Introduction

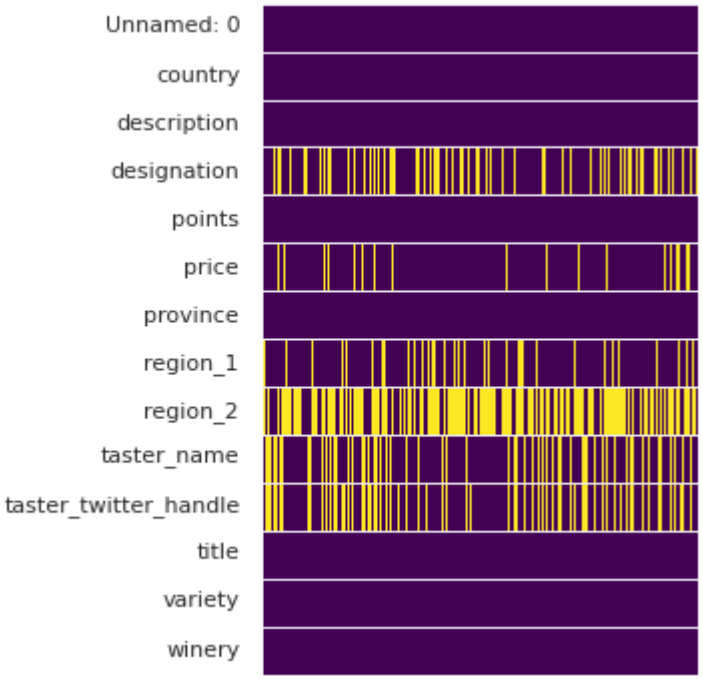
For a long time, wine has been a part of human culture. Wines have always been a topic of discussion, and thus the wine industry has also flourished. With an increase in demand for wine, the production has also increased. Furthermore, from such a large variety of collections, people like to choose the reviewed wine because no one wants to invest their money in something they will not like. Some interesting cases portray how critical wine reviewing has become. One of those is a man insuring his nose for $8mm in 2008.

Here, in this project, it was tried to understand whether the description of a wine, as given by an expert, can be used to understand the points received by it.

Exploratory Data Analysis

Data Cleaning

At first, all the duplicates in the data were dropped. On checking whether the data had null values or not, it was found that most of the null values were in price,taster_name,taster_twitter_handle, and designation. To deal with this, the price was filled with the median value of the price column. Then the unwanted columns were dropped, including designation and taster_twitter_handle. Then remaining null columns were dropped.



Country Feature

Interpreting country distribution, we can see that the USA is the most reviewed country. Using point plot() showed the distribution of country and price while analysing the plot, we can see that the wine price in England is high. Interpreting the distribution of country and price plotted using strip plot(), it was seen that France has the most expensive and cheap wines. Interpreting the distribution

of country and point plotted using point plot, we can see England, India and Austria at the top. While using a strip plot, we can see that Italy, France, the US, Portugal and Spain are the top scores.

Variety Feature

A variety of wine means wine made from a specific grape. When a wine bottle shows a varietal designation on the label, it means that the wine in the bottle is at least 75% that grape variety. While analysing the distribution of wine reviews of top 20 varieties, Pinot Noir was the most reviewed variety. Interpreting the distribution of variety and price Bordeaux-style Red blend is the most expensive grape type. Analysing in terms of points we can see that majority of the varieties got a high rating. While merging the two plots varieties who got the highest point and the grapes used the cheap wine, we get the Merlot, Chardonnay, Portuguese Red, Syrah and Shiraz are the best varieties to make wine.

Taster Feature

The most frequent taster in the dataset was Roger Voss, who tasted 10000 wines ahead of Michael Schachner, who is at second. Interpreting taster_name and points distribution plotted using box plot() we see that most of the reviewers have the same range in which they give points. There are a few reviewers who are below the range, but it can be because they have reviewed less number of wine.

