

### Forecast Commuters Inflow For Airline Industry Using Prophet Model With IBM Cloud

Team: CSE\_AIML\_C10



18481A05F8-O.Lakshmi Sameera 18481A05G1-P.Sowsilya 18481A05G5-P.Sindhu

18481A05F4-N.Varshini

# project



## INTRODUCTION

& OBJECTIVE

Air passenger traffic forecast is of great importance for airlines and civil aviation authorities. For airlines, accurate forecasts play an increasingly important role in revenue management. It helps to reduce the airlines’ risk by objectively evaluating the demand of the air transportation business. For civil aviation authorities, air passenger traffic forecast provides a concrete basis for planning decisions in air transport infrastructure. The main objective of this project is to build a prophet time series model that forecasts the passenger traffic for a given date



### PROBLEM STATEMENT

* We know, that passenger numbers appears to be highest sometimes after which we see dip in numbers due to implicit seasonality and trend in the context of air passenger traffic
* This leads to raise a questions like-

What capacity aircraft should they use?

When should they fly?

How many air hostesses and pilots do they need? How much food should they stock in their inventory?



### PROPOSED SYSTEM

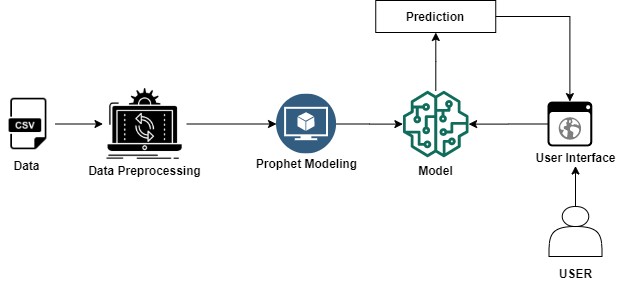
An airline company has the data of the number of passengers that have travelled with them on a particular route for the past few years. Using this data, they can forecast the number of passengers for the next months.

ADVANTAGES:

These forecasts will be a key element in the airport planning process. They are used for determining future airport requirements, analyzing alternative development plans, assessing the possible environmental effects of proposed plans, and determining the economic implications of future growth and development.

### PROJECT FLOW







DATA

#### We’re working with the Box and Jenkins (1976) Airline Passengers dataset, which contains time series data on the monthly number of airline passengers between 1949 and 1960.



MODEL BUILDING

The library we used is the Prophet library to predict future values of our time series. The authors of Prophet have abstracted away many of the inherent complexities of time series forecasting and made it more intuitive for analysts and developers alike to work with time-series data.

Steps for model building



Fit the model

Future prediction

Visualisation

Cross Validation

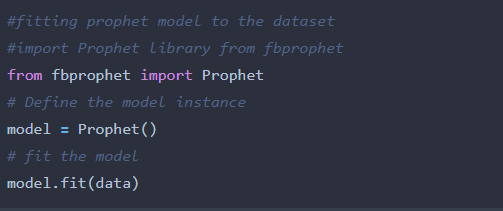
Evaluation

Save the model



FIT THE MODEL

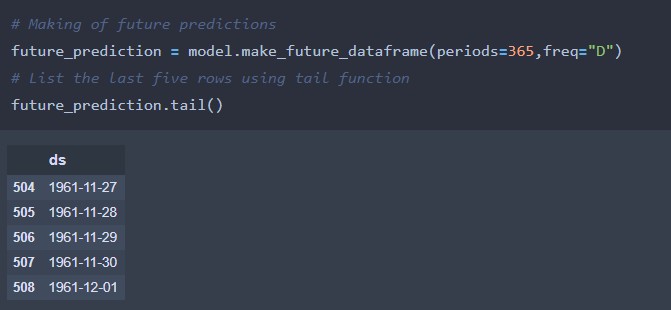
To begin, we must instantiate a new Prophet object. Now that our Prophet model has been initialized, we can call its fit method with our DataFrame as input.



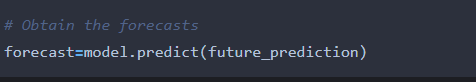


### FUTURE PREDICTION

* This is achieved using the Prophet.make\_future\_dataframe method and passing the number of days we’d like to predict in the future.



