Application of K-Means Clustering for Analog Game Attributes

Neil Opitz

Bellevue University

Abstract

This paper explores the composition of groupings for board game attributes and the group relationships to subjective ratings for board games. The data used is comprised of ratings and attributes from a database of over 100,000 game titles. Attributes include hard attributes such as publisher, designer, and component weights as well as soft attributes such as category, mechanic, and rating. Results suggest that specific attributes such as the theme of the game and the mechanics utilized in the design of the gameplay have an impact on user assessments.

Keywords: k-means clustering, explanatory data analysis, analog gaming, board game geek

Application of K-Means Clustering for Analog Game Attributes

The popularity of analog gaming (aka board games) has shown rapid growth since the turn of the century and board game sales have continued to increase (Booth 2020). Due to both market interest and personal interest, the goal of this project is to discover attributes of board games can be used to identify meaningful groupings of games that serve to amplify similarities of attributes within the groupings as well as amplify differences without the groupings. I will be using a K-means clustering algorithm for this purpose. What I hope to understand after this analysis is if there are clusters of board games that are grouped by specific attributes in the design that perform better for subjective assessments, and if so, can attributes in the groupings be identified that are associated with better subjective performance. Some of the attributes I am interested in researching include game designers, games that are successfully funded on the Kickstarter funding platform, and can this methodology assist in identifying board game 'flops' (defined as a board game that received a high number of ratings in a short amount of time but performed poorly in a subjective assessment score). An additional hypothesis I have for this dataset is that the mechanics that a game utilizes is more impactful on a game's ratings than the theme used for the game (the theme is the subject that the game uses to tell the story and/or design the components).

Data Source

There are several contributing factors that contributed to the growth of the board game industry, including improvements in manufacturing and shipping and evolution in game design mechanics. Also, the coronavirus pandemic is attributed with a significant increase in sales since March 2020 as households search for ways to pass the time while spending more time at home (Booth 2020). Arguably, the internet has been the single-greatest contributor to the increase that this industry has experienced. With the growth of the internet, board game publishers were able to reach an increasing population of experienced and new board game players. The interest in board games also spawned websites dedicated to enthusiasts and collectors. The most popular website for board game enthusiasts is board game geek (boardgamegeek.com). Board game geek is a website where users and publishers can create a webpage for a game (each game has a single webpage) and users can add their personal rating of the game and any comments associated with their rating. Games are rated on a scale from 1 to 10 with a higher rating indicating greater satisfaction levels with the game. In addition to game ratings, the board game geek website has data for the number of players, recommended minimum age for a player, and the expected game duration. The website also has data for a game's publisher, designer, and artist as well as date of first publishing, game mechanics and general categories that describe the game. All in all, the data collected for each board game on the website is substantial and descriptive of objective physical game attributes as well as subjective game assessments. A dataset containing this board game data is available on Kaggle (https://www.kaggle.com/seanthemalloy/board-game-geek-database). The Kaggle board game geek dataset contains data on more than 125,000 different board and card games. I will be using this dataset to identify clusters of games with similar attributes and comparing these clusters in an attempt to identify attributes outside of those utilized for clustering that assist in defining the grouping. I will be using a k-means clustering algorithm to identify the unique clusters. I also use a linear regression algorithm in the project to identify the relationships between variables identified in the groupings.

K-Means Clustering

K-means clustering is an unsupervised machine learning algorithm that uses distance from a central point to define membership in a cluster (Kumar 2019). The 'K' in K-means indicates the number of clusters into which the user desires to partition the data. After defining the number of clusters, the algorithm randomly selects k points (called centroids) in the data and measures the Euclidean distance between the point and each other point in the dataset. Points in the data become members of the centroid nearest to them. The algorithm iterates through different centroids with the goal of minimizing the distance between the k-centroids and member data points.

