Stat 208 Final Project

Ryan Hamlett

June 6, 2018

Introduction

One of the hallmarks of professional baseball are the pitchers' ability to throw many different pitches effectively. In fact, some pitchers throw different assortments of what are generally considered to be the same pitch, such as the different assortments of fastball (cut, two-seam, four-seam, and others). Different pitchers also tend to throw different types of pitches with different grips. All of this can seem a bit complicated and cumbersome for those who are not well-versed in baseball's pitching subculture. But one might wonder, are all of these pitches *actually* different?

As a huge baseball fan, I have wondered if it might be more prudent to classify pitches by spin, speed, and movement rather than simply by how the pitcher holds the ball when he throws it. For many avid baseball watchers, some pitchers throw a cut-fastball, some throw a slider. Both pitches, however, are generally in the range of 88-92 mph and move down and away from a right handed batter at roughly the same angle. In reality, it seems as if the only difference in these pitches is the name being given to them. To the hitter, viewer, and even most pitchers, the pitches are virtually identical. Simplifying classification to call both of these pitches a slider (or cut-fastbal or any other name) would seem to make life easier for casual fans and would likely do little to reduce the information they receive about the pitch being thrown.

Data Collection

All data has been collected using Major League Baseball's Statcast database, which records a number of interesting variables on every single pitch. These variables include "Spin Rate", "Pitch Velocity", "pf_x", "pf_z", "pitch type", and many others. The data pulled from the statcast database for this project is every pitch from the 2018 season through 5/23 this consists of n = 217,521 total pitches. By necessity, since we are using pitch movement as a clustering feature, I needed to limit the pitches to right-handed pitchers, though we could have chosen left-handed pitchers instead.. I do not think that we lose much, if any, in terms of limiting the dataset this way. I then simplified the dataset by removing pitches titled "null,"UN","EP", and"PO" since "null" and "UN" are both indicators that the pitch classification algorithm felt this pitch was unlike any pitch it recognizes, "EP" is an "eephus" pitch which is thrown about 10 times a year, and "PO, meaning pickoff, which isn't a real pitch. I then reduced the dataset further by making sure that every pitch included had a recorded"pitch velocity","pf_x","pf_z", and"spin rate" value. Ultimately, this resulted in a dataset of n = 152,509 pitches.

One worry I had was that some pitchers throw around 100 mph while others throw in the range of 90 mph. The changeup for a 100 mph pitcher is usually in the range of 88-90 mph. If we simply use velocity as a feature, we may run the risk of having a changeup and a fastball classified as the same pitch. While you could argue that a 100 mph fastball is a different pitch than a 90 mph fastball, I found it more intuitive to simply transform the velocity column into "velocity minus maximim velocity" where I found the maximum velocity for each pitcher in 2018 and subtracted each pitch from that value. This gives us a better measure of the relative velocity differences in pitches which I think is more valuable than pure velocity.

The variables "pf_x" and "pf_z" are considered "horizontal movement" and "vertical movement" respectively. Both values are measured relative to a theoretical pitch with no spin-related movement. For example, "vertical movement"" would be equal to 0 on a pitch that simply dropped the amount we would expect it to drop due to gravity alone.

In order for the clustering algorithm to work effectively, we also need to scale the data since Euclidean distances are being calculated. Obviously, since we are running a clustering algorithm, we scale this dataset

without the pitch classifications but we will hold onto the original classifications for later in order to analyze the sucess of our clustering algorithm. All of the work on the data set is included in the code in the appendix.

Methods

For this project, I have used what is likely the simplest clustering algorithm, the k-means clustering technique. In the k-means clustering algorithm, we first center and scale the data. We then pick k starting values as our initialized cluster centers. Each iteration through the algorithm, we calculate the Euclidean distance from all k centers for each point in our dataset. We assign each point to the cluster where the calculated Euclidean distance is smallest. We then redefine the k centers as the mean of each of the k clusters defined in the previous step. These last two steps are repeated until convergence.

While there are many possible clustering algorithms that we may use for smaller datasets, many of these algorithms require the construction of a "distance matrix" that increases exponentially in size as we increase n. For a dataset with n = 152,509, the required amount needed to hold the distance matrix was 86.6 GB. I considered choosing a subset of my data to analyze, possibly a week or two, but the dataset would still be massive and without at least a month's worth of data, we would be selectively sampling pitches that come from starters much more than that of relievers since relievers may pitch quite a bit less in any given two-week period. Thus, I felt it was best to simply run a k-means clustering algorithm with k ranging from 2 all the way to 12. The original number of pitch classifications was 12, as will be seen in the tables in the next section. Since our goal is to possibly reduce the number of pitch clusters, I decided not to go larger than k = 12.

