FOOD DEMAND FORECASTING FOR FOOD DELIVERY COMPANY

SMARTBRIDGE

Let's Bridge the Gap

Submitted in complete fulfillment

of the requirement

by

TEAM MEMBERS :

218X1A0508	CH. DIVYALEKHYA
218X1A0507	CH. SURESH
218X1A0530	K. MEGHANA

218X1A0514	D.ARSHITH TEJA

CONTENTS:

6. Project Files Submission and Documentation

7. Project Demonstration

1. Introduction

Project overviews:

- A food delivery service operates in a dynamic and challenging environment where efficient management of perishable raw materials is crucial for success.
- With the constant flow of ingredients and supplies, accurately forecasting daily and weekly demand becomes the most critical factor for such a company.
- Maintaining an optimal inventory level is a delicate balance. On one hand, too much inventory sitting in the warehouse increases the risk of wastage, leading to financial losses and environmental concerns.
- On the other hand, inadequate inventory levels can result in out-of-stock situations, disappointing customers, and potentially driving them towards competitors.

Objectives:

- The main goal of this project is to develop a suitable machine learning model that can accurately predict the number of raw material procurement orders over the next 10 weeks.
- To achieve this goal, it is important to collect relevant information about distribution centers, including regions, cities, and other relevant factors.

- In addition, detailed food information such as food category, subcategory, price, and specific discounts offered during a particular week are important inputs to the predictive model.

2. Project Initialisation & Planning Phase:

Problem Statement:

- Food Demand Forecasting for a Food Delivery Company using IBM Cloud involves leveraging advanced data analytics and machine learning algorithms to predict the future demand for various food items.
- By analysing historical data such as order volumes, time of day, day of the week, weather conditions, and special events, the system can forecast the demand accurately.
- This helps the food delivery company optimize its operations by efficiently allocating resources, reducing wastage, and ensuring timely delivery to customers.

Proposed Solution:

- A proposed solution involves the utilization of a machine learning model to predict the demand for the next 10 weeks (Weeks: 146-155) for different center-meal combinations.
- By leveraging machine learning techniques, the aim is to develop a predictive model that can automate the forecasting process and provide accurate predictions.

- The proposed solution would involve collecting historical data on center-meal combinations, including factors such as resource usage, perishability, staffing levels, and any other relevant variables.
- This data would serve as the training dataset for the machine learning model. The model would then learn patterns and relationships from the historical data to make predictions about future demand.

Project Planning:

1.Initial Project Planning involves outlining key objectives, defining scope, and identifying number of orders.

2.During this phase, the team establishes a clear understanding of the dataset, formulates goals for analysis, and plans the workflow for data processing.

It includes

1. Past customer orders (by location, time, and type of food).
2. Delivery times, peak hours, and delays.
3. Seasonal data (holidays, weekends, weather impact).
4. Promotional campaigns and their impact on demand.
5. External factors (e.g., local events, public holidays).

3. Data Collection &Preprocessing Phase:

Data Collection Plan and Raw Data source identified:

The dataset for "Food demand forecasting for food delivery company” is sourced from Kaggle. It includes order details and customer data. Data quality is ensured through thorough verification, addressing missing values, and maintaining adherence to ethical guidelines, establishing a reliable foundation for predictive modelling.

Datasets: https://www.kaggle.com/kannanaikkal/food-demand-forecasting?select=fulfilment_center_info.csv

Data Quality Report:

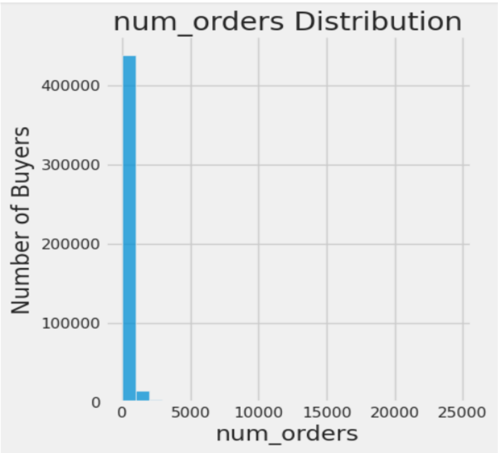
The Data Quality Report will summarize data quality issues from the selected source, including severity levels and resolution plans. It will aid in systematically identifying and rectifying data discrepancies.

