# Stakeholder Report - SDS 2019 - M1: Group Assignment

Group Jonas Røge Jepsen, Michael Bering Olesen, Tobias Maltha Christensen

**Problem statement:**

We are working a with coffee bean review dataset, and want to see if it is possible to predict the quality of the cupper (the person rating the coffee) based on the features in the dataset.

**Description of data acquisition:**

We found “Coffee Beans Review” on Kaggle. The data on Kaggle comes from a Github, which have scraped the reviews from the Coffee Quality Institute. The reviews are made by Cuppers, and they are therefore subject some subjectivity. However, these reviewers (Cuppers) are educated through courses and exams on how to rate coffee. The data contains 1319 rows and 44 columns. Some of the columns contains nulls or faulty data, which needs to be cleaned or filled.

**Data preparation**

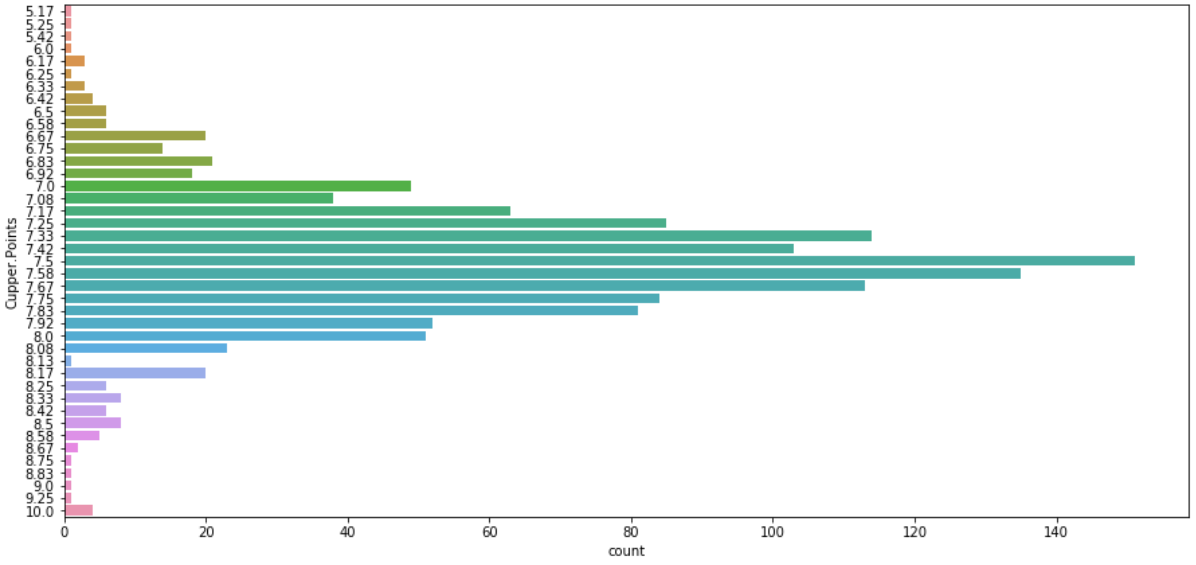
When looking at our data we observed different errors. These errors consisted of typing errors, wrong data types and missing values, which had to be dealt with, before we could move on and apply machine learning. Furthermore we analysed every column for relevance, data, datatypes and picked 17 columns out of the 44 columns. Through different filtering methods we ended up deleting 13 rows in total from our data set, keeping it intact for EDA, unsupervised- and supervised machine learning. We ended up with 1306 rows and 17 columns.