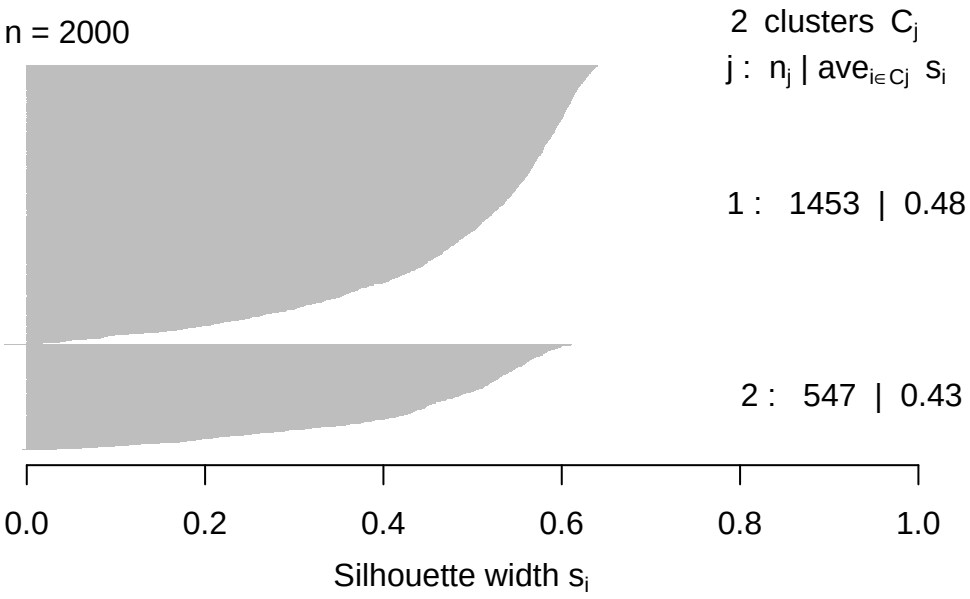
For each individual clustering algorithm, several things are calculated. First, I have computed silhouette diagrams for all 11 different clustering algorithms. Since for each clustering algorithm, a distance matrix is required, I have sampled 2000 data points randomly from the original dataset, with the proportion of each cluster in the 2000 randomly selected points equal to the proportion of each cluster in the overall dataset. This method is prone to error, so I have done this 20 times for each cluster and averaged the average silhoutte length to hopefully get a more stable estimate of average silhoutte length. The plots, however, are simply the plot of the 20th silhouette for each cluster and are available as a visual guide of what happens in run through each clustering algorithm. The second thing I have computed is the ratio of within-cluster sum of squares to total sum of squares for each clustering algorithm.

The reason I have calculated both of these is that I want to choose the appropriate number of k according to both of these criterion functions later on.

Results

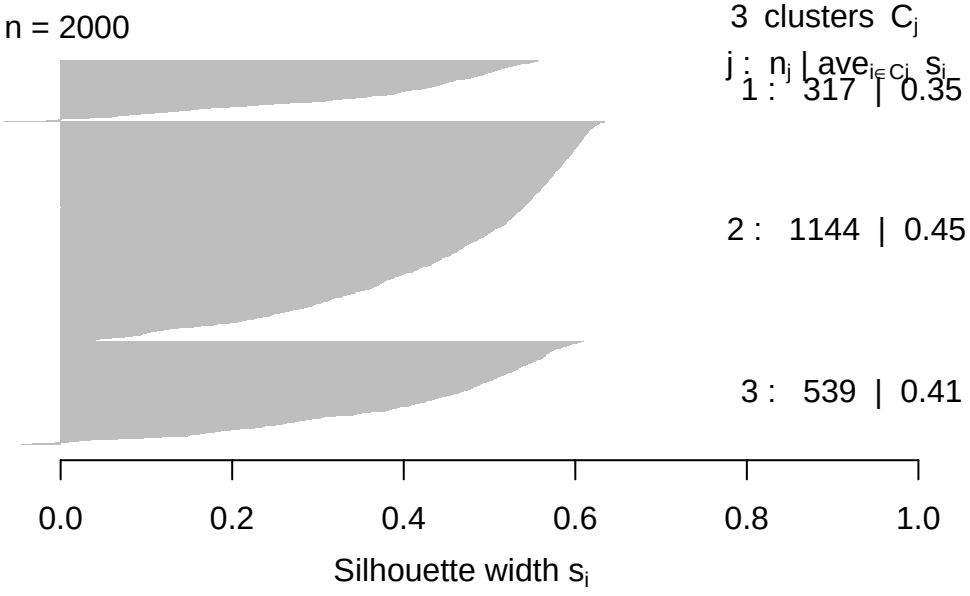
On the next few pages, you will see one of the silhoutte plots calculated for each of the clustering algorithms from k = 2 to k = 11. Directly after these silhouette plots, you will see a table that shows the mean average silhouette length for each k. One popular way of selecting k for clustering algorithms is choosing k to be the number of centers that produces the smallest average silhouette length. Once again, as was stated previously, these average silhouette lengths are prone to some error since we are only sampling a random subset of each cluster for each algorithm. Thus, I have calculated the mean average silhouette length over m = 20 different random subsets of the data in the hopes of reducing the variability of this estimate.

