

Project Report

# BoardGameGeek data analysis with Nonparametric Methods

Nonparametric Statistics

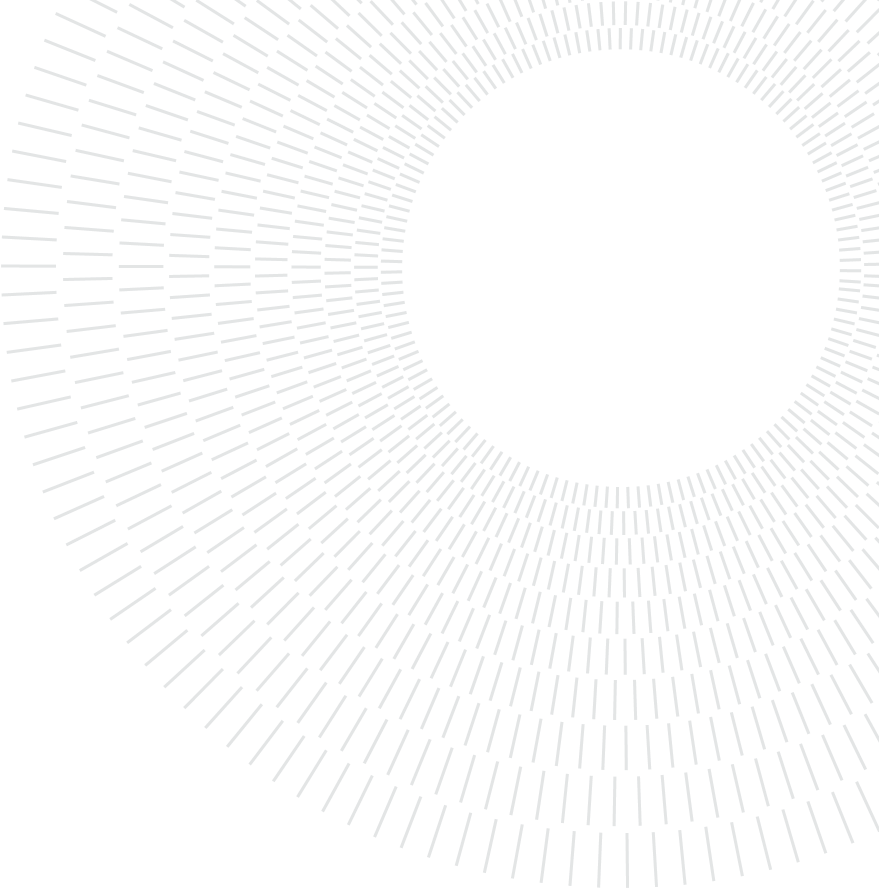
Authors: Davide Carrara and Francesco Romeo Academic year: 2021-2022

# Dataset Introduction

## Introduction

The data for this analysis is an expanded version of the one provided by the weekly updated project https://github.com/rfordat TidyTuesday: the complete dataset is available https://www.kaggle.com/datasets/jvanelteren/boardgamegeekreviews?select=games*detailedinfo.csvhere.*

The dataset corresponds to part of the database of the website https://boardgamegeek.com/BoardGameGeek, which is considered to be the most complete and recognized data source in the area of Board Games. BoardGameGeek is particularly relevant in the boardgame community since it provides a leaderbord of all the available games, and its evaluation of a game quality is usually trustworthy.

Specifically, the provided dataset, updated at January 2022, contains information regard 21630 games with a mixture of text data (mainly description of the games, user reviews, and general comments), numerical quantities (user ratings, information about players and more) and categorical variables.

## Main Features

Here are reported some of the main features of the dataset, along with their meaning and possible uses.

* id, name: Identificative for each game
* Image: Cover image of the game, that could be used for classification purposes
* Description: Text description of the game
* Year: The year of publication of the game
* minplayers, maxplayers, suggested num players: Information about the number of players
* minplaytime, maxplaytime: Information about the ideal playing time of the game
* minage: Minimum required age for the game
* category, mechanic, family, expansion: Categorical variables representing the topic of the game, its main playing mechanics and whether or not belongs to a particular family or expansion. Every game can (and usually does) have multiple categories and mechanics.
* designer, artist, publisher: Information about the people behind the game publication
* average, numratings: Average rating of each game on BoardGameGeek, among with the number of ratings
* averageweight, numweights: Complexity score assigned by users to the game, among with their numerosity
* owned, wishing, trading, wanting: Information related to the internal marketplace of BoardGameGeek.
* A second dataset contains all the reviews of the website, that could be use for sentiment analysis.