Exploring the Dataset

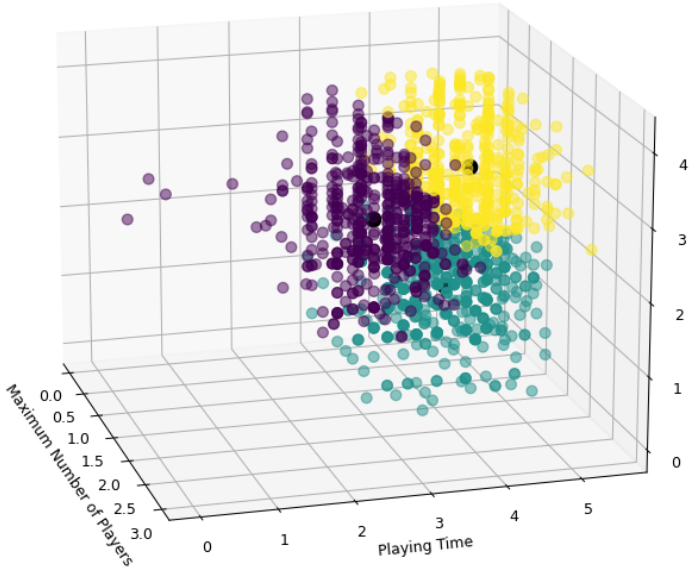
I have been playing board games for over forty years and have been a collector for almost thirty years. I also designed an award-winning game published by Gamewright games. I mention these points only to add credence to my claim of being a subject matter expert in the subject of board games. I used a Jupyter Notebook to open the data and store the dataset in a pandas dataframe. The pandas dataframe stores the data in a two-dimensional structure, includes the label associated with each variable, and allow for efficient utilization and manipulation of data using python commands (Stojiljkovic). A pandas dataframe is an ideal structure for performing exploratory data analysis on a dataset, which is the next step in the process. There are several methods in pandas that can be used to do initial exploration to get a better understanding of the dataset as a whole as well as for each individual variable. I used the info() method provides the size and number of columns and types for each column while the describe() method provides counts, means, and quartiles for the individual numeric variables found in the data (Rodriguez 2020). The dtypes attribute in pandas will provide detail as to what type each variable belongs to such as float, object, and integer. While there are several numeric variables in the dataset, the primary numeric variables that I am interested in for my analysis include the game rating, minimum and maximum number of players, playing time, number of ratings, and year published (I created a 'game age' variable to replace year published). I first created histograms for each of the numeric variables. The histograms revealed the bulk of games allow fewer than 20 players and have a playing time of 6 hours or less and are less than 75 years of age (publishing age). There are many games that were published more than 20 years ago that have served to inspire the design of modern games through re-purposing of mechanical game elements. After reducing the dataset based on these parameters and also removing games that have a small sample of ratings (less than 30) the dataset was reduced to approximately 23,000 cases. The removal of games with fewer than 30 ratings was the largest contributor to the reduction of the dataset. Retaining games in the dataset with a small sample of ratings could add bias to the ratings analysis. Reducing the dataset by 80% assisted in removing the outliers within each of the numeric variables. With the exception of game rating, these variables have a distribution that resembles a lognormal distribution. I used numpy.log to perform a lognormal transformation on these variables in an effort to work with distributions that more closely resemble normally distributed data. At this point I began exploring relationships between the numeric variables. I created scatterplots for each numeric variable pairing and also calculated a correlation matrix to explore these relationships. Some relationships were initially evident, including between a game's rating and its age, and also between a game's expected duration and its rating.

Analysis of the relationships led me to determine that an additional dimension would be required in order to generate more separation in the data points.

Clustering Algorithm

For the clustering algorithm I am using the KMeans cluster module from the sci-kit learn library which uses a random initialization of the k centroids as determined by the user (Versloot 2020). Random initialization indicates that the algorithm randomly assigns k points in the data to serve as the initial centroids. After this initial selection the algorithm will move each centroid iteratively until an optimum is achieved in which the differences between each point in a cluster and the centroid is minimized (Versloot 2020). I determined that the three numeric variables I would use for the K-means clustering algorithm would be the maximum number of players, expected game duration, and the age of the game. Review of the relationships found in the scatterplots during exploratory data analysis facilitated my decision in opting to use three dimensions and also which three variables to include. The decision to use three clusters was based on results from the elbow-method for different pairings among the numeric variables. In addition to using three variables I also opted to use three centroids in the K-means algorithm. I tested using 4, 5, and 6 centroids, however I found three to provide the greatest visible differentiation in the scatterplot. For plotting a 3-dimensional scatterplot I am using the matplotlib mpl-toolkits mplot3d extension. This module creates an interactive 3-dimensional scatterplot that allows a user to rotate the chart when used in a browser. The interactivity provided by mplot3d is very useful for exploring the relationships and cluster membership (Seif 2019). The scatterplot derived from the K-means clustering algorithm is shown below with the three clusters clearly differentiable due to the colors.

Figure 1

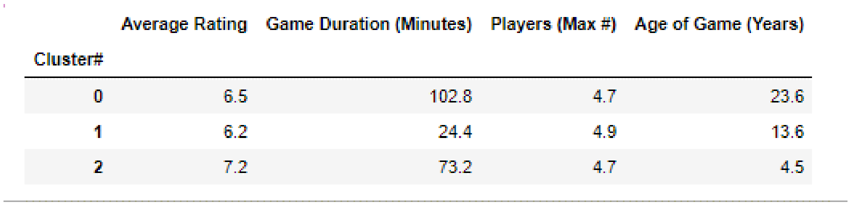


3-Dimensional scatterplot of K-means clusters. A reduced dataset of 1500 random cases was utilized for this visualization to assist in more easily differentiating the clusters and centroids.