Silhoutte plot for 2 clusters



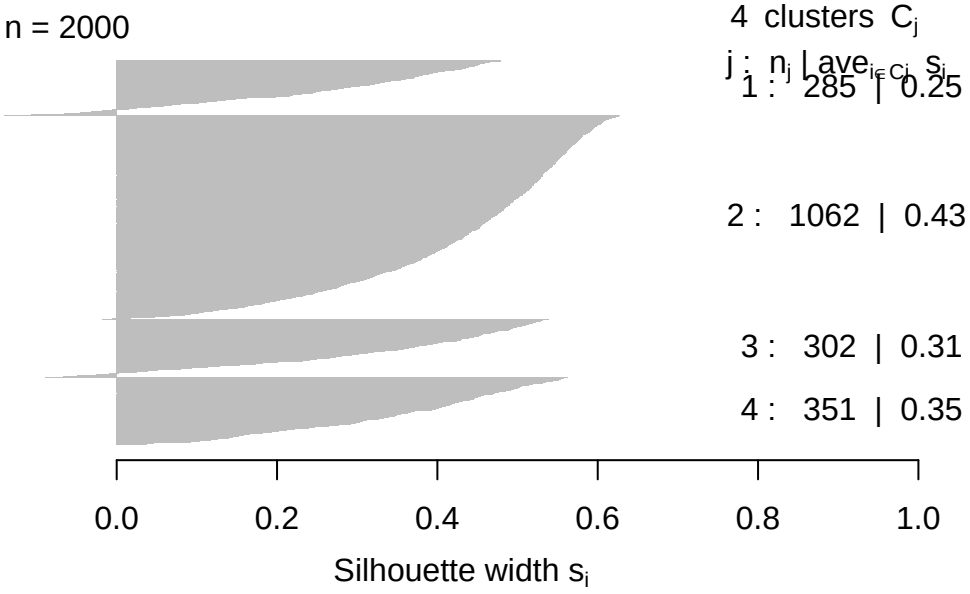
Average silhouette width : 0.47

Silhoutte plot for 3 clusters



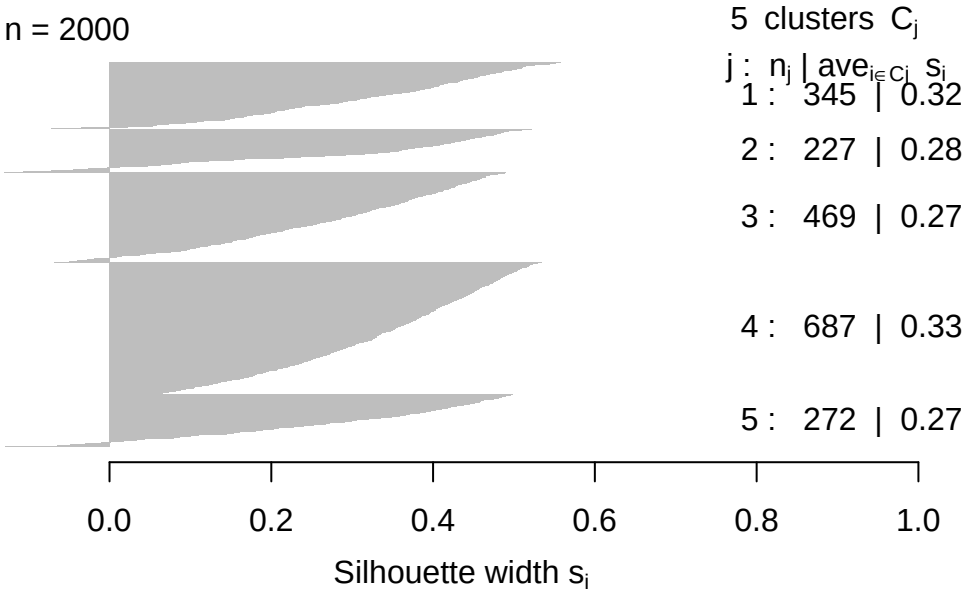
Average silhouette width : 0.42

Silhoutte plot for 4 clusters



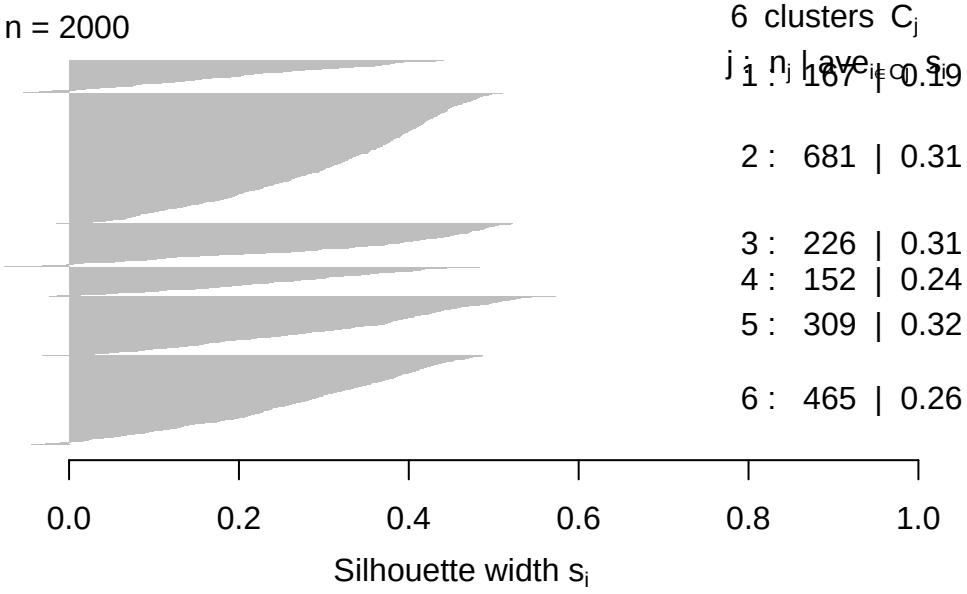
Average silhouette width : 0.37

Silhoutte plot for 5 clusters



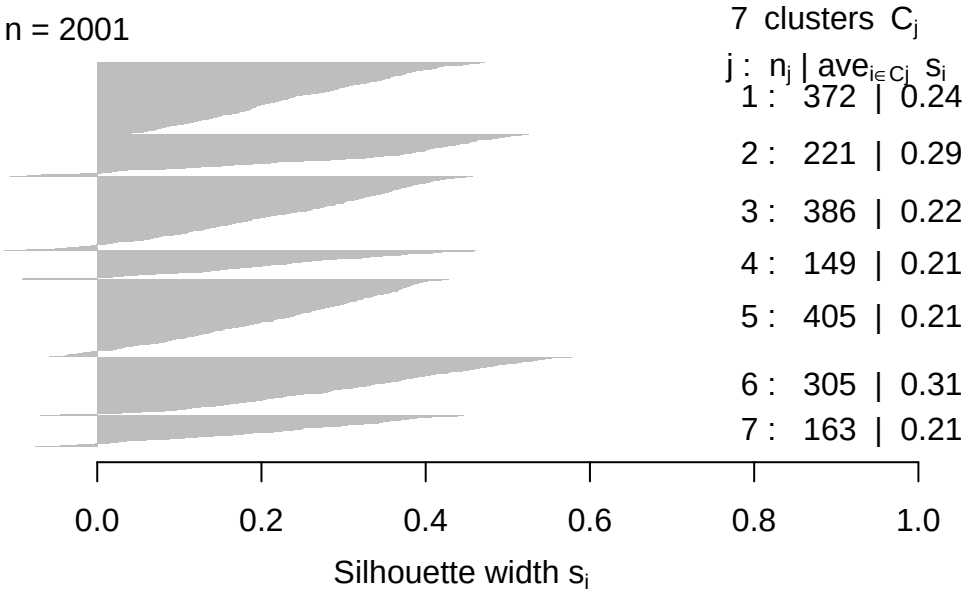
Average silhouette width : 0.3

Silhoutte plot for 6 clusters



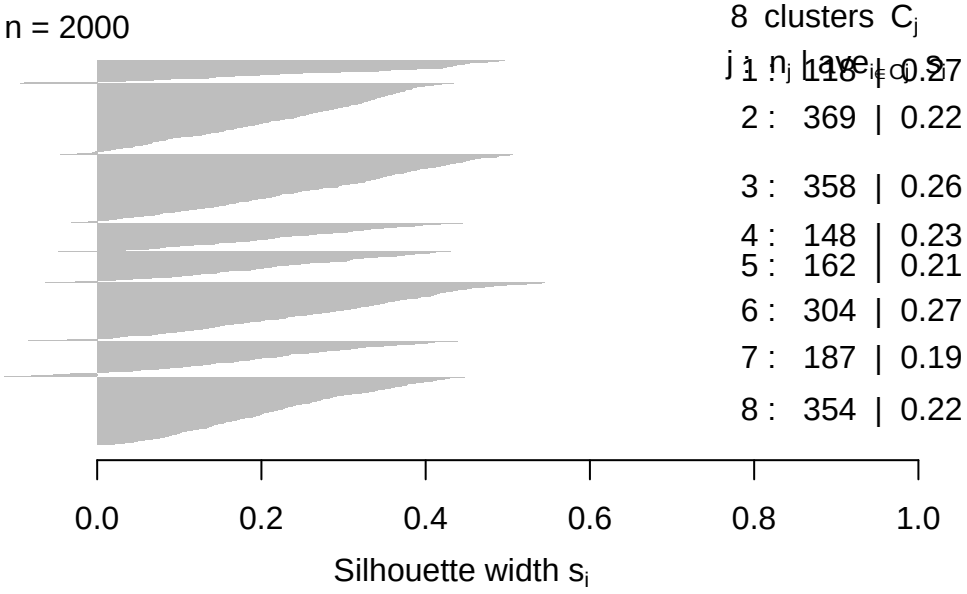
Average silhouette width : 0.28

Silhoutte plot for 7 clusters



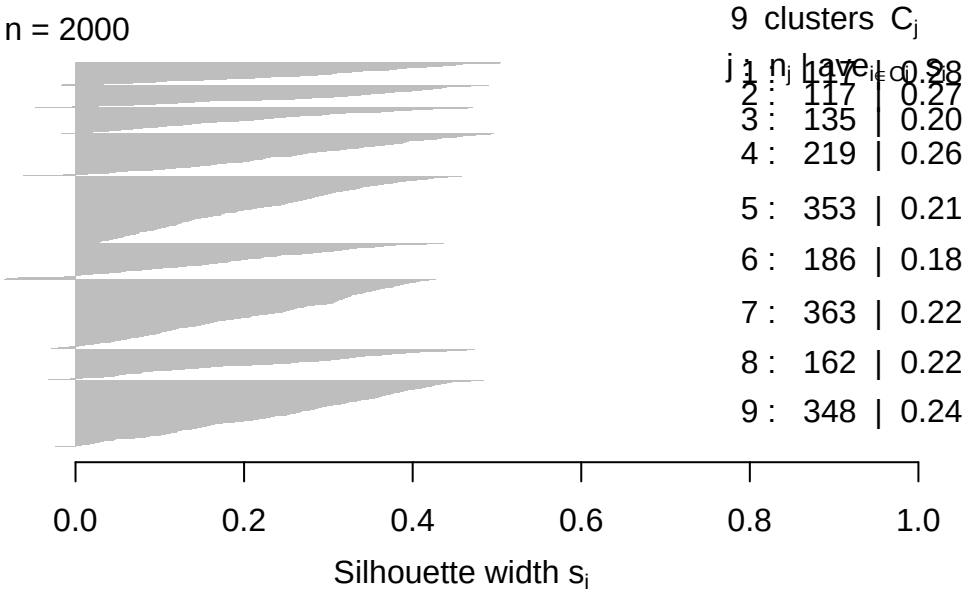
Average silhouette width : 0.24

Silhoutte plot for 8 clusters

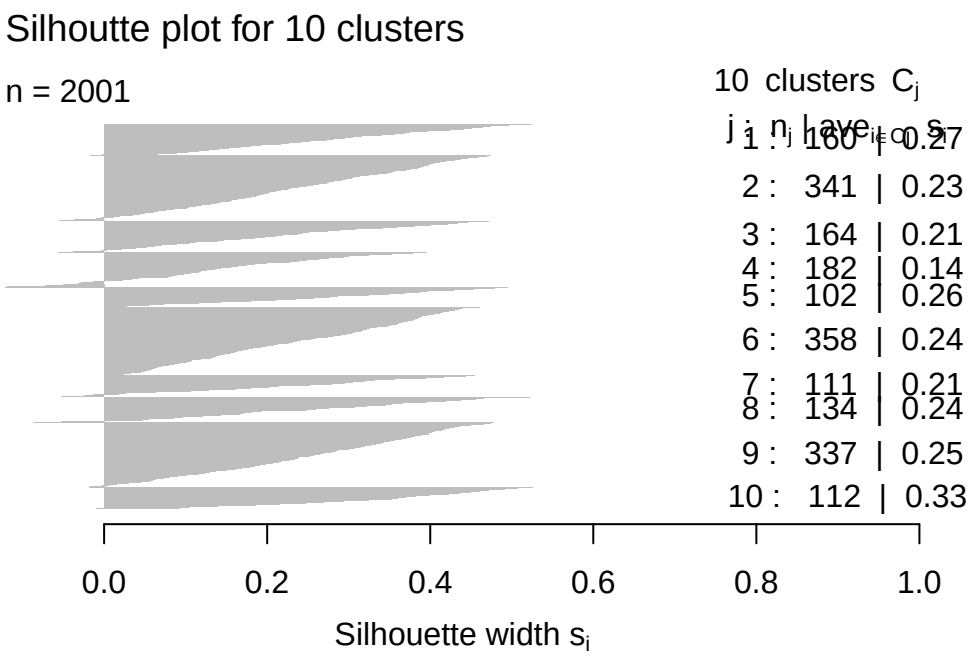


Average silhouette width : 0.23

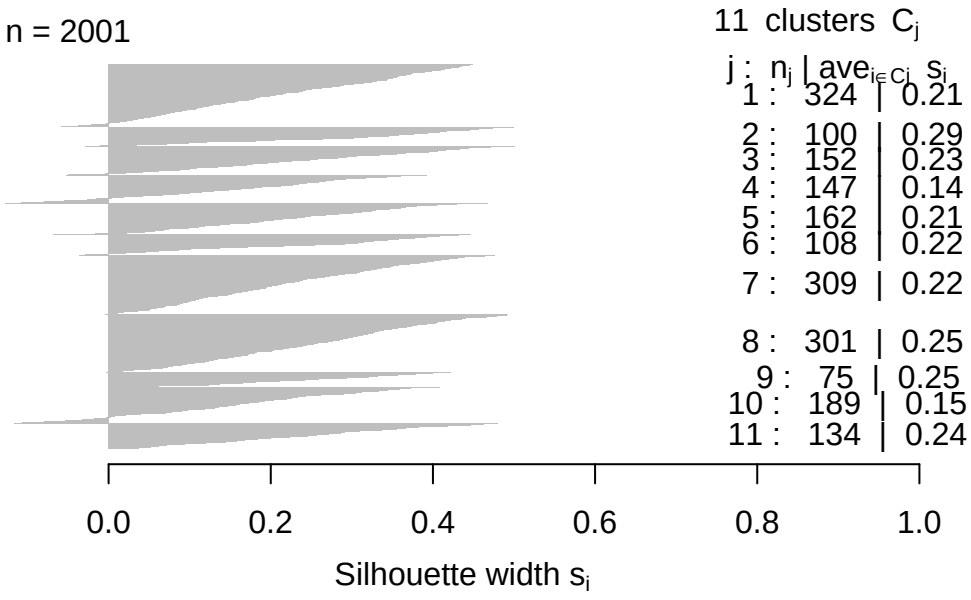
Silhoutte plot for 9 clusters



