**Data Science Project – Draft Paper**

**Problem statement and hypothesis**

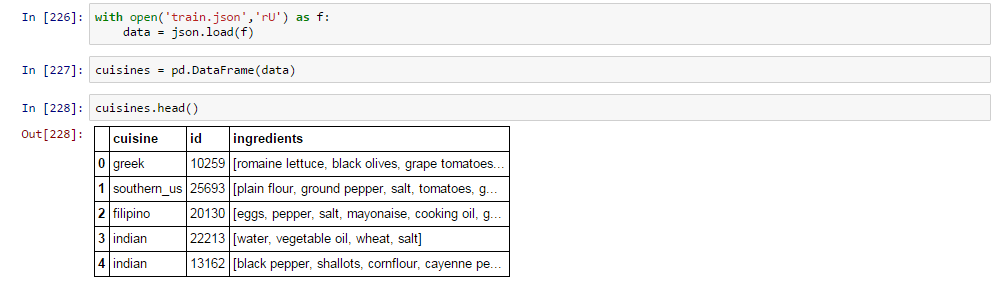
Predict the category of a dish's cuisine given a list of its ingredients

**Description of data set and how it was obtained**

This problem was obtained from a Kaggle competition. Kaggle has provided separate datasets for training and testing. Data was downloaded from the Kaggle website in the form of json files

**Pre-processing steps**

Data in the form of json objects was read into a pandas dataframe



Although Kaggle has provided a dataset for testing, the current dataset needs to be split into testing and training sets. This will help in evaluating the model



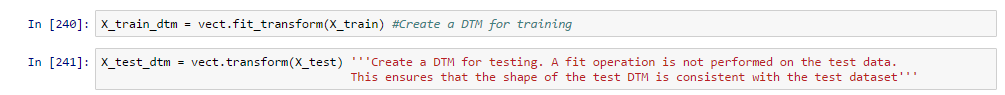
Since the dataset and prediction piece is text based, Natural Language Processing is the obvious choice for pre-processing the data. In order to prepare the data for the model, it is important to identify and weigh words in the text data. This was accomplished using Count Vectorizer.



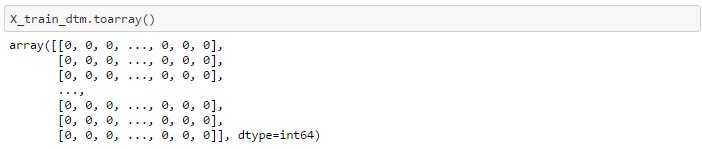
Count Vectorizer accepts data only in the string format. It is important to ensure that the dataset to be fit into Count Vectorizer has a data type ‘String’

Count Vectorizer parses the text data, identifies words in the data and fits it into a Document Term Matrix (DTM) that is built based on frequency of terms in the text piece to be analysed



This will print out a matrix of integers with rows corresponding to each row in the dataset and number of columns corresponding to the numeric value assigned to each word in the dataset

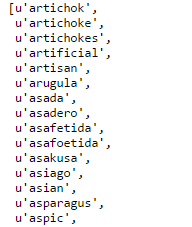


Data exploration and visualizations

Once the data is fitted, Count Vectorizer identifies individual words in the text data. It is possible to view a list of the words that Count Vectorizer has identified



This will print out individual words in the data. For example:



Some observations:

* Words are organized alphabetically from A-Z
* Count vectorizer automatically strips out special characters like (‘;’ ‘,’ ‘.’ ‘?’) etc.
* Count vectorizer splits words separated by a space in between them. Eg. ‘black pepper’ will be separated into ‘black’ ‘pepper’

**Feature selection for analysis:**

Features in this case were supplied by the Kaggle competition. Since the prediction is for the type of cuisine, the features provided are ingredients for each cuisine.

Although features have been specified, the data in the feature set will have to be filtered for best results. Since the problem here is classifying ingredients, common features do not really add value. There is definitely a lot of intersection between each cuisine which can be neglected.

**Modeling process and model evaluation:**

Since this problem is a classification problem, models under consideration are Naïve Bayes and Logistic Regression. Naïve Bayes was the first choice because:

* Processing speed faster than logistic regression
* Naïve Bayes is good in neglecting irrelevant features which is an advantage in this situation since the dataset has a lot of intersections

However, Naïve Bayes works best on small datasets. The dataset being evaluated is large, so accuracy will be impacted. Logistic regression is good in evaluating large datasets. For this reason, logistic regression may be considered as a viable model. Both the models need to be compared with each other

**Challenges and successes:**

Success upto this point has been fitting the data into the Naïve Bayes model and parsing the text data through count vectorizer

**Possible extensions or business applications of your project**

**Conclusions and key learnings**