Description Feature

Interpreting the distribution of word cloud of the description of lowest rated wine, we can see the words heavy, bitter, burnt and words like sour. The most expensive wine had the comments show, dense, elegant in the description. For the highest rated wine word description were vintage, age and ageing appear a lot; this proves that ageing is an essential aspect of the wine. For a wine to get a useful review, ageing, concentration and aroma play an essential role. Interpreting description length and price distribution, we can see that description length had a linear relationship with points.

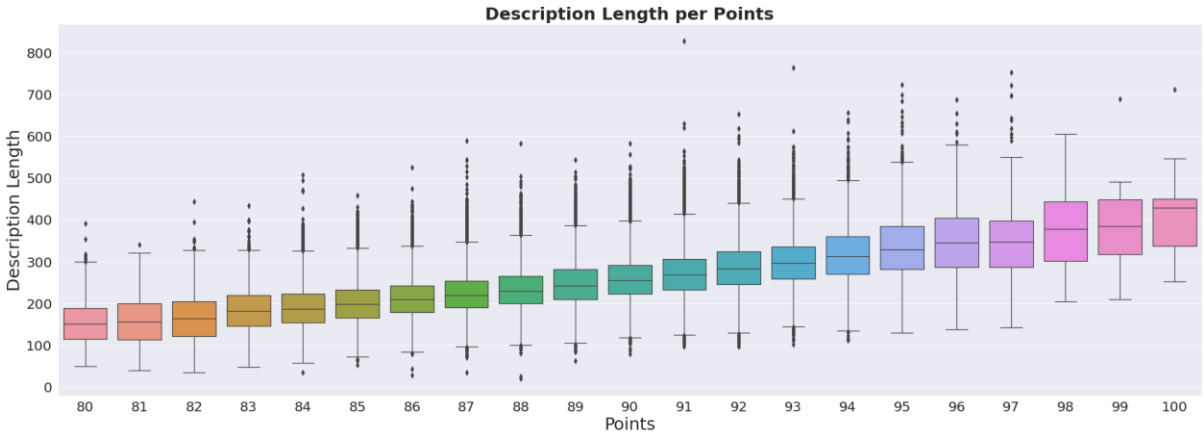
Point Feature

The graph of Points Count distribution looks like a normal distribution. Most of the wines had points between 82to95. Using regplot(), we can see that the highest rated wine is not the most expensive one.

Interpreting the heat map, it was seen there is a very weak positive correlation between price and point. There is a relationship between points and description length. Based on the description, we can create a recommendation system.

Model Building

On doing EDA, we could see the relationship between points given and description length. It can be inferred that an adequate amount of time was given for writing the description of good wines.



To analyse the description and use it for KNN, first, it was vectorised.

Description Vectorisation

Vectorisation is the process of turning a collection of words into feature vectors. This strategy is known as Bag of Words or Bag of n-grams representation. In this, documents are described by word occurrences. The relative position information of the word in the document is ignored.

For example, if our dictionary of words contains {Python, is, the, not, great}, and we want to vectorise the text “Python is great”, we would have the following vector: (1, 1, 0, 0, 1). A few vectorisation algorithms are available, the most famous being:

- CountVectorizer: vectorises a word weighted by word counting.
- TF-IDF Vectorizer: the weight increases proportionally to count, but is offset by the frequency of the word in the total corpus; this is called the IDF (Inverse Document Frequency). IDF allows the Vectorizer to adjust weights with frequent words like “the”, “a” etc.…
- n-grams
- stopwords

For this problem, Count Vectorizer was used.

CountVectorizer:

Count Vectorizer is a tool provided by scikit learn, that vectorises sentences based on the frequency of the word in the sentence. Count Vectorizer can create sparse matrix having columns of unique words, and rows representing documents/sentences/paragraphs.

