MLB Pitch Clustering Analysis with R's K-Means Clustering Algorithm

Jon Anderson

Summer, 2019

Pitch Clustering Introduction

Major League Baseball (MLB) makes their pitch-by-pitch data set available to the public. This data set contains 85 different columns of data about every pitch thrown in every MLB game. One part of this data set is the classification of each pitch (whether it be a fastball, curveball, slider, etc.) and the data about the movement of that pitch (velocity, horizontal movement, and vertical movement). This provides us with a nice oportunity to cluster pitchers together based on the attributes of their pitches. In this example, we will focus on all of the sliders thrown (this pitch typically is thrown in the mid to high 80's with lots of horizontal movement and not a lot of vertical movement, although these numbers vary from pitcher-to-pitcher).

Data Import

We will first load in the data, take just the needed columns (pitcher name, pitch name, pitch speed, pitch horizontal movement, and pitch vertical movement) into a new data frame. Then we will convert the number values (speed and movement) to numeric form, and then select only the sliders while ignoring all rows with a null value.

```r
data <- read.csv("G:\\Sports\\flb\\savant\\data\\2019pitches.csv")
df <- data[,c("player_name" , "pitch_name", "release_speed", "pfx_x", "pfx_z")]
suppressWarnings(df$release_speed <- as.numeric(as.character(df$release_speed)))
suppressWarnings(df$pfx_x <- as.numeric(as.character(df$pfx_x)))
suppressWarnings(df$pfx_z <- as.numeric(as.character(df$pfx_z)))

sliders <- df[df$pitch_name == "Slider",]
sliders <- na.omit(sliders)
sliders$pitch_name <- as.character(sliders$pitch_name)
```

Set Up

My data sample will be all of the MLB pitches thrown from the beginning of the 2019 season until early August, when I downloaded the data set (580,329 pitches with 98,818 being sliders). Right now we have a bunch of rows for each individual pitcher, so we need to make a new data set that has just one row per pitcher. We will use a loop to accomplish this. First we get a list of all unique pitchers in the data, and then we loop through each pitcher that has throw 200 or more sliders to create their row. We isolate a data frame with just that pitcher's sliders, and then take the median values (we chose median to control for the outlier pitches that may have been mis-recorded) of their velocity and movements. We add the rows to a master data frame at the end of every loop and then we have our data frame that we can do our clustering analysis on.

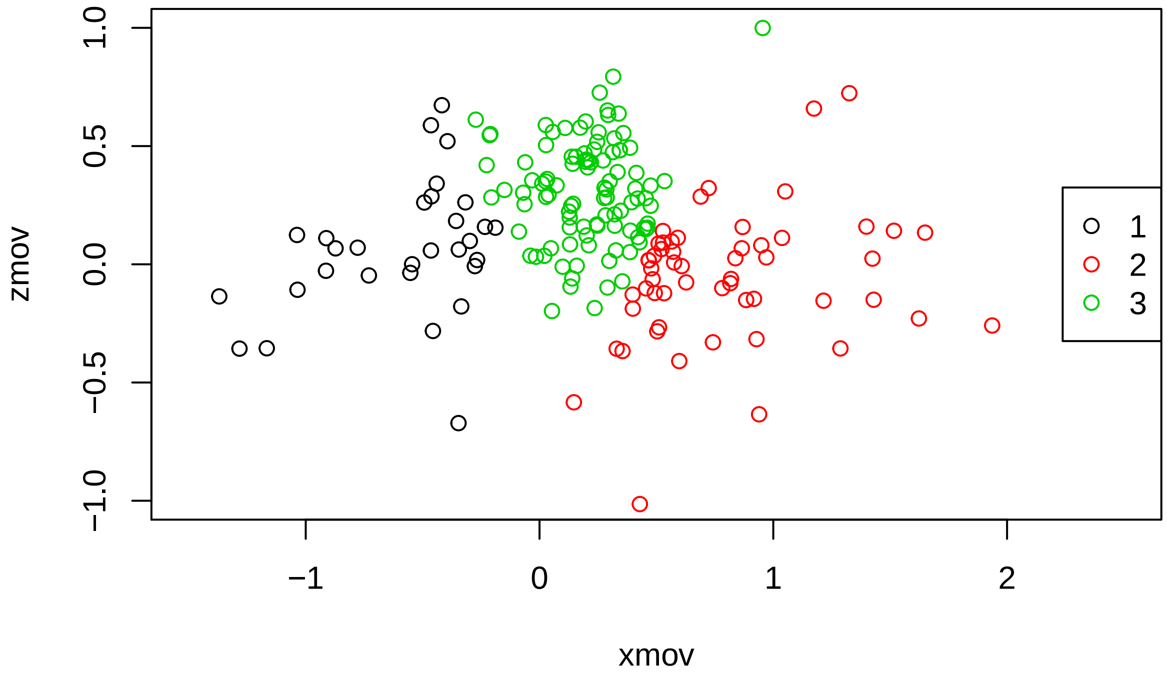
```r
sliderdf <- data.frame(player=factor(), velo=double(), xmov=double(), zmov=double())
pitchers <- as.vector(unique(sliders$player_name))
```

```r
for (value in pitchers) {
  tempdf <- sliders[sliders$player_name==value,]

  if (dim(tempdf)[1] > 200) {

    velo<-median(tempdf$release_speed)
    xmov<-median(tempdf$pfx_x)
    zmov<-median(tempdf$pfx_z)

    if (is.na(velo)) {velo<-0}
    if (is.na(xmov)) {xmov<-0}
    if (is.na(zmov)) {zmov<-0}

    newdf <- data.frame(value, velo, xmov, zmov)

    sliderdf <- rbind(sliderdf, newdf)
  }

}
```

