High Level Summary:

Players have been clustered in 5 groups. Orange, red, purple, green & blue.

We can split these 5 clusters into two groups. One being players who start & the other being situational players.
Starting groups: Orange, red, purple
Situational groups: Blue and green

Defining each starting group:

Orange players are the best. They are in their prime.
- Overall, they have most production
- They have the above average experience & are a little older
- From this we can infer that they would require the most money to sign to a roster

Red players are the most athletic
- They are on the younger side
- They tend to take more risks
- They are talented & productive, but they are less consistent

Purple players are the most risk adverse
- They are the oldest players
- They tend to take the least amount of risks
- They may not be as athletic or talented as the red group, but they are more consistent

---

From this point, each of these three groups will be referred to by their associated group name (orange/red/purple).
For example, if I say “Orange players are the most expensive” I mean “The best players who have the most production, experience & are in their prime are the most expensive”
Doing this to save word count.

Key Point:
- To keep roster costs down, a team should look to sign red players & then purple players. They should occasionally sign orange players if costs permit. The team should look to fill vacancies which cannot be filled by orange/red/purple with green players.

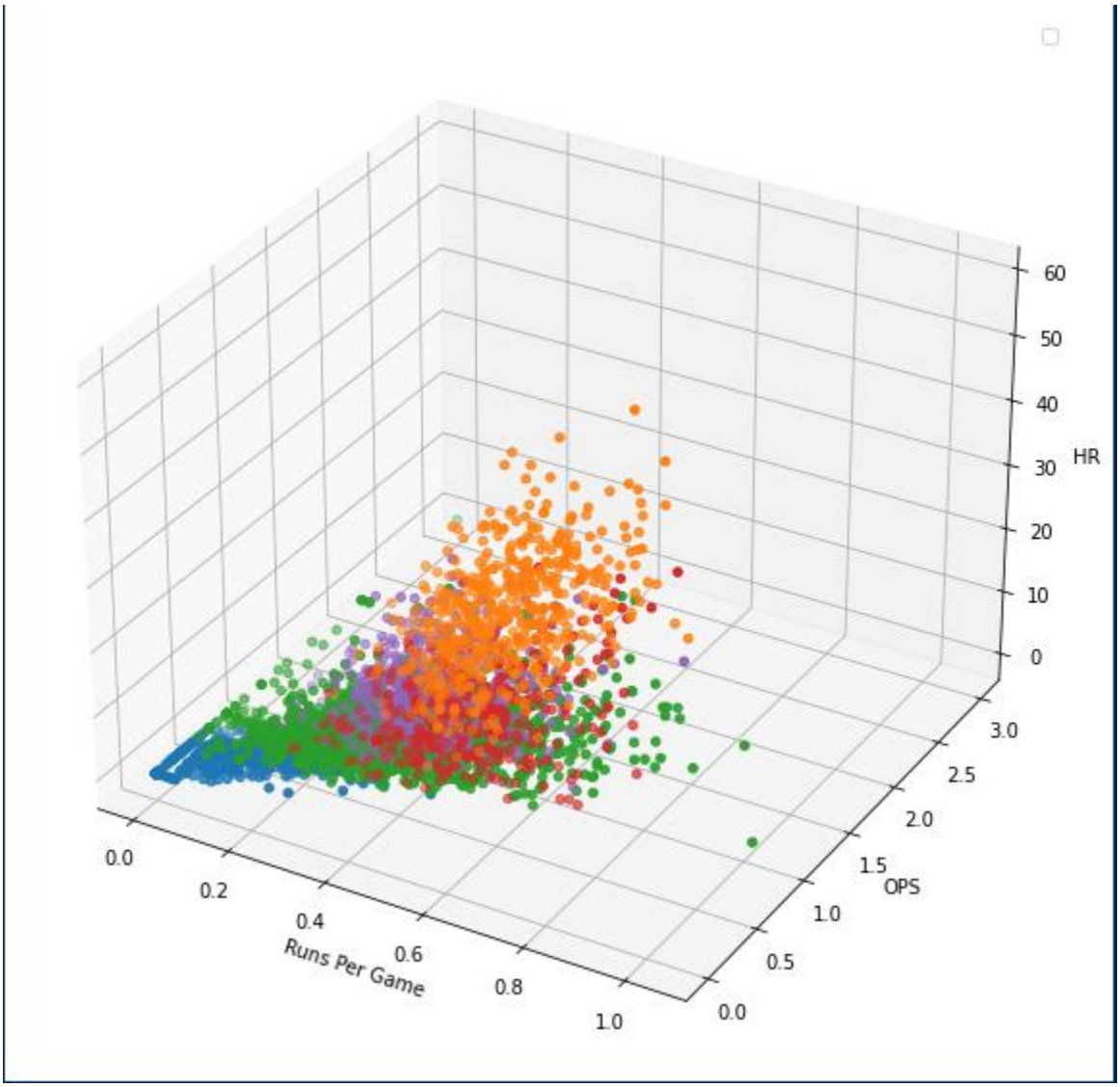
From the team building perspective, orange players will be limited. This is because they only make up 13% of the players and because they would require the most $ to sign since they are in their prime. In a perfect world, we could sign all orange players & have a great team, but that is unrealistic.

