

TANDON
SCHOOL OF
ENGINEERING



Date:

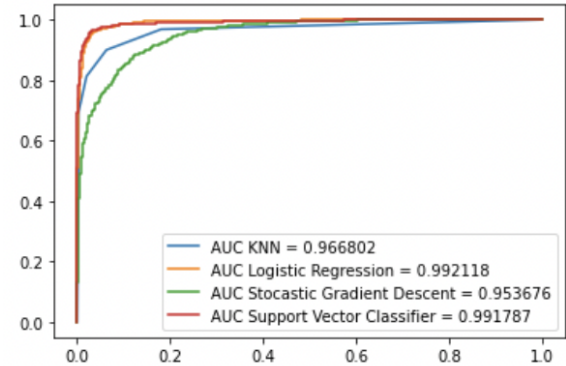
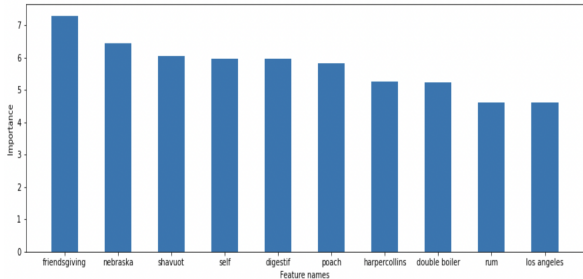
12/22/2021

Students:

Jaya Amit Sai Gurralla (jg6660)

Ashwin Suresh Babu (as14091)

QUAD CHART

<h2>Problem Statement</h2> <p>Given a list of recipes, can we link a new recipe with the existing food columns and help users to try out these new recipes. Can we generate new inferences from the existing recipe book?</p>	<h2>Why Machine Learning</h2> <p>A lot of categories are involved in the recipe book. There are chances where new categories can be included. It will not be possible to manually arrive at these patterns for different categories on a large scale.</p>
<h2>Results</h2> <h3><u>Dessert Category Classification</u></h3> <p>Comparison of different classification models</p>  <p>Top features</p> 	<h2>Conclusion and future scope</h2> <ul style="list-style-type: none"> • We have successfully implemented a prototype which classifies a relevant recipe as dessert and studied the features contributing to it. • We can extend this prototype to include various targets and build a system which associates various tags when a new recipe with existing tags are added. • We can implement the system to support learning new tags which the user adds along with some existing tags and use that new tag as a target for classifying future recipes once enough training data is available. • We can also build a visualization dashboard highlighting the popularity of recipes across different countries, top ingredients contributing to a recipe, top recipes during festive times and so on.

What did we do differently?

We will be using the below dataset from kaggle which has a sparse dataset of recipes and various categories.

<https://www.kaggle.com/hugodarwood/epirecipes>

We will be taking reference from one of the gold rated challenges in that dataset.

<https://www.kaggle.com/rtatman/regression-challenge-day-1>

The author attempts to throw light on different regression models that can be carried out and compares a target with a nutrient value.

Here we will be focusing on logistic regression and will be comparing the target with other similar categories instead to see if we can build a reliable system of recipe suggestions given a category. We will also check if any other models can be used in place of Logistic regression.

Abstract:

People in modern days are obsessed about food a lot. The Internet has provided them a platform to know about different recipes and the corresponding nutritional benefits they can gain. Given a list of recipes and the associated food columns, can we link a new recipe to the existing food categories with the help of machine learning algorithms and help the users to try out these new recipes. Is it possible to generate new inferences from the existing recipe book?

Introduction:

We are provided with different books of recipes which belong to a wide variety of cuisines. The recipes are associated with different categories like rating, calories, snack, wedding, cities, soya free, nutrients like proteins, carbohydrates, fats. These recipes are uploaded into a cookery website, which provides an access to people to try out these recipes.

People are also provided with an opportunity to upload their own recipes and tag the given dish with some of the categories available. Our machine learning algorithms will then try to associate the given dish with the existing food columns. This indeed will help people who are trying to filter the selected recipes.