Average silhouette width : 0.23



Average silhouette width : 0.23



Average silhouette width : 0.22

Analysis of K-Means Clustering Algorithm with k = 11

As we can see in the previous table, the mean average silhouette length favors k = 2 and 3 over the pack of k = 4 through 11 which are all in roughly the same range. We will proceed with an analysis of the clustering method for k = 2 first.

Table 1: Mean Average Silhouette Length for k = 2, ..., 12

2	3	4	5	6	7	8	9	10	11
0.4677	0.4285	0.379	0.2946	0.285	0.2443	0.2381	0.234	0.2334	0.221

Below we have three tables that describe the clustering method in different ways:

Table 1A: Proportion Within Cluster

- This first table shows the proportion of pitches classified under their original classification within both of the 2 new clusters. For example, "Pitch 1" is made up of 30% curveballs, 9% knuckle-curves, and 53% sliders. In other words, the row sums in this table are equal to 1.

Table 2: Table continues below

	Changeup	Curveball	Cut FB	Four-Seam FB	Forkball
Pitch 1	0.134	0.000	0.039	0.510	0.001
Pitch 2	0.007	0.299	0.073	0.002	0.000

Table 3: Table continues below

	Splitfinger FB	Two-Seam FB	Knuckle Curve	Knuckleball
Pitch 1	0.023	0.179	0.000	0
Pitch 2	0.002	0.001	0.088	0

	Sinker	Slider
Pitch 1	0.083	0.032
Pitch 2	0.004	0.524

Table 2A: Proportion Within Original Classification

- This second table shows how the original pitch classifications are distributed through the 2 new pitch clusters. For example, 98% of changeups are classified as "Pitch 2" and 2% are classified as "Pitch 3". In other words, the column sums in this table are equal to 1.

Table 5: Table continues below

	Changeup	Curveball	Cut FB	Four-Seam FB	Forkball
Pitch 1	0.982	0.002	0.587	0.998	0.984
Pitch 2	0.018	0.998	0.413	0.002	0.016

Table 6: Table continues below

	Splitfinger FB	Two-Seam FB	Knuckle Curve	Knuckleball
Pitch 1	0.963	0.999	0.002	0.75
Pitch 2	0.037	0.001	0.998	0.25

	Sinker	Slider
Pitch 1	0.983	0.138
Pitch 2	0.017	0.862

Table 3A: Middle 90% of Feature Space

- This third table shows the middle 90% of the feature space in each new pitch cluster. For example, in Pitch 1, the middle 90% of spin rate is between 2487.0 and 3114.1 reveolutions per minute (rpms), the middle 90% of horizontal movement ranges between 0.447 and 1.598 inches, the middle 90% of vertical movement ranges between -1.600 and -0.411 inches and the middle 90% of velocity relative to the maximum is -21.90 to -12.70 miles per hour (mph).