Red players are also limited, this is because they are young & ahead of the curve talent wise. They would not cost as much as orange players, but they are more rare to come by.
Knowing this, if the goal is to minimize costs, a team should first look to sign red players then purple players. The team should sign orange players if the opportunity presents itself, but this is unlikely.

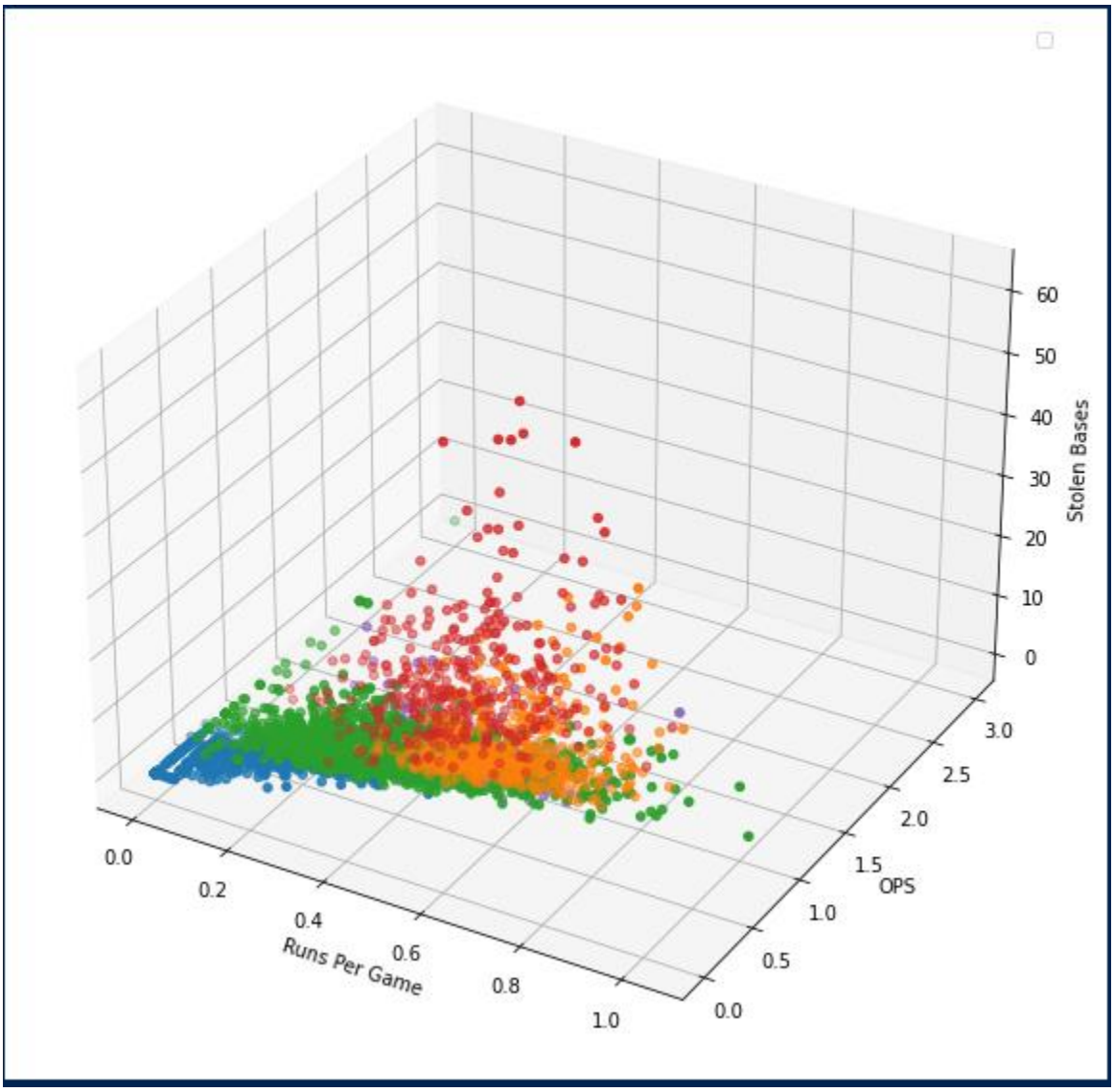
Signing red/purple players would allow a roster to have a balance of young athletic/talented risk takers with less experience who are also less consistent & older, more experienced players who are more consistent and less risk adverse. Purple players could provide roster stability while red players provide roster talent.