In addition to providing a visualization of the clusters, the KMeans module creates a numpy array of the cluster values for each case included as part of the algorithm. This is how the algorithm is able to associate each data point to each cluster. The cluster identifiers are going to be necessary for the additional analyses that will be performed on the dataset based on the cluster membership determined by the algorithm. The numpy array that contains the cluster value can easily be appended to the dataframe from which the data points for the clustering were derived. To add the array of cluster identifiers to the dataframe, I first converted the array to a list using the tolist() method, converted the list to a dataframe, then merged the dataframe to the existing dataset dataframe. Now that the cluster identifiers are in the dataframe, all the variables for each case in the data is now associated with a specific cluster value. Having the cluster values in the dataframe allows for a comparison of attributes in each cluster that were not utilized in the clustering algorithm. One of the primary attributes that I am interested in researching is the subjective rating for the games within each cluster. Using the groupby() method in pandas, I am able to group the entire dataset into three groups based on the cluster values, then calculate an average value of the user ratings for the clusters:

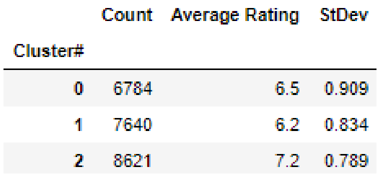
cluster_average = data.groupby(['clusters'])['average'].mean()

The results of the grouping are interesting as user ratings are significantly different when comparing the clusters:



Cluster#	Average Rating	Game Duration (Minutes)	Players (Max #)	Age of Game (Years)
0	6.5	102.8	4.7	23.6
1	6.2	24.4	4.9	13.6
2	7.2	73.2	4.7	4.5

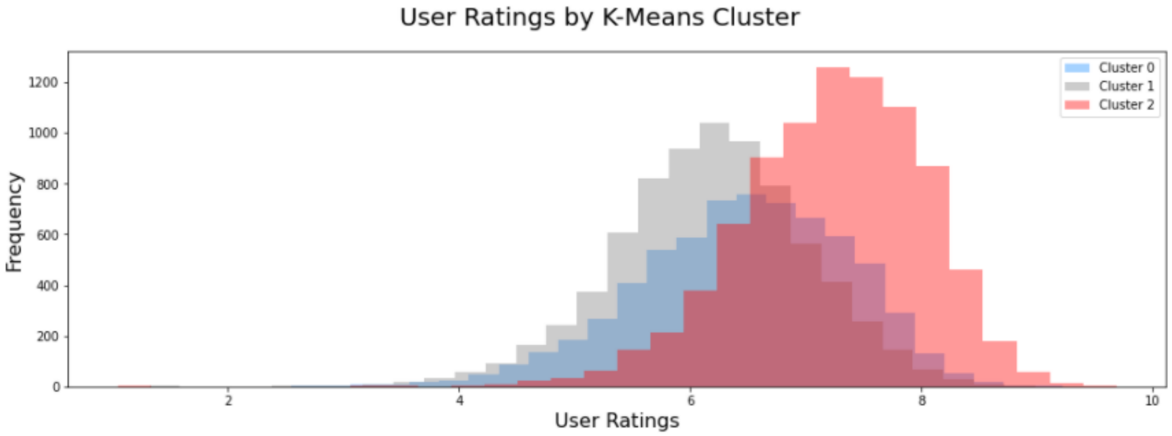
The games with membership in Cluster 2 have an average rating that is significantly higher compared to Clusters 0 and 1. To confirm that the difference in scores is significant, I utilized an independent samples t-test that is in the scipy.stats module. This module performs a two-sided test which utilizes the sample size, mean, and standard deviation for each mean to test if the two independent samples have identical averages for the null hypothesis (1). The results for the p-values for each pairwise comparison is <.05, therefore the hypothesis that the means are not different is rejected for each pairing of means. The results of the t-test can be found in the Appendix. The data in the table below can be used to perform the t-tests.



Cluster#	Count	Average Rating	StDev
0	6784	6.5	0.909
1	7640	6.2	0.834
2	8621	7.2	0.789

**The p-value for comparison of each mean pairing is <.05*

Further investigation into the difference in mean scores reveals that there is clear separation of the center for the distributions of the user ratings for each of the clusters. The histogram below, created with the alpha for each grouping set to 0.4, allows us to see how the ratings are distributed for each cluster.



In addition to user ratings, other attributes contained in the dataset can be analyzed using the groupby method. A table can be created in pandas that displays the averages for each of the variables that were used in the clustering algorithm. The table below includes these variables along with the average rating. It can be discerned from the results that there is a large variance for game duration and age of game compared to maximum number of players.

We can also make some generalizations from this table that games with a shorter duration (less than 30 minutes) or longer than 80 minutes are associated with lower ratings while games at the 70-minute mark have the highest. The table also reveals that newer games (game age less than 5 years) are associated with higher ratings. These results seem to indicate that game designed and published since 2015 are subjectively better games than those published prior to 2015 based on user ratings. Analysis of data grouped based on age (less than or equal to 4.5 years of age or greater than 4.5 years) indicates higher ratings but similarity in duration and number of players.

