M.AKASH REDDY
19R11A0576
CSE-TEAM : 09

Time Series Analysis For Car Sales Forecasting Using Prophet With IBM Cloud

1. INTRODUCTION

PROJECT:“ **Time Series Analysis For Car Sales Forecasting Using Prophet With IBM Cloud”**

a. Overview A brief description about your project

The aim of this project is to check in practice to what extent an ensemble forecast based on averaging the outcomes. Therefore, we use data of monthly new car registrations .

Forecasting or Predicting the sale value helps the investors to invest in such a time where profits can be maximum. This project provides guidance to individuals who are willing to invest or buy a car and help them in knowing the price of a day using the prophet library. It is built on the monthly sales data from 1960 - 1968. Time series analysis is made on the data for accurate predictions.

1.2 Purpose The use of this project. What can be achieved using this

This project provides guidance to individuals who are willing to invest or buy a car and help them in knowing the price of a day using the prophet library.

2. LITERATURE SURVEY

a. Existing problem Existing approaches or method to solve this problem

1.Exponential Smoothing

2. Autoregressive Integrated Moving Average

3. Artificial Neural Network

4. Vector Auto Regression

5. Theta

6. Random Forest

7. Generalized Linear Model

8. Naive Seasonal

2.2 Proposed solution What is the method or solution suggested by you?

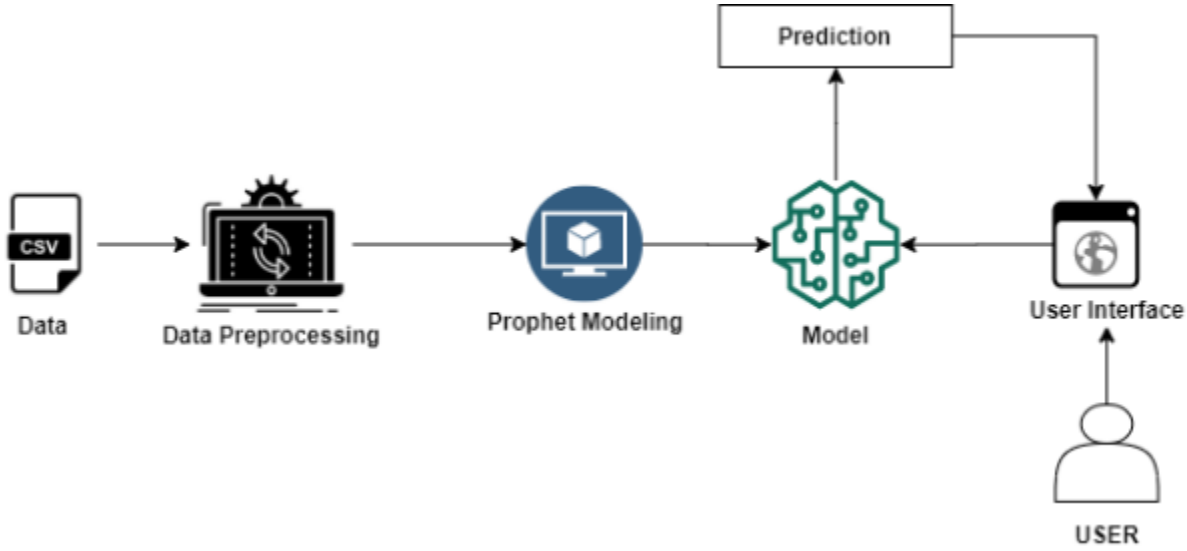
Prophet:

Time series forecasting can be challenging as there are many different methods you could use and many different hyperparameters for each method.

The Prophet library is an open-source library designed for making forecasts for univariate time series datasets. It is easy to use and designed to automatically find a good set of hyperparameters for the model in an effort to make skillful forecasts for data with trends and seasonal structure by default.

3. THEORITICAL ANALYSIS

a. Block diagram Diagrammatic overview of the project.



b. Hardware / Software designing

- **Hardware and software requirements of the project**

1.Installation of Anaconda IDE / Anaconda Navigator.

2.Installation of Python packages.