The in depth clustering summaries & explanations are on the following pages.

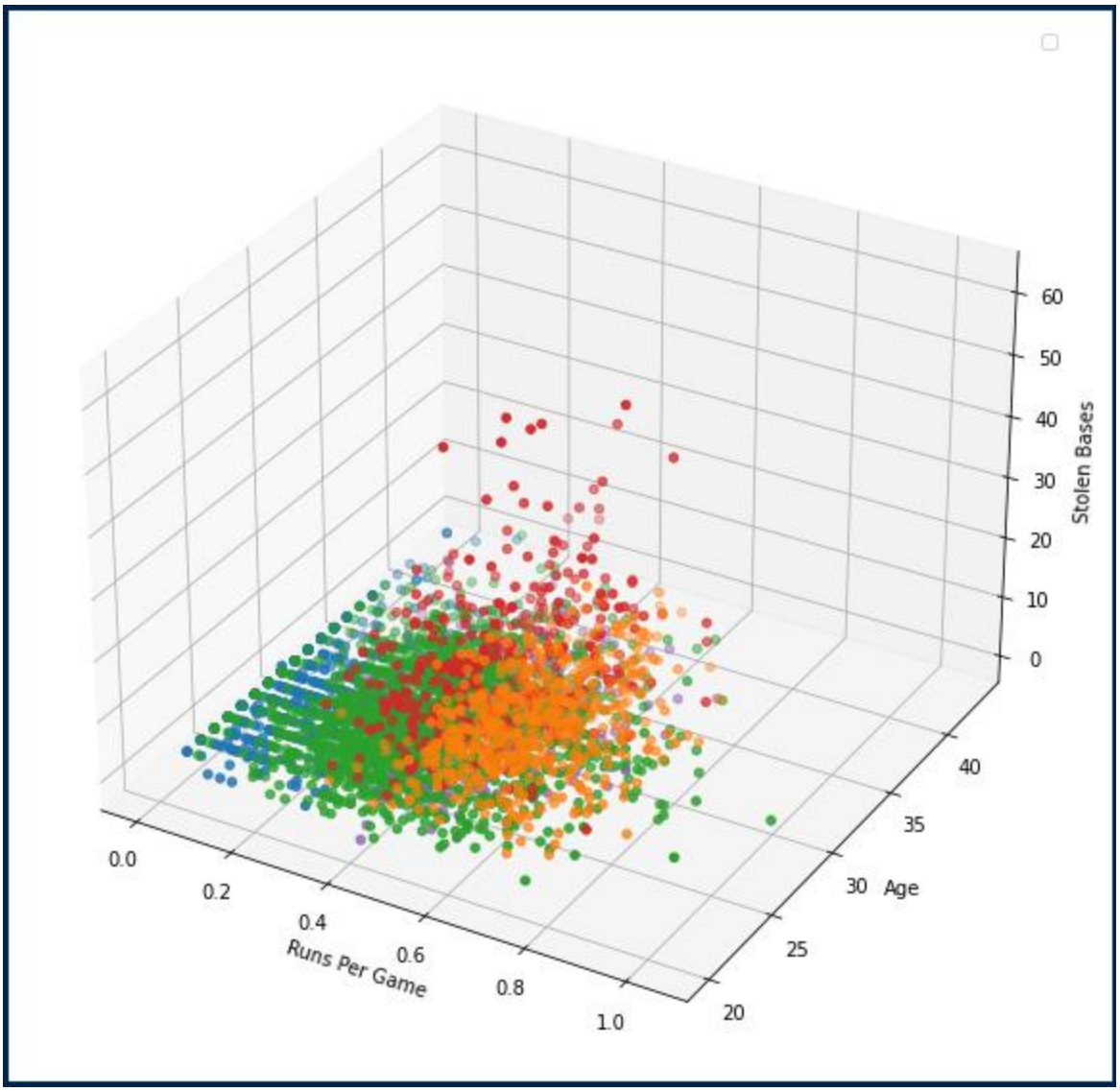
In-depth summary:
We have clustered each player into one of five clusters