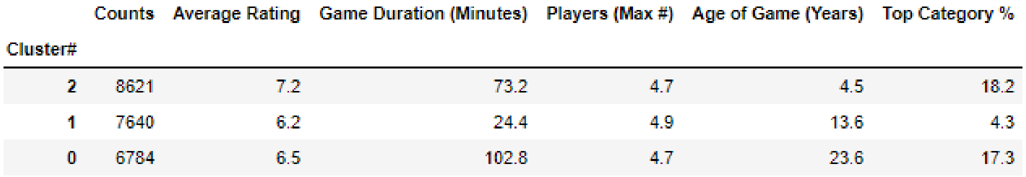
Attribute Analysis

There are several other board game attributes in the dataset that can be include when comparing clusters. Other attributes include the number of ratings, categorical rankings, designer, artist, game category, game mechanics, game family, publisher, and any awards received. Several of these attributes are likely highly associated with a games rating including categorical rankings and awards, so these variables will be excluded from further analysis of the clusters. As age of a game is among the variables utilized for clustering, I would anticipate there to be differences in the relative proportions of publishers, designers, and artists across the clusters as the contents of these variables will naturally shift over time. Analysis of the 'family' variables in the dataset indicate that there are many board games that have their own 'families'. For example, a very popular board game called 'Advanced Squad Leader' has about 60 games in its family that have been published as a result of the popularity of this game. Additionally, based on the number of observations, two of the top three families of games are 'Better Description Needed' and 'Unreleased Games'. I have therefore opted to exclude analysis on the 'family' of the game. The remainder of the analysis, therefore, will include the following attributes: game category, game mechanics, and number of ratings.

There are two primary attributes that are frequently referenced when discussing a game, this includes any discussion of a game such as a game review from a media source or media from a publisher, these are the ‘theme’ of the game and the ‘mechanics’ of the game. The majority of any media about a game will include these two attributes. This is because these two topics along with the game components developed to communicate these topics and the rules by which interaction with components and players occur, encompass the entirety of the interaction between the player and the game. Either of these attributes can ‘make or break’ a game. A designer or publisher who wants to develop a quality game will consider both of these attributes during the design and printing process. A game with a popular theme but poor mechanics, while it may have strong sales due to the publisher and the target audience, will not fare well in board game geek user ratings. A game with strong mechanics and a poorly tacked on theme will not suffer as much in board game geek user ratings as mechanics have a stronger association with user ratings (see Mechanic Analysis).

Category Analysis

In the board game geek dataset, there are more than fifty different categories for games with at least 30 observations. The categories with the highest associated user ratings include: ‘Expansion for Base-game’, ‘Book’, ‘Civilization’, ‘Miniature’, ‘Adventure’, ‘Environmental’, ‘Napoleonic’, ‘Age of Reason’, ‘City Building’, and ‘Collectible Components’. Each of these categories is associated with an average rating of 7 or higher. For purposes of this analysis I have excluded ‘Expansion for Base-game’. In order to determine the frequency of games in each cluster that belong to one of the top-rated categories, a new variable was created and set to zero. A loop was then utilized to change the value for this new variable to 1 if the game category belongs to one of the categories associated with an average rating of 7 or higher. Using the groupby method I calculated the frequency of games within each cluster that contain a 1 for the new variable based on the total count of games in each cluster. The table below shows the results for this process.

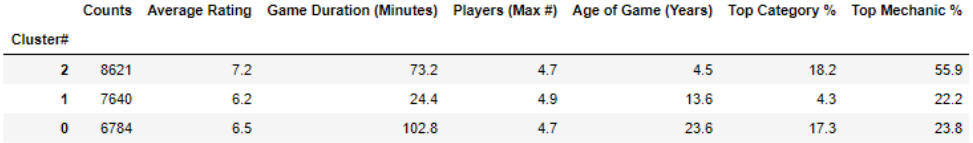


Cluster#	Counts	Average Rating	Game Duration (Minutes)	Players (Max #)	Age of Game (Years)	Top Category %
2	8621	7.2	73.2	4.7	4.5	18.2
1	7640	6.2	24.4	4.9	13.6	4.3
0	6784	6.5	102.8	4.7	23.6	17.3

The results indicate that the clusters with the top two average scores (Clusters 0 and 2) each have an approximately 17% to 18% rate of games that belong to one of the top rated categories. This is a much higher rate compared to the lowest performing cluster (Cluster 1 with a 6.2 average rating). Details for Cluster #1 indicate that just over 4% of games in this cluster belong to one of the top-rated categories. While this is an interesting finding, it does not fully explain why Cluster #2 performs significantly better than both of the other clusters. The next step will be to determine if the mechanics which a game utilizes are associated with a difference in the performance among the clusters.

