MS Applied Data Science Portfolio

2/6/2024

Brandon O'linn

2024

# INtroduction

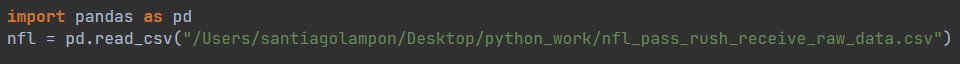
The Master of Science Program at Syracuse University uses a curriculum that focuses on the core theories of Data Science. It also has a strong emphasis on the application of the theories and lessons taught in the class. Many of the courses in the Data Science program involve project-based research and deliverables. The focus of these courses can vary drastically due to the diverse nature of the Data Science field. This diversity also means that to reach the learning objectives of these courses we must learn and adapt to many different software and coding languages. In this program there is a strong emphasis on the coding languages R, and Python. Although the content and software used in this program may be different, they all can be traced back to the core learning goals of the program. Those learning goals can be summarized as follows:

* Collect, store, and access data.
* Create actionable insight across a range of contexts using data and the full data science life cycle.
* Apply visualizations and predictive models to help generate actionable insight.
* Use programming languages such as R and Python to support the generation of actionable insight.
* Communicate insights gained via visualizations and analytics to a broad range of audiences.
* Apply ethics in the development, use and evaluation of data and predictive models.

The goal of this document is to demonstrate the achievement of these learning goals throughout my time in the program by providing examples and deliverables that align with each learning goal.

# Achievement OF Leanring Goals

Collect, store, and access data is something that is required for nearly every project in the program, however the way those steps are handled can vary greatly from project to project. The first example of collect, store, share comes from the final project of IST-652 Scripting for Data Analysis, called NFL Passing and Rushing Comparisons. This project alone leverages three different methods for collecting and storing our data. The first method used is one of the easier methods, we found a CSV provided by <https://www.advancedsportsanalytics.com/nfl-raw-data> , this file was downloaded straight from their website and was read in using the pandas package in python.



The next method we used was to import a python library known as nfl\_data\_py, this library contains NFL data that is sourced from nflfastR, nfldata, dynastyprocess, and Draft Scout. We used this library to bring in NFL play by play data, roster data, and team description data. These three separate tables were then joined to make one cohesive table.

A screenshot of a computer program

Description automatically generated

The last method we used to gather, store and access data was the most difficult of the three, it involved web scraping from <https://www.pro-football-reference.com/years/2019/passing.html> , to do this we used to python modules urllib\_request to open the URL, and BeautifulSoup to parse through the HTML. (To get data for different years we just changed the year in the above URL accordingly.

A screen shot of a computer

Description automatically generated

The above section of code opens the URL and passes the data through to the beautiful soup module.

A screen shot of a computer code

Description automatically generated

This section of code pulls all the table rows from the URL and finds the column headers and then the column rows, collects the data and then is eventually stored as a panda’s data frame.

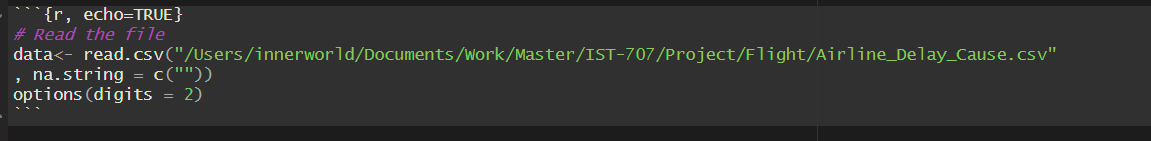
A screenshot of a computer

Description automatically generated

The result of the web scraping technique was the table shown above.

The other projects included in this portfolio used CSV files downloaded from Kaggle, one in Python and the other in RStudio.





Creating actionable insights across a range of contexts using data and the full data life cycle can be seen across the numerous projects completed throughout the course. The data science life cycle is mentioned in nearly every class and is the framework used when creating our projects. This life cycle starts with understanding the business for the project you are assigned. This often comes with a business question that you must find the answer for somewhere in the data. In this portfolio the projects included cover a variety of topics, so the questions being asked are very different. In IST-652 Scripting for Data Analysis we were looking for how the game of American Football has changed over time specifically on offense regarding rushing and passing. In IST-664 Natural Language Processing, we were looking for possible themes in the bible, as well as the overall sentiment of the entire bible and the sentiment of each individual book. In IST-707 we are looking at flight delay data to find out which airlines, dates, and times are the most susceptible to flight delays to minimize time spent at the airport.

