Written Paper – Portfolio for MS in Applied Data Science

Gabe Herz
gfherz@syr.edu

I. Introduction

The Master's of Science in Applied Data Science at Syracuse University's School of Information Studies is an interdisciplinary degree providing students the opportunity to learn in a broad range of areas pertaining to data science. The program outlines six main goals students will come away with at the conclusion of their degree. These goals are: collect, store, and access data; create actionable insights across a range of contexts; apply visualizations and predictive models to help generate actionable insight; use programming languages such as R, SQL, and Python; communicate insights to a broad range of audiences; and apply ethics in the development, use, and evaluation of predictive models and data. The 34-credit program contains 18 credits of core classes, six credits of secondary core classes (a concentration), nine credits of electives, and a one credit portfolio. In this paper, five courses will be highlighted as validation of successful fulfillment of the program's learning goals. These courses are IST 659 (Intro to Database Management); IST 687 (Intro to Data Science); IST 722 (Data Warehousing); IST 769 (Advanced Big Data Management); and IST 707 (Applied Machine Learning). Note that IST 659, IST 687, and IST 707 are part of the core classes while IST 722 and IST 769 constitute the secondary core concentration within the Data Pipelines and Platforms track.

II. IST 659: Introduction to Database Management

The first course to highlight is IST 659, Intro to Database Management. This course was taken in the fall of 2022, as one of the first core classes. SQL was the primary programming language used throughout the course and the majority of labs were completed within Microsoft SQL Server. The course was composed of weekly lab assignments, asynchronous lectures, in-person discussion sessions, and a final project. For the final project, a database for a Fantasy Basketball website was created. The use-case of the database was to serve as a repository for information on players, their statistics, their cost (known as "salary" in Fantasy Basketball), and their team's information. Using SQL, an up/down script was created to initialize tables, establish relationships between these tables, and finally, insert data into these tables. The blueprint for these tables were laid out within an entity-relationship diagram analysis done in excel, and were mapped artistically through Conceptual and Logical Model diagrams. These diagrams are shown below.