Mechanic Analysis

In the dataset, there are approximately fifty different game mechanics with at least 30 observations. The mechanics associated with the highest user ratings include: ‘Action Retrieval’, ‘Worker Placement’, ‘Campaign/Battle Card Driven’, ‘Action Points’, ‘Cooperative Game’, ‘Deck, Bag, and Pool Building’, ‘Communication Limits’, ‘Action Queue’, ‘Chit-Pull System’, ‘Area Majority/Influence’, and ‘Card Drafting’. Each of these mechanics is associated with an average rating of 7 or higher. Using the same methodology for game mechanics as was used for game category, a new variable was added to the dataset with an initial value of zero. The value for each case was updated to a value of 1 if the game mechanic for each case matched a mechanic that is associated with an average rating of 7 or higher. The groupby method was then applied to calculate the frequency of cases within each cluster that utilize a top-rated mechanic. The table below shows the results for this process.



	Counts	Average Rating	Game Duration (Minutes)	Players (Max #)	Age of Game (Years)	Top Category %	Top Mechanic %
Cluster#							
2	8621	7.2	73.2	4.7	4.5	18.2	55.9
1	7640	6.2	24.4	4.9	13.6	4.3	22.2
0	6784	6.5	102.8	4.7	23.6	17.3	23.8

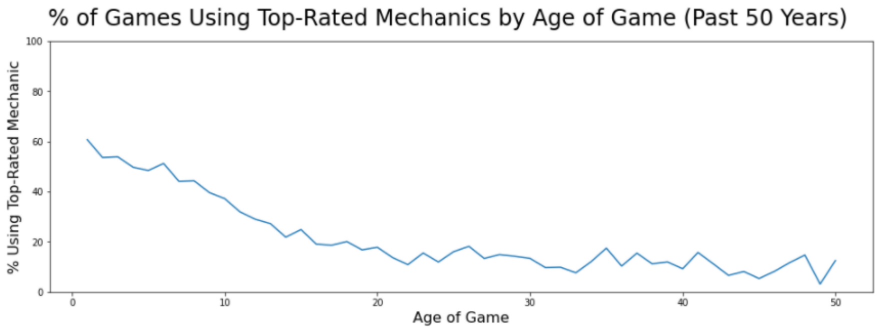
The table indicates that more than half of games in the cluster with the highest average game rating (Cluster 2) utilize a top-rated mechanic: 56% of Cluster 2 games utilize a top-rated mechanic compared to 24% of games in Cluster 0 and 22% of games in Cluster 1. This is a significantly higher proportion compared to Clusters 0 and 1. This finding strongly supports the hypothesis that the mechanics designed into a game have a greater impact on user ratings than does the theme selected for the game.

To further validate these findings, multiple linear regression was utilized in the Jupyter Notebook. I imported the LinearRegression module from the sklearn.linear_model package (2). The LinearRegression module can perform both simple and multiple linear regression. As I am using both category and mechanics as explanatory variables to predict the user rating outcome I will be using multiple linear regression for this purpose. After converting both of the explanatory variables to arrays, I fitted the model to calculate the coefficients. The results indicate that the mechanics for a game have nearly twice the impact on a game's rating compared to the theme. The coefficient for category is 0.39 compared to mechanics at 0.66. Based on the apparent relationship between the age of the game and user ratings, I ran an additional multiple linear regression and included the game age variable. Adding in the age of the game served to both reduce the impact of the mechanics variable (the impact of mechanics is still larger than category) and double the explanatory power of the model. With these three variables in the linear model, the regression formula is:

$$\text{Expected Rating} = 6.8 + .37(\text{Category}) + .47(\text{Mechanic}) - .03(\text{Age of Game})$$

A game with a top-rated category will increase the rating by about .4 points while a game with a top-rated mechanic will increase the score by about half a point. Each year that a game ages reduces it's score by approximately .03 points. The complete results of the linear regression can be found in the Appendix. Further analysis indicates that the mean age of mechanics that are not among the top-rated mechanics (15.7 years) is nearly twice that of top-rated mechanics (8.5 years). The averages were calculated based on the age of the game that uses the mechanic. There are many additional variables in the dataset that could be utilized in a multiple linear regression model. The addition of these variables is beyond the scope of this project.

With the advent of personal computing, the availability of custom component printing on standard and 3-d printers, and the assistance of crowdfunding sites such as Kickstarter, the number of games being produced by self-publishing methods and by publishing companies of all sizes has exploded over the past decade. Elements of game design including mechanics, components, and art have evolved as part of this explosion and several new and popular game mechanics have been introduced into board games such as cooperative gaming, communication limits, and deck building. The chart below reveals that the game mechanics with the highest associated ratings shows decline as the age of a game increases (meaning that the top-rated mechanics are used to a much larger degree with games that have been published more recently). As stated earlier, approximately half of the games published since 2015 utilize a top-rated mechanic compared to about ~30% from 10 years ago ~20% from 20 years ago.



Conclusions

Board games have many attributes that can be considered when determining the qualities that make a game desirable. Developing groupings of board games based on the maximum number of players, the expected duration of gameplay, and the number of years since first publishing has served to highlight interesting characteristics within the clusters. Board game ratings are subjective and influenced by many factors. The duration of a game may be a barrier to play for some players while thematic elements of a game may be undesirable for another group of players. Also, the access to higher quality graphics and component design for modern games may influence ratings for games that were published before these technologies were available. A game that is designed with both a popular theme in mind while using top-rated mechanics for gameplay will likely generate a higher rating on the board game geek website.