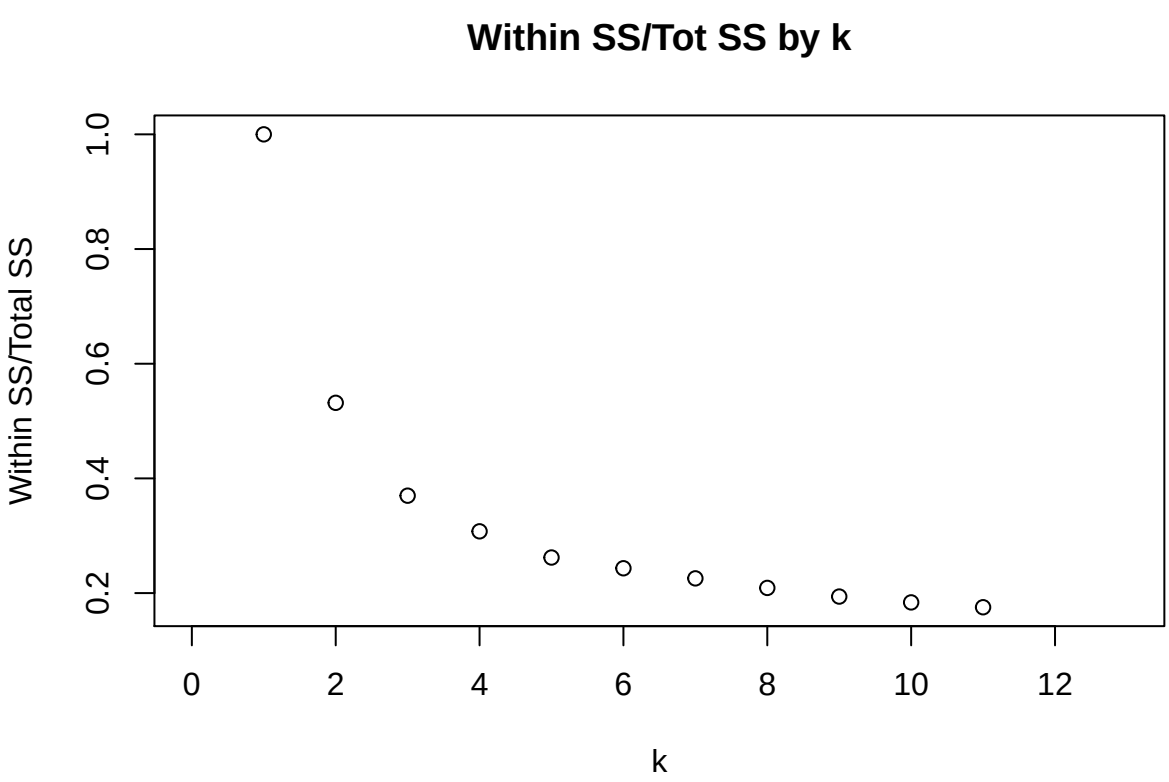
	Pitch 1 5%	Pitch 1 95%	Pitch 2 5%	Pitch 2 95%
Spin Rate	1539.000	2527.000	2070.000	2914.100
Horiz. Movement	-1.540	0.092	0.043	1.427
Vert. Movement	0.179	1.630	-1.311	0.661
Velo - Max	83.300	96.800	75.100	89.000

The most interesting part of this reclustering is that we have reclustered pitches into only two clusters. Originally, we started with 11 pitch classifications and the goal was to reduce the number of clusters to simplify pitch classification. What we have done here is recluster pitches into what I would consider "straight" pitches (i.e. fastballs, changeups, sinkers, splitters, etc.), and "breaking balls" (i.e. curveballs and sliders). This is interesting in that this algorithm seems to believe that the best way to reclassify pitches is just to classify pitches by splitting them based on movement in the x and y direction as seen in the previous table.

Analysis of K-Means Clustering Algorithm with k = 4

The primary goal of this project was to see if it was possible to reduce the number of pitch clusters, but reducing to only two pitches might be a little too simplistic. If we are to recluster all of these pitches in a more descriptive way, how should we determine the number of k to use if not via the average silhouette length as before?

One method of selecting k is what is called "The Elbow Method." This method only requires that we calculate the ratio of within-cluster sum of squares to the total sum of squares for each k. As we would expect with any sum of squares statistic, this value will always decrease as k increases. Thus, we need to select k where we start to see little-to-no decrease in our within-cluster sum of squares ratio. The following plot and table shows the relationship between k and within-cluster sum of squares divided by total sum of squares.



1	2	3	4	5	6	7	8	9	10	11
1	0.532	0.37	0.308	0.262	0.243	0.226	0.209	0.194	0.184	0.175

Looking at the plot, the argument could be made for $k = 3$, 4, or 5 in my opinion. The drop in the ratio of within-cluster sum of squares from $k = 2$ to $k = 3$ is substantial (about 0.15), and the drop from $k = 3$ to $k = 4$ is quite a bit smaller (0.06), and I think it would also be logical k to choose is $k = 5$ since the within-cluster sum of squares ratio drops consistantly as k increases up until we from $k = 5$ to $k = 6$ where the decreases is less than 0.02. At this point, I should also state that the number of clusters I had hoped to be able to achieve when I started this project is in the range of about 4 or 5. Thus, I will proceed with a similar analysis as with $k = 2$, but now with $k = 4$.

Table 1B: Proportion Within Cluster

The table below shows us several things:

1) "Pitch 1" is made up of 61% Changeups with a smaller number (11%) of split-finger fastballs. We can probably label this pitch as simply a "changeup", though we will investigate how these pitches truly look when we look at the feature space laer.

2) "Pitch 2" is made up of mostly curveballs, knuckle-curves, and some sliders. This pitch can probably be labeled more simply as your standard "curveball".

3) "Pitch 3" is made up mostly of four-seam and two-seam fastballs. Surely, this pitch would be called a "fastball" as a catch-all term.

4) "Pitch 4" is comprised almost exclusively of cut-fastballs and sliders. I think we could more accurately call this pitch a "slider" and remove the cut-fastball terminology.

