CSCI 114 Final Project
Hana Erickson

Introduction

The data set I chose for my final project was Ichiro Suzuki's batting statistics from when he played in the MLB. Suzuki began his professional baseball career in Japan and then played in the MLB from 2001-2019. He was famous for being an outstanding hitter which is why I chose to analyze this data set. I was curious to see if patterns or relationships of his hitting style changed throughout his career. Suzuki also had a mid-season trade so I was curious to see how that may have affected his career statistics.

I've always had a passion for sports and baseball in particular has been my family's favorite. Being in a Japanese American family, Ichiro Suzuki was a big conversation topic, and currently so has Shohei Ohtani as he is rising in the ranks. I have had a lot of friends go into career paths where they are sports analysts and I was curious to analyze a data set that was outside the stem field.

Hypothesis

My hypothesis is that Suzuki had increasingly better hitting seasons, until a peak, and then after that, his hitting success declined. A secondary hypothesis is also that he felt the most comfortable playing with the Seattle Mariners because that is where he played the most seasons. Typically athletes have a peak of performance and then it levels off after. I am curious to see when or if Suzuki has a hitting peak and if there is any correlation to the different teams he played on.

The data set many batting statics like batting average, grounding into double plays, sacrifices, etc. It also has how many hits, doubles, triples, and home runs Suzuki had each season. Along with the year, and his age for each season. With the different types of statistics for each season, there are many ways to plot this data to show how the hitting style changed over time. The parts of the data set I looked at to help resolve my hypothesis were the ratio of types of hits over time, grounding into double plays overtime, success rate of bases stolen over time, what year(s) was he sacrificing a lot, linear correlation between age and batting average, correlation between team and batting statistic, and a clustering graph to see what seasons his hitting styles were similar to each other. By interpreting these graphs I will be able to determine what seasons were his better-hitting seasons, if there was a peak, and how drastically was the drop off after the peak.

Data Analysis

I plan on using 7 data analysis techniques to detect hitting patterns in Suzuki's career if he peaked when he peaked, and what team he peaked. Firstly, I plan to plot bar graphs of the percentage of singles, doubles, and triples for each season. This will allow me to see how the ratio of types of hits changed over time. I next plan on looking at the correlations between extra-base hits and grounding into double plays along with how grounding into double plays evolved. This information will allow me to see on seasons where he had more extra-base hits, did he also had more hits grounding into double plays. Next, I will look at stealing over time because I am curious if his stealing was strong the same years his hitting was also strong. Next, I'll look at what year he was the best teammate – looking at sacrifice hits and sacrifice flies over time. Again, I am curious to see if the years he was a better teammate were also his better hitting years and if this has any correlation with the team he was on at the time. I also plan to see how the correlation between age and his batting average and how this deviates from the linear regression. Typically as one gets older their batting average will decrease. I also will plot bar graphs to see how batting average, on-base percentage, and slugging percentage change for the 3 different teams he played on. Finally, I plan on graphing the k-means clusters of hitting styles to see what years were similar to each other. This I plan to do after normalizing the data. I am curious if the clusters will be all on one team or if they will show a different pattern.

Plots and Results

Before I describe the plots I would like to note that I omitted the 2018 and 2019 seasons as Suzuki didn't play in as many games that season so I felt like that data would not be comparable to the other seasons where he played close to every game. I also would like to bring up that in 2012 Suzuki had a mid-season trade from the Mariners to theYankees. In this data set, they include this full season as the team “TOT”. If the plot was a team-independent statistic I would use the TOT row and omit the two half seasons.

Part 1:

My part 1 bar graph showed the percentage of singles, doubles, triples, and home runs per season. The percentage of singles peaked in 2004 and again in 2015. They were typically at least 75% of the hits that Suzuki would have each season. The doubles would be anywhere between 10-20% of his total hits, with his highest number of doubles being in the 2012 season. Triples and Home Runs made up less than 10% of each of the total hits. The only season where the number of triples was much larger than home runs were in 2015 and 2016. Otherwise, the two were similar or Suzuki had more home runs. In part 1 I also plotted the ratio of extra-base hits to total hits over time. 2012 was the season where Suzuki had the most extra-base hits with them making up close to 27% of his total hits. I would say the extra-base hits would oscillate with peak seasons being 2003, 2005, 2009, 2012, and 2016.