References

Booth, Paul (December 2020). *What's Old Is New: Board Games Can Be a Lifeline in Lockdown*. Retrieved from https://www.usnews.com/news/health-news/articles/2020-12-24/board-games-can-be-a-lifeline-in-covid-lockdown

Stojiljkovic, Mirko (Undated). *The Pandas DataFrame: Make Working With Data Delightful*. Retrieved from https://realpython.com/pandas-dataframe/

Rodriguez, Melissa (July 2020). *How to Summarize Data with Pandas*. Rerieved from https://medium.com/analytics-vidhya/how-to-summarize-data-with-pandas-2c9edffafbaf#9b41

Kumar, Abhishek (August 2019). *K-Means Clustering*. Retrieved from https://towardsdatascience.com/k-means-clustering-13430ff3461d

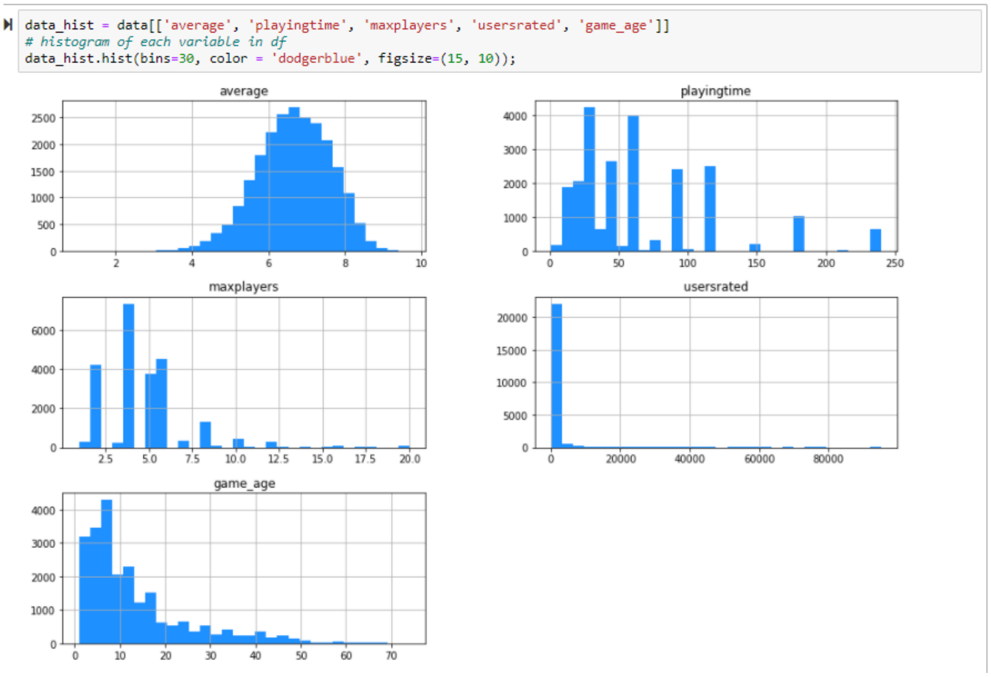
Seif, George (May 2019). *An easy introduction to 3D plotting with Matplotlib.* Retrieved from https://towardsdatascience.com/an-easy-introduction-to-3d-plotting-with-matplotlib-801561999725

Versloot, Christian (April 2020). *K-means Clustering Tutorial: example with Scikit-learn.* Retrieved https://www.machinecurve.com/index.php/2020/04/16/how-to-perform-k-means-clustering-with-python-in-scikit/

1. https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_ind_from_stats.html
2. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html

