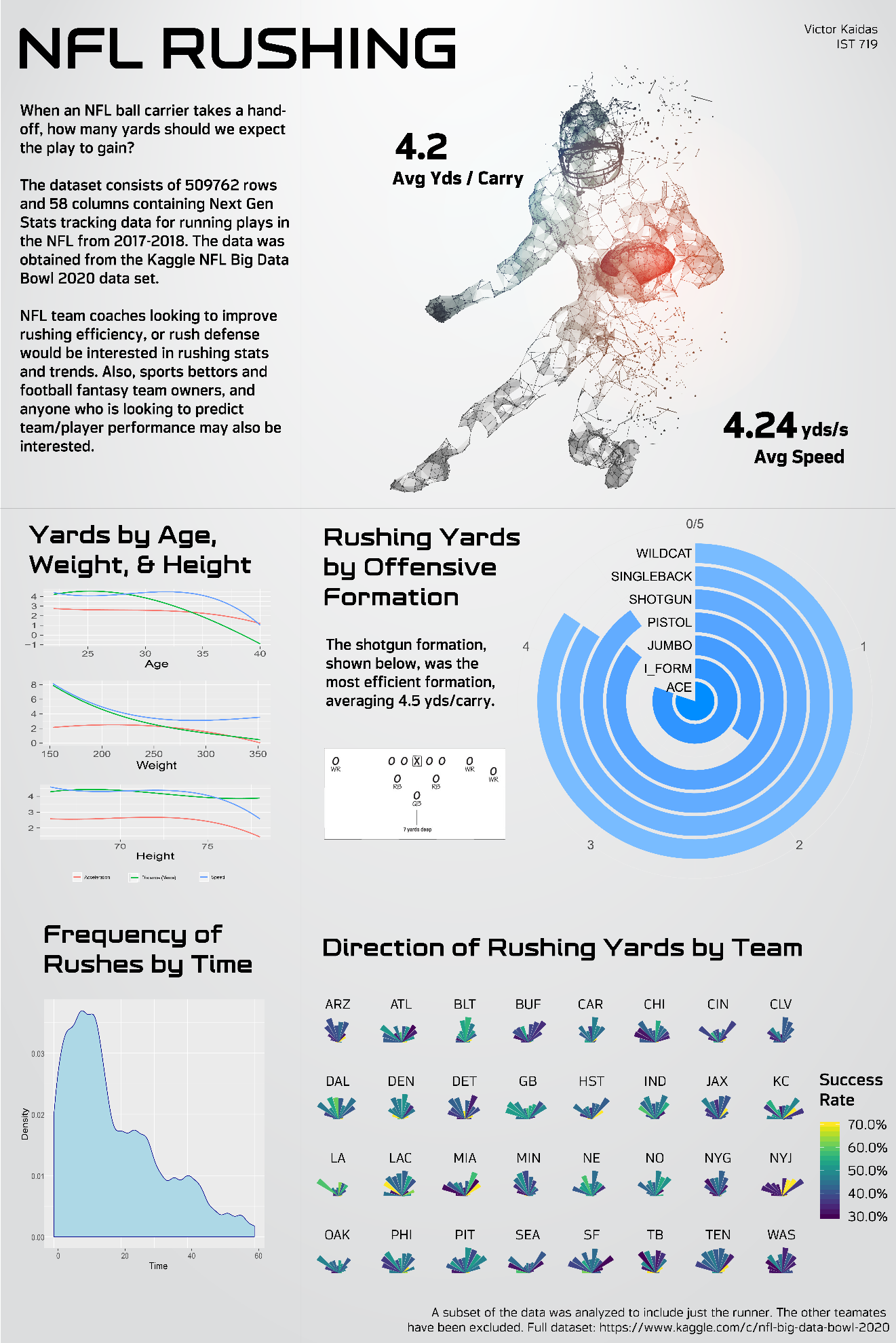
5a  
**  
Figure 1****

**Figure 2  
  
  
Figure 3**

**IST 659 – Database Administration Concepts and Database Management**  
Throughout the course, a fictional online video streaming platform was designed and implemented using a systematic approach, with each week building upon the week before. We were provided a spreadsheet view of the raw data. We identified problem spots and recommended improvements for a relational data model. We were provided narrative descriptions of organizational data needs for the video streaming platform and parsed them for business rules and facts. We modeled the conceptional entity-relationship diagram (ERD) based on these business rules. We used draw.io, an online software tool to create database model diagrams.

We then converted our ERD into a more fully dressed logical model diagrams. We learned to identify appropriate cardinality and degree of relationships from business rules, as well as identify data types using narrative elements and prior domain knowledge. We learned to normalize relations and identify candidate keys and functional dependencies and remove them from relations.   
  