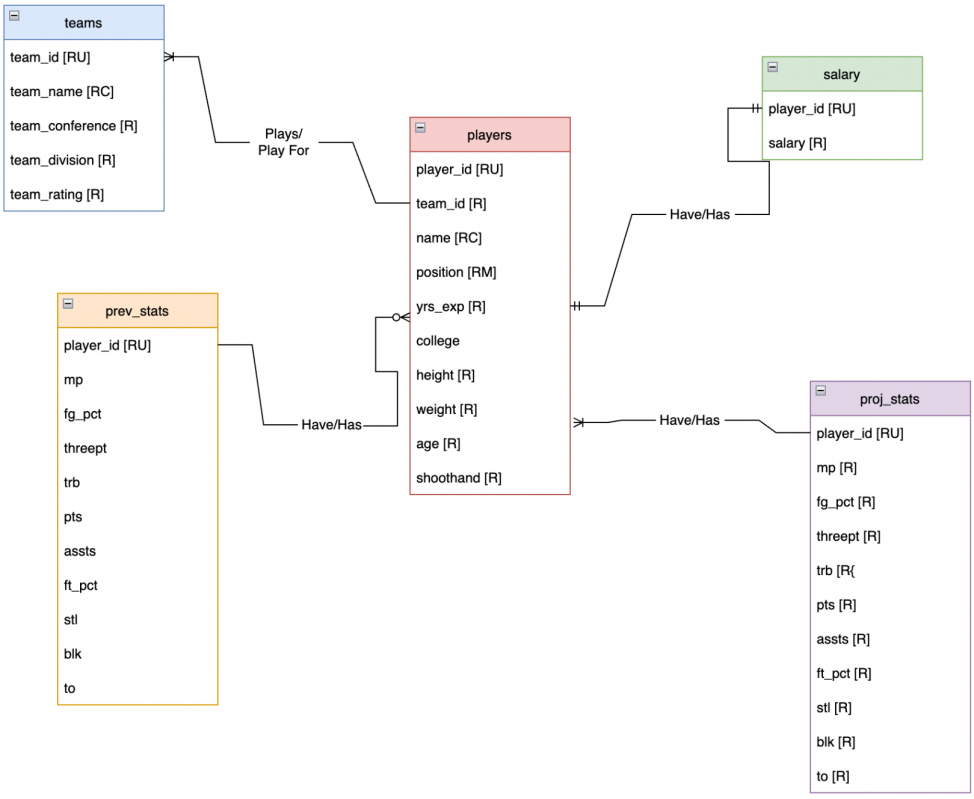

Table 1: IST 659 Conceptual Diagram

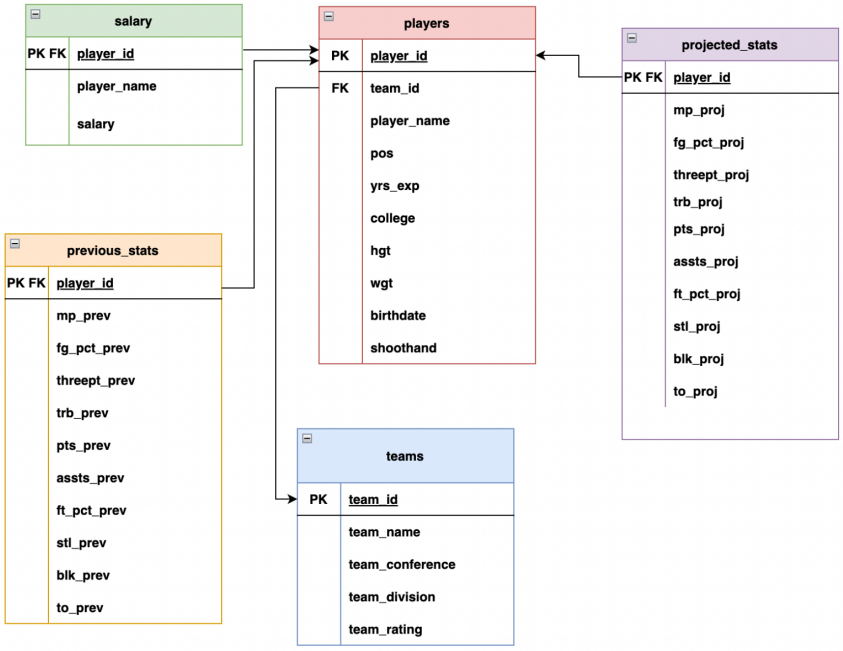

Table 2: IST 659 Logical Diagram

After the framework was built out within Microsoft SQL Server, the attention shifted toward communicating insights and demonstrating the use case of the database. One component of this analysis was the creation of an app demo. Using Canva, an animated walkthrough of the power app was created, showing a user querying basketball players based on certain criteria and the app returning their statistics. The other component was a comprehensive PowerPoint presentation explaining the purpose the database serves and how it's useful.

Reflecting on this project, it is clear how the course goals are highlighted. Data was collected, accessed, and stored using SQL commands within Microsoft SQL Server. The implementation of the power app allowed for the creation of visualizations, aiding the ability to create and generate actionable insights. The presentation helped explain some nuances of both database management and fantasy basketball in layman's terms, allowing the project to resonate with broad audiences. Additionally, the presentation provides snippets of SQL commands used to power the application on the back-end. All of the project information, aside from the SQL scripts (no access to rds for class) can be found at this link:
https://github.com/gherz24/MS-ADS-Portfolio/tree/main/Herz_IST659

III. IST 687: Introduction to Data Science

The next course to highlight is IST 787: Intro to Data Science. This course was taken during the fall semester of 2022, as one of the first core classes. The course consisted of a lecture class, a lab class, lab assignments, homework assignments, an exam, and the final project. For the final project, the aim was to conduct an analysis for an HMO (Health Maintenance Organization) on factors that lead to higher healthcare costs in the United States and how the HMO can adjust for these factors. This project tapped into all of the learning goals of the ADS program. In order to do the analysis, it was imperative to properly collect, store, and access data. Using R, R Studio, and the tidyverse package, a csv file was read into the code console. Once the data was properly loaded, data cleaning was performed on null values in the dataset. Then, using R code, the dependent variable, called "Expensive", was created. The variable was assigned a factor value of "Expensive" if that person's cost of healthcare was larger than $5,000 and was assigned "Normal" for any value less. The ethics of data science was very important within this project. When dealing with different demographics of people, it's imperative to make sure the source of data collection is accurate and all groups are represented proportionally. Otherwise, the data can be skewed and the insights from the machine learning analysis will be biased.

Once the variable was created, the data was partitioned into a training and test set. This ensures that the model is not biased, as it has a certain number of observations to test on as well as a significant enough amount to verify the finds on. Then, using R packages, two models were run on the dataset: Support Vector Machine (SVM) and Recursive Partitioning (Rpart). The results of the modeling showed the SVM model to have a higher sensitivity rate (ratio of predicted positive outcomes to true positives) making it the preferred model. Despite it being the inferior model, the Rpart model was helpful because it provided insights into which variables were important indicators of expensive healthcare, contrary to the SVM model which is blackbox in nature (no insight into predictor variables). The rpart decision tree showed that being a

smoker, having a high BMI, being older, having more kids, and not exercising were indicative of higher healthcare costs. To accompany the models, a slew of visualizations including boxplots, scatterplots, histograms, and maps were created and added to the presentation. These visualizations helped generate actionable insight and communicate these insights to the audience. The visualization below is an example of one of these graphics, depicting the relationship between age and the cost of healthcare.

