Department of Computer Science
Technical University of Cluj-Napoca

Intelligent Systems

Laboratory activity 2019-2020

Project title: Machine Learning - Unsupervised Learning
Tool: Scikit-learn

Name: Georgescu Vlad, Elekes Lukacs
Group: 30434

Contents

Chapter 1

Overview

1.1 Unsupervised Learning

Machine Learning Algorithms can be divided into categories according to their purpose. The main categories ar the following:

- Supervised Learning
- Unsupervised Learning
- Semi-supervised Learning
- Reinforcement Learning

Our focus is on Unsupervised Learning. Unsupervised learning is a machine learning technique, where you do not need to supervise the model. Instead, you need to allow the model to work on its own to discover information. It mainly deals with the unlabeled data.

Unsupervised learning algorithms allow you to perform more complex processing tasks compared to supervised learning. Although, unsupervised learning can be more unpredictable compared with other natural learning deep learning and reinforcement learning methods.

1.2 Scikit-learn

The library used in the development of the project is Scikit-learn.

Scikit-learn is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, K-Means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

Chapter 2

Main Functionalities

Scikit-learn provides some functionalities that allow us to use machine learning techniques. Among these features are the unsupervised learning category.

The goal in such problems may be to discover groups of similar examples within the data, called **clustering**; or to determine the distribution of data within the input space, known as **density estimation**.

Clustering algorithms may be classified as below:

1. **Exclusive Clustering**: where data is grouped in an exclusive way, so that if a certain data point belongs to a definite cluster then it could not be included in another cluster

2. **Overlapping Clustering**: which is opposed to the first one. It uses fuzzy sets to cluster data, so that each point may belong to two or more clusters with different degrees of membership

3. **Hierarchical Clustering**: it is based on the union between the two nearest clusters

4. **Probabilistic Clustering**: uses a completely probabilistic approach where each point have a certain probability to belong to a given cluster

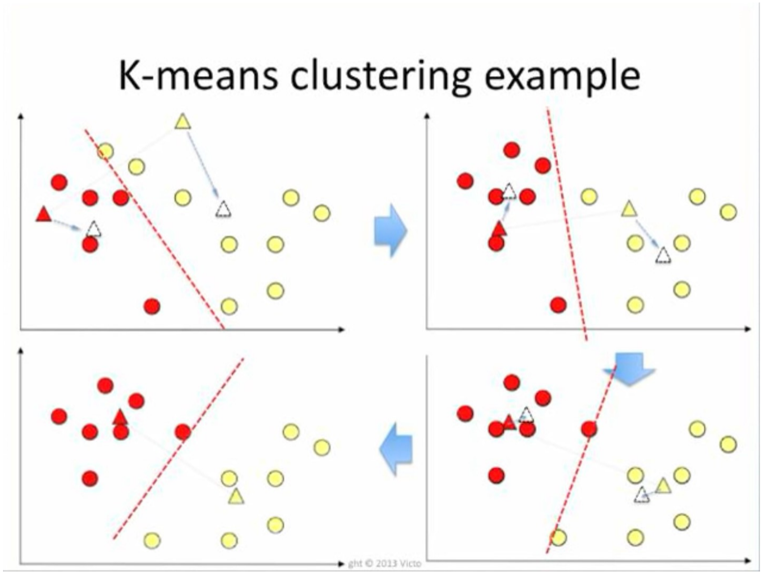
Chapter 3

K-Means Algorithm

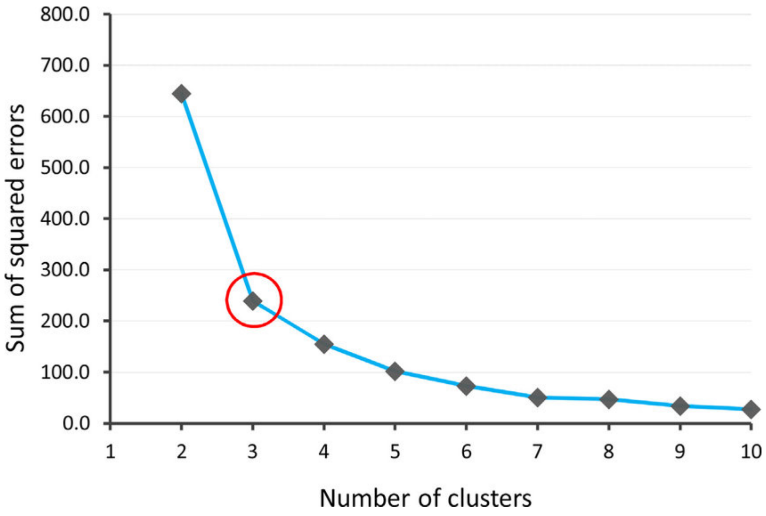
K-Means Algorithm is one of the Exclusive Clustering algorithm that scikit-learn provides. K-means algorithm is an iterative algorithm that tries to partition the dataset into K predefined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group. It tries to make the inter-cluster data points as similar as possible while also keeping the clusters as different (far) as possible. It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster's centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum. The less variation we have within clusters, the more similar the data points are within the same cluster.

The steps of the algorithm are the following:

1. It starts with K as the input which is how many clusters you want to find. Place K centroids in random locations in your space.
2. Now, using the euclidean distance between data points and centroids, assign each data point to the cluster which is close to it.
3. Recalculate the cluster centers as a mean of data points assigned to it.
4. Repeat 2 and 3 until no further changes occur.



To decide the optimal number of clusters, we can use the Elbow method. It involves running the algorithm multiple times over a loop, with an increasing number of cluster choice and then plotting a clustering score as a function of the number of clusters. The score is, in general, a measure of the input data on the k-means objective function i.e. some form of intra-cluster distance relative to inner-cluster distance. For example, in Scikit-learn's k-means estimator, a score method is readily available for this purpose.