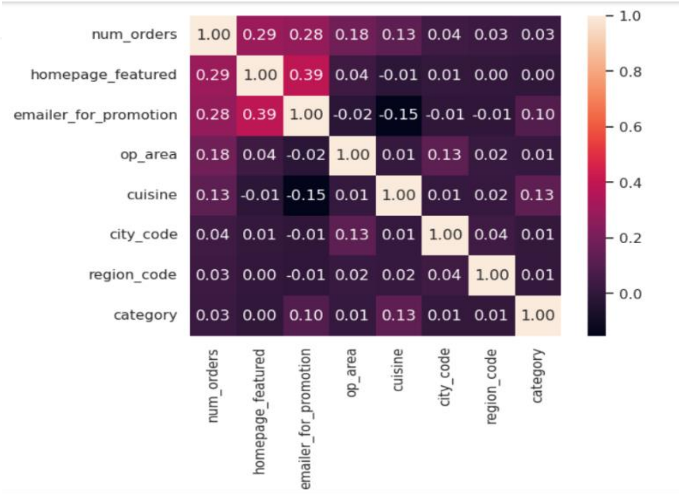
Checking for null values in the data by using isnull().sum() function and merging the files train.csv and meal_info.csv.

Data Exploration and Preprocessing:

1. Dropping the columns center_id and meal_id as there are not used in the data.
2. Data Exploration involves analysing the loan applicant dataset to understand patterns, distributions, and outliers.
3. Preprocessing includes handling missing values, scaling, and encoding categorical variables.
4. These crucial steps enhance data quality, ensuring the reliability and effectiveness of subsequent analyses in the food demand forecasting.

5. Label encoding is used to handle categorical variables. Data visualization helps the detection of patterns, trends, and correlations that might go undetected in text-based data.





4. Model Development Phase:

Feature Selection Process:

- The Feature Selection Report outlines the rational behind choosing specific features (e.g., id, week, center_id, meal_id) for the food delivery model.
- It evaluates relevance, importance, and impact on predictive accuracy, ensuring the inclusion of key factors influencing the model's ability to predict number of orders.
- The features (input variables) used by the model will heavily influence its accuracy. Start with these common features, and generate additional ones based on analysis.

- **Time-based features**:
  - ✓ Hour of the day, day of the week, month, season.
  - ✓ Weekends and holidays (binary feature).
- **Location-based features**:
  - ✓ Customer location (zip code, city, region).
  - ✓ Distance between the restaurant and the delivery location.
- **Weather-related features**:
  - ✓ Temperature, precipitation, snow, wind speed.
  - ✓ Weather conditions categorized (e.g., sunny, rainy, stormy).
- **Historical order data**:
  - ✓ Previous order volumes (lag features: order counts from the previous hour, day, week).
  - ✓ Order types (cuisine, menu items, order size).
  - ✓ Cancellations, refunds, and customer ratings.

Model Selection Report:

- The Model Selection Report details the rationale behind choosing Random Forest, Decision Tree, KNN, and XGB models for loan approval prediction.
- It considers each model's strengths in handling complex relationships, interpretability, adaptability, and overall predictive performance, ensuring an informed choice aligned with project objectives.

RMSE: The metric we used here is RMSE is the square root of the averaged squared difference between the target value and the value predicted by the model.

Model	Description	Performance Metrics (Ex: Accuracy, precision)
XGB Regressor	It excels at handling structured data, non-linear relationships, and large feature sets.	RMSLE: 69%
Linear Regression	It assumes a linear relationship between the dependent variable (food demand) and one or more independent variables	RMSLE:129%
Lasso Regression	To prevent overfitting and to improve the model's generalization.	RMSLE:129%
Elastic Net Regression	It's particularly useful when dealing with complex datasets where some features are highly correlated,	RMSLE:130%

	and feature selection is important.	
Decision Tree Regressor	Simple tree structure; interpretable, captures non-linear relationships.	RMSLE:62%
KNN	Classifies based on nearest neighbours; adapts well to data patterns.	RMSLE:66%
Gradient Boosting	Gradient boosting with trees; optimizes predictive performance, handles complex relationships.	RMSLE:97%

These are the models used for predicting the number of orders.