**Preprocessing the Data**

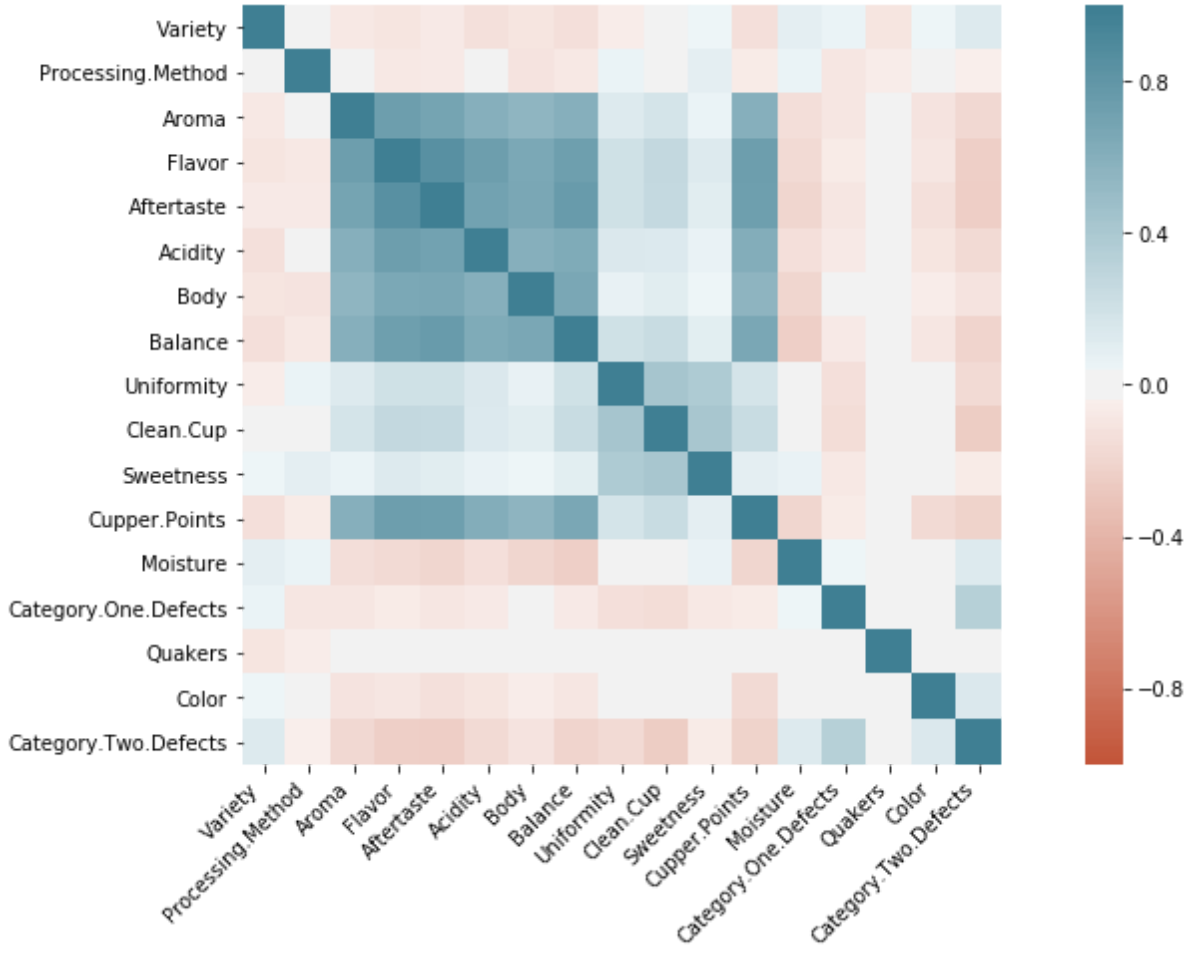
Before starting on unsupervised and supervised machine learning, it was necessary to do some preprocessing of the data. We split our data into X and y, X is the features that we use to predict and y is what we want to predict (Cupper.Points). We then scale the values in X, scaling ensures that the scales for each feature is the same, and that no features are given higher og lower weight based on the scale of their values. Finally we split the datasets into train and test datasets. The train dataset is used to train the algorithms and the test dataset is used to test how good the algorithms are at predicting. This results in our X data having 16 features.

**EDA - Exploratory Data Analysis**

In the Exploratory Data Analysis(EDA) we explored the dataset. We found some interesting statistics about the features of the data, such as the minimum, maximum and mean of the "Cupper.Points". It was noticeable that the minimum score given was 5.17 and the maximum was 10. Moreover, the mean score is 7.5. We did a visualization of the scores given, which made it possible to see that most of the given scores lies in the range between 7 and 8.



