DPR

DPR

BACK ORDER PREDICTION

Revision Number – 1.3

Last Date of Revision – 28/03/2022

Pichika Anil Naga Sai

1

DPR

Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Description | Author |
| 23-10-2022 | 1.0 | Abstract, | Anil |
|  |  | Introduction |  |
| 17-11-2022 | 1.1 | Deployment and | Anil |
|  |  | Process |  |
| 28-11-2022 | 1.2 | Q and A | Anil |

2

DPR

**Contents**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**Abstract**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.gjdgxs) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | [4](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.gjdgxs) | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| [**INTRODUCTION**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.30j0zll) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | [4](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.30j0zll) | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | [**Why this DPR Documentation?**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1fob9te) | | | | | | | | | | | | | | | | | | | | | | | | | | |  | | [4](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1fob9te) | |  |
|  |  | [*Key points:*](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.3znysh7) | | | | | | | | | | | | | | | | | | | | | | | | | | | |  |  |  |
|  |  |  |  | |  | |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| [**1 Description**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.2et92p0) | | | | | | | |  | | | | | | | | | | | | | | | | | | | | | | [4](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.2et92p0) |  |  |
|  | [**1.1**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.tyjcwt) | |  |  | [**Problem Perspective**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.tyjcwt) | | | | | | | | | | | | | | | | | | | | | | | | | [4](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.tyjcwt) |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | [**1.2 Problem Statement**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.3dy6vkm) | | | | | | | | | | | | | | | | | | | | |  | | | | | | | | [4](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.3dy6vkm) |  |  |
|  | [**1.3**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1t3h5sf) | |  | [**Proposed Solution**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1t3h5sf) | | | | | | | | | | | | | | | | | | | | | | | | | | [5](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1t3h5sf) |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | [**1.4**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.4d34og8) | |  | [**Solution Improvements**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.4d34og8) | | | | | | | | | | | | | | | | | | | | | | | | | | [5](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.4d34og8) |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| [**2 Technical Requirements**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.2s8eyo1) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | [5](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.2s8eyo1) |  |  |
|  | [**2.1**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.17dp8vu) | |  | [**Tools Used**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.17dp8vu) | | | | | | | |  | | | | | | | | | | | |  | | | | | | [5](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.17dp8vu) |  |  |
| [**3 Data Requirements**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.3rdcrjn) | | | | | | | | | | | | | | | | |  | | | | | | | | | | | | | [5](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.3rdcrjn) | |  |
|  | [**3.1 Data Gathering from Main Source**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.26in1rg) | | | | | | | | | | | | | | | | | | | | | | | | | | | |  | [5](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.26in1rg) | |  |
|  | [**3.2**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.lnxbz9) | |  | [**Data Description**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.lnxbz9) | | | | | | | | | | | | | | |  | | | | | | | | | | | [5](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.lnxbz9) | |  |
|  | [**3.3**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.35nkun2) | |  | [**Import Data into Database**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.35nkun2) | | | | | | | | | | | | | | | | | | | | | |  | | | | [6](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.35nkun2) | |  |
|  | [**3.4**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1ksv4uv) | |  | [**Export Data from Database**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1ksv4uv) | | | | | | | | | | | | | | | | | | | | | | | | | | [6](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1ksv4uv) | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| [**4 Data Pre-Processing**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.44sinio) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | [6](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.44sinio) | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| [**5 Design Flow**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.z337ya) | | | | | | | | | | | | | | | | | |  | | | | | | | | | | | | [6](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.z337ya) |  |  |
|  |  | | | |  | |  |  |  | |  | |  |  | |  | | | |  | | |  | | | | | | |  |  |  |
|  | [**5.1 Modelling**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.3j2qqm3) | | | | | | | | | |  | | | | | | | | | | | | | | | | | | | [6](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.3j2qqm3) |  |  |
|  | [**5.2**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1y810tw) | | | [**UI Integration**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1y810tw) | | | | | | | | | | | | | | | | | | | | | | | | | | [6](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1y810tw) | |  |
|  |  | | | |  | |  | |  | | | |  |  | |  | | | |  | | |  | | | | | | |  |  |  |
|  | [**5.3 Modelling Process**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.4i7ojhp) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | [7](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.4i7ojhp) | |  |
|  |  | | | |  | |  | |  | | | |  |  | | | | | |  | | |  | | | | | | |  |  |  |
|  | [**5.4 Deployment Process**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.2xcytpi) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | [7](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.2xcytpi) | |  |
|  |  | | | |  | |  | |  | | | |  |  | | | | | |  | | |  | | | | | | |  |  |  |
| [**6 Data from User**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1ci93xb) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | [7](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1ci93xb) | |  |
|  | | | | |  | |  | |  | | | |  |  | | | | | |  | | | | | | | | | |  |  |  |
| [**7 Data Validation**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.3whwml4) | | | | | | | | | | | | | |  | | | | | | | | | | | | | | | | [7](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.3whwml4) | |  |
|  | | | | |  | |  | |  | | | | |  | | | | | |  | | | | | | | | | |  |  |  |
| [**8 Rendering the Results**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.2bn6wsx) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | [7](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.2bn6wsx) | |  |
|  | | | | |  | |  | |  | | | | | | | | | | |  | | | | | | | | | |  | |  |
| [**9 Deployment**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.qsh70q) | | | | | | | | | | | | | | | | | | | |  | | | | | | | | | | [7](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.qsh70q) | |  |
|  | | | | |  | |  | |  | | | | | | | | | | | | | | | | | | | | |  |  |  |
| [**Conclusion**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.3as4poj) | | | | | | | | |  | | | | | | | | | | | | | | | | | | | | | [7](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.3as4poj) | |  |
| [**Q & A:**](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1pxezwc) | | | | | | |  | | | | | | | | | | | | | | | | | | | | | | |  | [7](https://docs.google.com/document/d/1qC8MubQJLeq1KMzIZ1mvfNNyKisjH5_L/edit#heading=h.1pxezwc) |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 3 | |