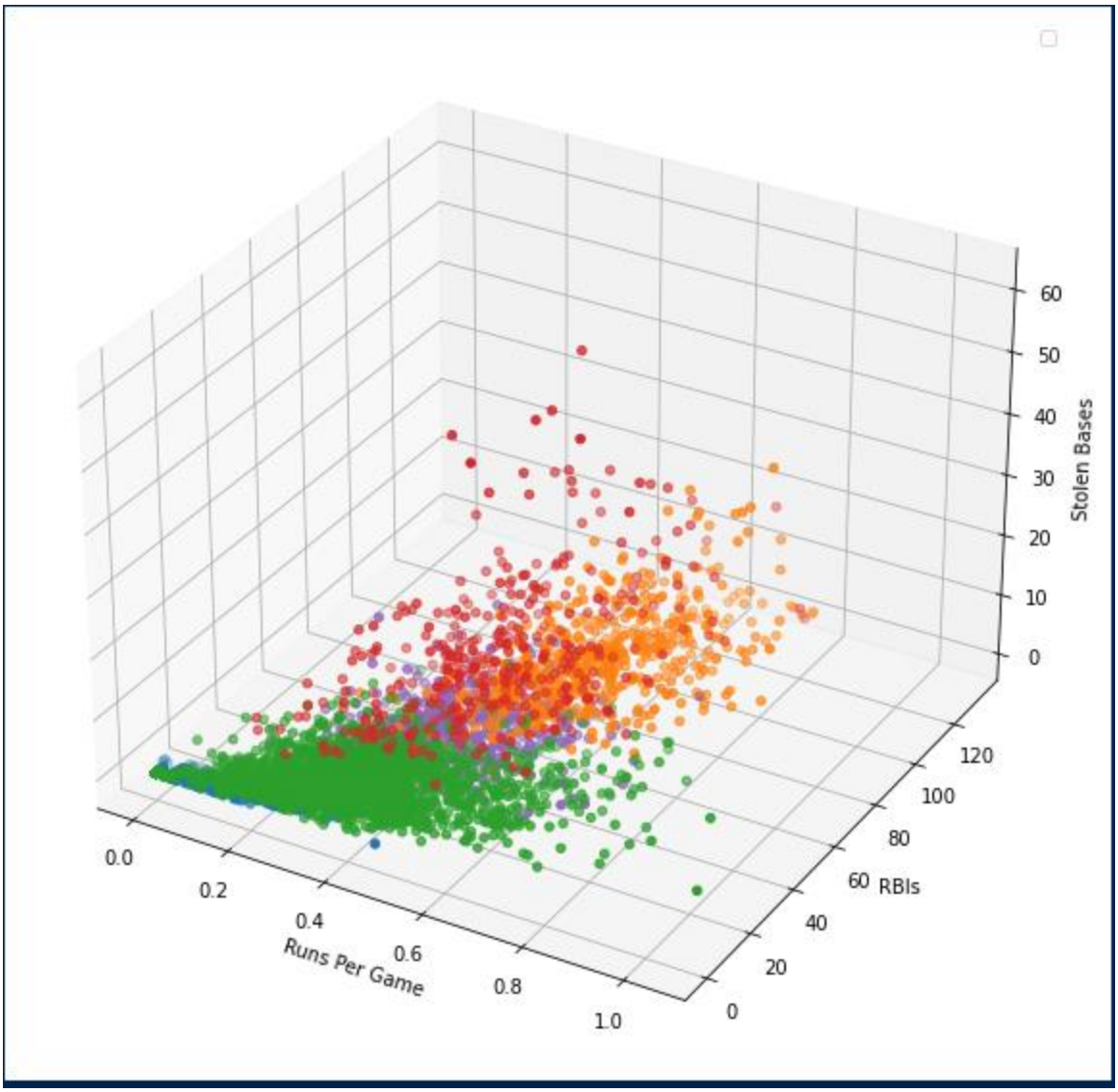
We can see that the orange group is the best regarding RPG, HRs & OPS. We can see that the blue group would be the worse. The orange group tends to have more RPG along with higher OPS statistics & also more homeruns. The red group is similar to the orange group, but they have less homeruns. The green & purple groups seem to be the average.