Initial Model Training Code, Model Validation and Evaluation Report:

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots…

```
#XGB Regression
XG = XGBRegressor()
XG.fit(X_train, y_train)
y_pred = XG.predict(X_val)
y_pred[y_pred<0] = 0
from sklearn import metrics
print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))
```

```
RMSLE: 69.32905885354447
```

```
#Linear Regression
LR = LinearRegression()
LR.fit(X_train, y_train)
y_pred = LR.predict(X_val)
y_pred[y_pred<0] = 0
from sklearn import metrics
print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))
```

```
RMSLE: 129.3075491948968
```

```
#Lasso Regression
L = Lasso()
L.fit(X_train, y_train)
y_pred = L.predict(X_val)
y_pred[y_pred<0] = 0
from sklearn import metrics
print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))
```

```
RMSLE: 129.01033731722407
```

```
#Elastic Net Regression
EN = ElasticNet()
EN.fit(X_train, y_train)
y_pred = EN.predict(X_val)
y_pred[y_pred<0] = 0
from sklearn import metrics
print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))
```

```
RMSLE: 130.98030233959565
```

```
#Decision Tree Regression
DT = DecisionTreeRegressor()
DT.fit(X_train, y_train)
y_pred = DT.predict(X_val)
y_pred[y_pred<0] = 0
from sklearn import metrics
print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))
```

```
RMSLE: 62.770325211941525
```

```
#KNN
KNN = KNeighborsRegressor()
KNN.fit(X_train, y_train)
y_pred = KNN.predict(X_val)
y_pred[y_pred<0] = 0
from sklearn import metrics
print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))
```

```
RMSLE: 66.93928614373267
```

```
#Gradient Boosting
GB = GradientBoostingRegressor()
GB.fit(X_train, y_train)
y_pred = GB.predict(X_val)
y_pred[y_pred<0] = 0
from sklearn import metrics
print('RMSLE:', 100*np.sqrt(metrics.mean_squared_log_error(y_val, y_pred)))
```

```
RMSLE: 97.97282613598486
```

5.Model Optimization and Tuning Phase:

The Model Optimization and Tuning Phase involves refining machine learning models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

Comparison Report:

- FOOD DEMAND FORECASTING FOR FOOD DELIVERY COMPANY is a Regression Model, as it involves regression models to predict a continuous outcome variable based on one or more predictor variables.
- The RMSLE for different models include
  - XGB Regression-62%
  - Linear Regression-129%
  - Lasso Regression-129%
  - Elastic Net Regression-130%
  - Decision Tree Regression-62%
  - KNN-66%
  - Gradient Boosting-97%

Final Model Selection Justification:

- The Final Model Selection Justification articulates the rationale for choosing Gradient Boosting as the ultimate model.
- Its exceptional accuracy, ability to handle complexity, and successful hyperparameter tuning align with project objectives, ensuring optimal loan approval predictions.

The best model for food demand forecasting for food delivery company is Gradient Boosting (Regression Model) that is 97%.

Source Code:

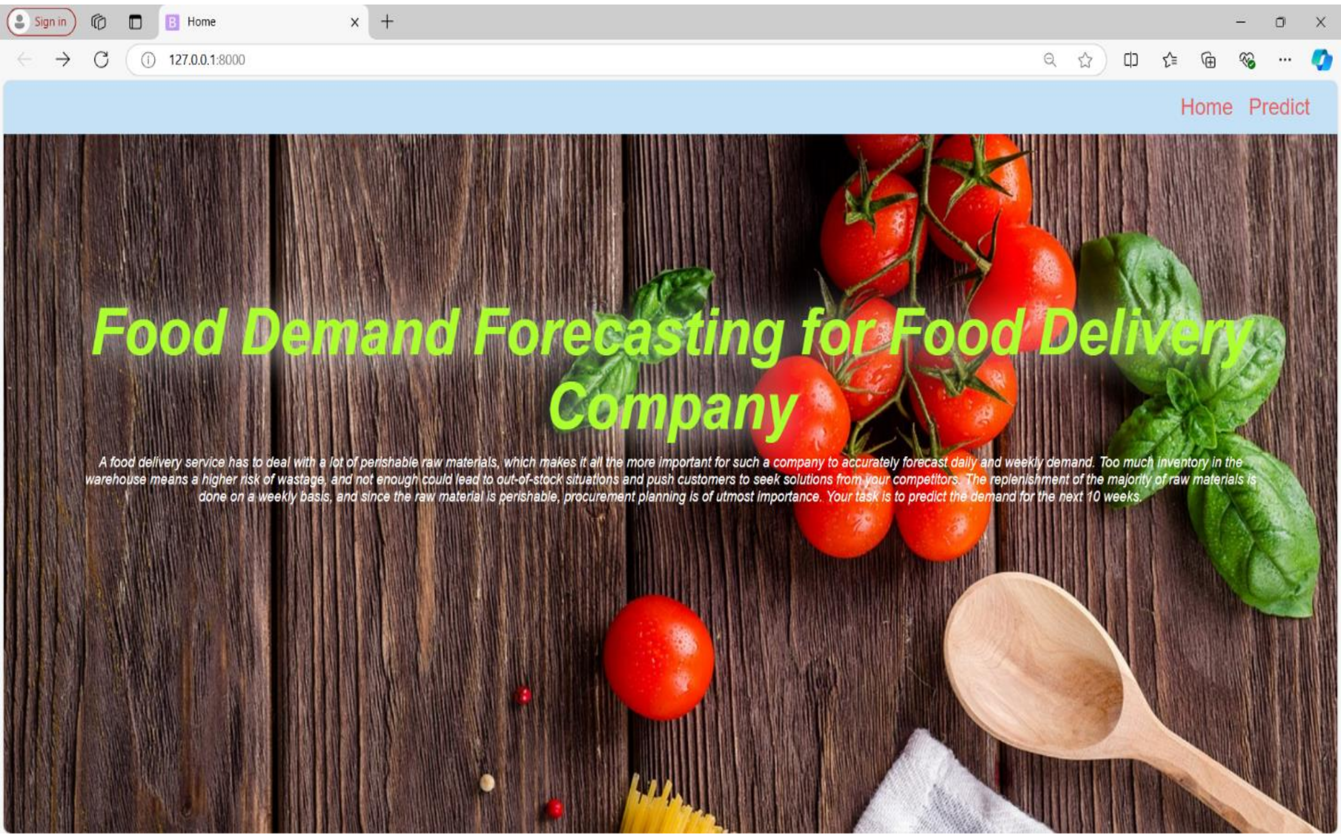
https://colab.research.google.com/drive/1TUiaYplREAbimhxPy8Z14aF4Pl4SydNK?usp=sharing

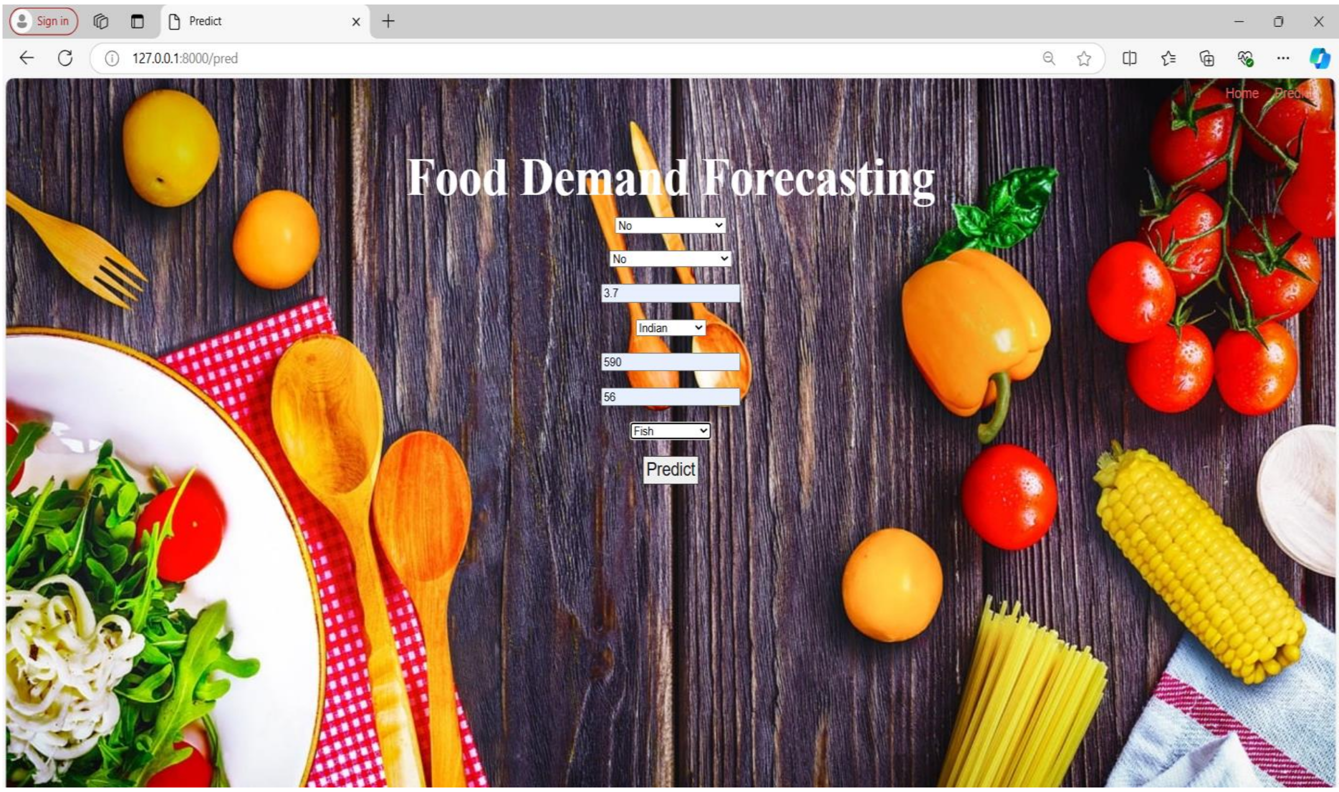
Project Files Submission and Documentation & Video demo:

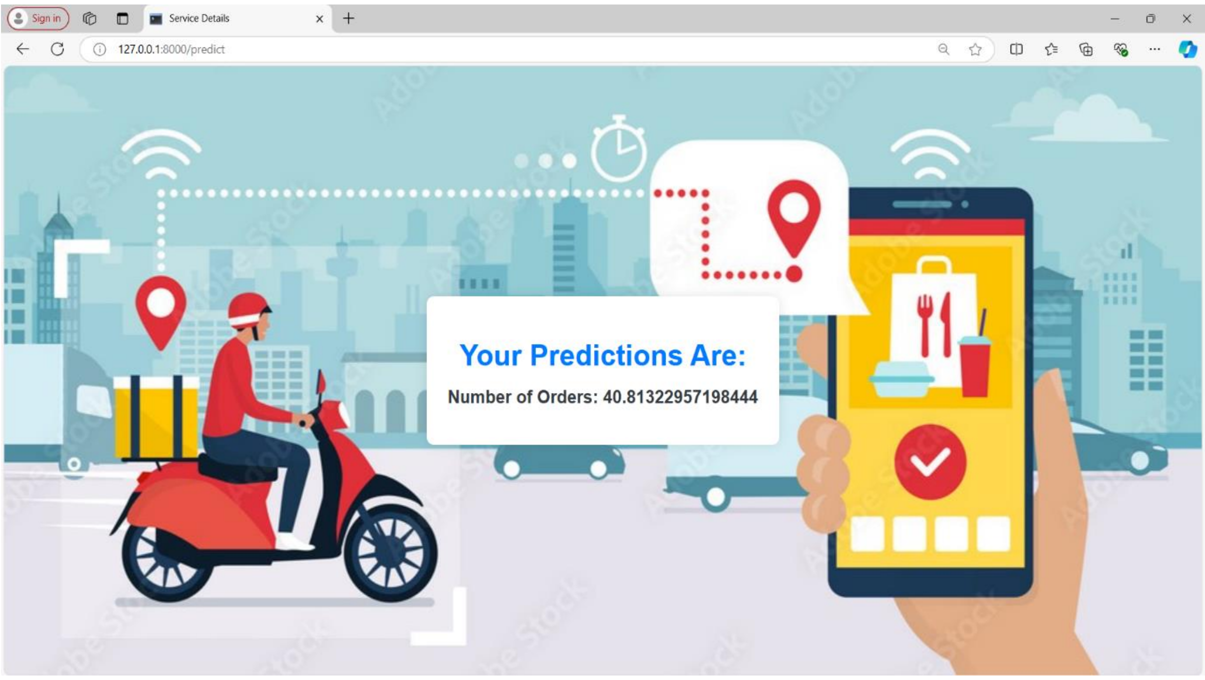
click the link and refer to the flow.

https://github.com/divyalekhya78/4592-Smartinternz-Food-Demand-Forecasting-for-Food-Delivery-Company

Result: Output Screenshots







Conclusion:

- Implementing accurate food demand forecasting for a food delivery company is essential for optimizing operations, enhancing customer satisfaction, and driving profitability.
- Lastly, we predict the number of orders by forecasting the demand.