3. fbprophet

4.spyder/jupyter notebook

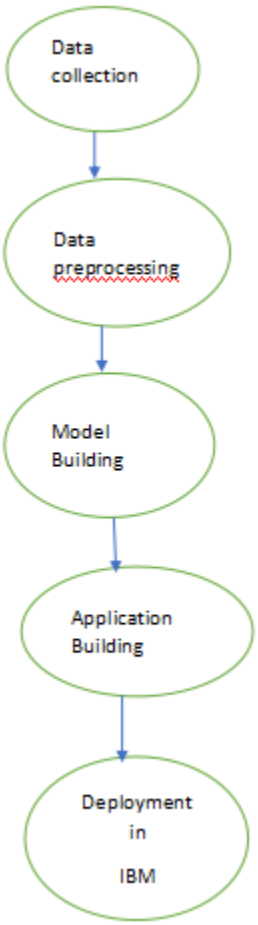
5.lite account in IBM

4. **EXPERIMENTAL INVESTIGATIONS**

Analysis or the investigation made while working on the solution.

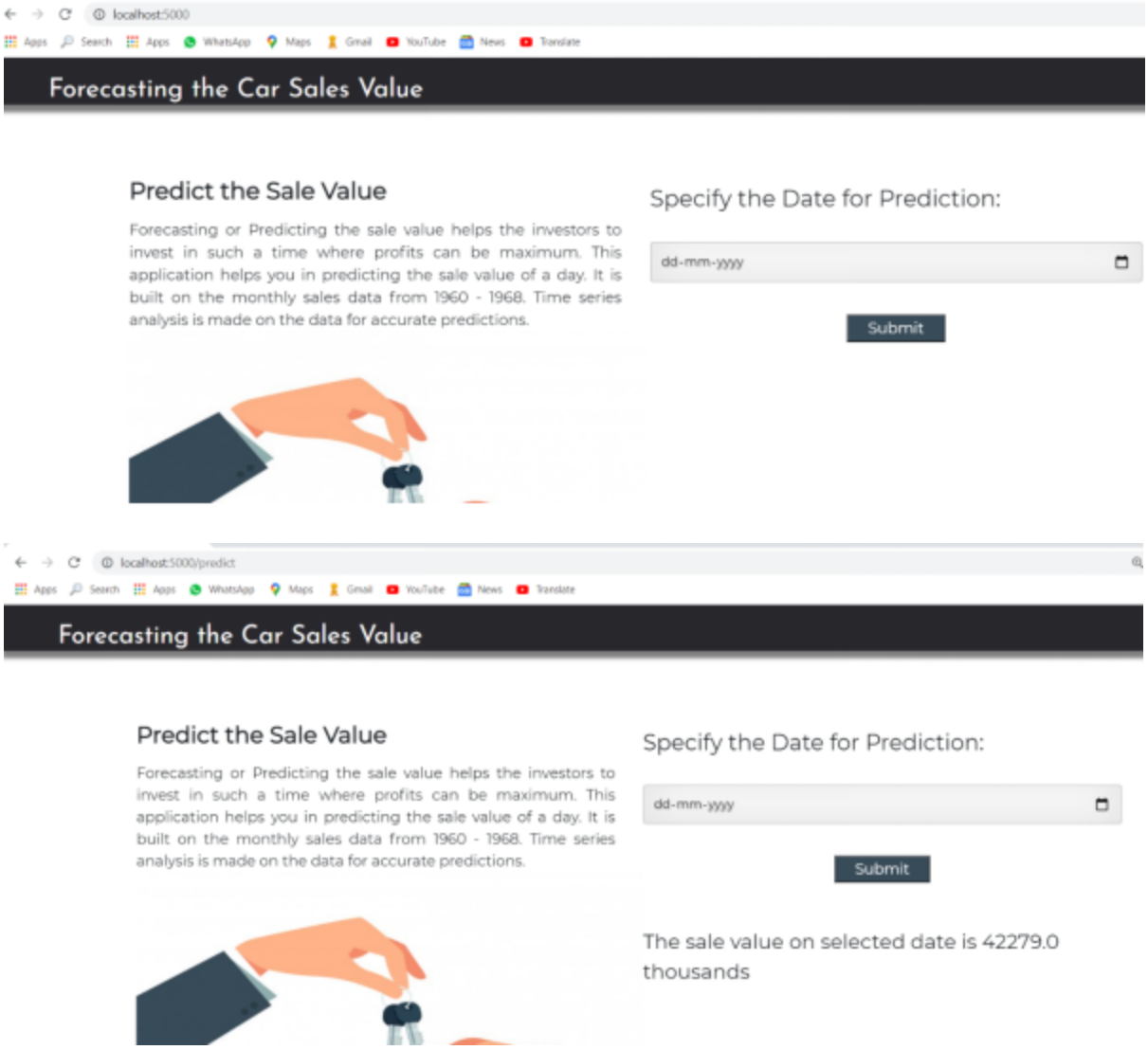
Made analysis of the given dataset and the profhet library.Also went through the videos related to forecasting,prophet library,deployement in IBM.As this is the guided project given to us went through all the information given to us.