During the process of uploading, if some of the pages are lost in the book, can we generate tags for the given recipes? If we want to search dishes based on occasion, given a simple query like *Christmas*, the dashboard should provide the user with recipes, associated with the keyword Christmas. The dashboard can also provide us with food items which are highly nutritious.

Description of Data Source:

The dataset which is used for the Cookery Book is taken from Kaggle's *Epicurious- Recipes with Rating and Nutrition*. The dataset was built, in order to understand the factors which humans consider while they are cooking or having the food. The dataset consists of 15000 sparse recipe entries which are tabulated based on rating, nutritional information and other categories. The information about the attributes is tabulated below.

Name of the attribute	Total No. of Columns	Attribute Type	Description about attribute
Title	1	String	Title of the recipe name
Rating	1	Float (0-5)	Average rating of the recipe
Nutritional Information	4	Float	Information about the nutritional values
Categories	674	Binary (0/1)	Binary variable: 1- if associated with recipe 0-otherwise

An instance of a tuple from the dataset is title - Potato and Fennel Soup Hodge, rating - 3.75, calories - 165, proteins - 6, fat - 7, sodium - 165 and the categories linked with the following recipe is ['fall', 'pasta', 'quick and easy', 'side', 'vegetable', 'vegetarian'].

Impact of Machine Learning:

Machine Learning has a huge impact on the food industry. It has a lot of applications like creation of new recipes based on nutritional values, food delivery business, ensuring safety of food and supply chain optimization - ensuring less waste and more transparency . The quality of food can be improved by lowering the production costs and analyzing the customer tastes using Machine Learning algorithms. It also helps them to understand what dishes should be present on the menu in order to increase the revenue. During this process a lot of categories are involved and new categories might be included in future, hence it will not be manually possible to arrive at interesting patterns for different categories on a very large scale.

Implementation Details:

The attributes considered for the classification are the categorical ones which include cakeweek, almond, alcoholic, anniversary, apple, apricot and many more. These attributes have ensured us with better performance when compared to the consideration of all attributes. We have considered *dessert* as our target variable in order to show how other attributes can also be used for carrying out classification and corresponding simulations. The target variable is appended to the end of the dataset, and the dataset is converted to a numpy array.

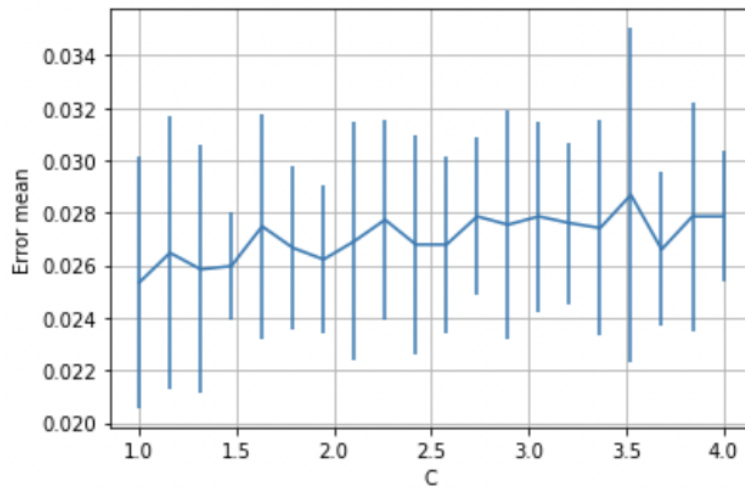
Initially logistic regression was carried out for the given dataset with default configurations. The following performance metrics have been reported.

Accuracy	Precision	Recall	F1-score
97.02 %	92.75 %	90.81 %	91.77%

We need to verify whether the model does well on different splits of train and test data. We carry out K-fold cross validation by considering 10 folds. K-fold cross validation generally ensures us with less biased results when compared to other cross validation methods. K-fold cross validation considers the fact that every observation from the dataset has an effective chance of appearing in the train and test set. The following average performance metrics have turned up.