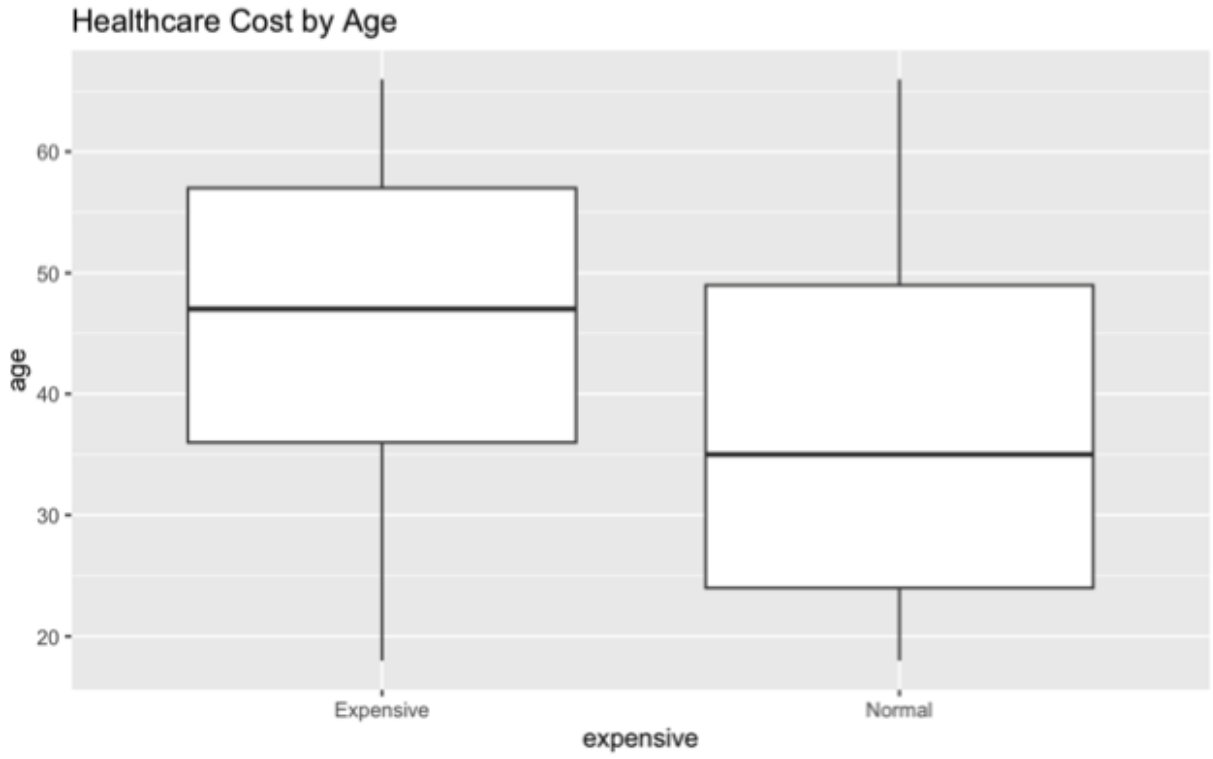

Table 3: IST 687 Healthcare Cost by Age Boxplot

An R shiny app was created in supplement to the presentation. The app served the purpose of allowing users to choose specific criteria and estimate the expected cost of healthcare. Overall, there were three significant recommendations to the HMO; Charge higher premiums to smokers; Charge higher premiums to those in New York; and charge higher premiums to older people. The presentation, Shiny App, code files, and data can all be found at the portfolio link here: https://github.com/gherz24/MS-ADS-Portfolio/tree/main/Herz_IST687

IV. IST 722: Data Warehousing

The next course to highlight is IST 722: Data Warehousing. This was a two-week intensive course taken in January, 2023. The project consisted of four days of eight-hour lectures, homework assignments, and a week-long final project. This was the first course taken in a series of two for the data pipelines and platforms secondary core track. For the final project a data warehouse and business intelligence analysis were produced for the merger of two companies: fudgemart and fudgeflix. Fudgemart was an amazon-like retailer that sold everyday goods. Fudgeflix was a Netflix-like service that sold subscription plans as the product. In order

to create the data warehouse, first, a high-level dimensional modeling worksheet was completed to map out what the target business process was and what attributes would be needed in the tables. For this project, the business process analyzed was the sales analysis of the merged companies. Next, a detailed dimensional modeling worksheet was completed, populating the columns of each individual table with the proper attribute and data type. Relationships were also established in this stage, connecting tables on foreign and primary keys. This worksheet contains macros that, on execution, creates a SQL script to initialize the tables. Once this step was done, the SQL script was copied into Microsoft SQL Server, creating the framework for the data warehouse. Next, Microsoft SQL Server Integration Services (MSSIS) was used to retrieve data from the sources to the stage tables, and then from the stage tables to the dimensional tables and fact table. Along the way, derived columns were used in certain instances to change names of columns to match the destination, as well as to add new columns, change data types, and assign each product to the portion of the merged company they originally belonged to. Below are examples of loading data from the source to stage table, from the stage to dim table, and what the contents of a derived column look like.