To specify the count vectoriser, ngrams also need to set. N-gram is a contiguous sequence of n items from a given sample of text or speech. The items can be phonemes, syllables, letters, words or base pairs according to the application. The n-grams typically are collected from a text or speech corpus. When the items are words, n-grams may also be called shingles.

```
vectorizer = TfidfVectorizer(ngram_range = (2, 3), min_df=5,
                             stop_words='english',
                             max_df=.5)
```

Here,

TfidVectorizer

It is a tool provided by scikit, which stands for “Term Frequency – Inverse Document Frequency”. TF-IDF is a numerical statistic which measures the importance of the word in a document. Where Term Frequency means the frequency of words in the documents. Inverse Document Frequency measures whether the word is rare or expected in the document.

Ngram_range

ngram range takes two parameters, ngram_range(2,3) means to take a minimum of 2 words as sequence and maximum 3.

min_df,max_df

where min_df means ignoring the words that occur too rarely. Moreover, max_df is used to ignore the words that occur too frequently.

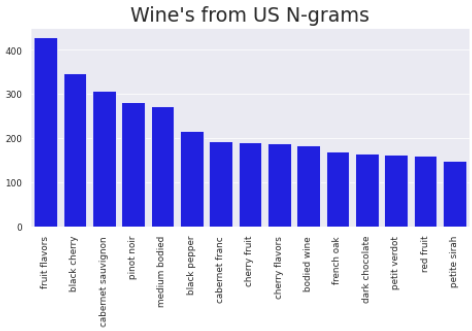
Here min_df = 5 means, ignore the words that occur in less than five documents.

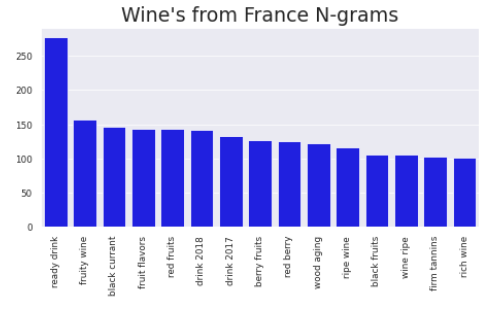
Furthermore, max_df = 0.5 means, ignore the words that occur in more than 50% of documents.

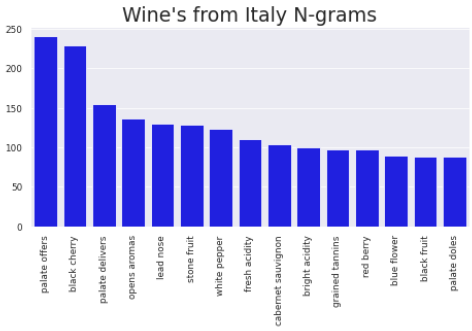
Stop_words

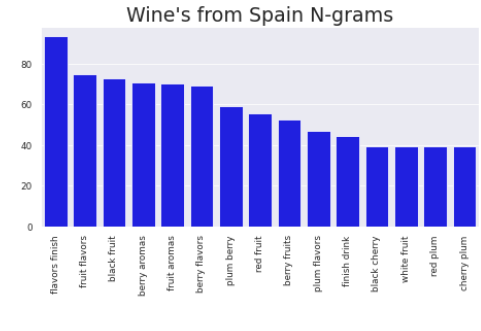
stop_words is used to filter out the common words that are mainly used in that language. For example, stop_words = ‘English’ will filter out words like, ‘to’, ‘the’, ‘in’, ‘of’ etc.

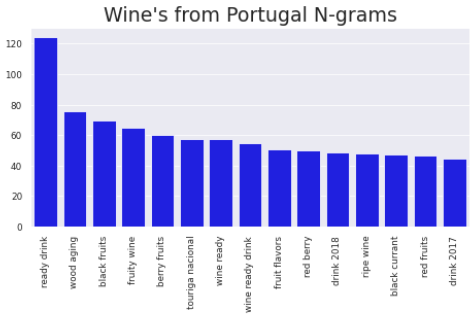
Barplots were made grouped by different countries and their vectorised words. Using bar plots, it could be seen that there were a few words that frequently appear in different countries. Similarities were also seen.

