Problem statement:

This dataset contains expert ratings of over 1,700 individual chocolate bars, along with information on their regional origin, percentage of cocoa, the variety of chocolate beans used, and where the beans were grown. We have to predict the coco percent by using a different model.

Column names	**Type of data**	**Relevance**
Company	Object	Names of the company
Name	Object	A Place in country
REF	iint	Reference id
Review	int	Review in year
Cocoa percent	Float	Percentage of cocoa bean
Company location	Objective	Company located place
rating	Float	Cocoa rating
Bean type	Objective	Type of cocoa bean
Origin	Objective	Coco produced personal names

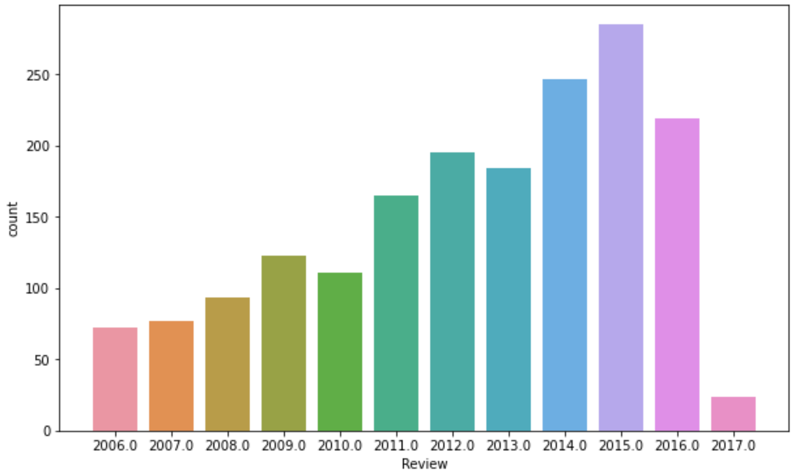
Data Pre-processing

- Except Class variable all the columns are numeric
- Checking if there are any null, nan, or duplicated values. the data type is numeric or objective. We found two null columns so filled null columns with ‘unknown’ .then we filled the bean &origin columns with the most frequent value. Finally, we don’t have any null values in the data
- Then I did normalize the data by using the norm function