1.3. Boh

# Data Exploration - Feature Engineering

## Removing useless features and cleaning

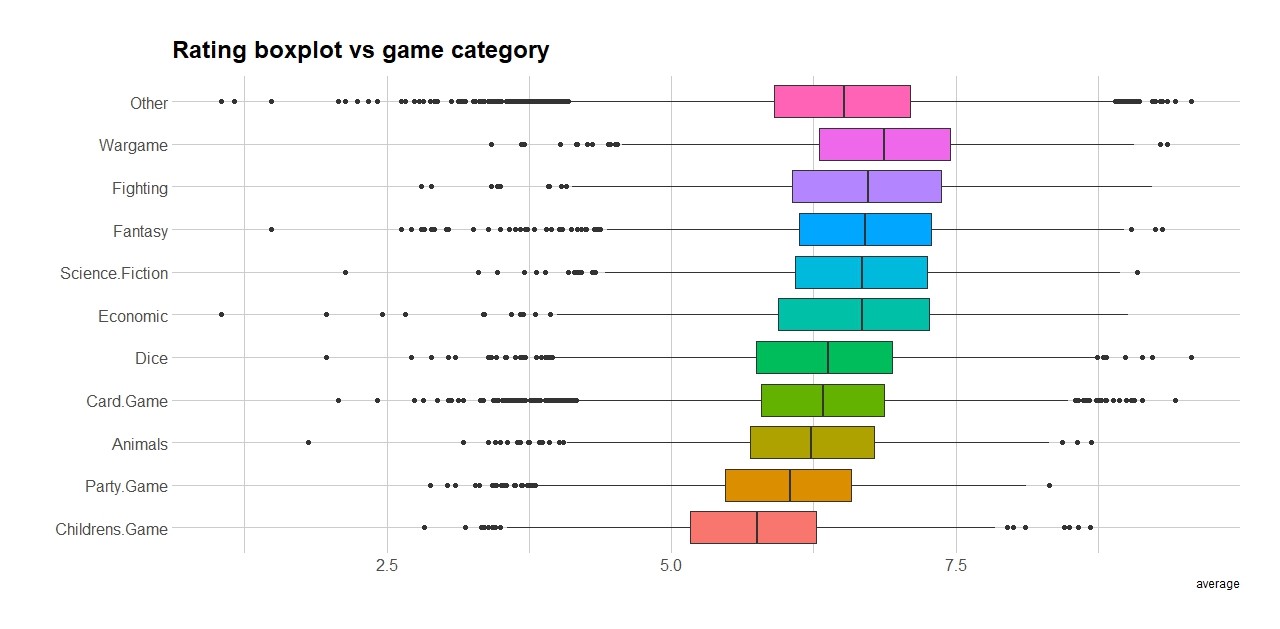
As a first approach, we removed some of the features that would not be useful for our model, or that we would not have time to effectively exploit. We thus removed family, publisher and artist, since they presented very few common instances, and also chose to ignore image and the reviews dataset, as we would have needed to use a completely different approach to treat them.

We also deleted clearly flawed game, having minage and playingtime equal to 0 or negative. In the same way, we chose not to consider games having publishing year before 1900, as we imagine their dynamics to be very different from more recent ones, and they do not correspond to the contemporary idea of boardgames.

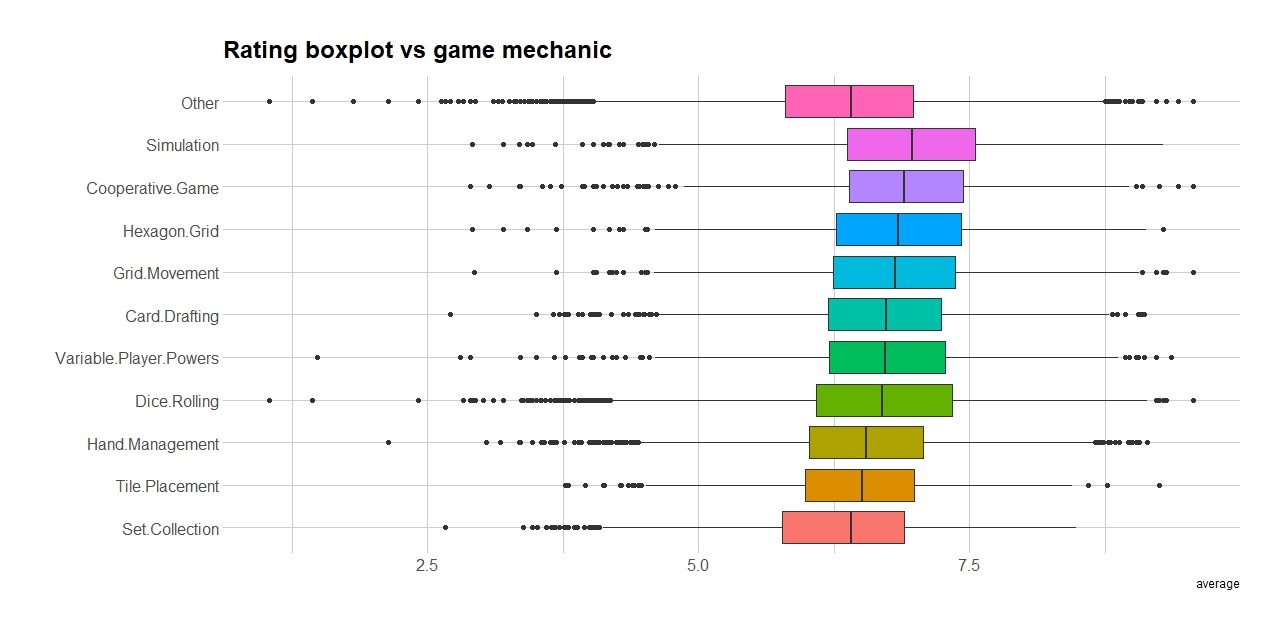
## Feature Extraction

### Category and Mechanic

We extracted the multiple mechanics and categories of each game into a dummy matrix, considering the fact that most games are characterized by more than one category (such as *Political, Fantasy, Children*) or mechanic (such as *Card drafting, Dice Rolling ...*). Only categories with more than 300 appearence ad mechanics with more than 600 instances were kept, to improve robustness.