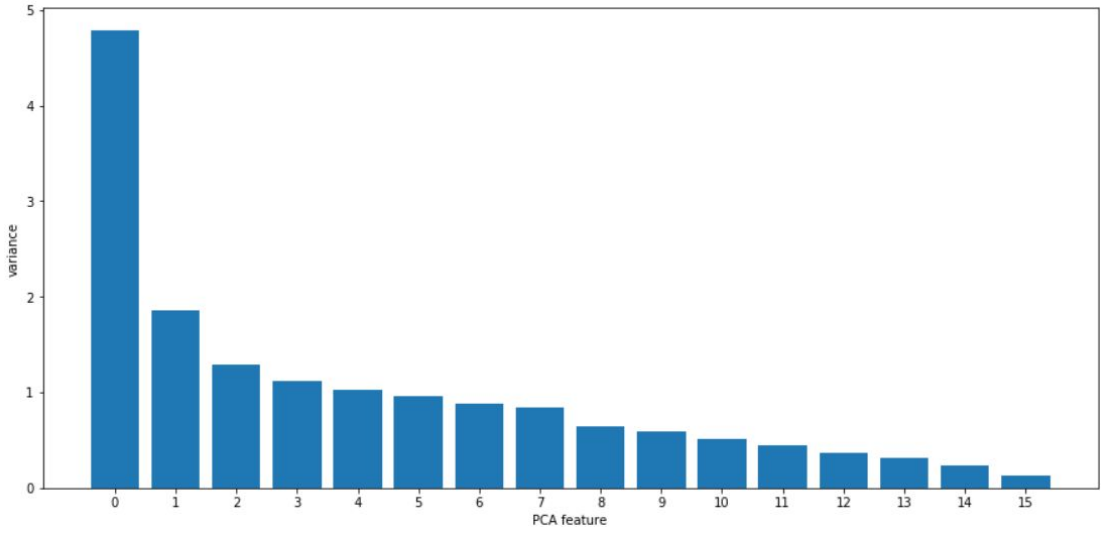

Furthermore in the EDA we created a heatmap based on the correlation of the features in the dataset, to find any correlations between them. The heatmap can be seen below.




The heatmap makes it possible to see a lot of different correlations between the features both positively and negatively. The most distinguishing positive correlation is possibly between our numerical values for describing the coffee together with the Cupper.Points. It it also noticeable that a feature such as Quakers have limited correlation to most of the other features. Cupper.Points being as correlated as it is, it is a great target valuable to predict.