Accuracy	Precision	Recall	F1-score
97.52 %	94.13 %	92.44 %	93.27%

We will now select an optimal 'C' which reduces the error rate using K-fold cross validation again and draw an error plot as below to compare different C values. We also refine the number of relevant features used by choosing a Lasso Regularizer.

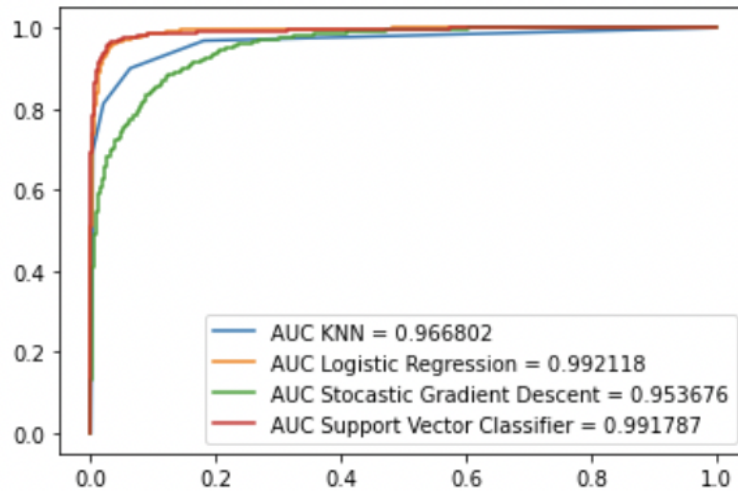


Performance Evaluation:

Later we analyze the performance of different classification algorithms for the given dataset. We have implemented Stochastic Gradient Descent, K- Nearest Neighbors, Support Vector Classifier using Grid Search CV. Grid Search CV uses all the combinations of hyperparameter values which are passed in the dictionary, compares the model performance against each combination using Cross Validation Method. We can observe during the training phase that KNN, SVC take a significant amount of time using Grid Search. Then we chose the best hyperparameters for respective models and carried out the classification for the given dataset. We have even reported detailed metrics as below for the Support Vector Classifier.

	Precision	Recall	f1-score	Support
0.0	0.98	0.99	0.98	4227
1.0	0.94	0.91	0.92	1009
accuracy			0.97	5236
macro avg	0.96	0.95	0.95	5236
weighted avg	0.97	0.97	0.97	5236

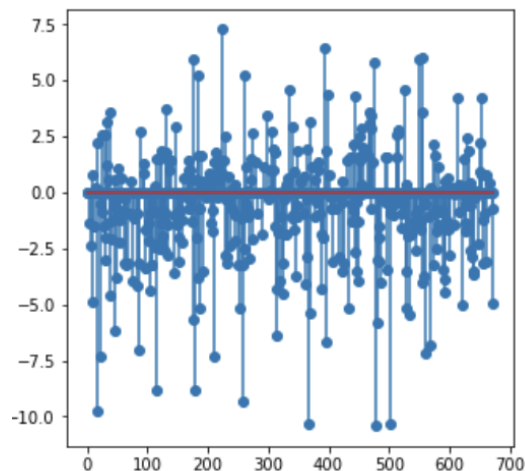
We now plot an AUC for all the models for better comparison. The AUC-ROC curve is always used as a performance metric for the classification algorithms using different thresholds. ROC is a probability curve which is plotted against TPR (y-axis), FPR (x-axis) and AUC determines the measure of separability. If the AUC value approaches closer to 1 it indicates the model has a good measure of separability. We have plotted an AUC ROC curve for different classification algorithms used in the project.



We can infer from the above AUC-ROC curve that Logistic Regression and Support Vector Classifiers have nearly the same performance while carrying out the classification. If we want our whole machine learning process to be less time consuming we can use Logistic Regression over Support Vector Machines.