### Figure 1: Boxplot of the 10 most common categories



### Figure 2: Boxplot of the 10 most common mechanics

As we can see from Figure 1 and 2, categories and mechanics seem to have a significant effect on the average ratings. However, even after reducing the numerosity, we still have 39 possible value for category and 21 for mechanic. we will thus need some other reduction technique.

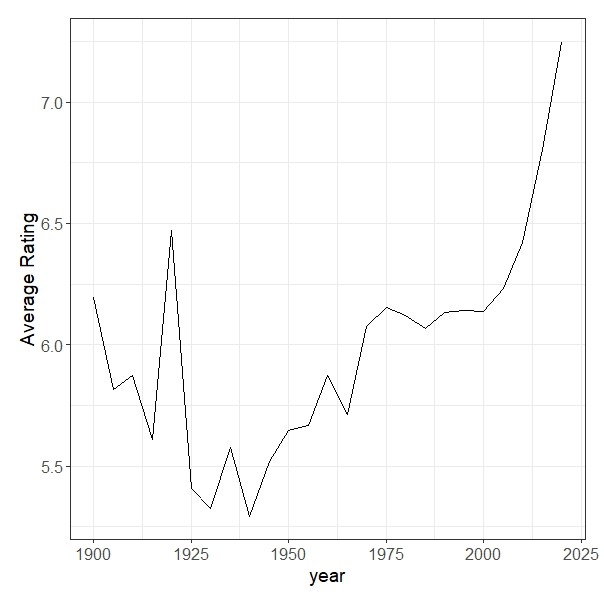
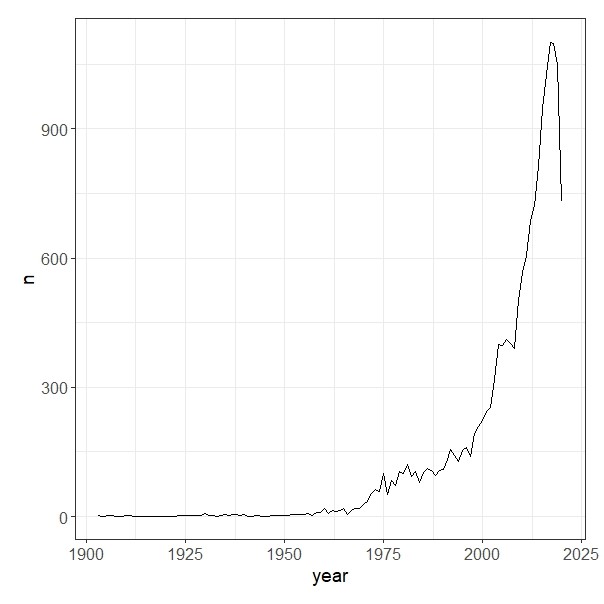
### Suggested Number of Players

The suggested number of players column is expressed as a list of evaluation from users. For each possible number of players we have an indication of how many users considered it the best possible one, an acceptable one, or one which is not adapt for the game. We chose to build a single indicator keeping only the number of players with the most votes, considering the ’best’ indication as a double vote.

## Exploration

### Evolution over time

As we can see from Figures 3a and 3b, time has a significant impact both on the game publication and their average. In particular, excluding the first part of the 20*th* century, when the game production is quantitatively negligible, we can see a clear upward trend for the average and the number of games produced. The downward trend that can be seen in more recent years is probably due to data not being consolidated on the website and to a smaller game production during the pandemic period.