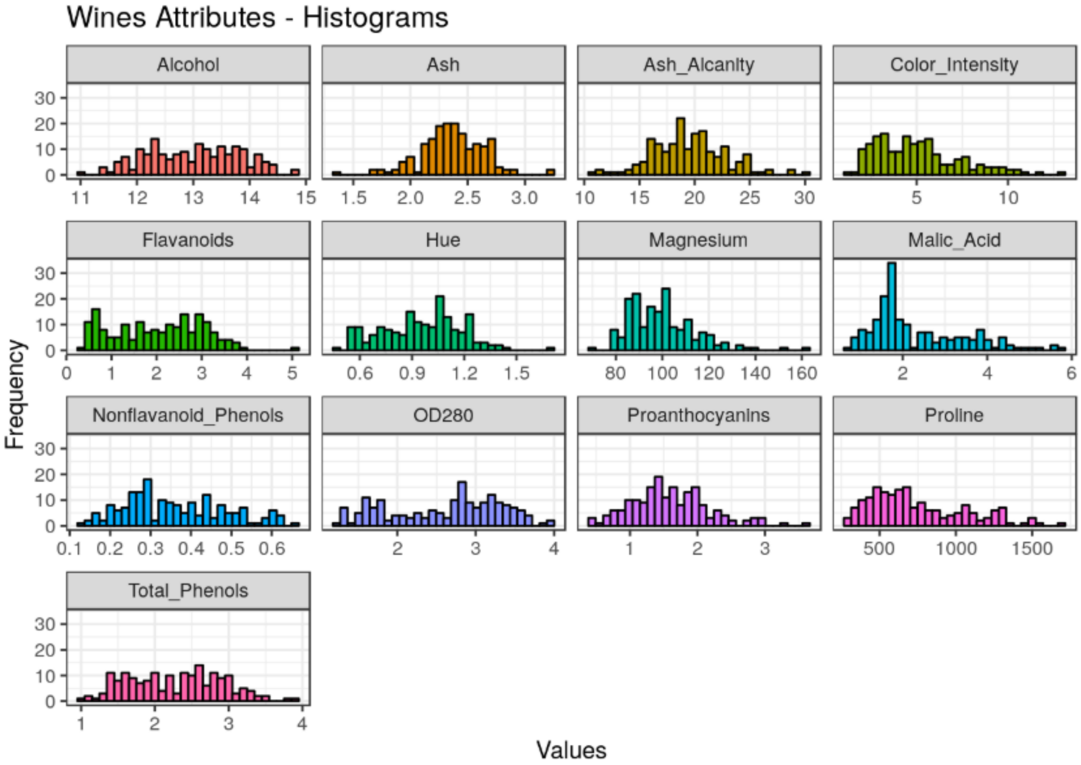


Chapter 4

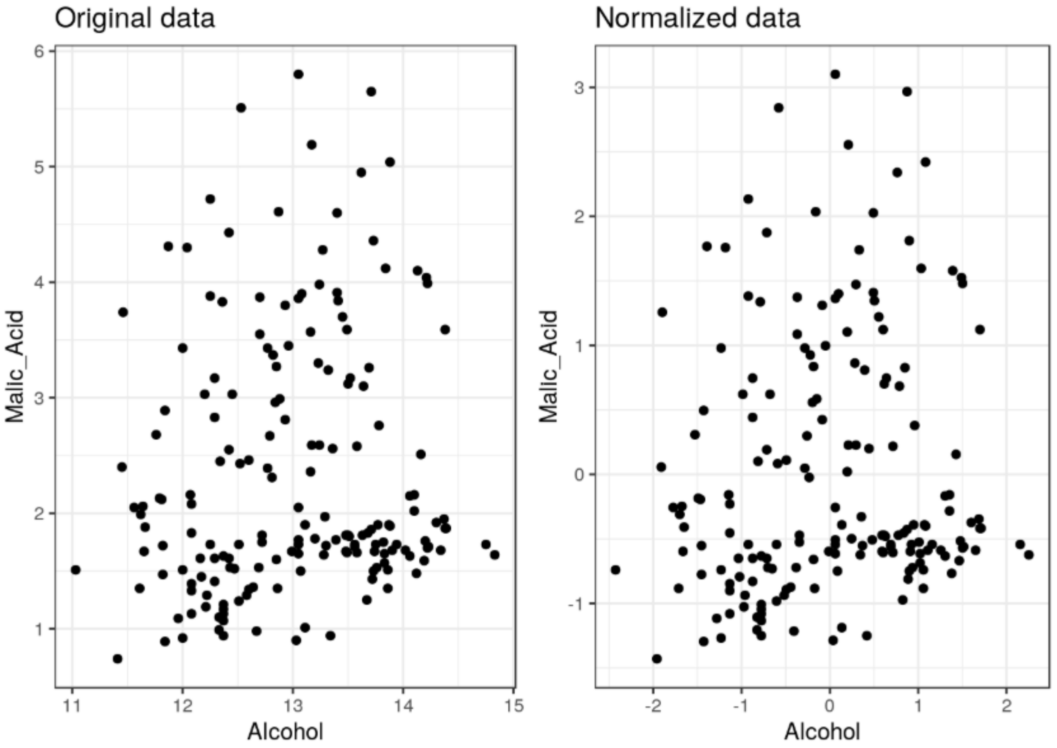
Examples

4.1 The Wine Dataset

We have analyzed an example that presents the clustering process on the famous wine dataset using K-Means clustering[1]. The wine dataset has 13 features like "Alcohol", "Color_Intensity", "Magnesium" and others. The feature are presented below using histograms to visualize the data.

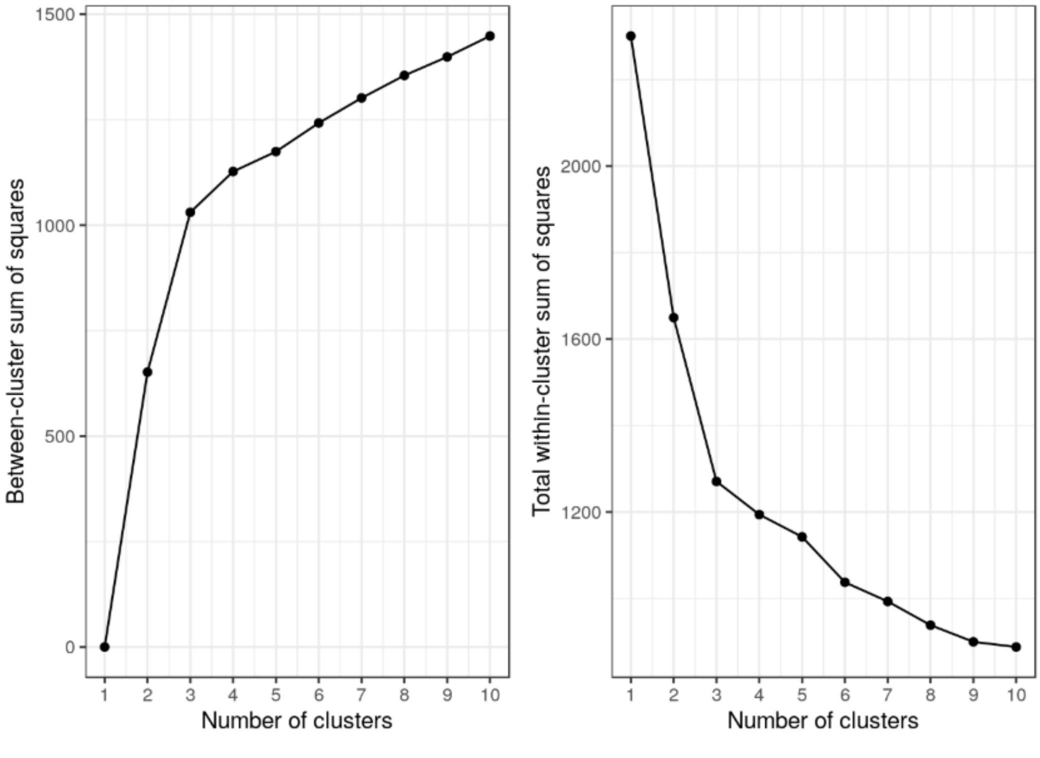


The data is normalized to express the variables in the same range of values.



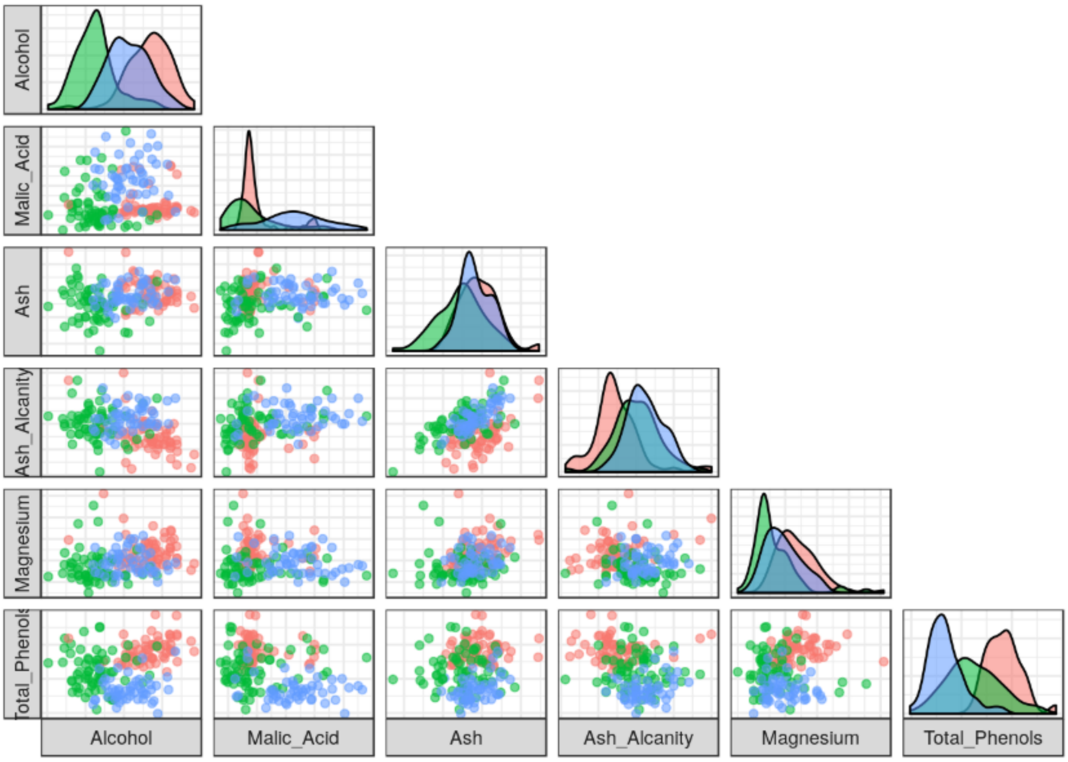
We can observe now that the data are plotted on a common scale.