Time To Cluster

The hard work is done, and now we can use the k-means clustering algorithm to do the heavy mathematical lifting for us. The K-Means algorithm looks at all of our individual points and clusters them together into X clusters (we provide the X) after picking X centers and deeming which points belong to which centers. We are using just the horizontal and vertical movement for this clustering example. Here is what it looks like when we choose three clusters.

```r
mov_df <- sliderdf[,3:4]
kmeans3 <- kmeans(mov_df,3)

tbl <- table(sliderdf$value, kmeans3$cluster)
write.csv(tbl,'three_clusters_results.csv')
plot(mov_df[c("xmov", "zmov")], col=kmeans3$cluster, xlim=c(-1.5,2.5), ylim=c(-1,1))
legend("right", legend = paste("",1:3), pch=1, col=1:3)
```



We see that this more or less just clustered the points together based on their location along the x-axis (the horizontal movement of the pitch). This makes sense since the there is a wider spread of values on the x-axis than the y-axis (there is a range of about 3.5 on the x compared to just 2 on the y.)

One thing we would expect to see is that the left-handed pitchers would be in a different cluster than the right-handed pitchers, since their sliders move the opposite way horizontally (their slider moves left to right from the pitcher's perspective, while a right hander's slider will move right to left). Let's see if that's true. Chris Sale, Matthew Boyd, Brad Hand, Max Fried, and Carlos Rodon are three left-handed pitchers known for big, sweeping sliders. They should all be in the same cluster here.