The rest of the data science lifecycle includes understanding the data that you will be using, preparing that data for analysis, then using exploratory data analysis to help understand the data and the relationships between different variables. Then comes finding the correct models to apply to the data to look for the answers to your questions. Then these models must be evaluated and fine-tuned to find the right parameters. Once that is done you can deploy the model to find answers. To best see the full data science lifecycle represented we will look at the project from IST-707 Flight Delays. In this project we first start forming questions, these questions included:

1. Which airlines typically have significant flight delays?
2. Which airports typically have significant flight delays?
3. Which time of year, and month typically have the most flight delays?
4. Which airlines typically have significant flight delays?

Then to answer those questions we begin by reading in our data sets and getting an understanding of what each variable represents in the Flight Delay Project we built a glossary of what each variable represented.

A screen shot of a computer

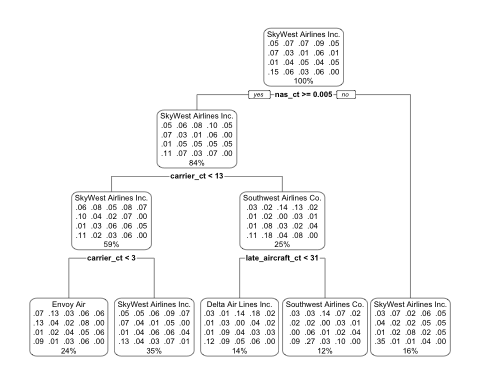
Description automatically generated

We then cleaned the data looked for duplicates and replaced “NA” values with “0” so as not to exclude any airports. Then simple visualizations such as box and whisker plots and scatter plots were used to further explore the data and help establish possible relationships between variables. Our team threw several different types of models at the data such as, APRIORI Association Rule Discovery, K-Means Clustering, Hierarchical Clustering, Decision Tree and Naïve Bayes. We trained and tested for some of these models like the Decision Tree:

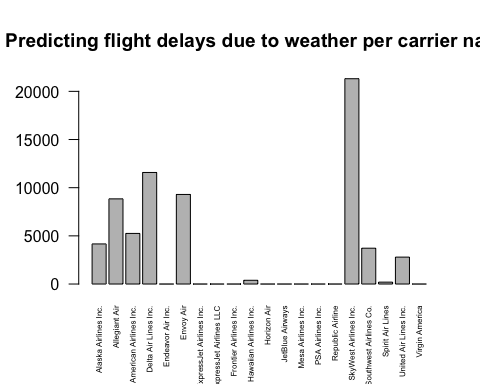
A black screen with white text

Description automatically generated

We deployed the model to make our decision tree:



Then used it to make predictions and visualized the results so that it can be more easily interpreted.



Through all the analysis in this project we were able to answer the questions shown above. The full results and analysis can be found in the RMD file 707 Project in the IST 707 – Flight Delays folder.

The achievement of the next two learning goals of the program, using visualizations and predictive models to gain actionable insight, and using programming languages such as R and Python to use actionable insight, will be demonstrated together, because all the visualizations in these projects were created in RStudio or Python (Jupyter Notebook). Starting with the Bible NLP project, the analysis for this project was completed using Python through Jupyter Notebook. Several different models and visualizations were used. We used Natural Language processing models to analyze different themes, generate genres and classify different chapters of the bible into the genres, and evaluate the overall sentiment of the bible. To find themes in the bible we used Python to evaluate the most common unigrams bigrams, and trigrams, and presented those results as simple bar graphs, then further evaluation was done using Non-Negative Matrix Factorization, and Latent Dirichlet Allocations. These two algorithms are used to find themes by evaluating topics based on the most used words. For Genre Classification we used Naïve Bayes to predict and classify each chapter of the bible into one of several genres: Apocalyptic, Acts, Epistles, Gospels, Wisdom, Prophets, Law, and History. The accuracy of this model was 72% and was visualized using a confusion matrix:

A screenshot of a graph

Description automatically generated

Then we visualized the results of the model using a simple horizontal bar graph.