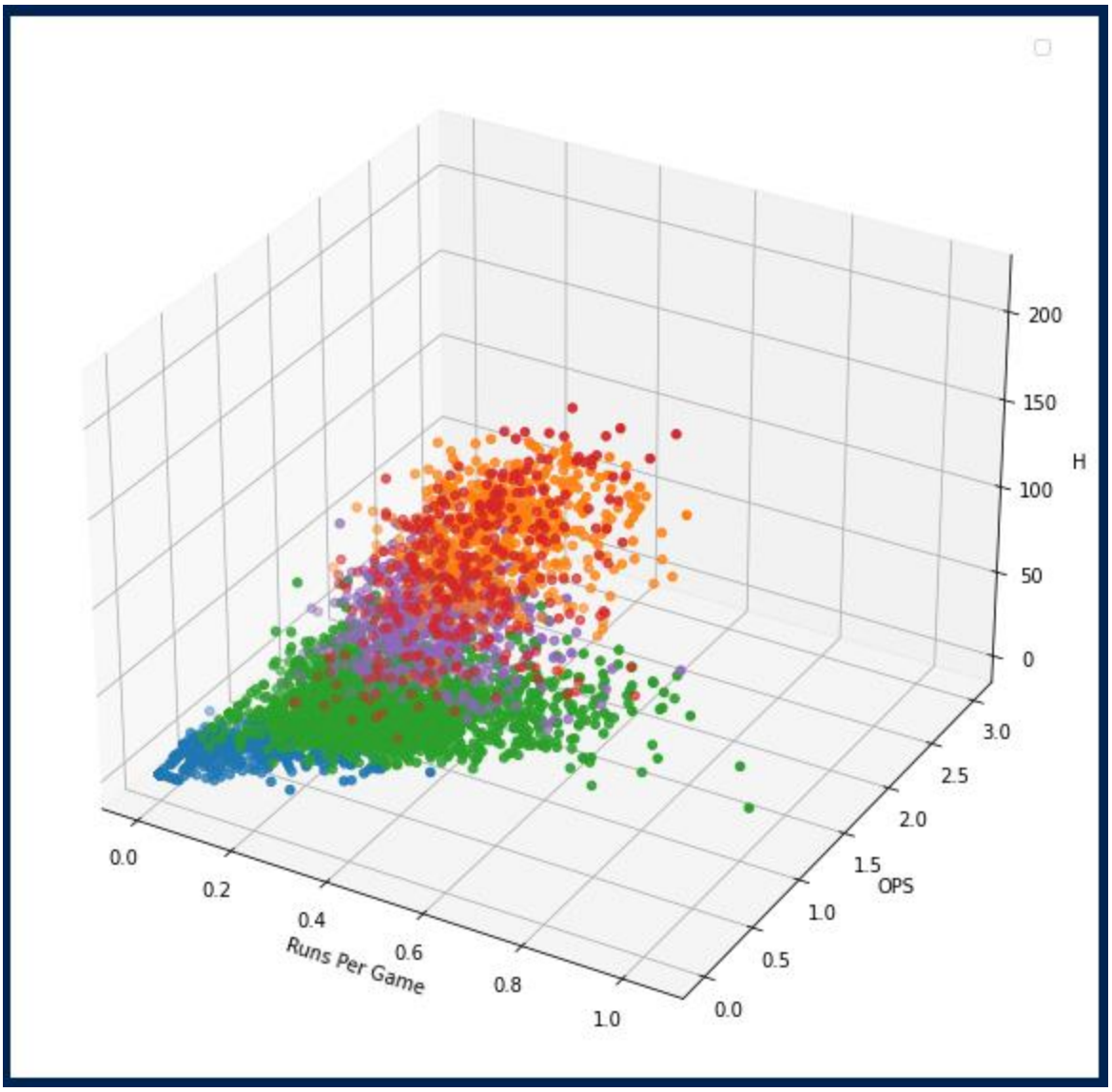
This is a similar graph to our first plot; except we have stolen bases along the y axis. This allows us to see that the red group is recording a higher number of stolen bases among any group, while having similar OPSs & RPGs. The blue & green groups did not change much. The blues still seemed to be the worst & the greens seems to stay at the average.