Table 4: IST 722 Source to Stage

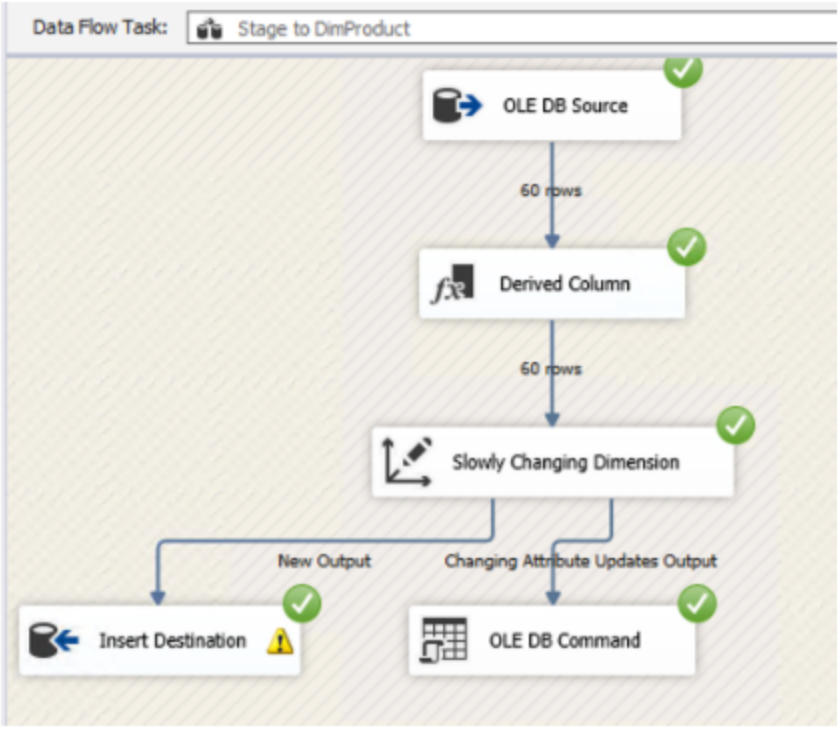

Table 5: IST 722 Stage to Dim Table

Derived Column Name	Derived Column	Expression
ProductID	<add as new column>	product_id
ProductName	<add as new column>	product_name
ProductRetailPrice	<add as new column>	product_retail_price
ProductWholesalePrice	<add as new column>	product_wholesale_price
ProductIsCurrent	<add as new column>	product_is_active

Table 6: IST 722 Derived Columns

With all of the data loaded into the dimension tables and fact table, the ROLAP cube was completed. ROLAP stands for Relational Online Analytical Processing and refers to data stored in tables, columns, and rows. The next step was to create a MOLAP cube. MOLAP, or Multidimensional Online Analytical Processing, stores data in multidimensional formatted databases called Data Cubes. The MOLAP cube was completed using Microsoft SQL Analysis Services and allowed for the connection of the data cubes to Power BI. With Power BI, visualizations were created to help communicate core findings of the project.