Part 2:

In part 2, I looked at extra-base hits over time along with grounding into doubles plays over time. I put the two plots on the same graph to see if there was any correlation between seasons where extra-base hits were high, and whether there were also higher rates of grounding into double plays. The extra-base hits oscillated overtime and the grounding into double plays decreased over time. From 2000-2006 Suzuki was grounding into double plays over 8 times per season. Then the number of grounding into double plays decreased until it peaked again in 2009. Then it continued to decrease, and from 2014 on as he never grounded into a double play more than 4 times per season. Extra base hits seem to increase in the seasons that Suzuki was grounding into double plays less and vice versa. The part that shows this the most evidently is 2011 and 2015. However, this is a general claim as in some seasons both extra-base hitting and grounding into double plays decreased like in 2012 or both increased in 2013.

Part 3:

In part 3, I looked at the success rate of stolen bases over time. Suzuki was known for his fast speed alongside his powerful hitting which made him such an outstanding player. In 2006, his stolen base success rate peaked at 96%. It was consistently over 70% for many seasons and only below 70% for three seasons. However, one of these seasons seems to be an outlier because it was his 3rd to last season where his percentage was 50%, and in the consecutive seasons after that, he was no longer playing in every game.

Part 4:

In part 4, I looked at what season Suzuki was a good teammate. I made bar graphs of the number of times he made a sacrifice fly and a sacrifice hit per season. I thought it was interesting how from 2001-2009 he had more sacrifice hits compared to sacrifice flies but then in 2010 they were the same and then after 2010 the relationship reversed and he had more sacrifice flies than sacrifice hits. Suzuki had sacrificed more in the earlier seasons compared to his later seasons with his most amount of sacrifices being in 2001.

Part 5:

In part 5, I plotted three different batting statistics, the batting average, on-base percentage, and slugging percentage. I made bar graphs of the averages of these statistics for the three different teams that Suzuki played on. Suzuki had the best average batting statistics on the Seattle Mariners which makes sense as he played there the longest and from the beginning to the end of his baseball career, when he was in his prime. On the Mariners his batting average was about 37%, his on-base percentage was 42% and his slugging percentage was about 80%. His stats on the Yankees after his mid-season trade were slightly less with a batting average of about 31%, his on-base percentage was 34% and a slugging percentage was about 65%. Finally, for his last season with the Miami Mariners, his batting average increased to 32%, his on-base percentage decreased to 33% and his slugging percentage stayed the same at 65%. In conclusion, he felt the most comfortable on his first team because he was there the longest and there during his prime.

Part 6:

In part 6, I looked at the age vs batting average on a scatter plot. I also plotted the linear regression to see how much age affects the batting average. As Suzuki got older his batting average got lower over time. Also, the plot deviated more from the linear regression as he got older which shows that age had a smaller effect on the batting average but does contribute to it decreasing over time. Meaning at a younger age, age has more of an effect on the batting average than age as a predicting ability at an older age. The R^2 is 0.513 showing that there is a semi- weak correlation between the age and the batting average.

Part 7:

In part 7 I plotted the K means clustering of Suzuki's hitting statistics. After looking at the differences between singles, doubles, triples, and home runs with various combinations I noticed that when plotting singles vs doubles there were 2 main clusters. These two main clusters show that Suzuki had 2 distinct hitting styles. This would make sense as his hitting oscillated and decreased over time. One cluster had more doubles and more singles and then other cluster had less doubles and less singles. This shows that the two clusters show that some seasons Suzuki had better batting statistics than others. The years he was batting better, the batting patterns were similar to each other and the years he was batting worse the batting patterns were also similar to each other.