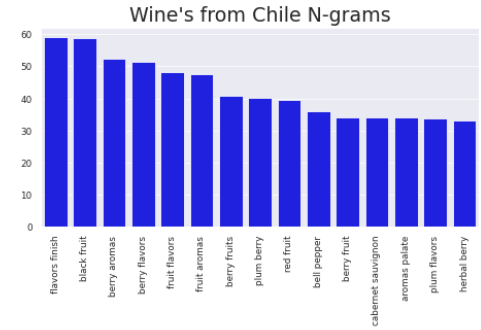

Wine's from US N-grams
fruit flavors
black cherry
cabernet sauvignon
pinot noir
medium bodied
black pepper
cabernet franc
cherry fruit
cherry flavors
bodied wine
french oak
dark chocolate
petit verdot
red fruit
petite sirah

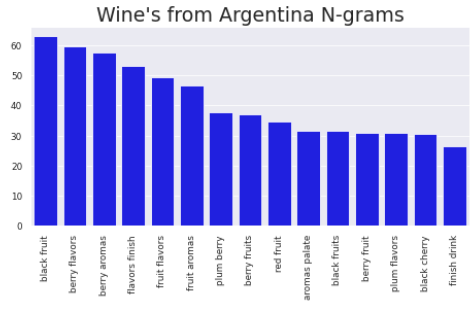

Wine's from France N-grams
ready drink
fruity wine
black currant
fruit flavors
red fruits
drink 2018
drink 2017
berry fruits
red berry
wood aging
ripe wine
black fruits
wine ripe
firm tannins
rich wine

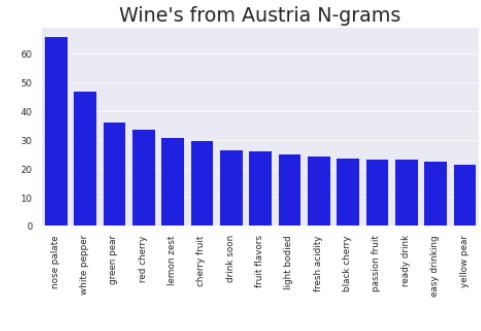

Wine's from Italy N-grams
palate offers
black cherry
palate delivers
opens aromas
lead nose
stone fruit
white pepper
fresh acidity
cabernet sauvignon
bright acidity
grained tannins
red berry
blue flower
black fruit
palate doles

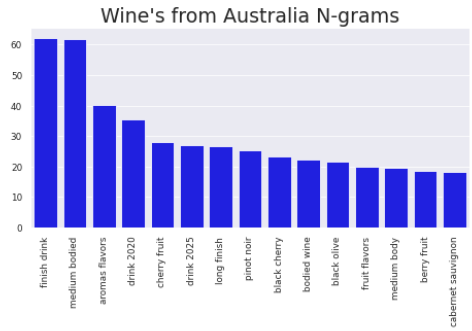

Wine's from Spain N-grams
flavors finish
fruit flavors
black fruit
berry aromas
fruit aromas
berry flavors
plum berry
red fruit
berry fruits
plum flavors
finish drink
black cherry
white fruit
red plum
cherry plum

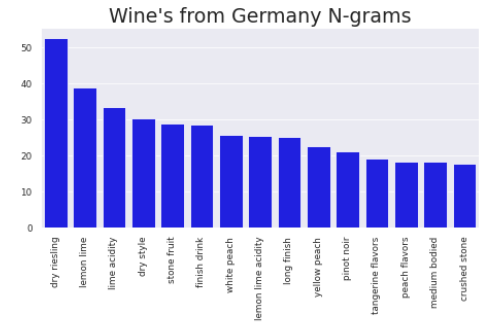

Wine's from Portugal N-grams
ready drink
wood aging
black fruits
fruity wine
berry fruits
touriga nacional
wine ready
wine ready drink
fruit flavors
red berry
drink 2018
ripe wine
black currant
red fruits
drink 2017