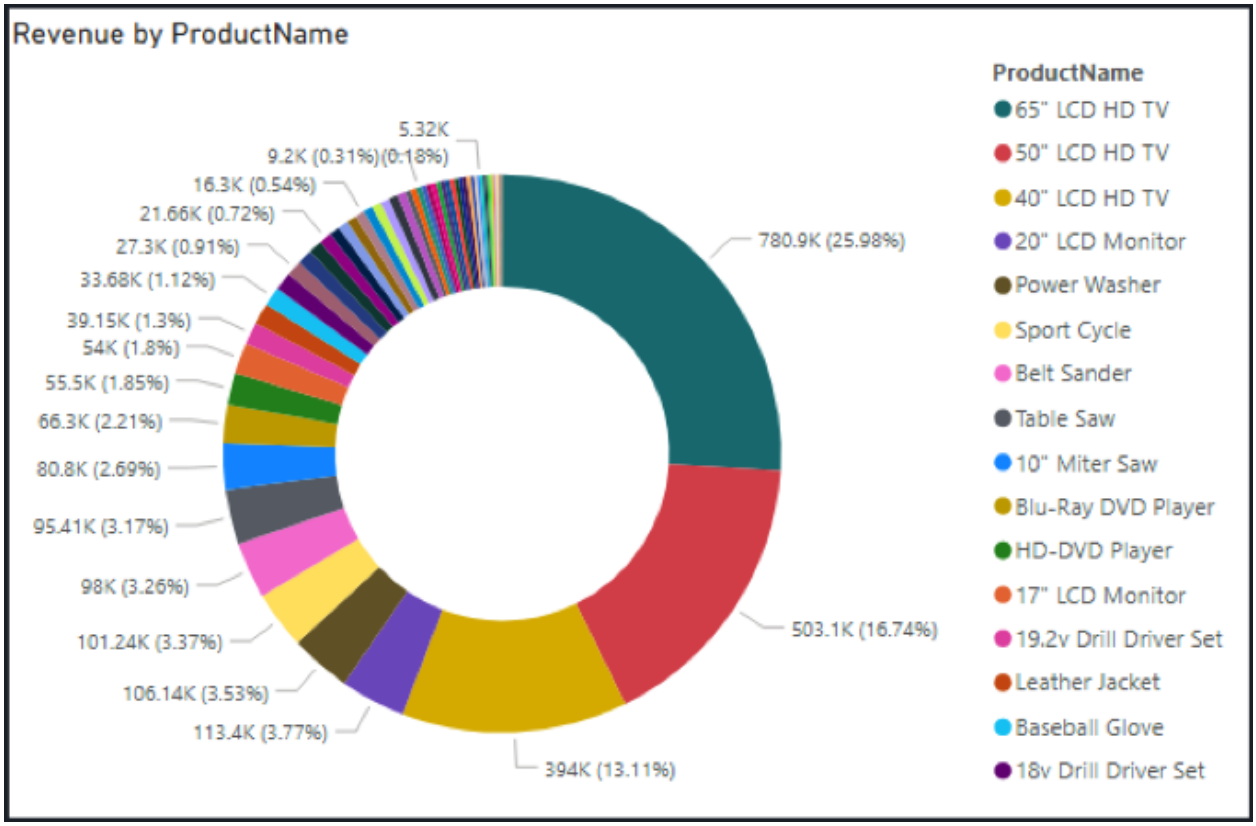

Table 7: IST 722 Donut Chart of Revenue by Product

Table 7 above shows an example visualization produced within Power BI. It shows the three HD TV's as the most revenue driving products for the merged company. All visualizations were incorporated within a presentation shared with the audience.

This intensive course and final project emphasized all of the key learning goals of the ADS program. Excel, Microsoft SQL Server, Microsoft Analysis Server, and SQL programming language were used to collect, store, and access data. Power BI aided in applying visualizations and creating actionable insights into the sales analysis of the merged company. Additionally, the presentation combined the use of these visualizations with examples from the ETL pipeline to communicate these insights to the audience in an easy to understand manner. The presentation, modeling files, SQL scripts, and ETL pipelines can be found in the portfolio repository at the link here: https://github.com/gherz24/MS-ADS-Portfolio/tree/main/Herz_IST722

V. IST 769: Advanced Big Data Management

Taken in the Spring of 2023, IST 769 served as the second and final course within the data pipelines and platforms secondary core track. This course featured asynchronous lessons, synchronous discussions, weekly lab assignments, two exams, and a final project. The course focused on a different NoSQL database management tool each week, including, but not limited to: MongoDB, Cassandra, Hadoop, Neo4j, and Minio. The course primarily utilized Python and Apache Spark as the programming languages. For the final project, a docker-compose file was created to initialize an environment containing Apache HBase, Jupyter Notebook, and Apache

Drill. Apache HBase is a wide-column datastore which organizes data into flexible columns that can be spread across multiple servers or nodes. An instance of Jupyter Notebook with Apache Spark built in was required to access the database through Python. An instance of Apache Drill was necessary to query from the Hbase database with basic SQL commands.

Within the docker-compose file, an image of Apache Drill, an image of Jupyter Notebook with pyspark built in (Apache Spark and python), and the latest image of Apache HBase were included. Additionally, Hadoop name nodes, data nodes, and a history server were added, as Apache HBase runs on top of Hadoop. Once the docker-compose file was complete, the environment was initialized through the windows powershell, opening a virtual environment containing all of these images. Using HBase's native CLI (Command-Line Interface) data was inserted into HBase. To verify the data was loaded correctly, Apache Drill was used to query the data back out using classic SQL commands. A Python script was created to randomly generate data to insert into the command line for repeatability. Another Python script was created to connect to the HBase tables within Jupyter Notebook. The connection was successful, however, it was not possible to create, read, update, or delete tables using pyspark. This is attributed to HBase being rather archaic in nature and not compatible with recent iterations of Python. The last portion of the project was a demo providing insight into what HBase is and how it could be used within the scope of the project.