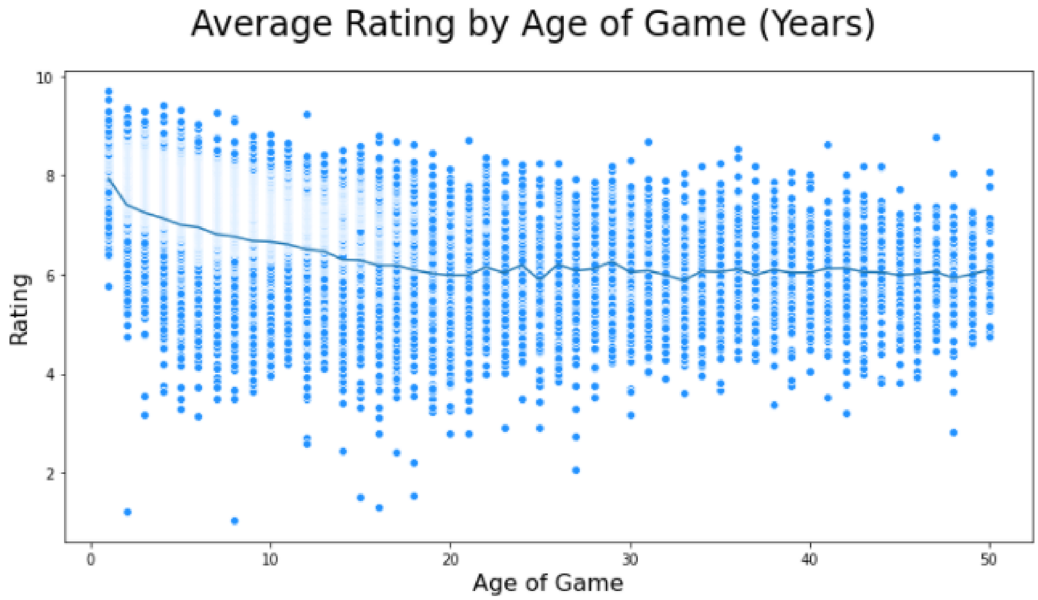
Appendix

Histograms of primary numeric variables in dataset

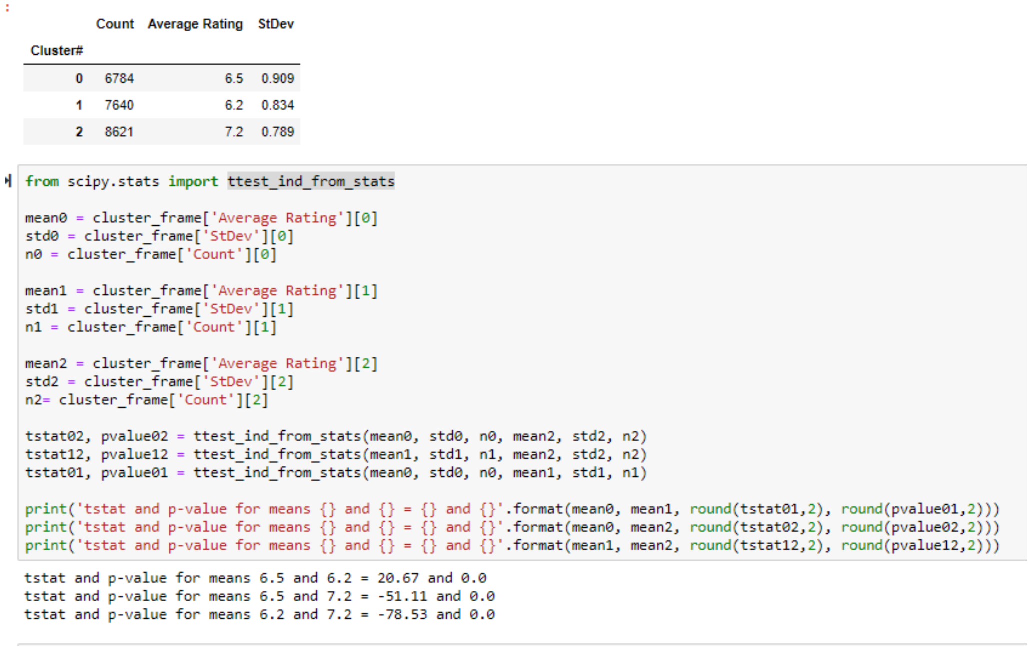


```
data_hist = data[['average', 'playingtime', 'maxplayers', 'usersrated', 'game_age']]
# histogram of each variable in df
data_hist.hist(bins=30, color = 'dodgerblue', figsize=(15, 10));
```