Wine's from Chile N-grams
flavors finish
black fruit
berry aromas
berry flavors
fruit flavors
fruit aromas
berry fruits
plum berry
red fruit
bell pepper
berry fruit
cabernet sauvignon
aromas palate
plum flavors
herbal berry


Wine's from Argentina N-grams
black fruit
berry flavors
berry aromas
flavors finish
fruit flavors
fruit aromas
plum berry
berry fruits
red fruit
aromas palate
black fruits
berry fruit
plum flavors
black cherry
finish drink


Wine's from Austria N-grams
nose palate
white pepper
green pear
red cherry
lemon zest
cherry fruit
drink soon
fruit flavors
light bodied
fresh acidity
black cherry
passion fruit
ready drink
easy drinking
yellow pear


Wine's from Australia N-grams
finish drink
medium bodied
aromas flavors
drink 2020
cherry fruit
drink 2025
long finish
pinot noir
black cherry
bodied wine
black olive
fruit flavors
medium body
berry fruit
cabernet sauvignon


Wine's from Germany N-grams
dry riesling
lemon lime
lime acidity
dry style
stone fruit
finish drink
white peach
lemon lime acidity
long finish
yellow peach
pinot noir
tangerine flavors
peach flavors
medium bodied
crushed stone

Sentiment Analysis

Sentiment Analysis is a technique used in Natural Language Processing, where a given piece of text is analysed to detect its sentiment.

For this problem, Sentiment Intensity Analyser from Vader was used.

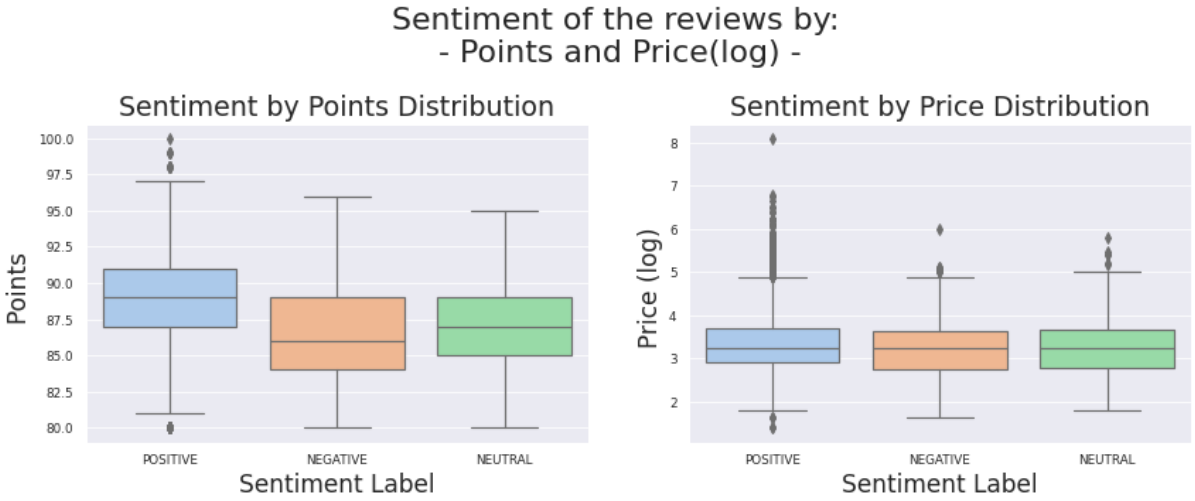
VADER

VADER (Valence Aware Dictionary and sEntiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media. VADER uses a combination of A sentiment lexicon is a list of lexical features (e.g., words) which are generally labelled according to their semantic orientation as either positive or negative. VADER is also capable of analysing emoticons like "😊","☹".

VADER does not require any training data but is constructed from a generalisable, valence-based, human-curated gold standard sentiment lexicon.