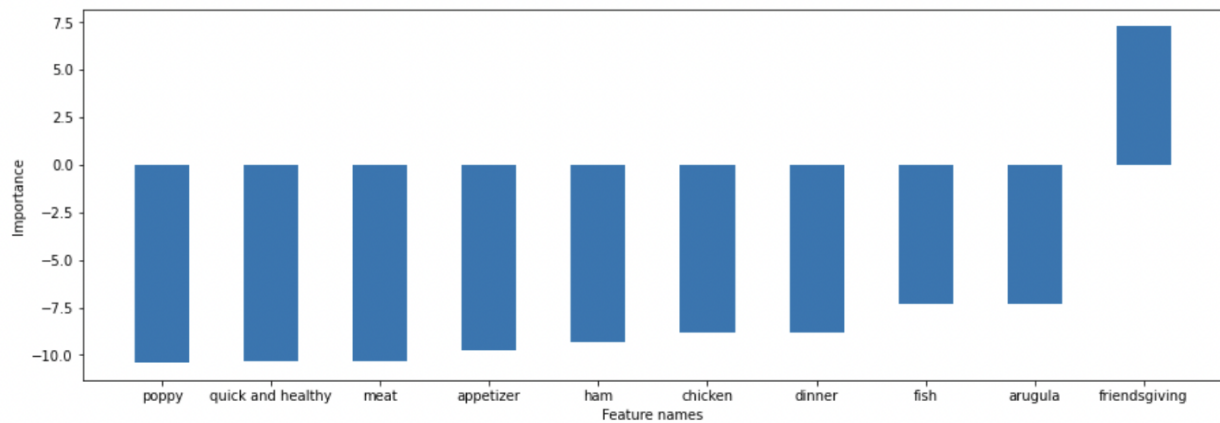
Analysis:

We now draw a stemplot of feature importance of dessert.



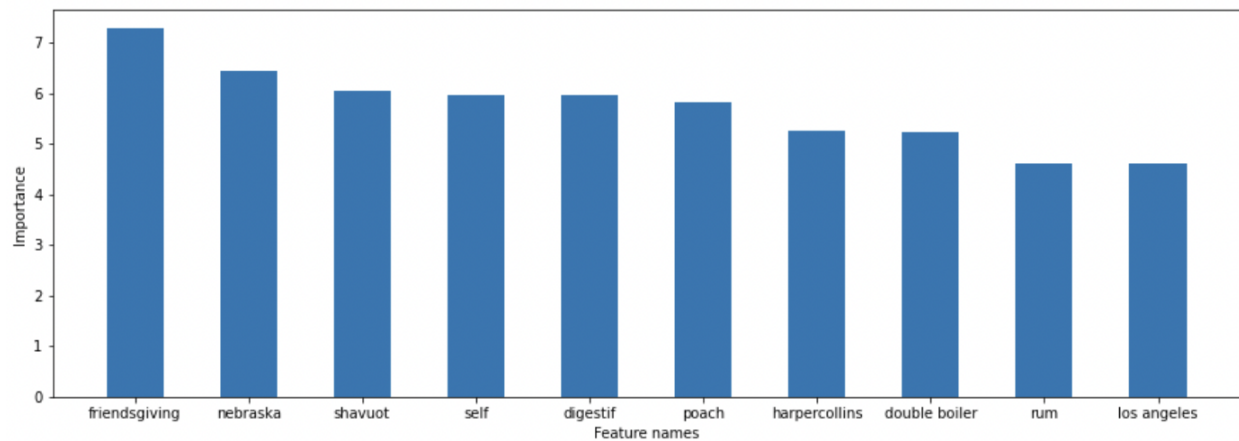
We see that only a few features contribute significantly. So we will analyze the features with higher correlation.

Non-absolute value of coefficient of features (in decreasing order)