Scatterplot of Game Age and User Rating



Jupyter Notebook t-test code and results

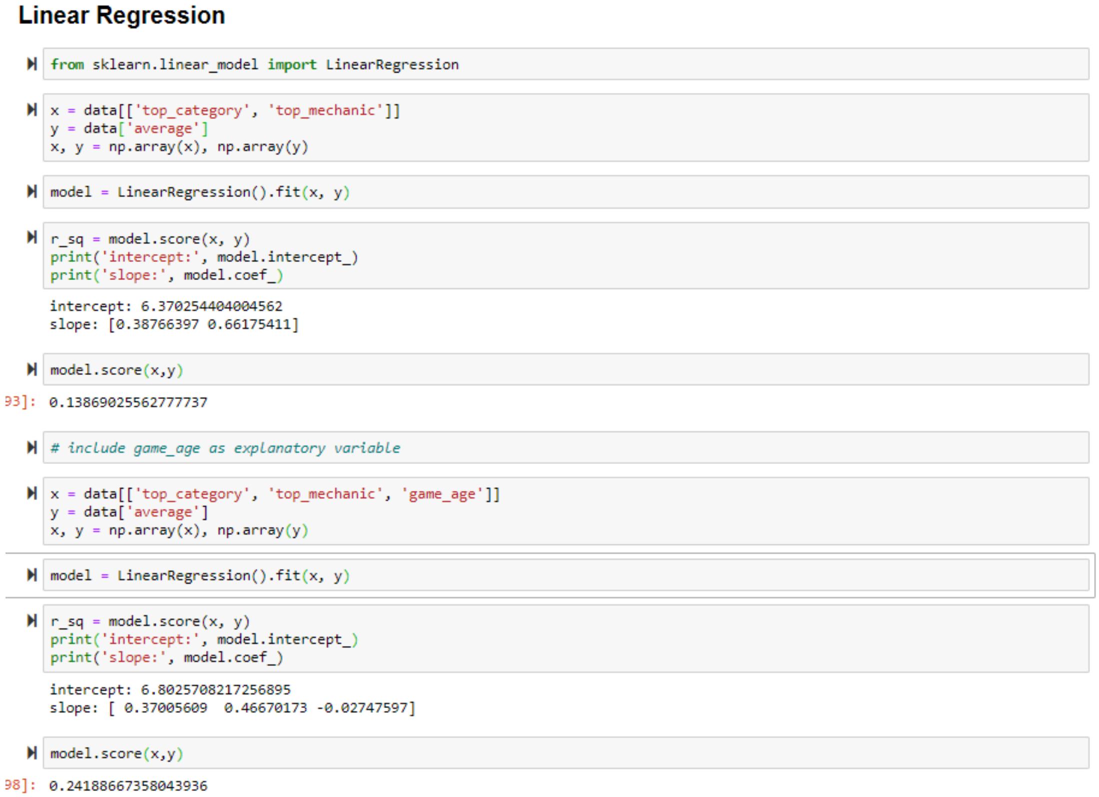


Cluster#	Count	Average Rating	StDev
0	6784	6.5	0.909
1	7640	6.2	0.834
2	8621	7.2	0.789

```
from scipy.stats import ttest_ind_from_stats

mean0 = cluster_frame['Average Rating'][0]
std0 = cluster_frame['StDev'][0]
n0 = cluster_frame['Count'][0]

mean1 = cluster_frame['Average Rating'][1]
std1 = cluster_frame['StDev'][1]
n1 = cluster_frame['Count'][1]

mean2 = cluster_frame['Average Rating'][2]
std2 = cluster_frame['StDev'][2]
n2= cluster_frame['Count'][2]

tstat02, pvalue02 = ttest_ind_from_stats(mean0, std0, n0, mean2, std2, n2)
tstat12, pvalue12 = ttest_ind_from_stats(mean1, std1, n1, mean2, std2, n2)
tstat01, pvalue01 = ttest_ind_from_stats(mean0, std0, n0, mean1, std1, n1)

print('tstat and p-value for means {} and {} = {} and {}'.format(mean0, mean1, round(tstat01,2), round(pvalue01,2)))
print('tstat and p-value for means {} and {} = {} and {}'.format(mean0, mean2, round(tstat02,2), round(pvalue02,2)))
print('tstat and p-value for means {} and {} = {} and {}'.format(mean1, mean2, round(tstat12,2), round(pvalue12,2)))
```

```
tstat and p-value for means 6.5 and 6.2 = 20.67 and 0.0
tstat and p-value for means 6.5 and 7.2 = -51.11 and 0.0
tstat and p-value for means 6.2 and 7.2 = -78.53 and 0.0
```

Jupyter Notebook linear regression results



Linear Regression

```
from sklearn.linear_model import LinearRegression
```

```
x = data[['top_category', 'top_mechanic']]
y = data['average']
x, y = np.array(x), np.array(y)
```

```
model = LinearRegression().fit(x, y)
```

```
r_sq = model.score(x, y)
print('intercept:', model.intercept_)
print('slope:', model.coef_)
```

```
intercept: 6.370254404004562
slope: [0.38766397 0.66175411]
```

```
model.score(x,y)
```

```
93]: 0.13869025562777737
```

```
# include game_age as explanatory variable
```

```
x = data[['top_category', 'top_mechanic', 'game_age']]
y = data['average']
x, y = np.array(x), np.array(y)
```

```
model = LinearRegression().fit(x, y)
```

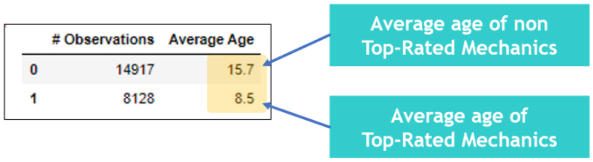
```
r_sq = model.score(x, y)
print('intercept:', model.intercept_)
print('slope:', model.coef_)
```

```
intercept: 6.8025708217256895
slope: [ 0.37005609  0.46670173 -0.02747597]
```

```
model.score(x,y)
```

```
98]: 0.24188667358043936
```

Jupyter Notebook average age games using non-Top-Rated mechanics (0) and Top-Rated mechanics (1)



	# Observations	Average Age
0	14917	15.7
1	8128	8.5

Average age of non Top-Rated Mechanics

Average age of Top-Rated Mechanics