DPR

Abstract

Inventory backorder prediction is widely recognized as an important component of inventory models. However, backorder prediction is traditionally based on stochastic approximation, thus neglecting the substantial amount of useful information hidden in historical inventory data. To provide those inventory models with a big data-driven backorder prediction, we propose a machine learning model equipped with an under sampling procedure to maximize the expected profit of backorder decisions. This is achieved by integrating the proposed profit-based measure into the prediction model and optimizing the decision threshold to identify the optimal backorder strategy. We show that the proposed inventory backorder prediction model shows better prediction and profit function performance than the state-of-the-art machine learning methods used for large imbalanced data. Notably, the proposed model is computationally effective and robust to variation in both warehousing/inventory cost and sales margin. In addition, the model predicts both major (non-backorder items) and minor (backorder items) classes in a benchmark dataset.

Why this DPR Documentation?

The main purpose of this DPR documentation is to add the necessary details of the project and provide the description of the machine learning model and the written code. This also provides the detailed description on how the entire project has been designed end-to-end.

***Key points:***

* Describes the design flow
* Implementations
* Software requirements
* Architecture of the project
* Non-functional attributes like:
  + Reusability
  + Portability
  + Resource utilization

1 Description

1.1 Problem Perspective

The back order prediction is a machine learning binary classification model which helps us to predict that the product will be gone in back order or not and helps the users to manage the inventory of their products.

1.2 Problem Statement

The main goal of the project is to create a user interface which provides the backorder of the store’s products by taking certain input from the user like national\_inv, lead\_time,min\_balance,forecasting,future sales etc.

1.3 Proposed Solution

The solution proposed to take the required input of user from the created interface and process all the provided data to meet the requirements of the machine learning model and finally display the output saying order is the predicted backorder or not.

4

DPR

1.4 Solution Improvements

We will even predict that how much order of a customer can be gone in a backorder and it will more help for the business to manage the inventory properly and having an advance inventory management in hand for the customer’s order to deliver it on a time.

2 Technical Requirements

There are no hardware requirements required for using this application, the user must have an interactive device which has access to the internet and must have the basic understanding of providing the input. and for the backend part the server must run all the software that is required for the processing of the provided data and to display the results.

2.1 Tools Used

* Python 3.9 is used as the programming language and frame works like numpy, pandas, sklearn and other modules for building the model.
* Visual Studio Code is used as IDE.
* For visualizations seaborn and parts of matplotlib are being used.
* For data collection SQL Server database is being used.
* Front end development is done using HTML/CSS.
* Flask is used for both data and backend deployment.
* GitHub is used for version control.
* AWS is used for deployment.

3 Data Requirements

The data requirement is completely based on the problem statement. And the data set is available on the Kaggle in the form of excel sheet(.xlsx). As the main theme of the project is to get the experience of real time problems, we are again importing the data into the SQL data base and exporting it into csv format.