Conclusion

In part 1 we see that his better extra base hit seasons were 2003, 2005, 2009, 2012, and 2016. From part 2 we learned that extra base hits seem to increase in the seasons that Suzuki was grounding into double plays less and vice versa. Indicating that his better extra base hits seasons generally also were his seasons he grounded into double plays less. In part 3 we see that he had an overall great stealing success rate with his peak of 96% in 2006.

In part 4, I learned Suzuki was a more sacrificing teammate in his early career. In part 5, we see his batting statistics were the best on the Seattle Mariners which was his first team, also the era when he was sacrificing a lot. In part 6, I learned that at a younger age, age affects batting average more than at an older age and that there is a correlation between the younger you are the better your batting average will be.

Finally in part 7, I looked at the K-means clustering and found 2 clusters that showed the seasons he hit better had similar patterns and the seasons he hit worse had similar patterns to each other.

In conclusion, Ichiro Suzuki had the best batting statistics in his early career on the Seattle Mariners, which is also the team he played the longest on. This was evidenced by his peak seasons, his stolen base success rate, and his ability to be a sacrificing teammate. This conclusion is strengthened by the fact that multiple of his batting statistics were the best on this team and the linear regression that younger age has a stronger correlation to better batting average. This conclusion can help me assume that the two clusters in the k means plot reflect the beginning of his career and the end of his career.

Step-by-Step Code

Part 1:

I first made vectors to hold the data as doubles for singles, doubles, triples, homeruns, extra base hits and years. I then used an ifile to input the data set that was in my github repository. I then made a string called line, that I used to get the header line via getline(). I also checked that the file worked by using the if statement if(!ifile.is_open()).

Then I had a while loop that kept going until there were no more rows to collect data from. By using getline I would get the whole line, and then using string stream and getline to the comma I was able to parse out the data. I used for loops when I wanted to skip over certain columns that were irrelevant to this part. I collected the data for hit, double, triple and home runs this way. I then used the total hits minus the doubles, triples and home runs to solve for the number of singles. Then I found their percentages by dividing each type of hit by total hits and I solved for extra base hits by adding the doubles, triples, and homeruns. Using push back I added these to the vectors I previously made. I also collected the year and added that to the vector as well. When using pushback I used stod to convert the string to a double.

Then using the erase function I was able to clear the seasons I wanted to omit. I counted back from the end for the last two seasons and counted from the beginning for the middle season I wanted to get rid of. I did this for all my vectors.

Then I made a vector of doubles called x that just had numbers 1-17 for the bars to be placed on the number line. I labeled them the year 2001, 2002, and so on but it was too spread out if I had kept them as their actual numbers, hence why I created x. I also made a vector of vectors of doubles that had all the hit ratios.

Then using the bar() function I plotted the hit ratio vs the years. I used the xticks and xticklabels to label the x values as the corresponding year. Then using ylabel, xlabel, and title I was able to label the axis and add a title. Then I used show() so the graph would show up and save so it would save. I repeated this process for extra base hits over time as well.

Part 2:

For part 2 the code until line 75 is very similar to above. I read in the file, made vectors of doubles for hit_vals, gdp_vals, extra_base_hits, and years, and then used a while loop of getline and string stream to parse out the data. I extracted hits, doubles, triples, home runs, gdp and years from the data file and then calculated extra base hits from there. I used pushback again to add to its proper vector. I then once again used the erase function to omit the seasons I was not planning on using.

Then I used the plot function to plot extra base hits vs years. I included "-xr" on this line to have red X's with a line connecting them. Once again I used title(), xlabel(), and ylabel() to create an axis. I then used hold so that I could add my second plot to the same graph, of gdp values to years. I used "-ob" so that it was blue and circles so I could tell the difference. I also set use_y2 to true so I could include a second y axis. Using y2label I labeled this second axis. Finally, I used show and save to see the plot after compiling and running the code.