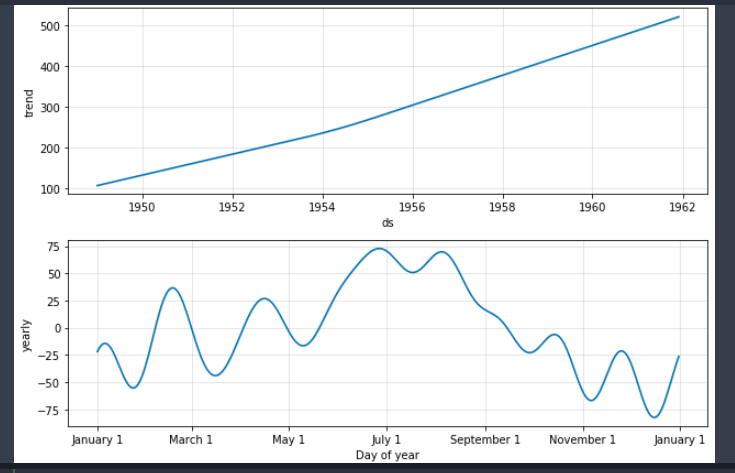
* Obtain the Forecasts





### VISUALISATION

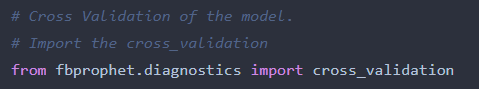
* + Visualising the component



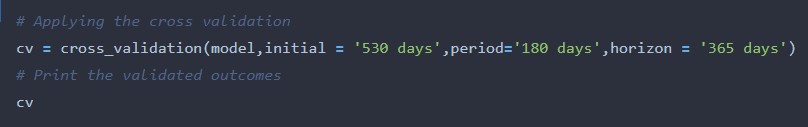


### CROSS VALIDATION

* + - Here we will validate the model for the accuracy.
    - Import the cross\_validation library using the below command.



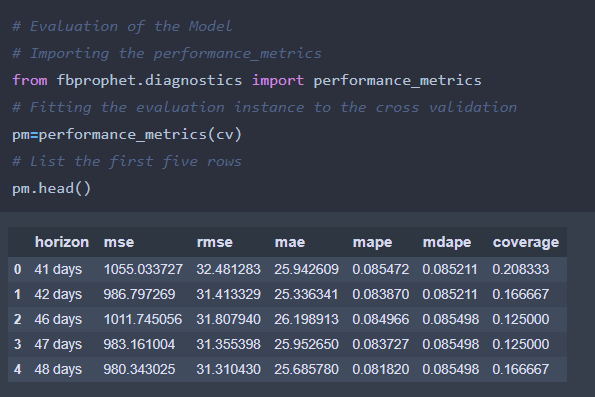
* + - Fitting the cross validation.





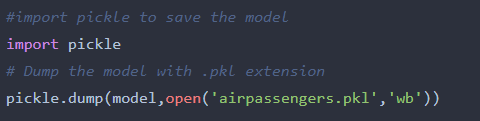
### EVALUATION

* Import the evaluation model libraries and fit it to the cross validated outcomes.
* The Outcomes of the evaluation will be like





### SAVE THE MODEL

This is the final activity of the model building, here we will the save the model to integrate into the web application.

After the model is trained we will create a Flask

## APPLICATION BUILDING



Web application where the user can select the specific date to forecast the passangers count on the selected date. To accomplish the task you should build the required HTML pages and styling sheets as well as backend scripting files



BUILD PYTHON CODE

#### We are using python for server side scripting. Let’s see

step by step process for writing backend code.



Importing the libraries

Importing flask module in the project is mandatory. An object of Flask class is our WSGI application. Flask constructor takes the name of current module ( name ).

Loading the Model

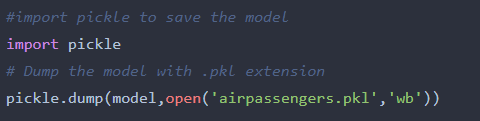
Loading the saved model using pickle library.

Model= pickle.load(open(‘airpassengers.pkl’ , ’rb’))



Rendering of home page Html

* + Here we will be using declared constructor to route to the html page which we have created earlier.



* + In the model, ‘/’ URL is bound with home.html function. Hence, when the home page of web server is opened in browser, the html page will be rendered. Whenever you enter the values from the html page the values can be retrieved using POST Method.