5 FLOWCHART Diagram showing the control flow of the solution



6 RESULT

Final findings (Output) of the project along with screenshots.


localhost:5000
Apps Search Apps WhatsApp Maps Gmail YouTube News Translate
Forecasting the Car Sales Value
Predict the Sale Value
Forecasting or Predicting the sale value helps the investors to invest in such a time where profits can be maximum. This application helps you in predicting the sale value of a day. It is built on the monthly sales data from 1960 - 1968. Time series analysis is made on the data for accurate predictions.
Specify the Date for Prediction:
dd-mm-yyyy
Submit
localhost:5000/predict
Apps Search Apps WhatsApp Maps Gmail YouTube News Translate
Forecasting the Car Sales Value
Predict the Sale Value
Forecasting or Predicting the sale value helps the investors to invest in such a time where profits can be maximum. This application helps you in predicting the sale value of a day. It is built on the monthly sales data from 1960 - 1968. Time series analysis is made on the data for accurate predictions.
Specify the Date for Prediction:
dd-mm-yyyy
Submit
The sale value on selected date is 42279.0 thousands

7 ADVANTAGES &DISADVANTAGES

List of advantages and disadvantages of the proposed solution

Advantages:

1.Open Source

2.Accurate and fast

3.Allows for a large number of people to make forecasts, possibly without training in time series methods;

4.Tunable parameters

5.Available for both Python and R

Disadvantages

Prophet was never meant to be an all-purpose predictive algorithm, and its downsides reflect its specialization.

- If you need to predict **hundreds or thousands of targets** simultaneously, Prophet will compute slowly. You will need to adapt it to multivariate forecasting by coding multiple loops and storing multiple predictions. And Prophet won't cross-learn potentially useful patterns from multiple targets.
- If you have **more meaningful features than just seasonality or special events**, Prophet won't help. That's a big limitation in a lot of situations. Even plain-vanilla demand prediction challenges often involve a multi-level product hierarchy and some contextual data.
- If you are exposed to **unexpected changes in the underlying structure of your data** (a very common risk with time series), you will need to manage Prophet's relearning procedure manually. The hassle intensifies exponentially when you multiply data types, prediction targets and prediction horizons.

8 APPLICATION

The areas where this solution can be applied

Prophet can be used. Including:

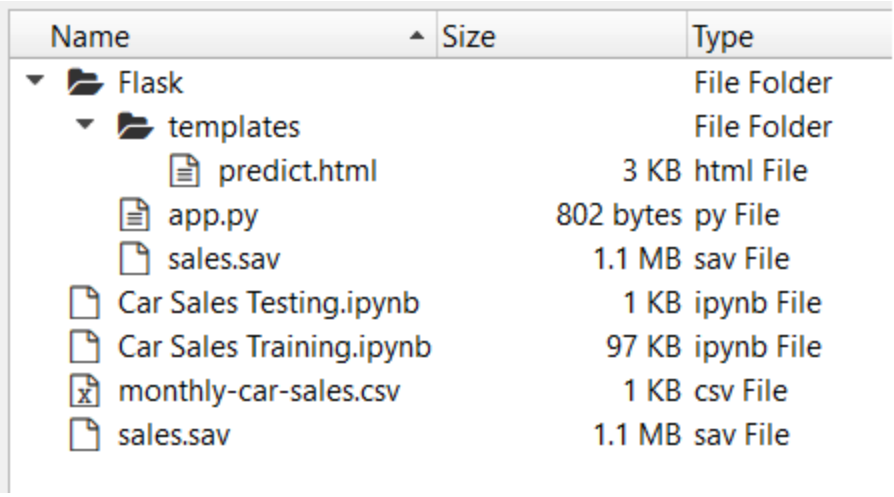
- **Looking forward to global insurance accounting (IFRS) developments**
- **The principles-based reserving world of valuations and projections has undergone some major transformations**
- **Prophet provides a single platform for ALM, pricing, planning and valuation**
- **US GAAP reporting including long duration targeted improvements**
- **Accommodating new and innovative demands in life insurance modelling**
- **Expanding Prophet into the general / P&C insurance industry**

9 CONCLUSION

Conclusion summarizing the entire work and findings.