A graph of a bar graph

Description automatically generated

For sentiment analysis we used the TextBlob function of the Natural Language Toolkit to perform sentiment analysis. This analyzes every word of a piece of text and assigns it a score with 0.0 being neutral and anything above 0 having a positive sentiment and anything below 0 considered to have a negative sentiment. We then did a time series analysis to show the sentiment of each sentence if you were to read the bible from start to finish. Each time series graph is a different book of the bible, the values on the x-axis represent what sentence of the bible is being evaluated, while the y-axis is the sentiment score.

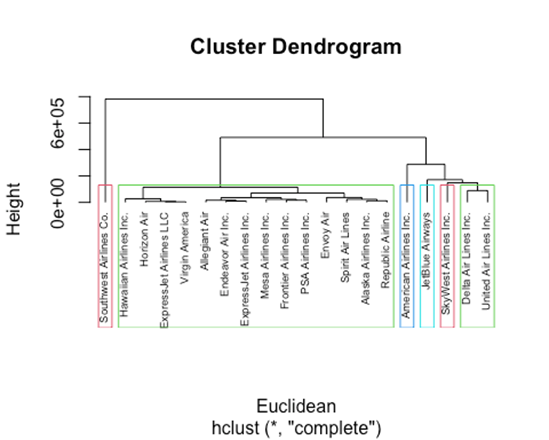
A screenshot of a graph

Description automatically generated

From the Flight Delays Project completed using RStudio, some other models and visuals I found to be interesting or insightful were the decision tree, shown on page 4, the cluster plot, and Cluster Dendrogram shown below.

A graph of a diagram

Description automatically generated



The learning goal: Communicate insights gained via visualizations and analytics to a broad range of audiences, is important because you need to be able to communicate the analysis to professionals in your field as well as people who may not understand the analysis but need to understand the results. Throughout the different projects in this program, we were required to make numerous presentations to our fellow classmates. Our classmates learned the same methods as us, but they did not know what our projects were about or anything about the data that was used in the analysis, so it was important to make sure our visuals and slides were easy to understand and interpret. To do this we took care to not make our slides overwhelming, with too much information, and when we had a complex model or visualization, we provided an appropriate amount of text to help the audience understand the point of the slide. The following slides from the NFL Rushing and Passing Comparison Project demonstrate how we approached making our slides audience friendly:

A screenshot of a computer

Description automatically generated

This slide gives a summary of what you can expect from the next slides in the presentation, explaining where the data came from, how it was used and what we pulled from it.

A graph showing a graph

Description automatically generated with medium confidence

The next slide has a line graph showing the number of yards thrown by all quarterback who threw over 4000 yards total in that season, but what viewers see is just a lot of lines that weave in and out of each other, so to help them we summarized the important take aways, which was who was the best of the best when it came to passing yards.

A screenshot of a graph

Description automatically generated

The above slide is another example of a slide where it may be difficult to understand what the two visualizations are trying to say, so to communicate our findings we put another summary, this one detailing the fall off of running backs that ran for over 1000 yards between 2009 and 2021.

The final learning goal of applying ethics when using and evaluating data and predictive models is important because we don’t want to use bias or manipulate the data to benefit our needs. This is important because when working with customers we do not want to provide them with false information that leads them to making the wrong decisions. An example where we see ethics put into action is in our final analysis of the NFL rushing and passing comparison project. Our hypothesis was that if teams are passing the ball more it means they are running the ball less, however in our analysis we found that to be false and reflected that in our analysis, instead of covering it up or finding a way to prove our hypothesis. In the following slide you can see us touch on this point and provide a possible explanation for why our hypothesis was false.

A black background with white text

Description automatically generated

# Summary

This program has several learning goals: Collect, store, and access data, create actionable insight across a range of contexts using data and the full data science life cycle, apply visualizations and predictive models to help generate actionable insight, use programming languages such as R and Python to support the generation of actionable insight, communicate insights gained via visualizations and analytics to a broad range of audiences, apply ethics in the development, use and evaluation of data and predictive models. Throughout the Applied Data Science program, we were taught about the Data Science Lifecycle and how to apply it to a data science project from start to finish. Through these teachings we were able to complete numerous projects including the ones provided in this portfolio. When you look at these projects and the evidence provided above, I believe that the achievement of the learning goals outlined in this program can be seen.