This project was different from most other projects within the program, and while it doesn't explicitly touch on all of the key goals, it provided a challenging avenue to highlight some of the key goals with a higher intensity than others. For one, the project required a lot of research, trial and error to collect, store, and access data. Creating a docker file and virtual environment from scratch was thought provoking, and having no prior knowledge of HBase made the feat of storing and accessing the data complex. SQL and Python were used heavily within this project to connect, create, modify, and query to and from data tables. The presentation served as an avenue to give insight into what a software foreign to the audience was and how it could be used properly. All project information, except for the demo (file size too large) can be found at the class folder in the project repository, here:
https://github.com/gherz24/MS-ADS-Portfolio/tree/main/Herz_IST769

VI. IST 707: Applied Machine Learning

The final course to be highlighted is IST 707: Applied Machine Learning. This course was taken during the Spring 2023 semester. The course consisted of weekly lectures, homework assignments, and a group project. Both Python and R were used in this course. For the final project, a comprehensive analysis was conducted across a variety of measures on a multi-million row dataset. The dataset, taken from Kaggle, contained every individual pitch thrown in Major League Baseball from 2015 to 2018. One component of the dataset contained the physical data of the pitches thrown, including the pitches start speed, end speed, vertical break, horizontal break, spin rate, spin direction, and break length. The dataset also included scenario specific data such as the count, number of outs in the inning, and dummy variables indicating if there were runners on base. Additionally, the dataset had a file containing the

outcome of each at-bat. Before modeling, Python was used to produce visualizations explaining the data. Useful graphics included boxplots that illustrated the different velocities and spin rates of each pitch type (there were 11 pitch-types in the initial dataset). Histograms were used to visualize how frequently each pitch was thrown. R was used to run Association Rules Mining algorithms and Python was used to run k-Means Clustering algorithms for this assignment. Association Rules Mining was used to answer several important questions: Which pitches are thrown most frequently in various game scenarios?; What physical properties of pitches lead to the most optimal outcome; What characteristics of fastballs and sliders make them more, or less, successful?