R**outing the prediction to the home page.**

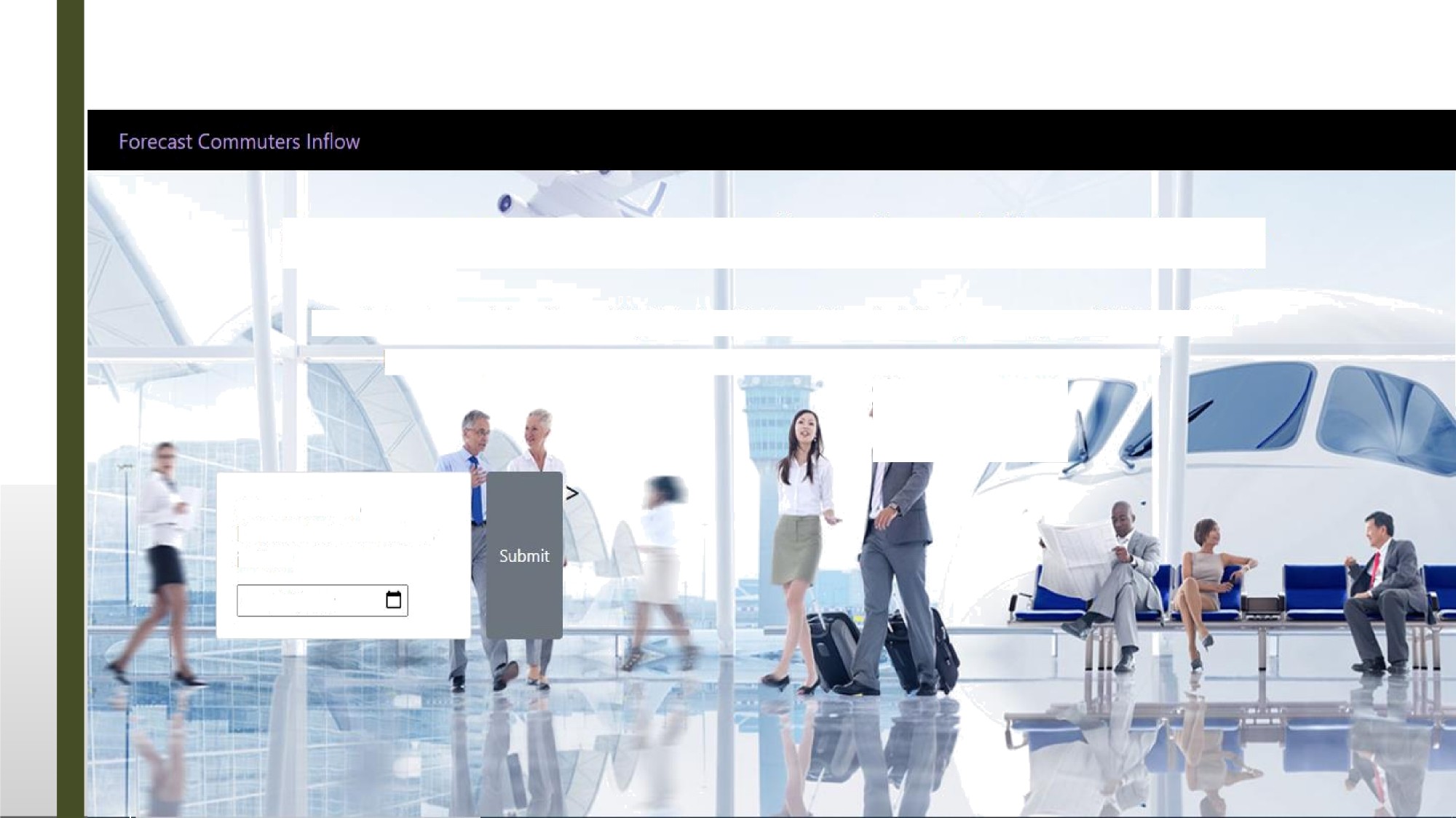
**Calling of main function**

This is used to run the application in local host.



RUN THE APPLICATION

HOME PAGE



F:recast Commuters Inflow for Airline Industry

The dataset consists of monthly totals of international airline passengers, 1949 to 1960.

Here, we predict the commutor inflow in thousands using prophet library.

Select date to Foretaste Commuters inflow

### PREDICTION PAGE