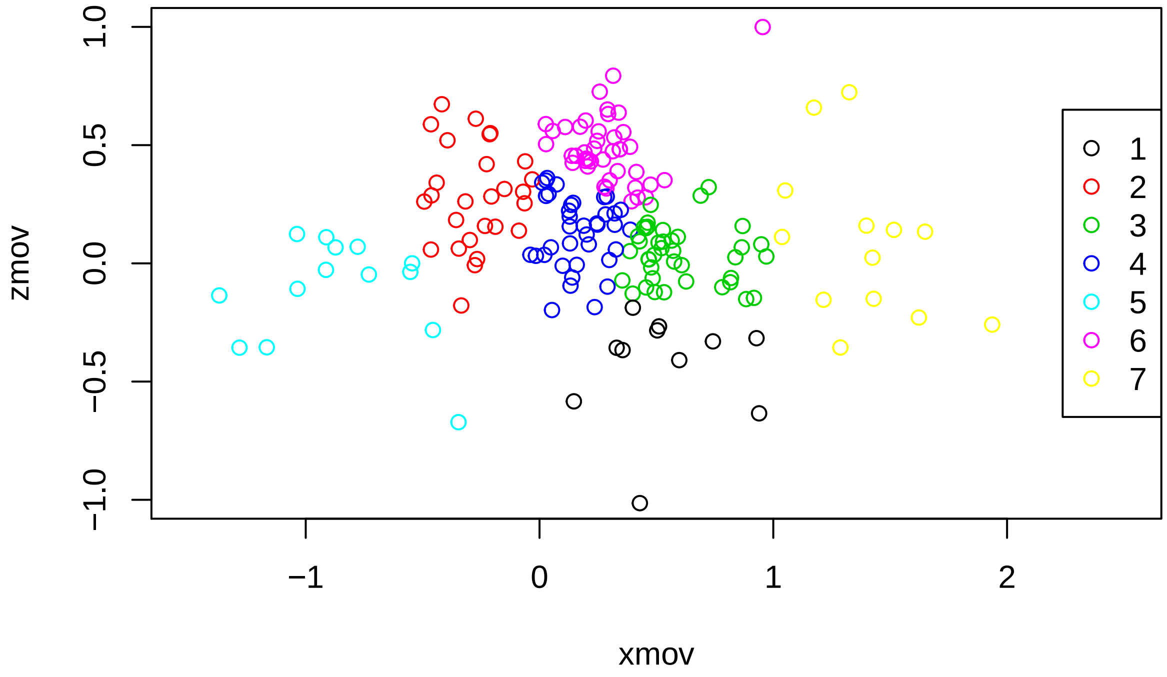
```
clusters <- read.csv("three_clusters_results.csv")
colnames(clusters) <- c("Pitcher", "Clus1", "Clus2", "Clus3")
lefty_df <- clusters[(clusters$Pitcher=="Chris Sale") | (clusters$Pitcher=="Matthew Boyd") |(clusters$Pi
print(lefty_df)
```

```
##         Pitcher Clus1 Clus2 Clus3
## 10    Max Fried     1     0     0
## 27   Chris Sale     1     0     0
## 36 Carlos Rodon     1     0     0
## 46    Brad Hand     1     0     0
## 66 Matthew Boyd     1     0     0
```

We find all three pitchers in the third cluster, confirming our suspicious that these five pitchers have very similarly moving sliders.

Let's do seven clusters and see what that looks like.

```
mov_df <- sliderdf[,3:4]
kmeans7 <- kmeans(mov_df,7)

tbl_7 <- table(sliderdf$value, kmeans7$cluster)
write.csv(tbl_7,'seven_clusters_results.csv')
plot(mov_df[c("xmov", "zmov")], col=kmeans7$cluster, xlim=c(-1.5,2.5), ylim=c(-1,1))
legend("right", legend = paste("",1:7), pch=1, col=1:7)
```



```
clusters7 <- read.csv("seven_clusters_results.csv")
colnames(clusters7) <- c("Pitcher", "Clus1", "Clus2", "Clus3", "Clus4", "Clus5", "Clus6", "Clus7")
lefty_df7 <- clusters7[(clusters7$Pitcher=="Chris Sale") | (clusters7$Pitcher=="Matthew Boyd") |(cluste
print(lefty_df7)
```

```
##          Pitcher Clus1 Clus2 Clus3 Clus4 Clus5 Clus6 Clus7
## 10     Max Fried     0     0     0     0     1     0     0
## 27    Chris Sale     0     0     0     0     1     0     0
## 36 Carlos Rodon     0     0     0     0     1     0     0
## 46     Brad Hand     0     0     0     0     1     0     0
## 66 Matthew Boyd     0     1     0     0     0     0     0
```

You can see that the seven cluster approach factors in the vertical movement a little more, breaking down a lot of those points in the middle of the graph into smaller sections along the Y-axis. It also further segments our chosen left-handed pitchers into different clusters.

Real World Use

While this is an example from baseball, and thus basically non-consequential, we can see the merit in clustering analysis. We are able to segment our data into more useable groups. There are all kinds of situations where it would help to cluster our data points together, that makes it easier to direct whatever strategies or business approaches at that type of data point, rather than each individual data point.

This could be useful for a baseball manager when deciding which players to start on a given day. You want to know how your hitters are likely to fare against the pitcher they are facing that day, but those hitters have almost surely not faced that individual pitcher enough times to give you a big enough sample to learn anything from. If we cluster all pitchers into a handful of groups based on their pitch arsenals, we could then see how a hitter has done against the pitchers in that cluster, and then have a pretty good idea of how they perform against similar pitchers.