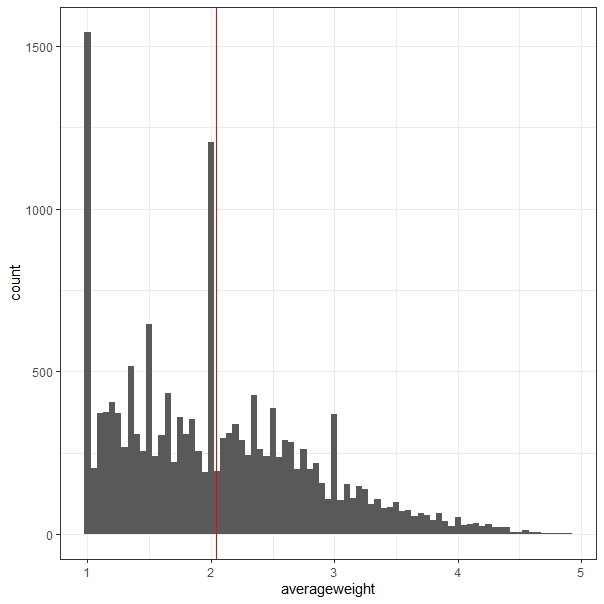
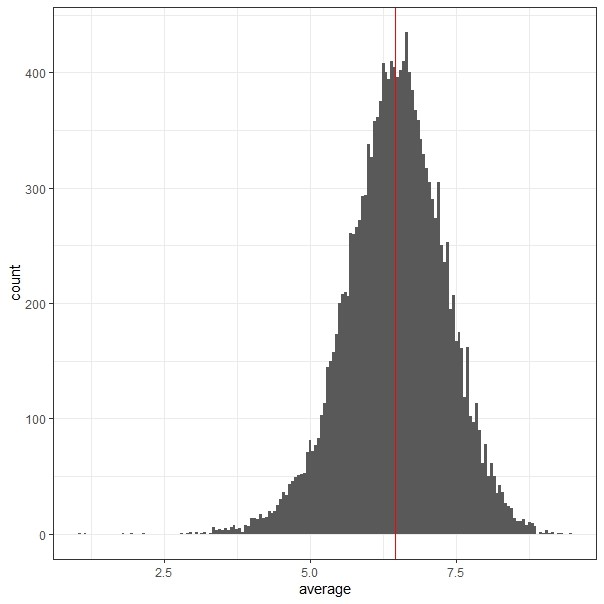


(a) Number of games over time (b) Average rating over time

Figure 3: Trend over time

### Ratings and Complexity

The following plots 4a and 4b represent the ratings given by the users and their evaluation of the game complexity. The distribution of the games ratings seems almost symmetric around a mean value of 6.46 on a 1-10 scale. On the other hand the complexity presents a much more skewed distribution, with mean value of 2.04 on a 1-5 scale.



(a) Histogram of Users rating (b) Histogram of Games Complexity

# Robust Analysis

In order to have a more robust and effective analysis, we wanted to recognize outliers among our dataset. However, the different nature of our features requires dirrent robust approaches, that will be explained in the following paragraphs.

## Coupled Biased Random Walks

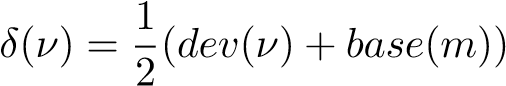
When considering fully categorical data, the literature regarding robust approaches is not as established as in the case of quantitative features. In particular, we would like to identify outliers based on uncommon associations of games categories, such as, for example Childrens games and Warfare or Crime games.

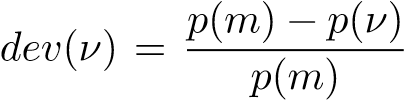
In order to face this problem, we will use the techniques explained in [1] and implemented in the package cbRw.

### Theoretical Aspects

Couple Biased Random Walk is an unsupervised outlier detection method, that will end up assigning an outlierness score to each data point by considering both intra-feature and inter-feature couplings. In particular, the model uses an iterative algorithm to propagate outlierness between feature values, basing on the assumption that outlying values will have strong couplings with other outlying values, while they will have weaker couplings with noisy values.

The categorical data is firstly mapped into a weighted graph, where each node represent a feature value. Each node is assigned a property value, which is based on the frequency of that node inside the feature and the frequency of the mode *m* for the same feature. In particular, the intra-feature outlierness of the value *ν* is given by the following formula:



where and *base*(*m*) = 1 − *p*(*m*). On the other hand the entry of the adjacent matrix

for the graph A(u, v) is equal to *p*(*u*|*v*), and represent the weight of the edge. This can be interpreted as the outlierness propagation value: if u has an high outlierness and u,v are strongly coupled then this propagation value is high. Since *p*(*u*|*v*) = 0 when *u,v* belong to the same feature, we can consider A as the inter-feature outlierness matrix.

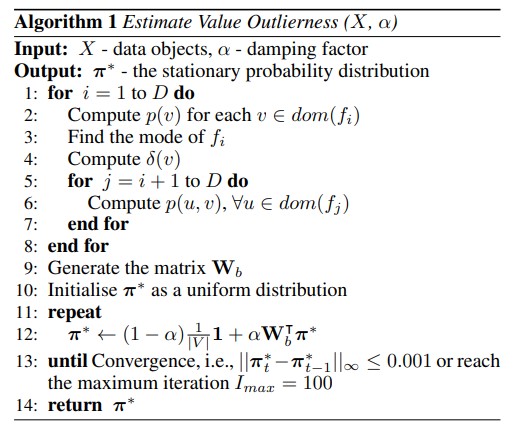
In order to build an unbiased random walk the transition matrix W = AD−1 would be used, where D corresponds to the diagonal matrix of A. However, a biased version is used with the following transition matrix:

*δ*(*ν*)A(*u,v*)

W*b*(*u,v*) = P*v*∈*V δ*(*ν*)A(*u,v*)

The outlierness score of each value is then obtained as *π*∗(*v*), representing the value of the stationary probability density in v.

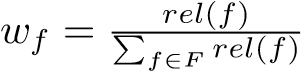
The pseudocode algorithm is taken from [1] and reported in Figure 5 below.



### Figure 5: CBRW Algorithm

Some minor tweaks are then applied to assure convergence in case of a periodic or reducible graph.