In the example studied the data are clustered into two clusters, but the ideal number of clusters is not analyzed yet, we don't know how good is the result. So a plot is added that shows the *between-cluster sum of squares* and the *total within-cluster sum of squares*.



Using the Elbow method, we can say that 3 for the number of clusters is a good choice.

In the following some plots are shown to visualize the data in the 3 clusters that were obtained.

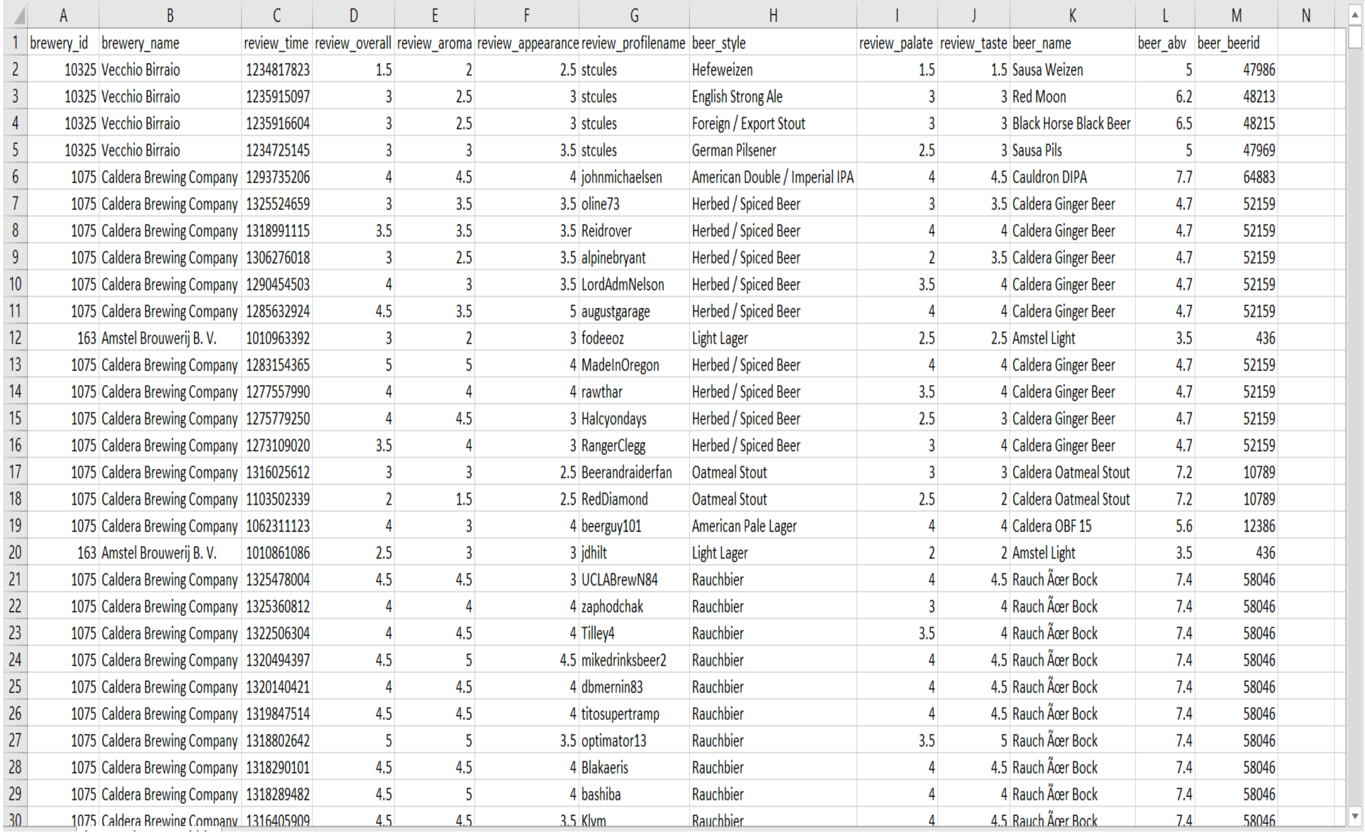


Chapter 5

Proposed Problem

5.1 Dataset

We are using a dataset obtained from the website Kaggle[2] consisting of over 1.5 million entries. These entries represent user reviews about different beers. The dataset is called ”beer_reviews” and contains the following features: ”brewery_id”, ”brewery_name”, ”review_time”, ”review_profilename”, ”review_overall”, ”review_taste”, ”review_appearance”, ”review_palate”, ”review_aroma”, ”beer_style”, ”beer_name”, ”beer_beerid” and ”beer_abv”. Originally, this is how a very small part of the dataset looks like:

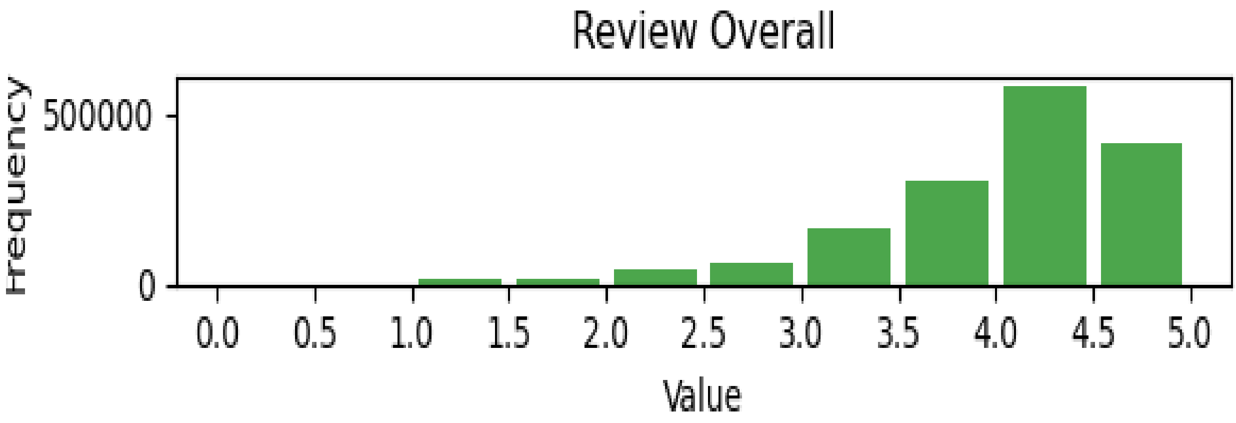


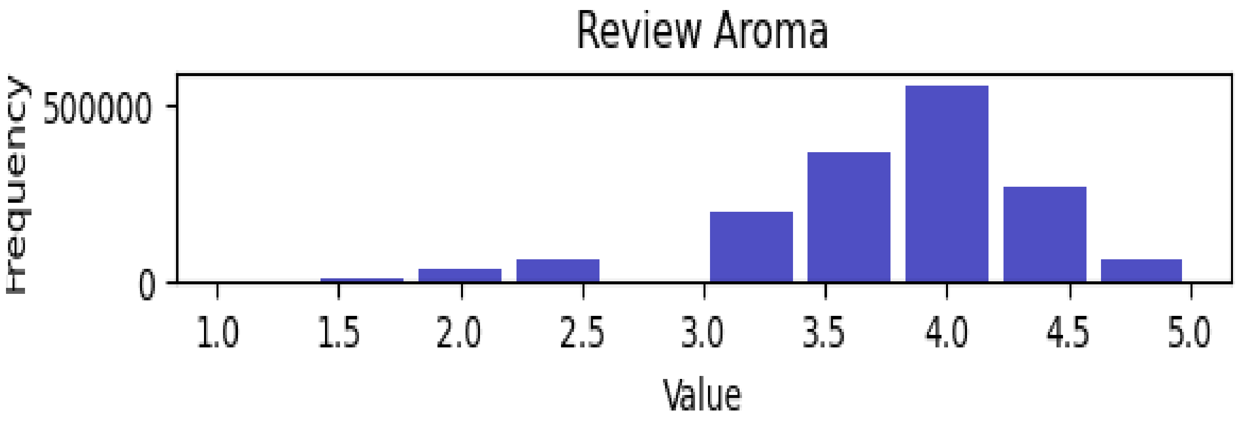
	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	brewery_id	brewery_name	review_time	review_overall	review_aroma	review_appearance	review_profilename	beer_style	review_palate	review_taste	beer_name	beer_abv	beer_beerid	
2	10325	Vecchio Birraio	1234817823	1.5	2	2.5	stcules	Hefeweizen	1.5	1.5	Sausa Weizen	5	47986	
3	10325	Vecchio Birraio	1235915097	3	2.5	3	stcules	English Strong Ale	3	3	Red Moon	6.2	48213	
4	10325	Vecchio Birraio	1235916604	3	2.5	3	stcules	Foreign / Export Stout	3	3	Black Horse Black Beer	6.5	48215	
5	10325	Vecchio Birraio	1234725145	3	3	3.5	stcules	German Pilsener	2.5	3	Sausa Pils	5	47969	
6	1075	Caldera Brewing Company	1293735206	4	4.5	4	johnmichaelsen	American Double / Imperial IPA	4	4.5	Cauldron DIPA	7.7	64883	
7	1075	Caldera Brewing Company	1325524659	3	3.5	3.5	oline73	Herbed / Spiced Beer	3	3.5	Caldera Ginger Beer	4.7	52159	
8	1075	Caldera Brewing Company	1318991115	3.5	3.5	3.5	Reidrover	Herbed / Spiced Beer	4	4	Caldera Ginger Beer	4.7	52159	
9	1075	Caldera Brewing Company	1306276018	3	2.5	3.5	alpinebryant	Herbed / Spiced Beer	2	3.5	Caldera Ginger Beer	4.7	52159	
10	1075	Caldera Brewing Company	1290454503	4	3	3.5	LordAdmNelson	Herbed / Spiced Beer	3.5	4	Caldera Ginger Beer	4.7	52159	
11	1075	Caldera Brewing Company	1285632924	4.5	3.5	5	augustgarage	Herbed / Spiced Beer	4	4	Caldera Ginger Beer	4.7	52159	
12	163	Amstel Brouwerij B. V.	1010963392	3	2	3	fodeeoz	Light Lager	2.5	2.5	Amstel Light	3.5	436	
13	1075	Caldera Brewing Company	1283154365	5	5	4	MadeInOregon	Herbed / Spiced Beer	4	4	Caldera Ginger Beer	4.7	52159	
14	1075	Caldera Brewing Company	1277557990	4	4	4	rawthar	Herbed / Spiced Beer	3.5	4	Caldera Ginger Beer	4.7	52159	
15	1075	Caldera Brewing Company	1275779250	4	4.5	3	Halcyondays	Herbed / Spiced Beer	2.5	3	Caldera Ginger Beer	4.7	52159	
16	1075	Caldera Brewing Company	1273109020	3.5	4	3	RangerClegg	Herbed / Spiced Beer	3	4	Caldera Ginger Beer	4.7	52159	
17	1075	Caldera Brewing Company	1316025612	3	3	2.5	Beerandraiderfan	Oatmeal Stout	3	3	Caldera Oatmeal Stout	7.2	10789	
18	1075	Caldera Brewing Company	1103502339	2	1.5	2.5	RedDiamond	Oatmeal Stout	2.5	2	Caldera Oatmeal Stout	7.2	10789	
19	1075	Caldera Brewing Company	1062311123	4	3	4	beerguy101	American Pale Lager	4	4	Caldera OBF 15	5.6	12386	
20	163	Amstel Brouwerij B. V.	1010861086	2.5	3	3	jdhilt	Light Lager	2	2	Amstel Light	3.5	436	
21	1075	Caldera Brewing Company	1325478004	4.5	4.5	3	UCLABrewN84	Rauchbier	4	4.5	Rauch Ãœr Bock	7.4	58046	
22	1075	Caldera Brewing Company	1325360812	4	4	4	zaphodchak	Rauchbier	3	4	Rauch Ãœr Bock	7.4	58046	
23	1075	Caldera Brewing Company	1322506304	4	4.5	4	Tilley4	Rauchbier	3.5	4	Rauch Ãœr Bock	7.4	58046	
24	1075	Caldera Brewing Company	1320494397	4.5	5	4.5	mikedrinksbeer2	Rauchbier	4	4.5	Rauch Ãœr Bock	7.4	58046	
25	1075	Caldera Brewing Company	1320140421	4	4.5	4	dbmernin83	Rauchbier	4	4.5	Rauch Ãœr Bock	7.4	58046	
26	1075	Caldera Brewing Company	1319847514	4.5	4.5	4	titosupertramp	Rauchbier	4	4.5	Rauch Ãœr Bock	7.4	58046	
27	1075	Caldera Brewing Company	1318802642	5	5	3.5	optimator13	Rauchbier	3.5	5	Rauch Ãœr Bock	7.4	58046	
28	1075	Caldera Brewing Company	1318290101	4.5	4.5	4	Blakaeris	Rauchbier	4	4.5	Rauch Ãœr Bock	7.4	58046	
29	1075	Caldera Brewing Company	1318289482	4.5	5	4	bashiba	Rauchbier	4	4	Rauch Ãœr Bock	7.4	58046	
30	1075	Caldera Brewing Company	1316405909	4.5	4.5	3.5	Klym	Rauchbier	4	4.5	Rauch Ãœr Bock	7.4	58046	

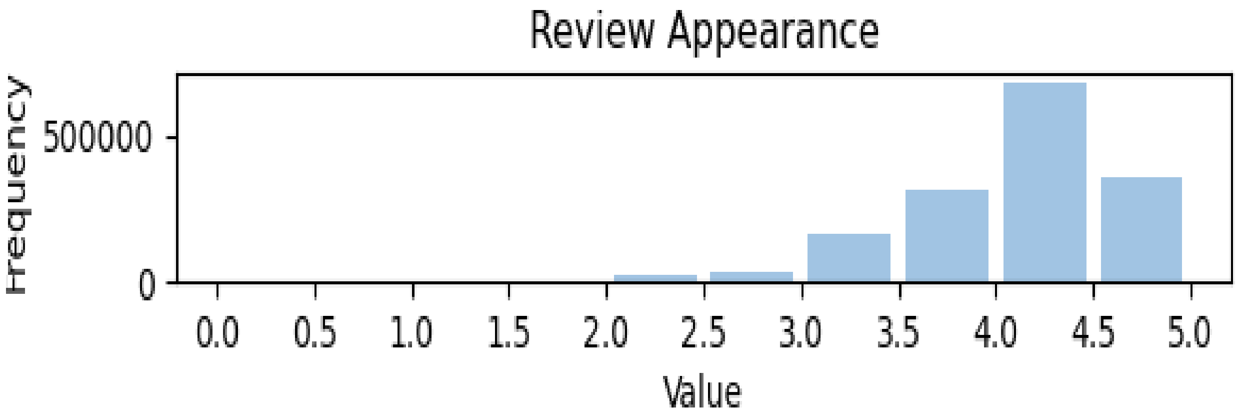
Some of this information was irrelevant for our project so we dropped some columns from the dataset. Columns that stored text information and ids were dropped since we did not need labeled data as in the case of supervised learning, but we needed numerical data in order to process it and cluster it. So we got rid of the following columns: ”brewery_id”, ”brewery_name”, ”review_time”, ”review_profilename”, ”beer_style”, ”beer_name”, ”beer_beerid”.