Part 3:

In part 3 the beginning is similar again, I opened the file, made vectors of doubles called YEAR and sucess_rate. I used getline in a while loop and string stream to parse out the data and I extracted the year, # of stolen bases, and # of bases caught stealing. I divided #of stolen bases by their sum to get the success rate. I used pushback to add the success rate and the year to their respective vectors.

Again, I used erase to omit the seasons I didn't want to include. Then similarly to part 2 I used plot() to plot success rate vs year and used “xr-” to have red X’s. I labeled the axis and then used show() and save().

Part 4:

In part 4, the beginning is similar to the other parts. I imputed the file, made vectors sac_hits, sac_flies, total_sacs, and years. Then I used getline and string stream in a while loop to parse out the data. I extracted the sacrifice hits, flies, and year and used pushback to its respective vector. I also calculated total sacrifices by adding them together and pushing that back onto its vector. Again, I also used erase to omit the seasons I did not want to include.

Similar to part 1, I made a bar graph with the Y values being the sac hits and flies and then x values and labels being the years. Since it was only 2 bars I could use the actual year and it wasn't too spread apart. I used the xticks, xticklabels, xlabel, and title to label my bar graph and show and save to see it after compiling and running. I repeated this to plot total sacrifices over years.

Part 5:

In part 5, I imputed the file the same as the other parts. I then made a vector of strings to hold the team names, along with vectors of doubles to hold batting average, on base percentage, and slugging percentage values. I then made a map of string to doubles for each statistic and the season count to be able to link the team to each stat. I used the while loop again to get the data from the file but this time I used if statements on the years to get rid of the “TOT” row which combined the stats from the full year during Suzuki's mid-season trade. I also omitted the last two seasons again. I extracted the team, batting average, on base percentage, and slugging percentage. Then using the map for each statistic I added the batting average, on base percentage, and slugging percentage to their respective teams map. I also increased the season count for this team by using the szn_count map. Then I iterated through the Batting Average vector which had each of the three teams, extracted the first value which was the team name and pushed it back onto the vector. I then made doubles and averaged all the stats for each team by dividing the summed stat from the while loop by the season count which was incremented anytime the sum was re-calculated. Then I used push_back to add these values to their respective vectors.

Then I made a vector of vector of doubles called data points that are the vectors of the averaged values for each team. Then using bar() I plotted the data and added labels using xticklabels. I labeled the axis and used show() and save() to see the plot after I compiled and ran the code.

Part 6

In part 6, I used a calc_mean function that took in a vector of doubles and summed the values and divided it by the vector's size to find the mean of the values in the vector. I also had a function to calculate beta using a vector of x and y values and the mean of the x and y values. Using a for loop it summed the numerator and denominator which was the x at i-th position minus the x mean times the y at the i-th position minus the y mean and then square of the x at i-th position minus the x mean respectively. Then it returned the numerator sum divided by the denominator sum. I also had a function to solve for alpha which was just the y mean minus the beta times the x mean. I also included a function to calculate the SS_res, which iterates through the fi_datapoints size(), and subtracts the fi_datapoints at the ith position from the y value at the ith position. Then it would sum this value after it was squared. There was also a function to calculate SS_tot which was going through the y values and subtracting the y mean and then squaring this value and summing it. I made all these functions to make my code easier to read and because I made them in a previous lab and found them very helpful.

In main, the beginning is very similar. I imputed the file and made vectors of doubles called age and batting average. Then using getline and string stream in a while loop I extracted batting average and age and pushed it back onto their vector. I omitted the seasons I didn't want and made a vector of doubles for the batting avg regression values. I then used calc_mean on age and batting average to get their mean values. Along with calc_B and calc_A to get the alpha and beta values for this data. I iterated through age.size() number of times to solve for the linear regression values and pushed these values back onto the batting average regression vector.