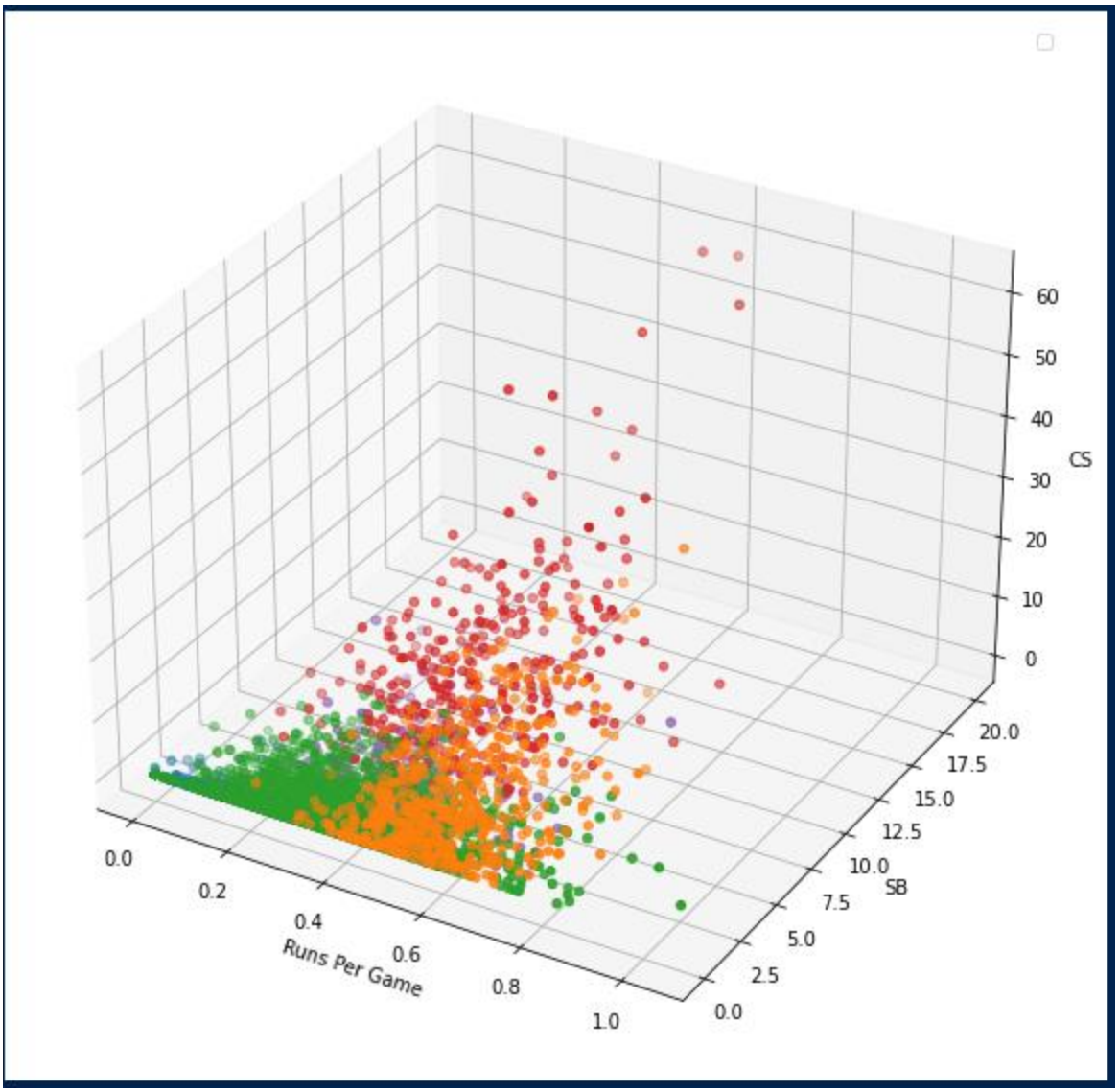
We built off the previous visual by adding Age in the place of OPS. This shows that the ages among groups is quite dispersed & there is no “young” or “old” cluster. Although red does seem to be on the younger side when compared to the dispersion of all other clusters.



This graph is another similar representation, with RPG along the x axis, stolen bases along the y – but we include RBIs along the z. This graph shows us that the green and blue groups have lower RBIS & the orange group tends to have higher RBIs.
Overall, we can see the orange group has higher RPGs & RBIs, but less stolen bases than the red group.



Here we can see that the orange and red groups are both accumulating higher hit totals when compared to the other groups. Followed by the purple, then green & blue. The hit totals for the orange and red groups are similar, but the orange group has a higher OPS. This would be skewed by games played.***



This graph is probably one of the most illuminating since we have a measure for successful stolen bases along with times the player was caught stealing. We can see that the blue and green groups did not change much, but the reds and oranges began to separate. As discussed before, the orange group has a higher RPG value than the red group & the red group has more stolen bases than the orange group.
But the red group also has a much higher caught stealing value. We can interpret this as the red group being higher risk takers, or the orange group being more risk adverse. This may be due to the orange group being older with more experience, but we can verify by further investigating the two groups. When looking at the summary statistics by group, we see that the red group had the lowest average age. So, this could be used to support the statement above that the younger players were less risk adverse, so they stole more bases – thus causing a higher number of caught stealing instances.

Key Items Found in Grouping Summary:

- Ages: Red, Blue, Green, Orange, Purple (Youngest to Oldest)
- Orange & Red played the most games (Orange was the highest)
    - Blue had a very low number of games played
- Orange had by far the most average homeruns
- Reds average stolen bases were triple that of orange
    - Red also had by far the highest average caught stealing instances
- Orange had the highest batting average, followed closely by red
    - Blue had a very bad batting average
- Orange had the highest on base percentage followed closely by red and purple
- Orange had the highest OPS/OPS+ followed by purple then red
- Compared with orange and red, purple had the lowest ground into double play instances. So we can infer purple is rather consistent & conservative.
- Orange had the highest RPG followed by red & purple

From all the above, we can essentially split the 5 clusters into 2 groups and then subgroup from there.