VADER classifies the sentence into positive, negative or neutral. VADER also takes into account, punctuation marks like '!', capitalisation like WOW, emojis and slangs like 'lol' are taken into account, which increases the sentiment.

On plotting sentiment analysis on the description concerning price and point, the following graph was obtained:



Here, it shows that price does not have much effect based on the sentiment of the description, which is obvious. However, it can also be seen that sentiments from description affect the point.

So, the recommender system based on the sentiment and the point involved can be created.

K Nearest Neighbour

KNN algorithm is one of the most straightforward Machine Learning Algorithms that used Supervised and Unsupervised Learning techniques. The optimal choice of value is highly data-dependent. KNN algorithm classifies data into categories based on the similarity between them. KNN algorithm is used for classification and regression, but it is more prevalent in classification problems. KNN algorithm is non-parametric in nature, which means it does not assume anything about the underlying data.

The similarity is calculated using Euclidean Distance, Minkowski Distance, Manhattan Distance, Cosine Distance, Jaccard Distance and Hamming Distance. The cosine distance metric is used to calculate the similarity between two vectors.

It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in the same direction. It is often used to measure document similarity in text analysis.

It is also used in text analytics to find similarities between two documents by the number of times a particular set of words appear in it.

$$\cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \cdot \|\vec{b}\|}$$

The value ranges from 0 to 1, where 0 means the two vectors are not at all similar, and one means the vectors are 100% similar.

```
knn = NearestNeighbors(n_neighbors=7, algorithm = 'kd_tree',metric = 'cosine')
model_knn = knn.fit(wine_pivot_matrix)
```

Here,

NearestNeighbours

Implements unsupervised nearest neighbours learning

n_neighbors

The number of neighbours that are to seen for distance calculation.