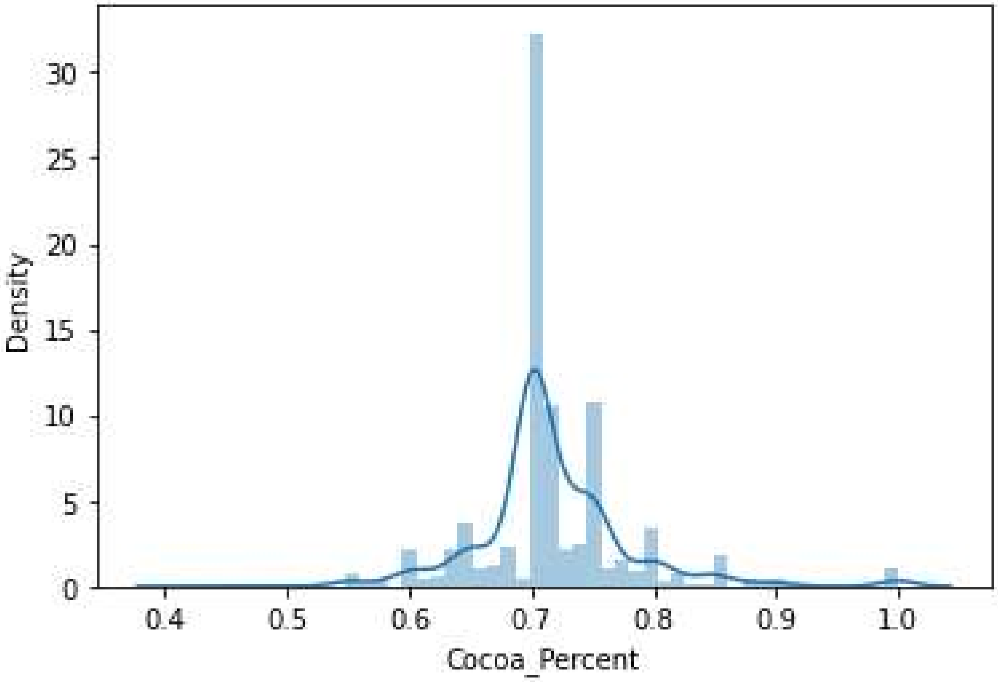
Exploratory Data Analysis

Univariate Analysis

- We saw reviews by year using a plot diagram

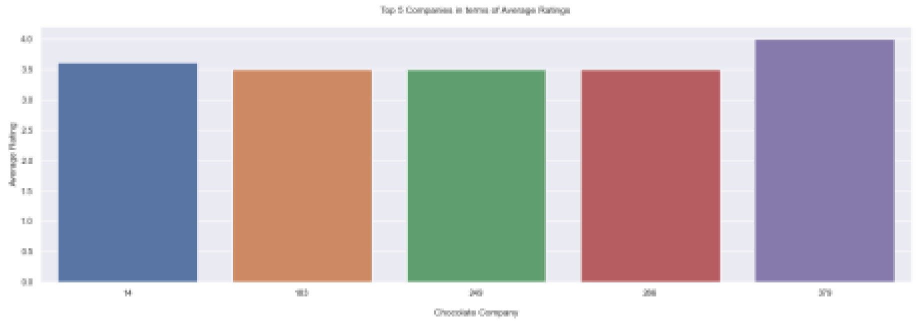


- In the above plot we can see that 2015 has a mostly reviewed year and 2017 has a very less number of reviews by customers.
- Distplot represents the univariate distribution. data distribution of a variable against the density distribution and it seems to be a lower whisker.



- By the above plot shows that the highest cocoa percent was 0.7 some companies are provided that was the highest density of cocoa percent value.
- Then we did label encoding for objected data that convert into numeric data.

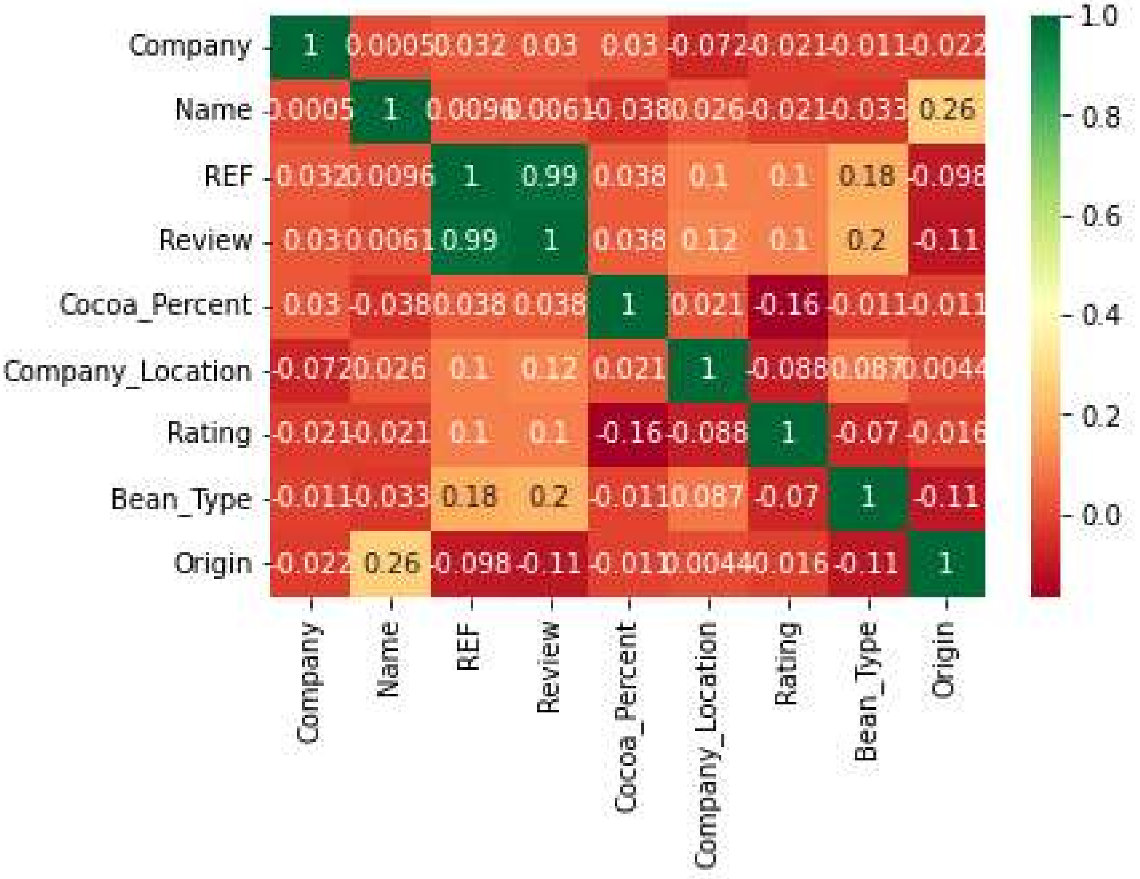
- By using the histogram shape of each column visually.
- The hist plot to see the distribution of the dataset but this time we are using this visualization to see the changes that we can see after those null values are removed from the dataset and we can see the difference
- We performed the scale function to normalize the values except the rating column because it was the target variable.