3.1 Data Gathering from Main Source

The data for the current project is being gathered from Kaggle dataset, the link to the data is:

https://www.kaggle.com/competitions/untadta/data

3.2 Data Description

There are about 1m+ records of sales information such as sku,national\_inv,lead\_time,min\_bank,forecast\_3\_month,sales\_3,6,9\_months etc.

5

DPR

3.3 Import data into Cassandra

Created an API for the upload of the data into the SQL Server database, steps performed are:

* Connection is made with the database.
* Created a database with name ineuron.
* Create command is written for creating the data table with required parameters.
* And finally, a insert command is written for uploading the dataset into data table by bulk insertion.

3.4 Export Data from Database

In the above created api, the download url is also being created, which downloads the data into a csv file format.

4 Data Pre-Processing

Steps performed in pre-processing are:

* First the data types are being checked and found only the price column is of type integer.
* Checked for null values as there are few null values, those rows are dropped.
* Converted all the required column into the date time format.
* Performed one-hot encoding for the required columns.
* Scaling is performed for required data.

And, the data is ready for passing to the machine learning algorithm.

5 Design Flow

5.1 Modelling

The pre-processed data is then visualized and all the required insights are being drawn. Although from the drawn insights, the data is randomly spread but still modelling is performed with different machine learning algorithms to make sure we cover all the possibilities. And finally, as expected LIGHT GBM Classifier performed well and further hyperparameter tuning is done to increase the model’s accuracy.

5.2 UI Integration

Both CSS and HTML files are being created and are being integrated with the created machine learning model. All the required files are then integrated to the app.py file and tested locally.

6

DPR

5.3 Modelling Process&5.4 Deployment Process

PYTHON

SQL SERVER

Data (CSV)

EDA

DATA PREPROCESSING

IMPORT PYTHON LIBRARIES & READ DATA

FEATURE SELECTION

FITTING MODEL WITH LIGHT GBM

FEATURE ENGINEERING

CREATING A WEB PAGE FOR DEPLOYMENT

SAVE MODEL IN PICKLE FILE

HYPER PARAMETER TUNNING

EXPORT DATA BACK TO SQL SERVER

DEPLOYMENT ON LOCAL HOST USING FLASK AND VS CODE

DEPLOYMENT ON HEROKU

CREATING POWER BI REPORT

INPUT VALUE & PREDICT FINAL RESULT

ARCHITECTURE

6 Data from User

The data from the user is retrieved from the created HTML web page.

7 Data Validation

The data provided by the user is then being processed by app.py file and validated. The validated data is then sent for the prediction.

8 Rendering the Results

The data sent for the prediction is then rendered to the web page.

9 Deployment

The tested model is then deployed to Heroku. So, users can access the project from any internet devices.

Conclusion

The back order prediction can predict the order based on the trained data set in the algorithm.

Hence the user can know that their product is gone in backorder or not .

Q & A:

Q1) What’s the source of data?

The data for training is provided by the client in multiple batches and each batch contain multiple files.

Q 2) What was the type of data?

The data was the combination of numerical and Categorical values.

7

DPR

Q 3) What’s the complete flow you followed in this Project?

Refer Page no 6 for better Understanding.

Q 4) After the File validation what you do with incompatible file or files which didn’t pass the validation?

Files like these are moved to the Achieve Folder and a list of these files has been

shared with the client and we removed the bad data folder.

Q 5) How logs are managed?

We are using different logs as per the steps that we follow in validation and

modeling like File validation log, Data Insertion, Model Training log, prediction log

etc.

Q 6) What techniques were you using for data pre-processing?

* Removing unwanted attributes
* Visualizing relation of independent variables with each other and output variables
* Checking and changing Distribution of continuous values
* Removing outliers
* Cleaning data and imputing if null values are present.
* Converting categorical data into numeric values.

Q 7) How training was done or what models were used?

* Before dividing the data in training and validation set, we performed pre-processing over the data set and made the final dataset.
* As per the dataset training and validation data were divided.
* Algorithms like Logistic regression, Decision Tree Classifier, Random Forest Classifier, XGBoost Classifier,LGBM Classifier were used based on the recall, final model was used on the dataset and we saved that model.

Q 8) How Prediction was done?

The testing files are shared by the client. We Performed the same life cycle on the provided dataset. Then, on the basis of dataset, model is loaded and prediction is performed. In the end we get the accumulated data of predictions.

Q 9) What are the different stages of deployment?

* First, the scripts are stored on GitHub as a storage interface.
* The model is first tested in the local environment.
* After successful testing, it is deployed on Heroku.

8