Next, we used structured query language (SQL) data manipulation commands to query and modify data in the video streaming database. We learned to create tables, columns, and constraints, as well as insert, update, query, and delete data. We performed basic data analysis using descriptive statistics provided by SQL aggregate functions. Next we learned to code views, functions, and stored procedures to solve the problems presented. Next, we created database users and administered their user privileges on the database objects. Lastly, we used a variety of applications to connect to our database, such as Microsoft Access, RStudio, and Open Database Connectivity (ODBC). Figure 1 shows the ERD of the streaming video database.  
As a final project, a database was designed and implemented to store data about upcoming and completed National Football League (NFL) games. The audience, sporting bettors, would be able to discover and track various trends in attempt to predicting a winner in upcoming games. First, a conceptual model was created to identify the highest-level relationships between the different entities and the relationships among them. To form this model, several data questions were defined, along with the audience and various user roles. An Entity-Relationship Diagram (ERD) was created to describes the process flow of the sports data, as shown in figure 2.   
  
Next, a logic model was created to describes the data in as much detail as possible, as show in figure 3. The logic model is a better representation of the data in its database form. In addition to all entities and relationships among them, it includes all attributes, primary and foreign keys, for each entity, and normalization of the data. Next, a SQL script was created to generate the database tables, columns, relationships, views, and pre-populated data. The creation of the ERD and logical diagram made it easy to create SQL script. An uninstall script was also created to completely remove the data.  
  
In addition to the database, a web-based user interface was created to perform CRUD operations on the data, along with provide reports to answer the data questions defined earlier. Figure 4 shows a report generated to answer some of the data questions.

  
**Figure 1**  
  
  
**Figure 2**

  
**Figure 3** **Figure 4  
  
  
IST 719 – Information Visualization**  
The Information Visualization course provided a broad introduction to data visualization for data scientists. We developed a portfolio of resources, demonstrations, recipes, and examples of various data visualization techniques through the use of R programming language and Adobe Illustrator. The skills we learned include using data cleaning techniques, developing custom plots, visually exploring data, using design concepts to visually communicate the story in the data, and discussing issues related to the ethics of data visualization. Conceptual themes were presented alongside technical aspects of data visualization.   
  
We discussed the different types of data, places to find it, such as Kaggle.com, and how to import them in R. We learned to clean and manipulate data in R with functions such as the aggregate/apply functions and the tidyr package. We identified stories in data sets through exploration using R to create appropriate rough plots to identify distributions and relationships in the data. We built pie charts, bar charts, histograms, line, scatter, and box plots. In addition to using the built-in R functions to create these plots, we also were able to draw them ourselves by drawing shapes. We built multi-dimensional plots.   
  
We learned to tell a story by selecting a color scheme. We discussed using appropriate colors together, and the meaning behind various colors. We discussed ways to manage overplotting, such as the use of opacity and changing the point size and styles. We were able to display density using a bagplot and smoothScatter plot, as shown in figure 1. We used Adobe Illustrator to enhance our plots from R. We learned to use vector images, instead of raster images, for more flexibility in Illustrator, and to avoid lack of image quality.  
  
We talked about layout, to include hierarchies and grids used to tell the story. We learned to use contrast between colors and sizes, in addition to whitespace to create a hierarchy. We learned about the use of composition, to include the use of the Rule of Thirds as well as the Golden Ratio to enforce flow and the use of lines to point to an area of focus. We learned about the importance of selecting a font and font-size and style, and to divide an audience in a room by distance and target them specifically by font size. We worked in R and Illustrator to create these hierarchical elements.  
  
We analyzed social media data, and were able to visualize the word distribution in a logged density line plot and a word cloud. We dealt with categorical data by plotting an alluvial plot, tree map, and river plot. The alluvial plot shown in figure 2 shows wine sales by region, color, and type. We were introduced to the ggplot library. We learned to draw a map and control the zoom. We created points and colored regions to form choropleth maps to analyze various data sources.   
  
We dove more into social networks and discussed ways to study relationships to discover who is important or who is influential, observing the degree, betweenness, and closeness centrality measures. We evaluate nodes and structures, we developed network maps using the igraph package. Figure 3 shows a social network map, where the size of the node is based on how many links the person has, betweenness is captured by how big the name is, and the color captures how close nodes are to other nodes. We found ways to identify clusters in the networks and created a bipartile plot to visualize them.   
  
We worked to generated three-dimensional (3-D) plots. Using the RGL package, we plotted a network map in 3-D. We produced a frame animation of the map, moving to various controlled camera angles. Lastly, we worked to create various interactive dashboard apps using the shiny package.  
  