Lastly, the relevance *rel*(*f*) = P*ν*∈*dom*(*f*) *value*\_*score*(*v*) is computed giving higher relevance to features where it is possible to effectively distinguish the outliers.

In conclusion, an outlierness score is computed for each data point, with *score*(*x*) = P*f*∈*F wf* ×*value*\_*score*(*v*), where. Outlying objects will thus have higher outlierness score and can be treated as necessary.

### Application

In the specific case of our dataset, we chose to apply CBRW to the matrix of our categorical features, representing whether a game presented a certain category or was played with a certain mechanic. The matrix is thus composed by 59 features (39 categories and 20 mechanics), with binary values.

We flagged as outliers games belongint to the top 0.5% quantile of the outlierness score distribution: we chose to use a conservative approach since we do not have a clear way of evaluating the overall result of the algorithm in terms of it finding uncommon categorical associations. In general

# Modelling

The cleaning of the data set through robust analysis and the creation of clusters for categories and mechanics are essential for the models we decided to build. Our point of view, as stated before, is the one of the boardgame producers : which dynamics or categories make a game more popular? How can we predict the rating of a game and its commercial success? We tried to address these and other questions, whose answers can be important for the company in order to produce a very successful game.

In this section we will discuss the methodologies that we implemented in order to solve these questions, such as permutational anova tests and nonparametric regression techniques. In particular, we will focus on three quantities that can be interesting for the company: the average rating obtained by the game, the average complexity of the game as the users perceive it (column averageweight in the dataset) and the appeal of the game (columns wanting + owned in the dataset). Our aim is to explain these quantities using the characteristics of the game: the suggested number of players, the minimum age required to play the game, the playing time, and so on. These are indeed quantities that the producers can establish during the design of the game itself. Moreover, it is also interesting to understand how the quantity of interest mentioned above are distributed among the clusters created on the categories and the mechanics. This can help us understand which are the most appreciated categories.

## Permutational Anova Techniques

The first step in the design of a successful game is to understand which type of game are most appreciated by the users. We decided to obtain this information by means of permutational anova tests; in particular we investigated how the average rating, the complexity and the appeal of the game vary over the categories and the mechanics of the game.

The procedure that we implemented is the following: first perform a two-way anova using the clusters of the categories and the mechanics, then if we find a significant difference in one of the two groups for one of the three quantity of interest, we investigate this difference by performing an anova multiple ways, in which the i-th factor represents the membership to the i-th cluster. The two-way anova test can be formulated as follows: indicating with *Xijk* the *k* − *th* statistical unit belonging to *i* − *th* cluster on categories and *j* − *th* cluster on mechanics, the assumptions of the model are:

|  |  |
| --- | --- |
| *Xijk* = *µij* + *ϵijk* with  and the hypothesis of the test are:  • For the interaction term: | *µij* = *µ* + *τi* + *βj* + *γij* |
| *H*0 : *γij* = 0 ∀*i,j* • For the factor 1: | *H*1 : ∃ *i,j s.t. γi,j* ̸= 0 |

*H*0 : *τi* = 0 ∀*i H*1 : ∃ *i s.t. τi* ̸= 0

• For the factor 2:

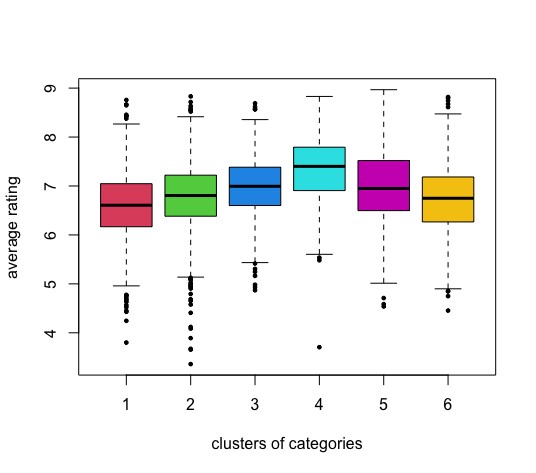
*H*0 : *βj* = 0 ∀*j H*1 : ∃ *j s.t. βj* ̸= 0

This formulation can be generalized also for the Anova multiple ways test that follows the two way anova, with the exception that, for the sake of simplicity and interpretability, we didn’t consider the interaction terms. The anova multiple ways helps us to understand which is the group that really affects the difference in the mean for the quantity of interest by testing the significance of each factor. In this case, since we will perform six tests, we decided to correct the level of significance of the test through a Bonferroni correction, that is, we considered as level of significance 0.1 divided by the number of clusters (6). This allows us to have an overall significance of 0.1 for the test related to the significance of the clusters.