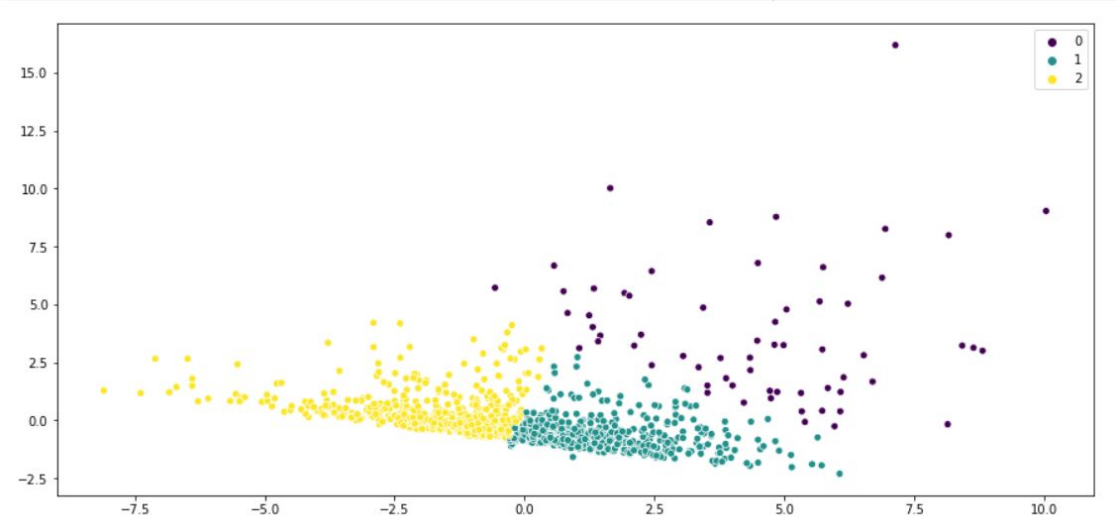
**Unsupervised Machine Learning**

For further data exploration and pattern recognition we used different methods within unsupervised learning. The first method we used was ‘Principal Component Analysis’(PCA) which calculates the optimal size for datasets, while not losing important data.




An analysis of how many components we should use for our PCA resulted in us using three components. By including more than three components we get a diminishing amount of variance for each feature.

The other method is KMeans, which clusters the data. To determine how many clusters we should use on KMeans, we used the elbow method. This resulted in us using three clusters, and based on that the following graph was created, which shows the clusters by color.

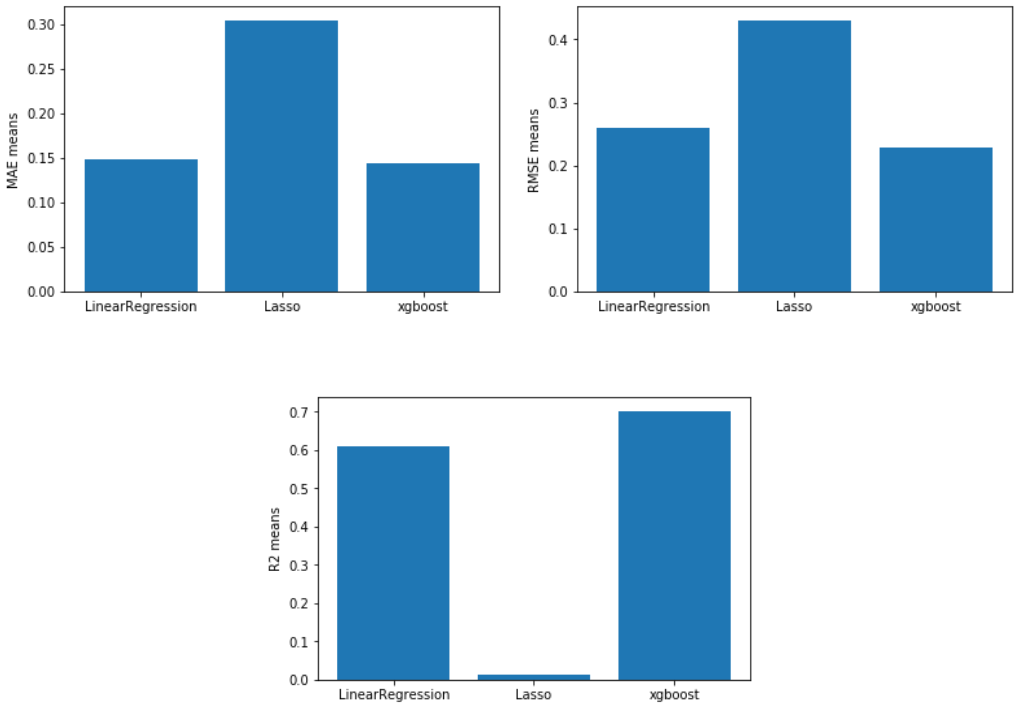



By further analysis of the KMeans with plots and cross tabulation, we could not find any significant patterns. However, the rating features such Aroma, Flavor, Acidity etc.  from our

dataset showed some minor patterns in regard to the clusters, but not strong enough to make a conclusion.

**Supervised Machine Learning**

During supervised machine learning we tried to predict Cupper.Points by using three different algorithms. Because we tried to predict Cupper.Points, which is a continuous variable and not a category, we were working with what is called a regression problem. Based on this we choose to work with the three algorithms **LinearRegression, Lasso and Xgboost.** The three algorithms were all trained using the data, tested and evaluated in regard to their performance. This was done by for instance using cross validation, which insures that the results obtained are not influenced by the way the dataset is split in train and test. Further the algorithms was evaluated based on the three metrics MAE, RMSE and R2. The result obtained in regard to each algorithm is displayed in the figures below.