After the elimination of the unnecesarry columns of the dataset, we remained with the following columns of data for processing: ”review_overall”, ”review_taste”, ”review_appearance”, ”review_palate”, ”review_aroma”, ”beer_abv”. After this process, we noticed that some of the cells in ”beer_abv” were not containing any value so we filled them by the mean value of the ”beer_abv” column in order to obtain accurate results. After all these operations, our dataset was processed and ready to operate with.

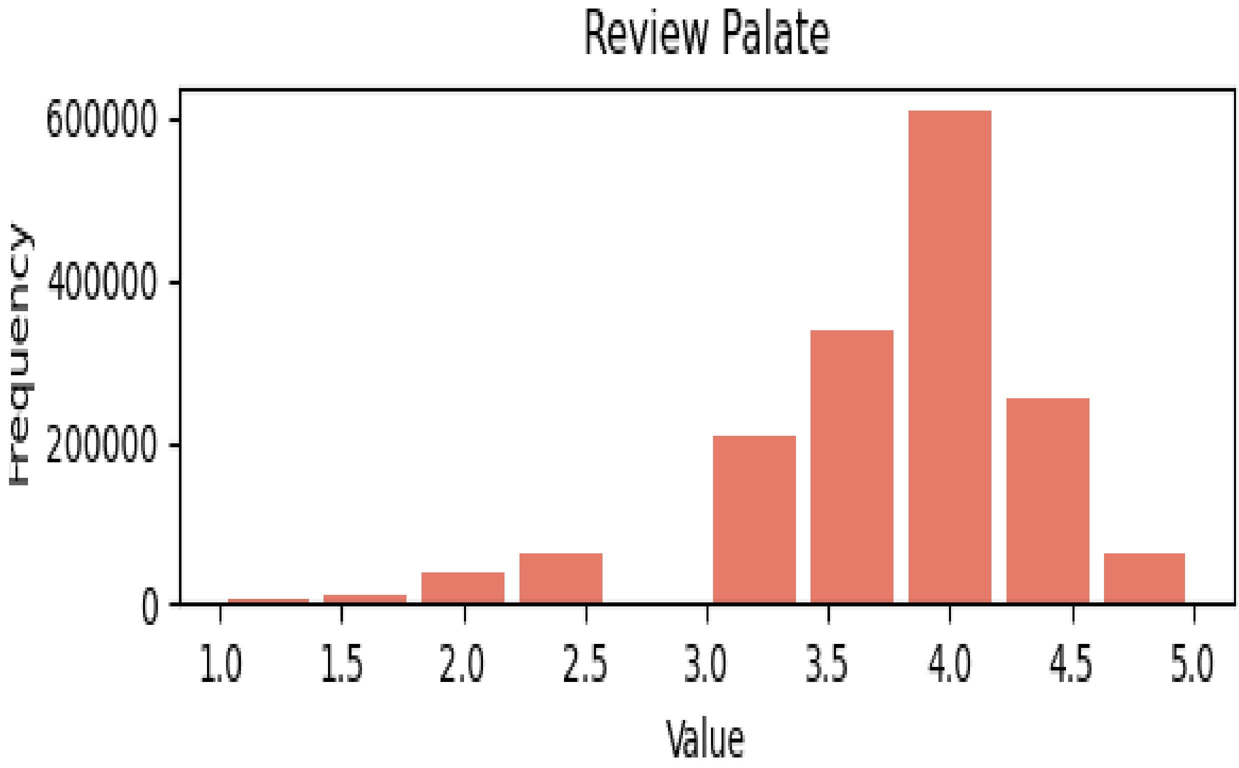
The visualization of the dataset was highlighted by means of some histograms for all of our interest features. Below, there can be observed the ratio between the frequency and the value of ”review_overall”, ”review_aroma” and ”review_appearance” features:

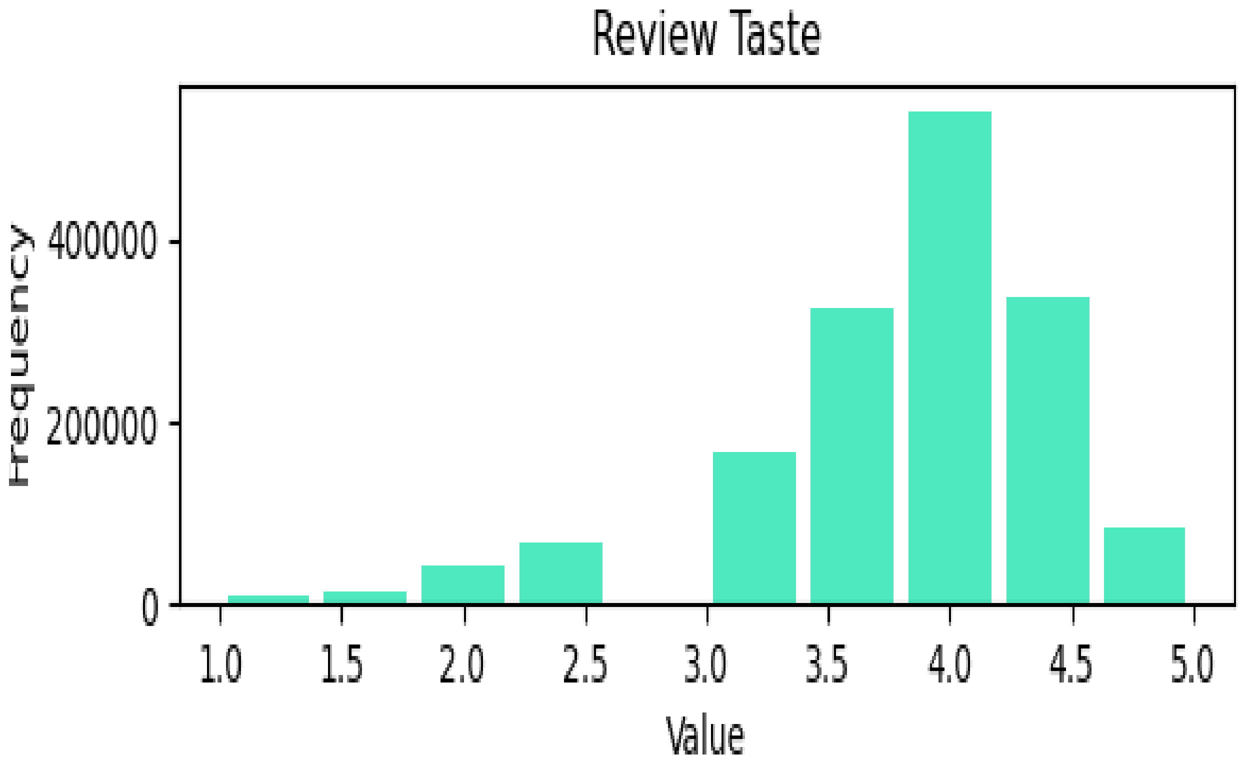




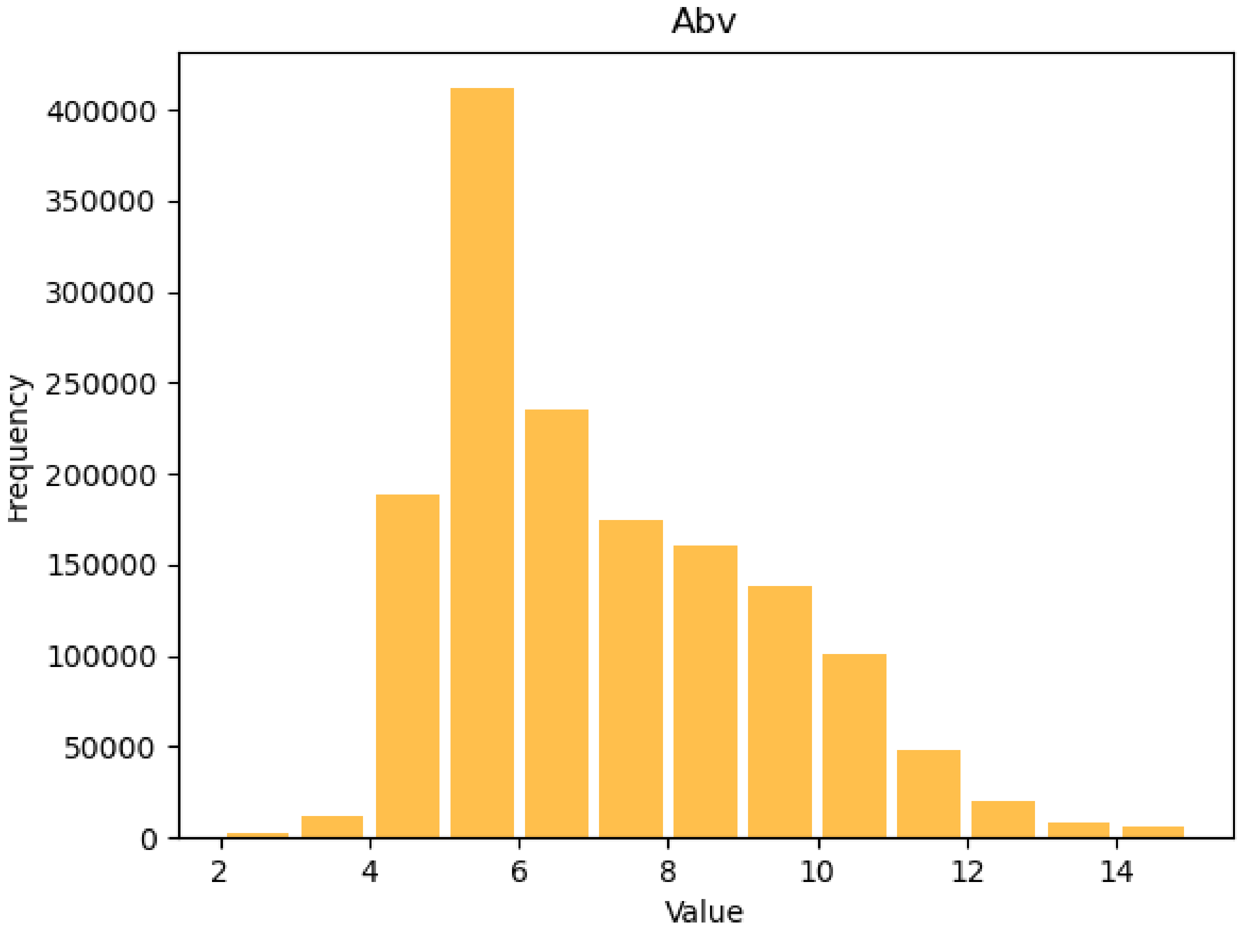


Furthermore, the ratio between the frequency and the value of ”review_palate” and ”review_taste” can be noticed in the following histogram:





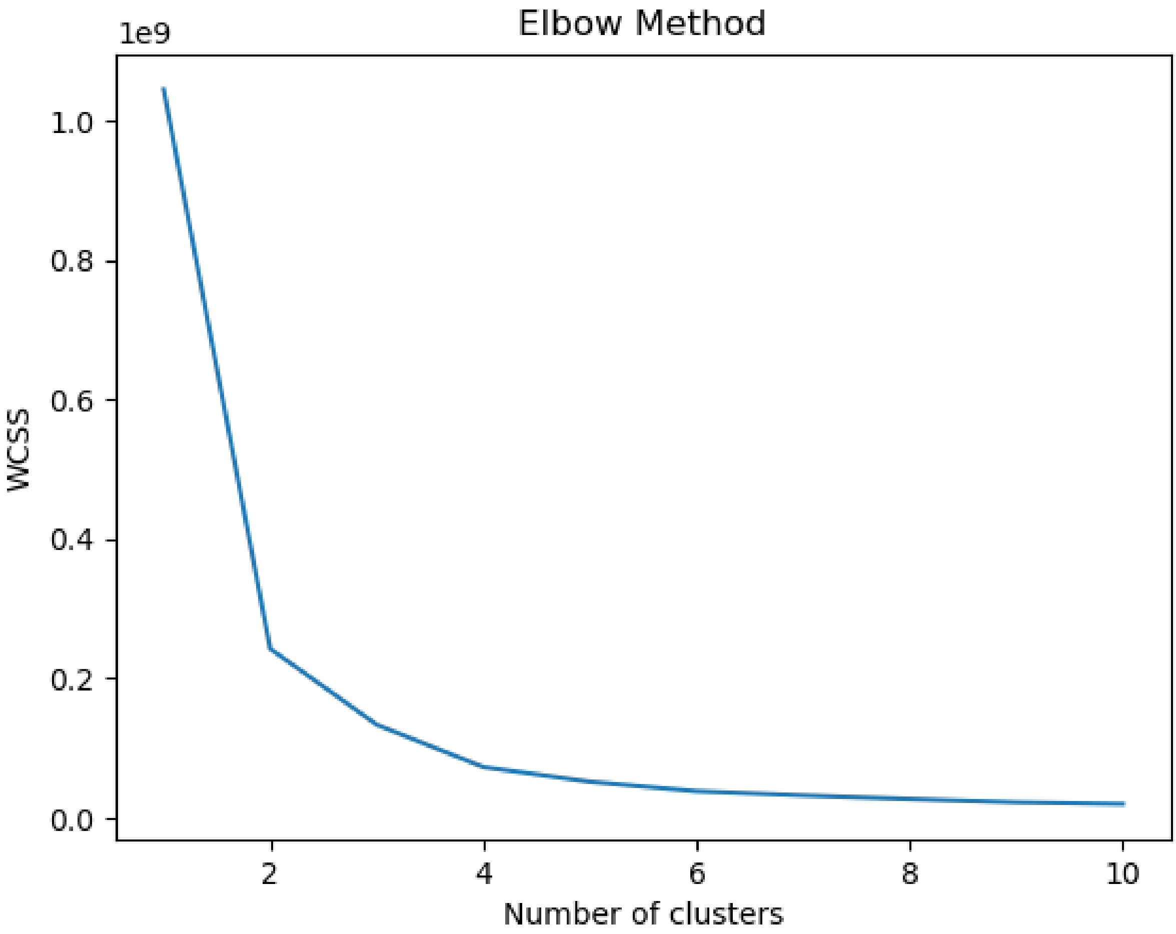
Moreover, a very important feature of the dataset is the beer alcohol. A very interesting statistics can be observed below with the help of the following histogram that is presenting the he ratio between the frequency and the value of ”beer_abv” column:



5.2 KMeans Algorithm

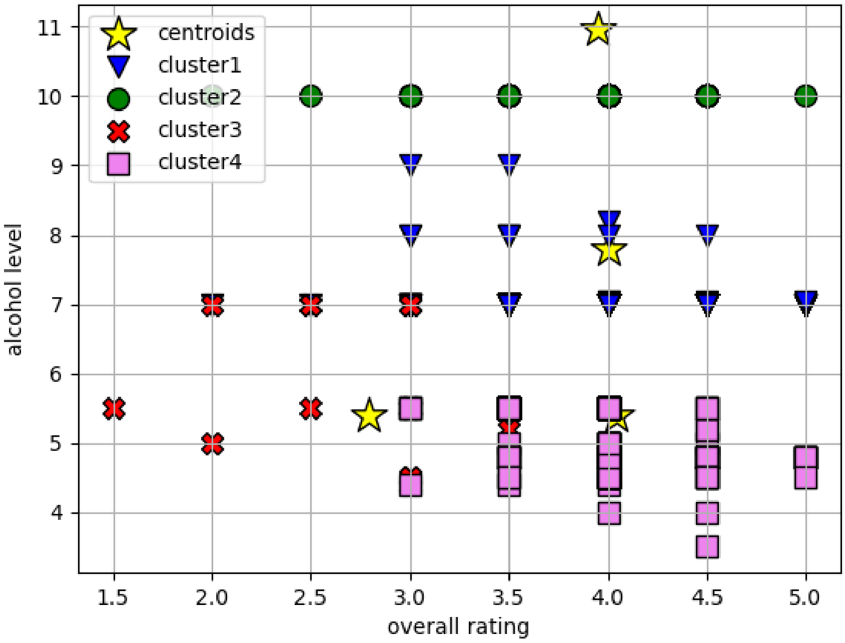
We used the KMeans algorithm in the development of the project in order to classify the data into a certain number of clusters. At the beginning, we made some attempts in order to understand better the mechanism and to observe the behavior of the algorithm on the dataset on different number of clusters.

Afterwards, we applied the elbow method in order to find the ideal number of clusters that are necessary for our data classification. We applied the KMeans algorithm from 1 to 11 clusters in order to observe the evolution of data. We noticed that the point after which the data is being stabilized was not so obvious and we decided to choose the number of clusters to be 4 after we made several attempts and noticed the data evolution. Finally, our Elbow method graphic looked as below:

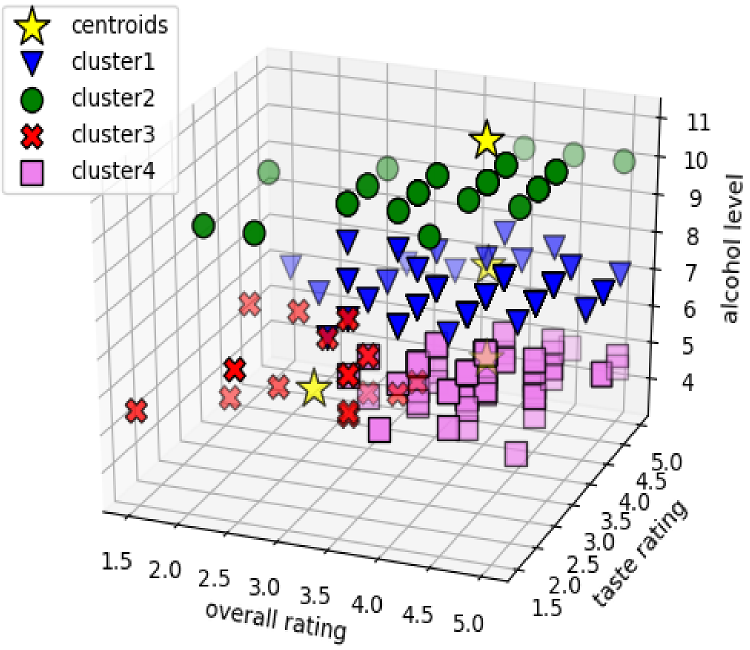


With n=4 clusters we obtained the following results:

2D Visualization of KMeans algorithm with n=4 clusters



3D Visualization of KMeans algorithm with n=4 clusters



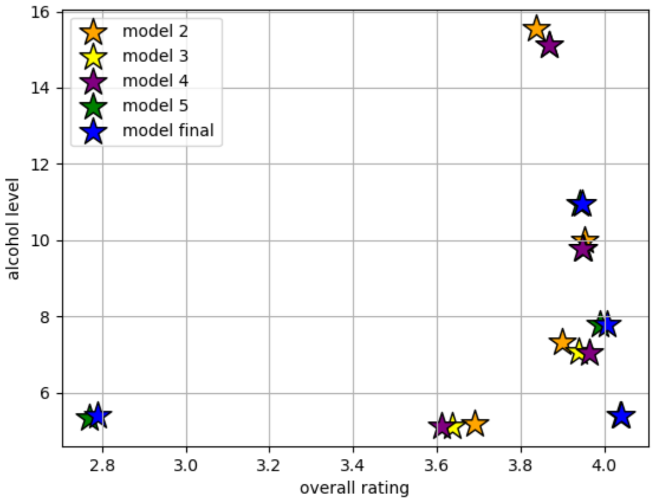
5.3 Data models construction

The next step in the flow of the project was to construct the data models. We began this construction from one feature and we added progressively, one by one, the rest of features, observing their behavior. We were interested in the cluster centers and at each step we visualized the cluster centers and their evolution.

The first model we took care of was "beer_abv". Then, we included also "review_overall" feature and up to this point the model contained "beer_abv" and "review_overall" features. "review_aroma" was the next column introduced, followed by "review_appearance". It was the "review_taste" feature which completed our data model that finally contained "beer_abv", "review_overall", "review_aroma", "review_appearance" and "review_taste" features. With the model finalized, we were able to visualize all the cluster centers and the model behavior.

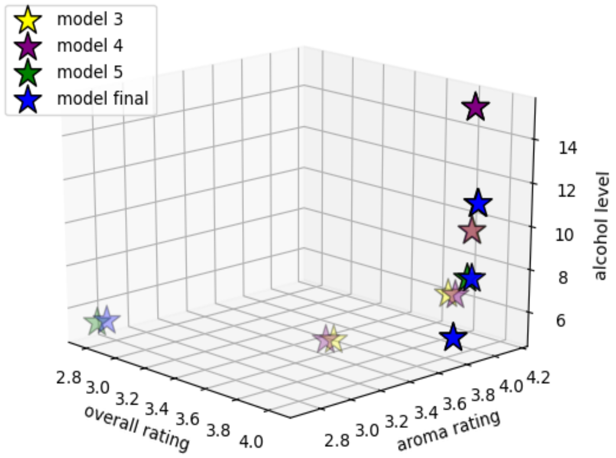
Below, we can notice the cluster centers in 2D of all models except the first:

2D Visualization of Cluster Centers



We can also visualise the cluster centers in 3D:

3D Visualization of Cluster Centers

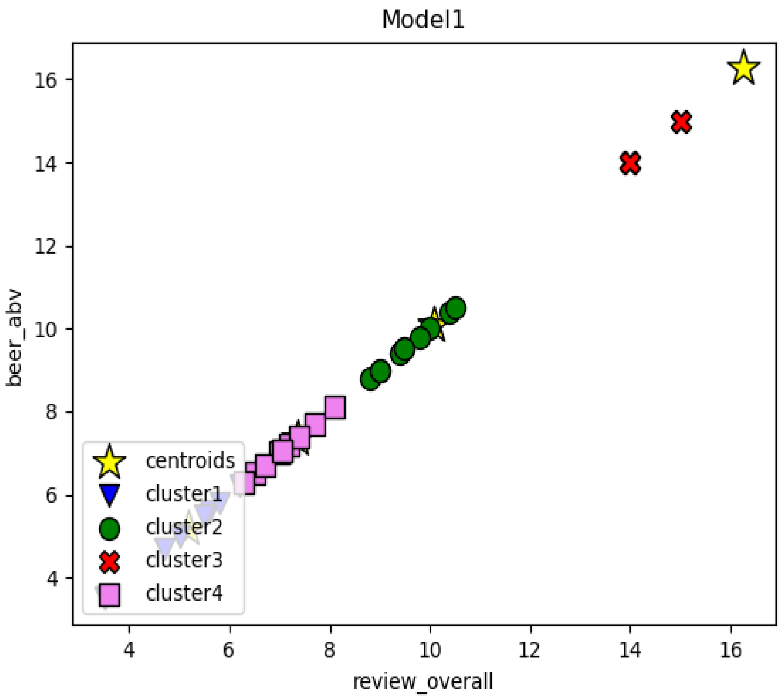


5.4 Models visualization

In this section we will present some results that we obtained during the process of model construction.