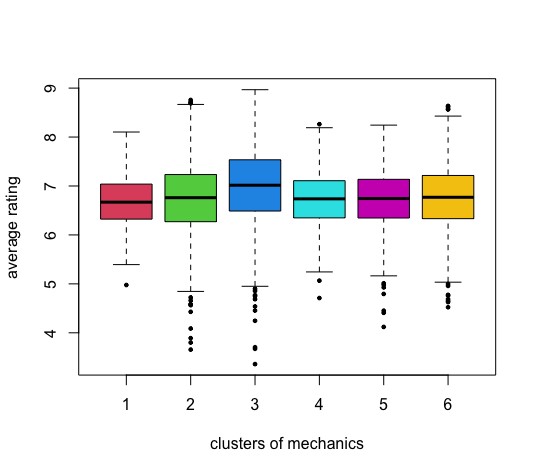
We decided to use the procedure described above because there is no post-hoc analysis for permutational anova tests that could have helped us in understanding which are the clusters that lead to a difference in the mean. It is worth to mention that in this section we used the Kmedoids clusters for our analysis. However, doing the same tests on the features clusters leads to the same results.

### Permutational Anova for the average rating

We start by showing a boxplot of the quantity of interest over the clusters created for the categories of the game and for the mechanics:



### Figure 6: Boxplots of average rating over the clusters of categories



### Figure 7: Boxplots of average rating over the clusters of mechanics

There seem to be a difference in the average rating among the clusters for the categories but not among those for the mechanics. Performing a permutational two-way anova we obtained that we have statistical significance to assess that the mean of the average rating differs only on the clusters for the categories. Indeed, for the interaction term we obtained a pvalue of 0.001, but for the term related to the clusters on mechanics the pvalue was equal to 0.82, while for the first factor, the clusters on categories, the pvalue was 0.009. Thus, we considered only the first factor as significant and we proceeded as described above performing an anova multiple ways in which each cluster is a grouping factor.

As a result, we obtained that belonging to fourth cluster increases the mean of the average rating of the game; we also obtained that the third cluster is not significant for the average rating, while the fifth has a coefficients that is close to zero, and the others have negative coefficients. Investigating over the fourth cluster, we noticed that the three most common categories are: War games, World War II, Ancient. These categories are not present in the other clusters, thus we can think that they overall increase the average rating significantly.

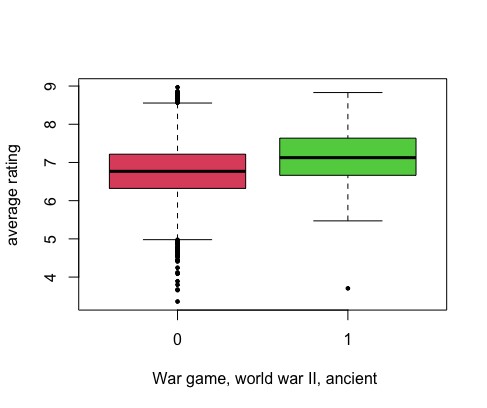
To confirm this qualitative result, we performed another anova test in which we distinguished between games that belongs to at least one of the three categories mentioned before. This test corresponds to an Anova one way, that can be desribed as follows: indicating with *Xik* the *k* − *th* statistical unit belonging to *i* − *th* group, the assumptions of the model are:

*Xik* = *µi* + *ϵik* with *µi* = *µ* + *τi*

and the hypothesis of the test are:

*H*0 : *τi* = 0 ∀*i H*1 : ∃ *i s.t. τi* ̸= 0

The boxplot related to this test is the following:



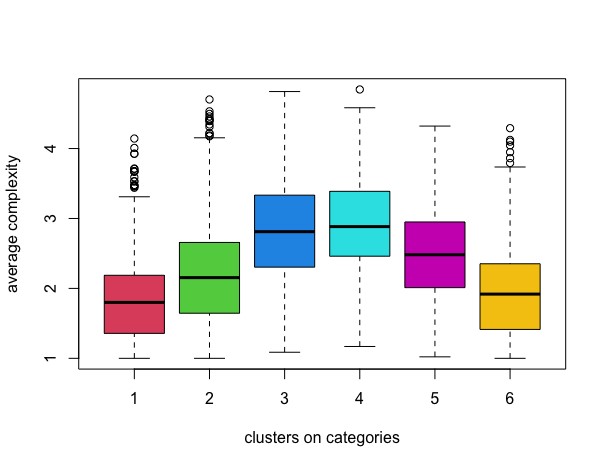
### Figure 8: Boxplots of average rating for War games, World War II, Ancient vs other categories

and the permutational anova gives a pvalue equal to zero, thus we can conclude that there is statistical significance to reject *H*0.

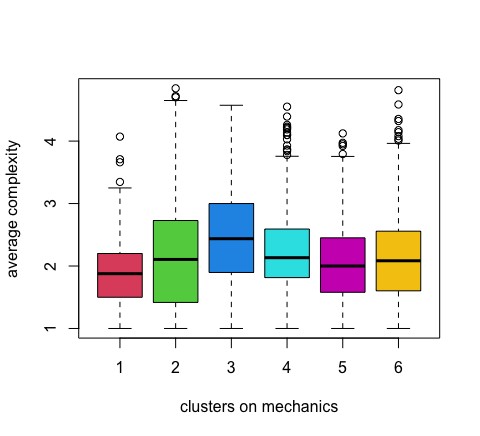
To conclude, we can say that the categories that leads to an higher average rating are War games, World War II, Ancient.

