

## PROJECT: Food Demand Forecasting

By: Madeti Karthikeya

In this project, our aim is to forecast the number of orders for a food delivery service to understand the demand and arrange and gather raw materials for the next ten weeks.

Dataset: We have collected the dataset from the Kaggle :  
[https://www.kaggle.com/kannanaikkal/food-demand-forecasting?select=fulfilment\\_center\\_info.csv](https://www.kaggle.com/kannanaikkal/food-demand-forecasting?select=fulfilment_center_info.csv)

We have merged the data of different datasets in order to predict the orders with all the possible features. We have to predict the number of orders.

### Preprocessing Step:

In this Step I found that: the train dataset was having 456548 rows with none of the columns containing null values. I then merged the train dataset with the meals information and fulfillment center information datasets in-order to get a dataframe with all the possible features together to predict. Converting the object type datatype into numeric was for essential for using the data in ML algorithm. Hence, I used one hot encoding for columns like center\_type, cuisine, category to numeric type using the label encoder class of sklearn library and removed all the unnecessary features from the dataframe making the dataframe ready for prediction.

Visualization the data of number of orders using the histogram gave me insight that majority of the customers were having their orders less than 972. Using the correlation metrics we found that 'homepage\_featured', 'emailer\_for\_promotion', 'op\_area', 'cuisine', 'city\_code', 'region\_code', 'category' were the top 7 columns which are positively correlated with the number of orders column.

Homepage\_featured and emailer\_for\_promotion are having the highest positive correlation. This shows that if homepage is featured about the products and promotion emails are sent to customers then more orders can be expected from the customers.

### Developing the Model:

Before training and building the model, I dropped the num\_orders columns from the train dataframe and divided that dataframe into X\_train and X\_val dataframes using the train\_test\_split method of sklearn library. This division will help me to understand and compare which ML algorithm would work with high accuracy for this dataset. Then I used the ML algorithms: Linear Regression, Lasso Regression, ELasticNet Regression, Decision Tree Regression, KNeighbors Regressor, Gradient Boosting Regressor, XGB

Regressor.

After fitting, predicting and checking the root mean square error of each of the mentioned algorithm, I found that decision tree algorithm is having the least root mean square error and hence understood that it is the algorithm with highest accuracy. I then saved the decision tree model using the pickle library.

I finally, used the model to predict the test dataframe and export the final document.

Application development:

I developed 2 html pages with home and upload as their file names. Home page contains the introduction and link to the prediction page i.e., upload page. On the upload page the user can input the values and get the final prediction of the number of orders. The two web pages and the model we have saved as the pickle file are integrated with the flask library of python. The flask code is run using using the local host which gives us the url from where we can open our home page, navigate to predict page, enter values and get our final prediction value. The final predicted value is shown on the predict page at the bottom.

Training the model on IBM:

As I have already registered on the IBM cloud, I created the Watson studio, cloud object storage and ML services. Deployed the model on IBM and integrated it with flask using the api keys.