Table 9: Table continues below

	Changeup	Curveball	Cut FB	Four-Seam FB	Forkball
Pitch 1	0.002	0.539	0.002	0.000	0.000
Pitch 2	0.010	0.000	0.011	0.679	0.000
Pitch 3	0.608	0.004	0.001	0.030	0.003
Pitch 4	0.012	0.027	0.241	0.033	0.000

Table 10: Table continues below

	Splitfinger FB	Two-Seam FB	Knuckle Curve	Knuckleball
Pitch 1	0.000	0.000	0.156	0
Pitch 2	0.001	0.214	0.000	0
Pitch 3	0.105	0.103	0.000	0
Pitch 4	0.005	0.002	0.011	0

	Sinker	Slider
Pitch 1	0.000	0.300
Pitch 2	0.081	0.004
Pitch 3	0.120	0.026
Pitch 4	0.001	0.669

We have to be wary with the proportions in this table, however. It might be more useful to look at how each of the original pitch classifications are distributed across the new pitch clusters

Table 2B: Proportion Within Original Classification

If we look at the table below, we can see that most changeups, forkballs, splitfinger fastballs, and knuckleballs are classified as "pitch 1", most curveballs and knuckle curves are classified as "pitch 2" (with some of the knuckleballs and sliders mixed in), most four-seam and two-seam fastballs are classified as "pitch 3", most cut-fastballs and sliders are classified as "pitch 4".

Table 12: Table continues below

	Changeup	Curveball	Cut FB	Four-Seam FB	Forkball
Pitch 1	0.003	0.937	0.005	0.000	0.000
Pitch 2	0.051	0.000	0.120	0.972	0.000
Pitch 3	0.925	0.007	0.003	0.012	0.984
Pitch 4	0.021	0.057	0.872	0.016	0.016

Table 13: Table continues below

	Splitfinger FB	Two-Seam FB	Knuckle Curve	Knuckleball
Pitch 1	0.002	0.000	0.919	0.25
Pitch 2	0.017	0.877	0.000	0.00
Pitch 3	0.933	0.120	0.003	0.75
Pitch 4	0.048	0.003	0.078	0.00

	Sinker	Slider
Pitch 1	0.000	0.257
Pitch 2	0.704	0.013
Pitch 3	0.294	0.024
Pitch 4	0.002	0.706

Based on these results, the naming mechanisms I have proposed in the previous section still seem to be reasonable. The last thing we should do, however, is investigate the feature space of each of our new clusters to see how these pitches act in general.

Table 3B: Feature Space

A few notable things about the feature spaces of these pitches seen in the table below.:

- Pitch 1 has a noticeable difference in spin rate from all other pitches. A pitch under 1500 rpms will almost always be classified as pitch 1.
- Pitch 3 is where most of the high velocity pitches are classified.
- Pitch 2 is where most of the pitches have significant downward and horizontal movement but have a similar speed to Pitch 1 are classified.
- Pitch 4 is where the pitches with a higher velocity but with horizontal movement are classified, typically.

Table 15: Table continues below

	Pitch 1 5%	Pitch 1 95%	Pitch 2 5%	Pitch 2 95%
Spin Rate	2116.000	2989.000	2002.000	2535.000
Horiz. Movement	0.201	1.566	-1.516	-0.092
Vert. Movement	-1.442	0.141	0.558	1.663
Velo - Max Velo	73.800	85.000	89.000	97.200

	Pitch 3 5%	Pitch 3 95%	Pitch 4 5%	Pitch 4 95%
Spin Rate	1198.000	2117.000	2050.000	2750.000
Horiz. Movement	-1.622	-0.419	-0.140	0.814
Vert. Movement	-0.287	1.210	-0.207	1.058
Velo - Max Velo	79.200	90.900	81.200	91.600

Ultimately, after analyzing the feature space of these pitch clusters, the naming conventions I have proposed continue to seem reasonable. Thus, in the case where we choose $k = 4$, we end up with only 4 total types of pitches in baseball: A changeup, slider, curveball, and fastball. This naming convention simplifies the nature of pitching instead of having multiple names for pitches that look and act the same to most observers.

Discussion

Pitching in baseball can seem quite complicated and convoluted to the casual fan. Understanding things like pitch sequencing is even further muddled by the current naming conventions of pitches in the Major League Baseball. The goal of this project was to reduce the complexity of the naming structure for casual fans and see how much, if any, information we lose.

Using k-means reclustering, out of necessity, I used two different methods to determine k. One method, average silhouette lengths, led me to choose $k = 2$, a significant simplification upon the naming conventions we started with. This seems like an effective way to classify pitches extremely simply, especially to casual

fans, since "breaking ball" and "fastball" are relatively easy to understand, even for a beginniner. It would be nice to find a happy medium between classifying to benefit a beginner and the many classifications we have now. Thus, reclustering with a slightly larger k seems reasonable and not particularly harmful if we choose k in an intelligent way.

Using the "elbow method" to select k, I chose $k = 4$. This clustering algorithm was more insightful than at $k = 2$, since we managed to reduce the types of pitches down to changeup, slider, curveball, and fastball.

Ultimately, this project was insightful in that I confirmed a lot of suspicions about which pitches are similar to each other and how the naming conventions in modern baseball seem precise but can be convoluted and imprecise as well.