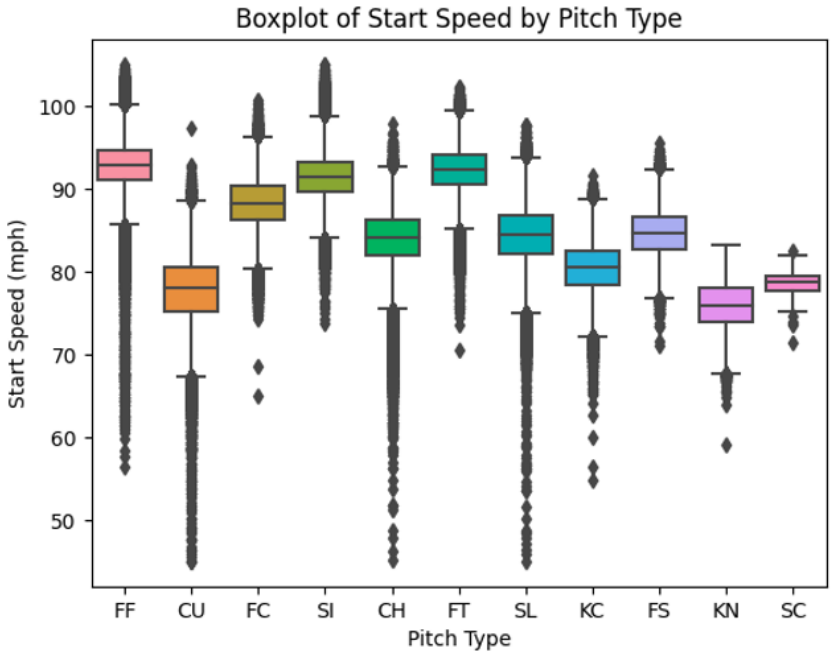

Table 8: IST 707 Boxplots of Start Speed by Pitch

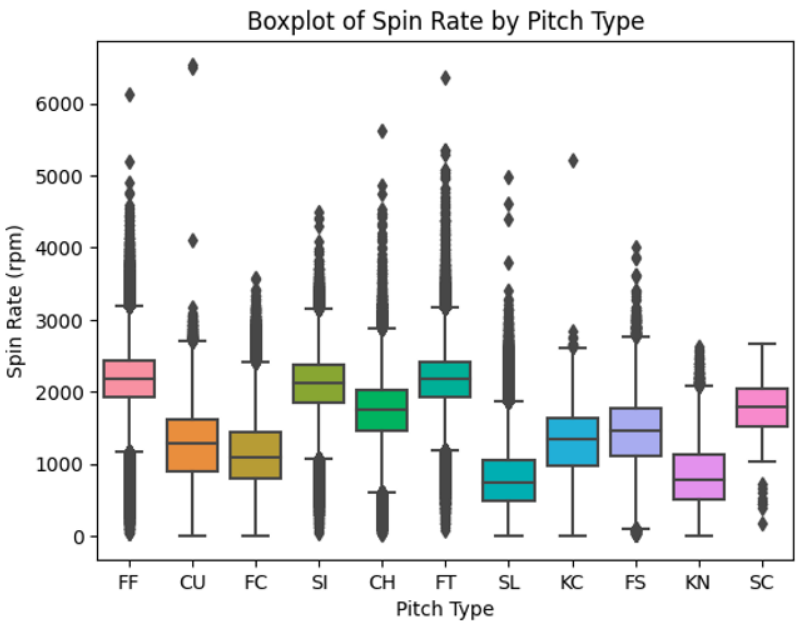

Table 9: IST 707 Boxplots of Spin Rate by Pitch

From the Association Rules Mining algorithm, it was observed that pitchers are extremely likely to throw a fastball in high ball counts (3-0, 3-1) and are extremely likely to throw a slider or breaking ball in high strike counts (0-2, 1-2). When focusing primarily on fastballs (the most common pitch thrown) it was observed that fastballs are more likely to result in a strikeout when thrown high and above the strike zone. Fastballs with velocities greater than 93 MPH were shown to result in a higher proportion of strikeouts, while slower ones ended in more hits. There was a correlation between spin rates and success, as fastballs with more spin led to more strikeouts while fastballs with less spin led to more hits. For sliders, they were observed to be most lethal when thrown low and beneath the strike zone, particularly on the inside portion of the plate.

K-Means Clustering was used to help teams prepare better for upcoming opponents. With 30 teams across two leagues and three sub-divisions in each league, most teams don't play each other. By clustering pitchers by how they throw their pitches, teams can better prepare for matchups against unfamiliar pitchers by comparing them with previously-faced pitchers in a similar bucket. Out of the 11 pitch-types in the dataset, clustering algorithms were run for nine of them, with the omission of the knuckleball and screwball. Those pitch-types were excluded due to the small number of pitchers who threw them. This can be seen in the histogram below.

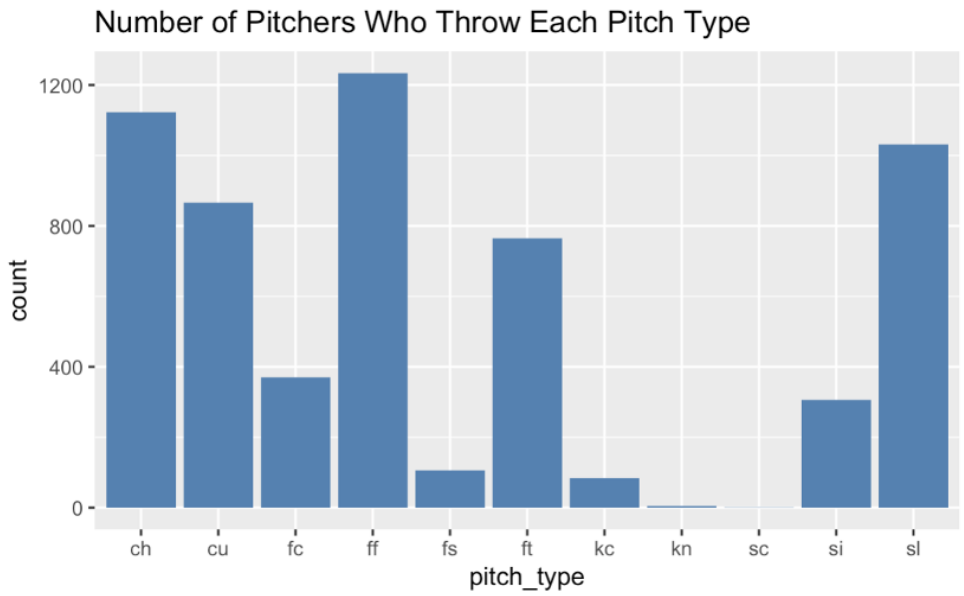

Table 10: IST 707 Histogram of Pitchers Who Threw Each Pitch Type

From the clustering analysis, meaningful clusters were made from seven of the nine pitch types. The written report focused on the takeaways from the three most commonly thrown pitches: Four-Seam Fastball, Slider, and Changeup. Using the four-seam fastball as the example, three clusters were produced. Cluster 1 was associated with slower pitches and less spin. Cluster 2 was associated with the fastest pitches, most spin, and most vertical movement. Cluster 3 was associated with average velocity and average spin, but the most horizontal movement. The highest-caliber pitcher would find himself in Cluster 2, while the lowest would be in Cluster 1. The cluster plot below illustrates this.