I then did some calculations for the R^2 value. I made a vector called fi_datapoints which was the same size as the age vector. I went through and set fi_datapoints at the ith position to alpha + beta * age at the ith position. Then I used the fi_datapoints and batting average to solve for SS_res and batting average and batting average mean to solve for SS_tot. Then I solved for R^2 by dividing the two and subtracting it from 1.

Then I plotted batting average vs age similarly to 1 with plot() and “xr-” for red X’s. I used the second axis to also plot the batting average regression on the same plot in blue. I labeled the axis and added the R^2 value and linear regression equation. I used the text function to do so.

Part 7

In Part 7, I made a struct called DataPoint which held a vector of doubles called features and an int called cluster. The vector held the stats like singles, doubles, triples, and homeruns and the cluster was my guess of how many clusters there would be on this plot. There was also a struct called Mean which held a vector of doubles also called features similar to above. The mean represented the centroids and the DataPoints represented just the data points on the graph.

I had a function that found the index of the smallest value in a vector called indexOfMin. It iterated through and used if statements to compare. I also had a distance function that solved for the distance between all the features in the DataPoint to the mean (the centroid). I also had a normalizing function which took the data and normalized it by using a double for loop so each feature, singles, doubles, triples, home runs was normalized. To normalize the values that were summed, the mean was found, the square of the standard deviation was solved using the sum and mean. Then the variance was solved for by taking the square of the sum of the deviations divided by the number of seasons - 1. Then the standard deviation was solved by taking the square root of the variance. Finally each point was normalized by taking the data point, subtracting the mean and dividing it by the standard deviation.

I had the kMeans function next. It started by having a vector of ints of k size all initialized to 0. Then I had the srand() function to get a random number. I randomly initialized the centroids with the rand() divided by the data points size so it was within range. Then set the centroid features to whichever data point it selected. Then made the boolean changes set to true to be used to track if there were changes on the clusters, and made the iterations set to 0 to count the number of iterations for converges. Then within a while loop, changes were set to false at the beginning of each iteration. There was a for loop iterating through the size of the data points vector. There was a vector of distances which was filled with the distance between the centroid and each data point. Then indexOfMin was found on the vector and the centroid was to the minimum value if it wasn't already. If it was changed, then changes were set to true. Counts and features are then reset for the next iteration. Then a vector of vectors of doubles was made called new_centroids. It was initialized to zero and size k by size data points. Also, count was set to 0 of size k. The new centroid was calculated by averaging the features of all the data points within each cluster. The centroid was updated with the new averaged values. Once there were no more changes or iterations exceeded 100 the loop stopped.

Next there was the plot kMeans function which had vectors that held the x and y coordinates for each cluster. Then it looped through the data points and pushed back the values onto the vector. This was the area I could alternate which statistic (singles, doubles, triples or home runs were determined by features[0], features[1], features[2], features[3]) to look at. Then I plotted using a for loop k times with if statements to make sure the plots were the correct color between the different clusters. Then I plotted the centroids, and labeled the axis.

In main, I loaded in the data the same as the other parts. I made a vector of vector of doubles called data and and vector of doubles called years. I extracted year, hit, double, triple, and home run directly from the data set and calculated the singles. I made a vector of doubles called season data that had all the batting stats for that year. Then this vector was pushed back onto the data vector, and years was pushed back onto the year vector. The seasons I didn't want to use were once again removed.

I pushed the data vector through my normalization function to normalize all the hitting statistics. I then made a vector of DataPoints called data_points. Then for data.size() times I made a data point of the data at the ith position with the cluster value set to -1.

I made k = 2 (after a couple plots and trial and error), and made a vector of Mean called centroids set to 0. Finally, I passed in data_points and centroids into kMeans and plot_kMeans.

Code

Shown in Project Folder in GitHub.

Instructions

Download Data Set:

https://www.kaggle.com/datasets/jarredpriester/ichiro-suzuki-mlb-career-stats?resource=download

Compile:

```
clang++ -std=c++17 -I/opt/homebrew/include/ -L/opt/homebrew/lib -lmatplot part7.cpp -o part7
```

Run:

```
./part7
```