For the final project, I created a visualization poster and presented the work during a live session. For my poster, I analyzed and created visualizations in R and Adobe Illustrator, regarding NFL Rushing data from the NFL Big Data Bowl 2020 competition on Kaggle. The poster, shown in figure 4, was designed to provide a hierarchal flow. The rule of thirds alignment was used, along with adequate spacing for rest of eye. The overall font was selected to provide a football look-and-feel. A mix of single and multidimensional plots were used to answer the data questions.   
  
The poster supports the "3 distances" of audience. The football rusher image was added to the top right to attract audience across the room and hint about the topic of the poster. Flashy charts, such as the circular bar chart and sonar charts, along with the mid-sized headings were added to intrigue the middle-distance audience. The granular details offered below the title and surrounding the charts are offered for audience up-close. Some of the analytics offered are direction of yards by team, yards by age, weight, and height, and yards by offensive formation.   
  
  
**Figure 1**  
**Figure 2**  
**Figure 3**  
  
  
  
  
  
  
  
   
  
**Figure 4**  
  
**III. Conclusion**As a student of the MS Applied Data Science program at Syracuse University, I have been exposed to a broad range of areas related to data science. I’ve collected data, either by using the DMAIC process to define and measure data in the process improvement project in MBC 638, by hiring workers on Amazon’s Mechanical Turk website to manually classify movie reviews and calculating Kappa scores to measuring agreement among the workers in IST 687, to using a variety of tools to import, clean, and transpose various types of data files in all of the courses taken in the program.  
  
I have identified patterns in data by using R to create visualizations such as bar plots, pie charts, box plots, scatter plots, line charts, geographical maps, and network maps. I used bar charts known as “sonars” when displaying an NFL rusher’s direction by team. Trendlines were added to visualize the prevailing direction of the data, while horizontal and vertical bars were also added to show thresholds, such to represent the freight forwarder’s expected shortfall and value at risk with metals in FIN 654.   
  
I was able to apply data mining techniques by performing statistical analysis on many assignments to observe distributions and relationships, such as when performing descriptive statistics on customer satisfaction towards Airlines in IST 687 or by performing descriptive and inferential statistics when comparing total dissolved solids between control and treatment group in IST 772. Machine learning was applied to various datasets, such as to predict country happiness in IST 707, predict sentiment analysis on MLB tweet data in IST 687, and identify the anonymous authors of the Federalist Papers by applying clustering and decision tree models in IST 687.  
  
I was able to develop alternative strategies based on the data from the final project in FIN 654, where we were able to recommend a new strategy for Gateway First Bank to win more bids on home loans. Additionally, in IST 687, our team was able to make recommendations to the airline company on the factors deemed most influential to customer satisfaction. I developed a plan of action to implement the business decisions derived from the analyses in the personal improvement project from MBC 638. Since certain arcade games were only profitable at specific thresholds, I was able to create a new process flow to avoid these games at these thresholds. Additional, by using Google Analytics in SCM 651,I was able to develop recommendations to improve Whitman School of Management’s marketing campaigns.  
  
I’ve acquired communication skills as a data scientist, either through IST 772 by using language that accurately describes the uncertainty of inferential statistics, or through live-session presentations in IST 707, IST 687, and IST 719. Through the NFL rushing poster created in IST 719, I was able to target audience from different spots in the room and create a visual hierarchy to tell a story.   
  
We held multiple discussions on ethics in the various classes. Privacy issues, for example, could be violated in the health care industry when predicting when a patient will come for a visit due to health issues. One of the newer ideas is to monitor vital signs 24/7 of a potential patient, outside of a hospital. While data mining may be able to accurately predict the next doctor’s visit, the data collected on an individual may be considered intrusive. Another discussion we had on ethics was on machine learning bias, and how data scientists should be cognizant of this. An example is that when facial recognition is more likely to recognize Caucasians than African Americans.

**References**  
School of Information Studies, Syracuse University. (2019, December). MS in Applied Data Science.  
 Retrieved December 17, 2019 from <https://ischool.syr.edu/academics/graduate/masters-degrees/ms-in-applied-data-science> Sustainable Development Solutions Network. (2017, February). World Happiness Report, Version 2. Retrieved December 17, 2019 from <https://www.kaggle.com/unsdsn/world-happiness>.  
  
The National Football League. (2019). NFL Big Data Bowl 2020, Version 1.   
 Retrieved December 17, 2019 from <https://www.kaggle.com/c/nfl-big-data-bowl-2020>.  
  
Twitter. (2019, December). Retrieved December 17, 2019 from <https://www.twitter.com>.  
  
Nathan Nunn & Diego Puga, 2012. "Ruggedness: The Blessing of Bad Geography in Africa," The Review  
 of Economics and Statistics, MIT Press, vol. 94(1), pages 20-36, 08.