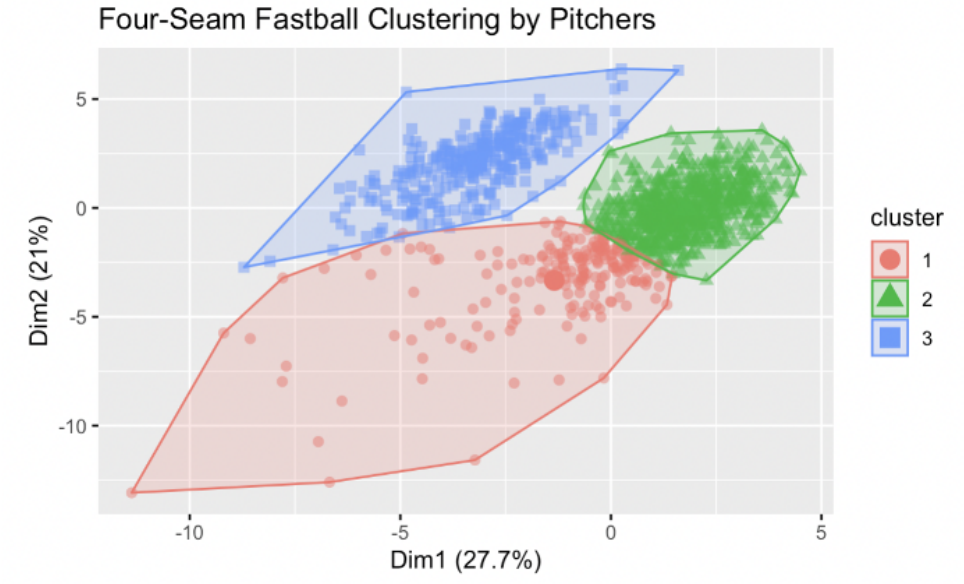

Table 11: IST 707 Cluster Plot for Four-Seam Fastball

This project not only included all six of the learning goals of the program, but provided a creative avenue to dive into a real world problem and provide actionable insights. The takeaways of the project included significant findings that could be genuinely useful for Major League Baseball teams. The Association Rules Mining analysis could help teams identify what pitches to have their pitchers throw to maximize outs, as well as helping their hitters prepare for pitches they will most likely see in specific game scenarios. The Clustering analysis could help teams bucket opposing pitchers into various skill categories, helping to improve how they prepare for upcoming matchups, especially in an absence of data. Collecting, storing, and accessing data was of the utmost importance throughout this project, as this was the largest dataset used throughout the ADS program. Managing, editing, and joining multiple data files with millions of rows within two distinct code environments (R Studio and Jupyter Notebook) demonstrated mastery of the skill. Histograms, box plots, and cluster plots were generated in both R and Python in an effort to create actionable insights and supplement the insights shown in raw form by the algorithm outputs. The written report and PowerPoint presentation were necessary in translating a complex project and subject matter to an audience largely unfamiliar with the sport of baseball. Paying deep attention to the datasets to ensure there was no bias was very important, given the vast differences in observations per pitch. Without doing so, certain pitch groups would have been evaluated improperly. The written report, presentation, code files, and link to the data files (too large so can't include directly) can be found at the portfolio repository here: https://github.com/gherz24/MS-ADS-Portfolio/tree/main/Herz_IST707

VII. Conclusion

This portfolio has demonstrated thorough understanding and completion of six learning goals of the Applied Data Science program. Proficiency in data collection, storing, and access across Python, R, and SQL was shown within each project. The ability to create actionable insights across a range of contexts was evidenced in the analysis of projects in topics ranging from database creation, ETL development, docker environment creation, healthcare analysis, and a sport analytics analysis. Mastery of applying visualizations and machine learning models was shown, through visualizations produced within Python, R, and Power BI, as well as various machine learning models including Recursive Partitioning models, Support Vector Machine models, Association Rule Mining algorithms, and k-Means Clustering algorithms. Fluency in R, SQL, and Python was demonstrated through three SQL-based projects (including a SQL based concentration), two Python-based projects, and two R-based projects. Insights were effectively communicating to a broad range of audiences, with one or more written reports, demos, and presentations completed for each project. This skill was highlighted most prominently within IST 722, where an entire new software was explained and demoed to the class, as well as IST 707, where a complex concept and unfamiliar sport were explained hand-in-hand to the audience. The application of ethics in the development, use, and evaluation of predictive models and data was applied to each project, but was most prominent within IST 687. Understanding the limitations associated with pre-collected demographic data was imperative to provide unbiased and ethical takeaways.

The Master's of Science in Applied Data Science at Syracuse University's School of Information Studies provides students the opportunity to learn in a broad range of areas pertaining to data

science, while gaining acumen across a variety of analytical tools and contexts. With thorough experience in SQL, Python, and R, exposure to database management, machine learning, and visualization creation, students leave the program with the technical expertise suited for a wide-range of data science jobs. Combined with the communication skills learned through presentations, written reports, and demonstrations, students walk-away with the ability to be an impact difference-maker in the professional world.