Orange, Red, and Purple are players who get playing time
Green players are situational players, who rack up some playing time
Blue players do not start much.

Regarding the cluster who have a decent amount of playing time

- Orange players are the best
    - They tend to have more experience along with better statistics
    - They are less risk adverse & a little older.
- Red/Purple players are a second best
    - Red players are the youngest
        - Red players tend to take more risks & are talented, but not as consistent as the orange/purple players
        - Red players have the highest average triples, so we can infer that they have power & are more athletic
    - Purple players are the oldest
    - Purple players are the most risk adverse

From a scouting standpoint, a lot would weigh on the contracts $ amount. But we can infer that the orange clusters are the players in their prime who would have the highest $ contracts. Red and purple players would be on the lower side since one group is younger & one group is on the older side. Knowing this, the red & purple groups may make a good combination of young productive risk takers, who may be a little less consistent & older players who are more consistent & take less risks.

Hierarchical Clustering:

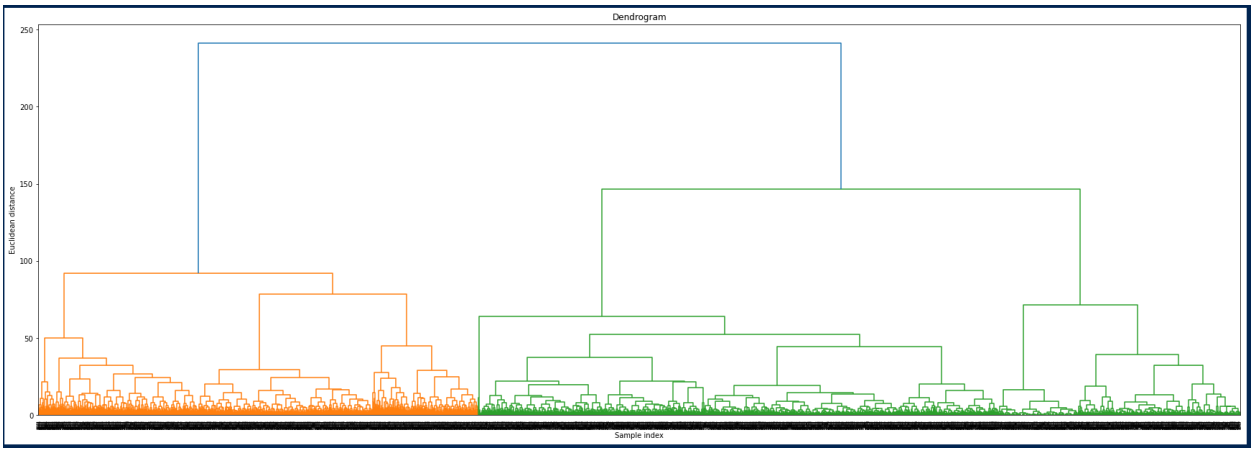
Agglomerative Hierarchical Clustering:
A bottom-up clustering approach. Each observation starts as a “unique” single cluster & then is grouped with similar clusters until all clusters have been merged and there is only one cluster.
This leaves us with a hierarchy of clusters, which resembles a tree-like structure.

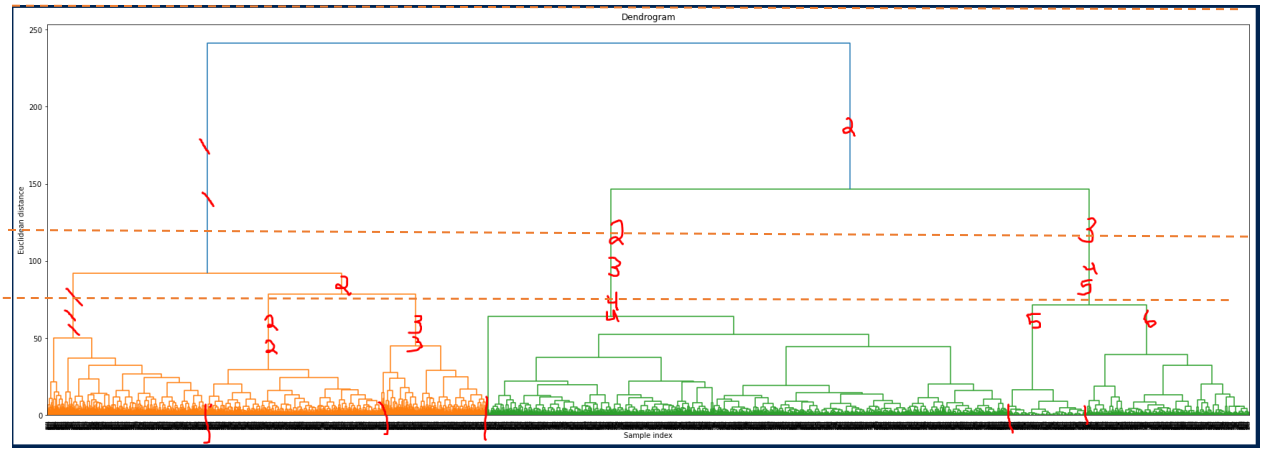
Steps & Walkthough:

- Import the data

- Drop duplicates/Nas
- Remove unwanted features
- Create the runs per game variable from the supervised project
- Remove outliers
- Drop players who did not play
- Standardize the data
    - Since we are going to use distance measurements with clustering
- Separate the data

Create the Dendrogram





How Agglomerative Hierarchical Clustering works:

- The goal is to group observations into similar clusters
- We do this by first, identifying each point as its own cluster
- Then we begin grouping each individual clusters together with similar clusters until we are left with one cluster
- Similarity between clusters is measured by their distance.
- We have already discussed different types of distances such as Manhattan & Euclidean in past projects.
    - The most common and default way to measure distance in Hierarchical clustering is Euclidean distance. (Affinity)
    - This is the metric used to compute linkage

    - This is what we will be using
- Once we have a distance metric selected (Euclidean), we select a linkage criterion.
    - The default linkage used in sklearn AgglomerativeClustering is Ward.
        - For an example, we can think of affinity as "centimeters" and linkage as "knuckle to wrist" hand length measurement for someone's "size of hand". (Thought of this example from the NFL draft & Kenny Pickett's hand size – there are a million ways the NFL can measure someone's hand size)

<u>Euclidean distance & Ward Linkage:</u>

Euclidean distance

- "Straight line" distance between two points.
- Can be calculated by cartesian coordinates, which are just specified unique numerical coordinates on a coordinate system.
- Can be applied to higher n-dimensions, but the foundation of Euclidean distance is $d(p,q) = \sqrt{(p-q)^2}$ for a single dimension
- And $d(p,q) = \sqrt{(p_1-q_1)^2+(p_2-q_2)^2}$ for a two-dimensional space
- And so on for higher n dimensions, we get $d(p,q) = \sqrt{(p_1-q_1)^2+\ldots(p_n-q_n)^2}$
- It also can be expressed in a simpler way
    - If each observation is described by 3 features, each observation can be described in a 3D space, and so on. For N features, we have an N-dimensional space
        - Each observation has a value for each feature (or X_1 to X_n)
        - So, if we have two observations (j & k) & are finding the distance for a given variable 1…n.
            - X_{vj}-X_{vk}: Or, the value of feature V for observation j minus the value of feature V for observation k.
            - Then we can write Euclidean distance as
            - $D_{jk}^2 = \sum(X_{vj}-X_{vk})^2$ -> squared, so we would have the take the sqrt after.
                - This is saying the distance^2 between observations j & K is equal to the sum of all of the squared differences across the respective n dimensions.
                    - Where n dimensions is defined as $X_1 \ldots X_n$
                        - Where X is the feature value & 1…n are the features
            - This is how Euclidean distance is measured & is the metric used in Ward's Linkage discussed below.

Wards Linkage

- Instead of directly measuring distance, as the other parameters (complete, average, single) would, Ward analyzes the variance of clusters.
    - The variance is expressed by the sum of squared errors (distance from the center of a cluster squared) for each cluster
    - Ward's method says that the distance between two clusters, A and B, is how much the sum of squares will increase when we merge them. (stat.cmu.edu --- Murphy 2004)
        - In Ward's method, each unique cluster has a center (M) for N points in the cluster.
        - Each of these N points will have a distance from the center M
        - Each distance will have a Squared Distance

        - And for all the N points in the cluster, there will be a Sum of Squared distances from center point M
        - This will be that unique clusters sum of squares
        - To calculate distance between two clusters (A & B), we start with each cluster's sum of squares.
        - We then combine the clusters A & B, to form cluster C. Now, we can find cluster C's center & compute the Sum of Squares for all N points in cluster C
        - Distance between A & B is then measured by calculating the increase in Sum of Squares from A & B to cluster C
- With hierarchical clustering – the sum of squares starts at zero since each observation is its own cluster.
- In our case, the distance terms used above are measured using Euclidean distance
- Sklearn will merge cluster pairs to minimize the linkage value & do this continually, providing us with a Dendrogram
- From there we can determine the appropriate number of clusters and group the observations accordingly
- Then we can use the newly categorized observations & extract meaning