Project Structure

Let us introduce you to the main project folder downloaded by prerequisites.



In order to proceed with this milestone, arrange all your project files in the below format.

- All the above files will be used to develop a flask application.
- In the templates, you will store all the rendering files and HTML pages.
- The sales.sav is the saved model file
- app.py is the python script (flask file)

steps:

1.Datacollection

This dataset contains two columns many columns. But we are interested in two columns for time series analysis

- Month
- Sales

2.Datapreprocessing

Data Pre-processing includes the following main tasks:

- Importing the required libraries
- Importing the dataset
- Analyze the data
- Resampling the dataset
- Preprocessing the data
- Taking care of Missing Data
- Prophet Library naming convention
- Data visualization

3.ModelBuilding

Model Building Includes:

- Model Fitting
- Making Future Predictions

- Obtaining the Forecasts
- Plotting the Forecasts
- Model Evaluation
- Saving the model

4.ApplicationBuilding

This activity lets you create a Flask Web application where the user can select the specific date to forecast the sales on the selected date. To accomplish the task you should build the required HTML pages and styling sheets as well as backend scripting files.

5.Deploy in IBM

In this milestone, we will be deploying our flask app in IBM Cloud as a cloud foundry application.

By the end of this project you will:

- know fundamental concepts and techniques of time series forecasting
- gain a broad understanding of time series data.
- Gain knowledge on fbprophet library.
- You will be able to know how to find the accuracy of the model.
- You will be able to build web applications using the Flask framework.

10 FUTURESCOPE

Enhancements that can be made in the future.

Vehicles purchasing will be done in few seconds due to this kind of forecasting .people will be investing more on vehicles and there will be demand for vehilcles and absolutely business will be expanded more employees will be joined and countrys income also increase.

11 BIBILOGRAPHY

Refered:

https://smartinternz.com/Student/guided_project_info/2439

https://machinelearningmastery.com/time-series-forecasting-with-prophet-in-python/

APPENDIX

_A. Source code

```
import pandas as pd

import numpy as np

from matplotlib import pyplot

from pandas import to_datetime

from fbprophet import Prophet

dataset=pd.read_csv(r"C:\Car sales\monthly-car-sales.csv")

dataset.info()

dataset.head()

dataset.tail()

dataset.isnull().any()

dataset.plot()

pyplot.show()

from pandas import to_datetime

dataset.columns=['ds','y']
```

```
dataset['ds']=to_datetime(dataset['ds'])

model=Prophet()

model.fit(dataset)

import joblib

joblib.dump(model,"sales.sav")

future = list()

for i in range(1,13):

   date = '1969-%02d' % i

   print(date)

   future.append([date])

future = pd.DataFrame(future)

future.columns = ['ds']

future['ds'] = to_datetime(future['ds'])

forecast = model.predict(future)

forecast[['ds','yhat','yhat_lower','yhat_upper']]

model.plot(forecast)

import joblib

import pandas as pd

from pandas import to_datetime

model=joblib.load('sales.sav')

a={'ds':[to_datetime('1975-1-9')]}
```

dg=pd.DataFrame(a)

op=model.predict(dg)

op.iloc[0,15]

Flask app code (python)

```
import joblib

import pandas as pd

from flask import Flask, request, render_template

from gevent.pywsgi import WSGIServer

import os


app = Flask(__name__)

model = joblib.load('sales.sav')

@app.route('/')

def home():

    return render_template('predict.html')

@app.route('/predict',methods=['POST'])

def y_predict():

    if request.method == "POST":

        ds = request.form["date"]

        a={"ds":[ds]}

        ds=pd.DataFrame(a)

        prediction = model.predict(ds)

        print(prediction)

        output=round(prediction.iloc[0,15])

        print(output)
```

```
        return render_template('predict.html',output="The sale value on selected date is {} thousands".format(output))

    return render_template("predict.html")

port = os.getenv('VCAP_APP_PORT','8080')



if __name__ == "__main__":

    app.secret_key = os.urandom(12)

    app.run(debug=True,host='0.0.0.0',port=port)
```