### Permutational Anova for the average complexity

The average complexity of the game perceived by the user is difficult to evaluate for the producers, yet it can be very interesting. This is why we first wanted to see if some games are considered more complex than others only based on the mechanic of the game or on the type of game, and to do so we exploited a permutational anova test. Below we reported the boxplots of the average complexity as it is distributed among the clusters for categories and mechanics:



### Figure 9: Boxplots of average rating over the clusters of categories



### Figure 10: Boxplots of average rating over the clusters of mechanics

As before, performing a permutational two-way anova we obtained that we have statistical significance to assess that the mean of the average rating differs only on the clusters for the categories, which is consistent with the boxplots. In particular, the pvalues for the interaction term, the cluster on categories and the cluster on mechanics are respectively 0.036, 0.005 and 0.89. Thus we can focus on the factor related to the cluster on categories. Performing an anova multiple ways in which each cluster is a grouping factor, we obtained that belonging to 3rd, 4th or 5th cluster increases the mean of the average rating of the game. The most common categories in these clusters are

* Cluster 3: Economic, Card.Game, City.Building
* Cluster 4: War games, World War II, Ancient
* Cluster 5: Fantasy, Fighting, Exploration

The 4th cluster indeed contains categories of games that can be considered complex games, and the same holds for the third cluster. But what is better: to have a more or a less complex game? We will answer later to this question, by building a model that tries to link the average complexity with the average rating.

### Permutational Anova for the appeal

The appeal of the games is measured using the information contained in the columns *wanting* and *owned*, which express the number of users of the website that want to buy the game and the number of users that already have it respectively. This can be a significant indicator of how much the game has been successful, thus we would like to explain it in terms of category of the game and mechanic of the game through a permutational anova test.

Below we reported the boxplots of the appeal in the different clusters:

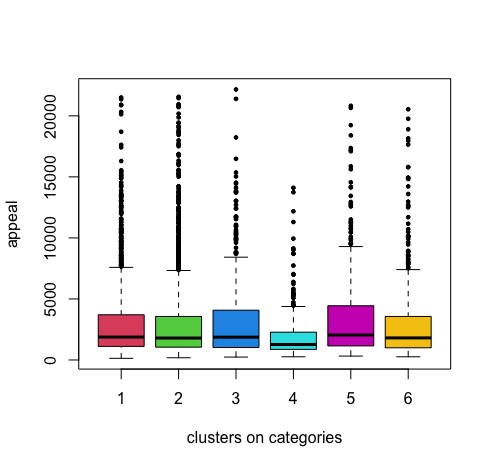
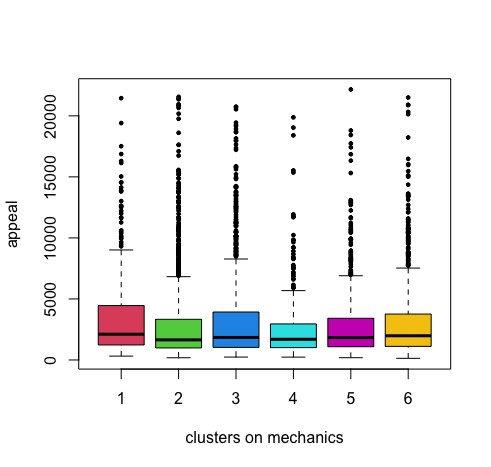


Figure 11: Boxplots of appeal over the clusters of categories



### Figure 12: Boxplots of appeal over the clusters of mechanics

This time the clusters doesn’t seem to help for distinguishing games with higher or lower appeal. Indeed, from the permutational two way anova test we don’t have statistical significance to reject H0 neither for the interaction term nor for the single terms. In particular:

* The pvalue of the test related to the interaction term is 0.20
* The pvalue of the test on the term related to the clusters on categories is 0.80
* The pvalue of the test on the term related to the clusters on mechanic is 0.90

So we can say that there is no statistical evidence to assess that the appeal differs on average among these clusters.