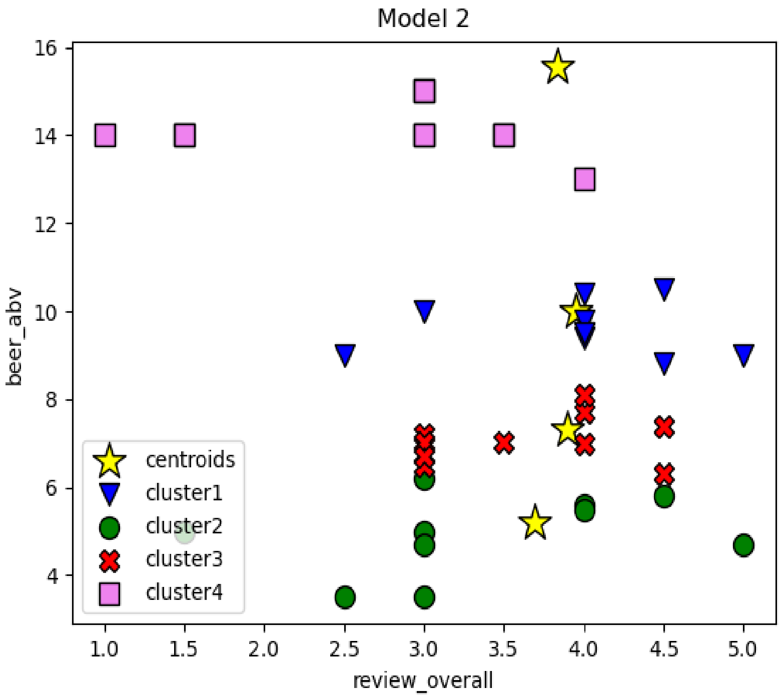
Firstly, we can observe our first model that contained the feature ”beer_abv”:

Visualization of First Model



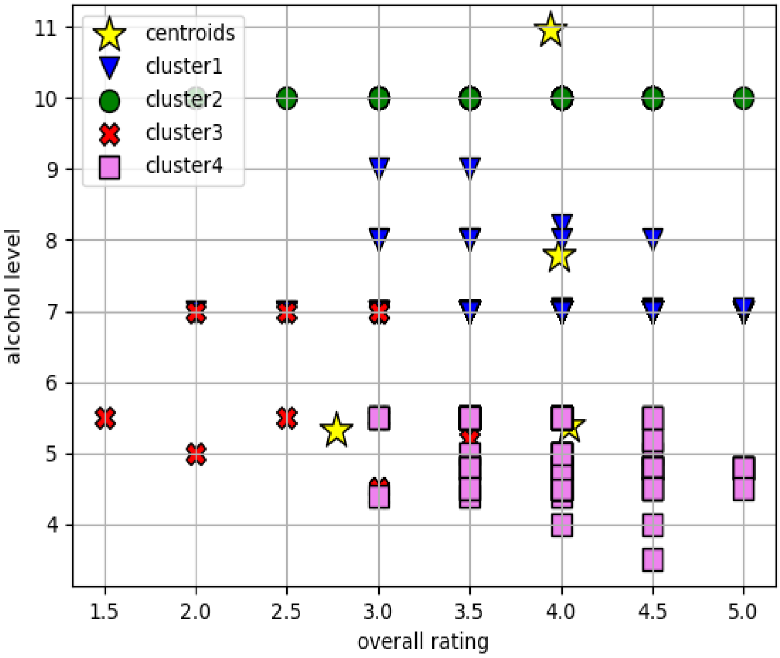
Secondly, we are able to see the second model, which included "beer_abv" and "review_overall":

Visualization of Second Model



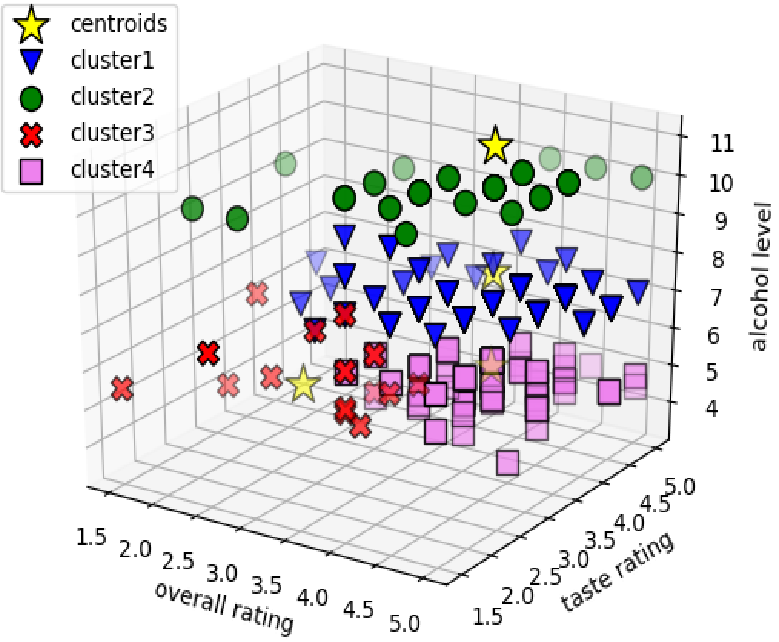
At the end of the process of model construction, we were able to see the model containing the features: "beer_abv", "review_overall", "review_aroma", "review_appearance" and "review_taste". Below, a 2D visualization is presented:

Visualization of Final Model 2D



Also, we obtained a 3D visualization:

Visualization of Final Model 3D



5.5 Beer Recommendation

The beer recommendation system is using a data set belonging to a specific user containing reviews that the user has given. The system uses that data set and based on the clusters obtained from the original data set by applying the K-Means algorithm tries to recommend several beers for the users.

The input, which is containing user reviews is processed. All the entries are predicted using the K-Means model, then a histogram is calculated, the number of elements in each cluster is found. At this point we have information that tells us from which cluster the user has drunk most of the beers. Processing also includes finding the entry with highest review, so we know the beer that user like the most.

After having the highest rated beer, the clusters and number of beers in descending order we can start looking for recommendations in the original database. We considered the search itself could consist of finding several beers following the rules:

- 4 beers from the cluster that contains the highest rated beer
- 3 beers from the cluster with the highest number of beers
- 2 beers from the cluster with second highest number of beers
- 1 beer from the cluster with third highest number of beers

having these constraints, the set of recommended beers varied, since there are beers that probably the user would like and items that can be discovered. The variety is also ensured by starting the search from different places in the original database.

The found beers are enumerated in a simple, readable format which is: brewery, beer name, beer type, abv.

Chapter 6

Bibliography

[1]: https://www.kaggle.com/xvivancos/tutorial-clustering-wines-with-k-means?scriptVersionId=30795
[2]: https://www.kaggle.com/rdoume/beerreviews
https://towardsdatascience.com/unsupervised-learning-and-data-clustering-eeecb78b422a
https://scikit-learn.org/stable/unsupervised_learning.html#unsupervised-learning
https://scikit-learn.org/stable/tutorial/index.html
https://www.guru99.com/unsupervised-machine-learning.html