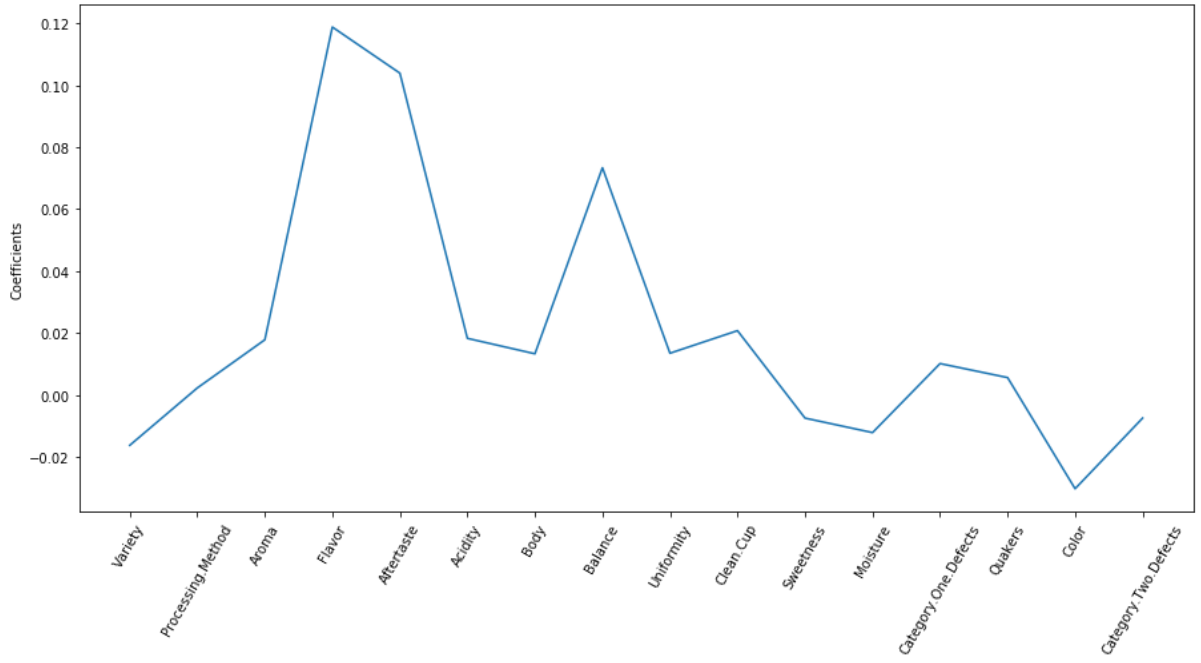



Based on these metrics Xgboost seems to be the algorithm that performs the best, and in relation to predicting the value of Cupper.Points and the value of MAE we can expect our prediction using this algorithm to be off by 0.1436 on average, meaning that if the actual

value of Cupper.Points is 6.3 a prediction based on Xgboost is likely to be 6.4436 or 6.1564 on average.

**Hyperparameter tuning**

It is possible to improve the performance of an algorithm by tuning the algorithms hyperparameters. These can be tuned by finding the best possible value of the parameter. We did this for the Lasso algorithm, which was the one that performed the worst. After performing hyperparameter tuning on it was able to perform at the same level as our LinearRegression algorithm. The hyperparameter tuning resulted in the algorithm giving each feature a different weight, how it weighs each feature after hyperarameter tuning can be seen on the below graph.




As can be seen, the improved lasso algorithm gives a higher weight to Flavor, Aftertaste and Balance, and gives little weight to acidity and body.

**Conclusion**

Our goal and problem statement was to see if it was possible to predict the Cupper.Point (The quality of the person reviewing the data). When using the Xgboost algorithm the score of

MAE was 0.1436, meaning that the prediction would be of by 0.14. Therefore it is possible for our algorithm to predict the quality of the cupper with some error.