## Regression Models

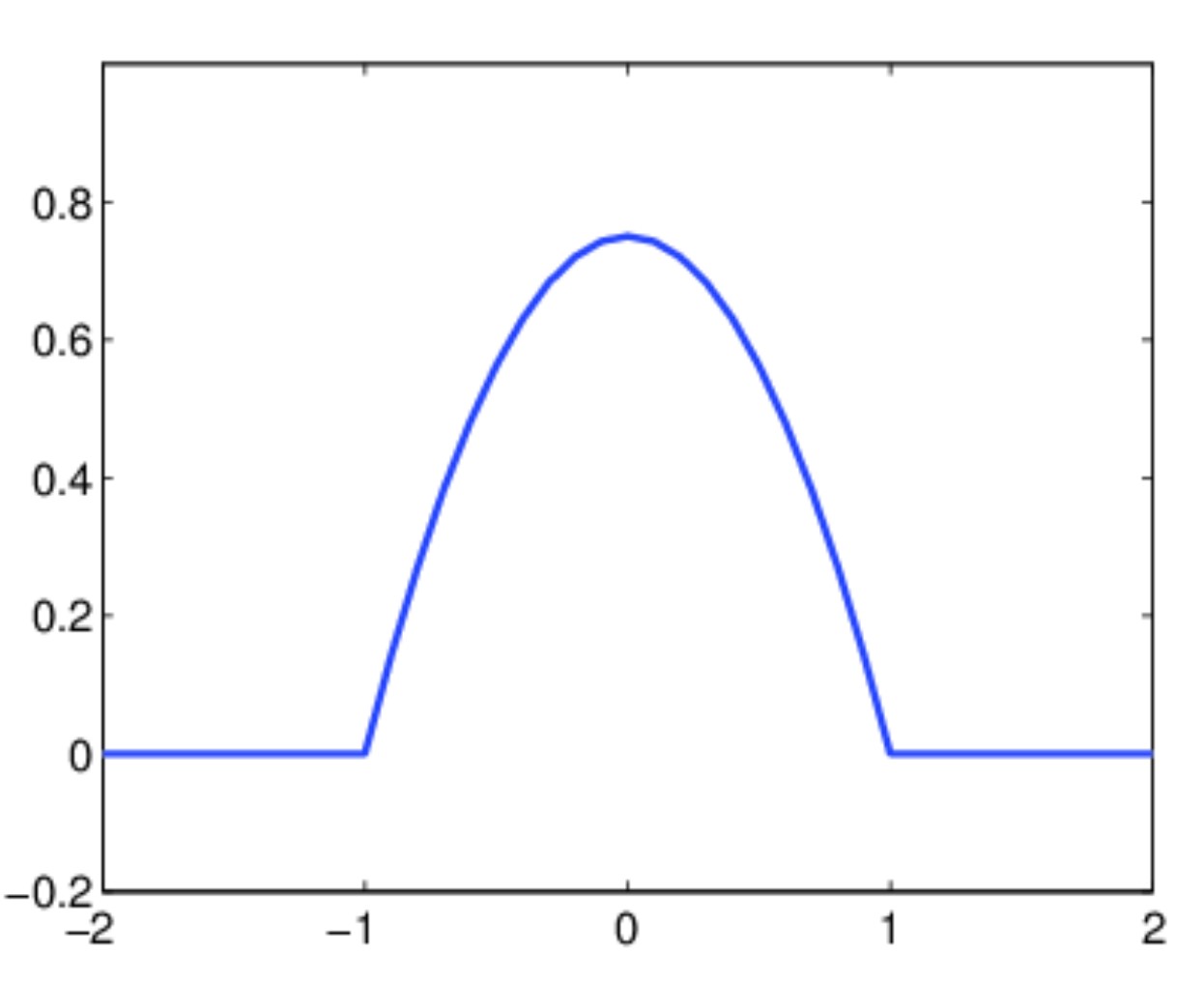
In this section, we will discuss the nonparametric regression techniques that we adopted in order to try to explain either the average rating of the game or the average complexity. The former is important because, if well explained, can help the producers to understand how much the game will be appreciated by the users. The latter is as well fundamental to be explained because it is somehow related to the former, and we will see that usually the more complex the game, the higher the average.

The models that we estimated are the following: weighted local averaging through epanechnikov kernel, generalized additive models, xgboost. Before seeing the result it is worth to mention some theoretical notions related to these three models.

### Theoretical Framework

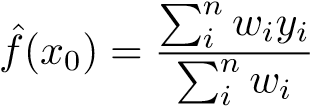
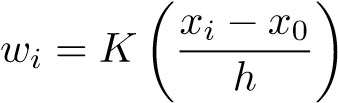
The three models we aim at implementing are completely different from each other. The first one is a nonparametric method for regression: given a focal point *x*0, local averaging outputs as *f*ˆ(*x*0) the weighted average of the points that are inside a neighborhood of the focal point itself. The weights of the points in the neighborhood are decided according to a kernel, and the bandwidth is determined by the kernel and a scaling factor called *bandwith*.

We decided to use the epanechnikov kernel, whose graphical representation is the following:



### Figure 13: example of epanechnikov Kernel

As we can see from the plot, the closer the point to the focal point, the higher the weight. Moreover, we can see that the bandwith is not infinite, since this function has a finite support. This means that some points will have zero weight, thus they will not have any importance in the prediction in correspondence of the focal point. Thus, given the kernel function *K*(*.*) and the bandwith *h*, the estimation given by this model in correspondence of a focal point *x*0 is given by:

 where 

*yi* are the values associated to the statistical units *xi* in the dataframe. This formula holds in the univariate case, which is the framework in which we will use this model. However, we would like to use more complex models, and not only univariate ones.

Nonparametric models suffer from curse of dimensionality, thus it is better to avoid considering multivariate models, but we can focus on functions of the following type: *f*(*x*) = *f*(*x*1) + *f*(*x*2) + *...* + *f*(*xp*) where p is the number of covariates we want to use.

This trick allows us to move to the second type of model we would like to use, which are the generalised additive models (GAM). This type of models focuses on functions of the shape indicated above, meaning that we can estimate each univariate model *f*(*xi*) and then combine them by summing them.

# References

[1] Guansong Pang, Longbing Cao, and Ling Chen. Outlier detection in complex categorical data by modelling the feature value couplings. IJCAI’16. AAAI Press, 2016.

# Conclusions 6. References

• The GitHub repository from which we extract the data can be found https://github.com/pcm-dpc/COVID-

19here