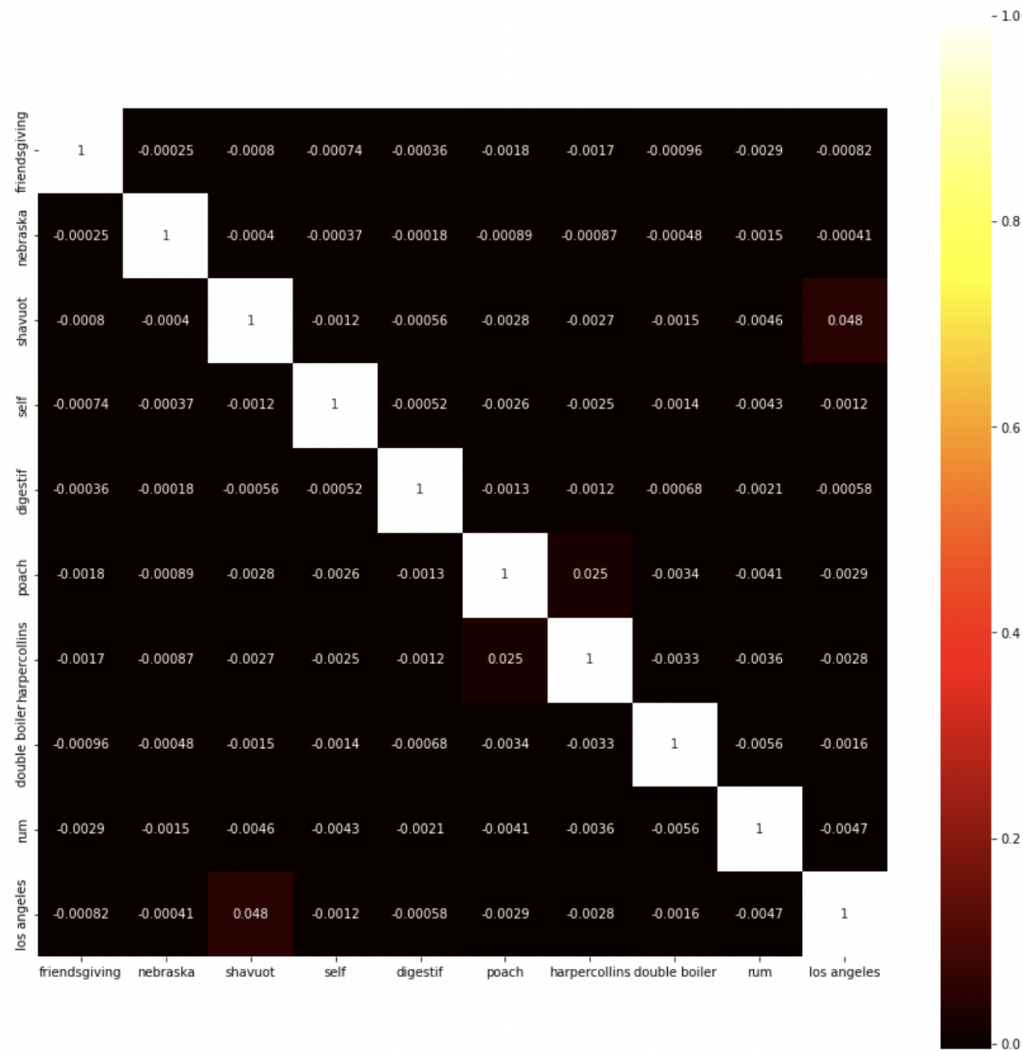
Take meat for instance. It is almost totally unrelated to a dessert so it has a high negative correlation. We will now further analyze the top positively correlated features.

Absolute value of coefficient of features (in decreasing order)



We see that there are a number of factors contributing to a recipe being classified as dessert such as places, ingredients and how it is cooked.

We can further draw a correlation heatmap of the top categories above and study if there are any underlying relation between them.



For instance, shavuot is a feast observed in Los Angeles. In general if we see very high correlation between features, we can combine them for more efficiency.

Conclusion and Future Scope:

We have been able to successfully implement a prototype which classifies a relevant recipe as dessert and studied the features contributing to it. We will further extend this prototype to include various targets and build a system which associates various tags when a new recipe with existing tags are added. The users can now search for a tag available in the system and get relevant recipe recommendations. Furthermore, we can implement the system to support learning new tags which the user adds along with some existing tags and use that new tag as a target for classifying future recipes once enough training data is available.

We can also build a visualization dashboard highlighting the popularity of recipes across different countries, Top ingredients contributing to a recipe, top recipes during festive times and so on.