- Above plot shows that people rated the coco mostly of the people rated cocoa that 3 to 4. very less rating of people 5 &1

Bivariate analysis.

- By using the map function we can see how much each column is co-related to another column. Because we can see how each column is dependent on another column



Model Building:

we split the data into 80% for training and 20% for testing and then we performed a different type of model.

Staking:

- Stacking or Stacked Generalization is an ensemble machine learning algorithm. It uses a meta-learning algorithm to learn how to best combine the predictions from two or more base machine-learning algorithms

LogisticRegression	=70%
KNeighborsClassifier	=71%
DecisionTreeClassifier	=66%
SVC	=70%
GaussianNB	=71%
get_stacking	=71%

-

- We are using models in staking are Knn, logistic-regression, Decissiontree, SVC, gaussian, and get_stacking models.

- After performing these all models we are getting good accuracy for knn and Bayes models compare to others.

Voting:

- A voting classifier is a machine learning estimator that trains various base models or estimators and predicts based on aggregating the findings of each base estimator.

Hard Voting: Voting is calculated on the predicted output class.

- In hard voting using classifiers are knn, linear and SVM model
- For testing the model we get an accuracy was 69%

Soft voting: Voting is calculated on the predicted probability of the output class.

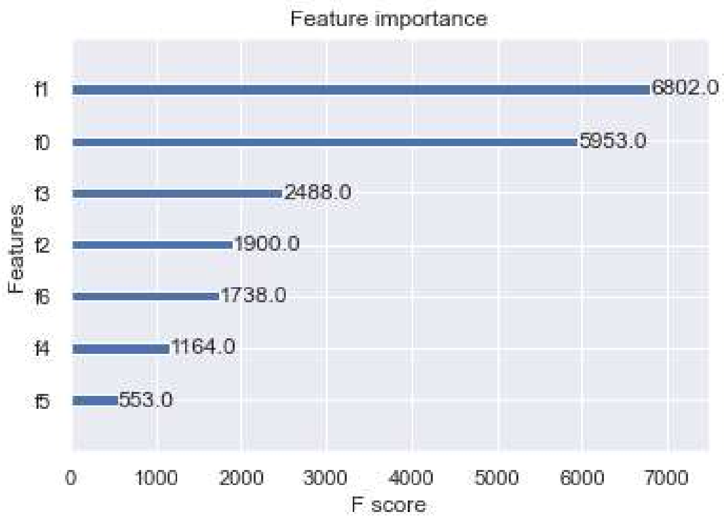
- For soft voting we are using classifier models knn, SVM, and NB models.

- For test data we got an accuracy was 70%

Bagging classifier: Bagging involves fitting many decision trees on different samples of the same dataset and averaging the predictions.

- For predicting test data accuracy was 69%
- For train predicting data was 100% which means it was overfitting.

XGBoosting: Extreme Gradient Boosting (XGBoost) improves gradient boosting for computational speed and scale in several ways. The key features of XGBoost are parallelization, distributed computing, cache optimization, and out-of-core processing.



- In this predicting model very bad results of we are getting like -ve prediction accuracy. So this model is not supported cocoa rating

Gradientclassification model:

- GB does not give incorrectly classified items more weight. This method attempts to generate accurate results initially instead of correcting errors throughout the process, like AdaBoost.
- For test accuracy prediction we are getting 68% and train prediction 80%

Hyperparameter tuning:

In this model finding a set of optimal hyperparameter values for learning the algorithm while applying this model to any data set optimized the data. This model maximizes performance and minimizes errors.

- For test data prediction we are getting 69% accuracy
- For train data prediction 70% accuracy. this random forecasting model gives good result

Conclusion & Benifits:

- After using all these cocoa ratings we can build a machine-learning model Stacking given high-accuracy prediction.
- Because of performing model building we know which company gives more cocoa flavor than other companies and people like how much cocoa percent. Due to this company can provide the best to customers they can get more customers and clients to expand their business all over the world.