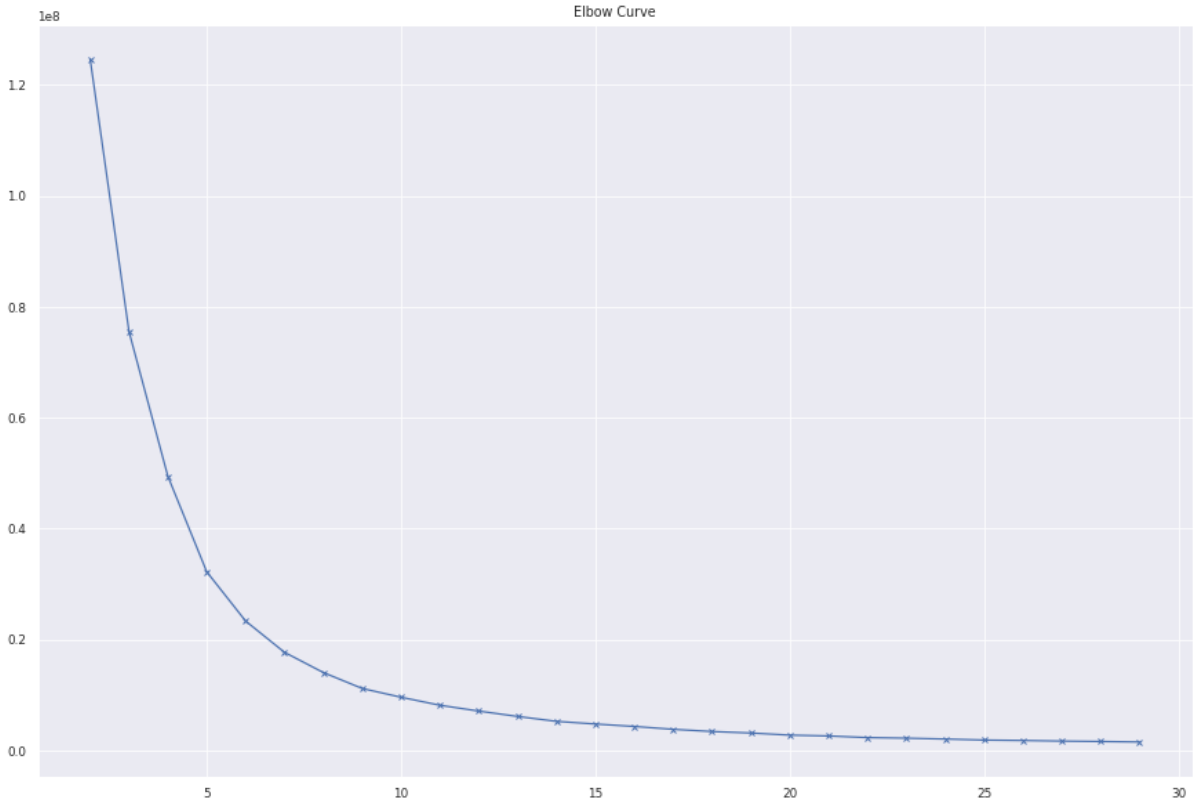
Algorithm

Three algorithms are provided by sklearn library, Brute, KD Tree and Ball Tree. In this problem, KD Tree was used because the amount of data was high, and it was not high dimensional. In the KD tree, various tree structures techniques are used in finding the distance. A general idea to explain this

that, suppose point A is very near to B, and B is very far from C, then A is very far from C, without explicitly calculating them. This reduces the computational cost to $O[DN\log(N)]$ or better.

Choosing K

Since KNN algorithm is a naïve algorithm, so it works for any k. However, for the optimal value of k, the elbow curve is used.



An elbow curve is plotted between the number of clusters and distortions.

```
X = data
distortions = []
for k in range(2,30):
    k_means = KMeans(n_clusters = k)
    k_means.fit(X)
    distortions.append(k_means.inertia_)
```

Here k is iterated from 2 to 38. The values of distortions are calculated for each value of k and distortion and inertia for each value of k is also calculated in the given range.

Here,

Distortion is calculated as the average of the squared distances from the cluster centres of the respective clusters. Mostly Euclidean distance metric is used.

Inertia is the sum of squared distances of samples to their closest cluster centre.

Output and Inference

```
Recommendation for ## Garnacha Blanca ##:
1: Carignan-Grenache with distance: 0.18354918074003612
2: Grenache-Carignan with distance: 0.18986752394464101
3: Trepat with distance: 0.2889539973907437
4: Cariñena-Garnacha with distance: 0.2889539973907437
5: Macabeo with distance: 0.2889539973907437
```

Smaller distance means more similarity. Hence it can be said that *Garnacha Blanca* is more similar to *Carignan-Grenache* that *Trepat*.

References

1. Schurig, R., 2019. Roaldschuring/Wine_Recommender. [online]

GitHub. Available at:

<https://github.com/RoaldSchuring/wine_recommender>

[Accessed 3 December, 2020].

2. Forbes.com. 2008. [online] Available at:

<https://www.forbes.com/2008/03/22/gort-winemaker-nose-face-cx_vr_0319autofacescan02.html#62c6004352f4>

[Accessed 5 December 2020].

3. Wine Enthusiast. 2014. Bordeaux Red Blend Wine Reviews |

Wine Enthusiast Magazine. [online] Available at:

<https://www.winemag.com/varietals/bordeaux-style-red-blend/>

[Accessed 5 December, 2020].

4. Wine Enthusiast. 2016. Terre Rouge 2013 Vin Doux Naturel Muscat Blanc À Petits Grains (Shenandoah Valley (CA)). [online] Available at: <https://www.winemag.com/buying-guide/terre-rouge-2013-vin-doux-naturel-muscat-blanc-a-petit-grain-Shenandoah-valley-ca/>
[Accessed 10 December 2020].

5. Wine Enthusiast. 2017. Von Schleinitz 2015 Apollo Dry Riesling (Mosel). [online] Available at: <https://www.winemag.com/buying-guide/von-schleinitz-2015-apollo-dry-riesling-Mosel/>
[Accessed 12 December 2020].