AIM-5014-100

Time-Series Prediction—Python versus DataRobot

By Zeyu Wang

Fall 2021

In this project, I’ll achieve time series by python and complete-time aware modeling. Because I don’t have the access to run time series modeling on DataRobot so I just can choose the relative time aware modeling.

First of all, my data comes from Kaggle(https://www.kaggle.com/szrlee/stock-time-series-20050101-to-20171231), which is a dataset describes the DJIA 30 Stock Time Series. I downloaded the dataset and chose google stock data. In Google stock data, I just selected date and close columns. I will focus on the times series on the Google DJIA stock.

Python

Graphical user interface, application, Word

Description automatically generated

Here is the data I chose. Two columns’ data including the date and Close data. There is a total of 3019 rows from 2006 to 2017.

Chart, scatter chart

Description automatically generated

I drew the plot, and we can see that it has the time repeat trends and its total trends are growth.

At the next step, I need to split the data into train and test data.

Graphical user interface, text, application

Description automatically generated

I wrote a function to set a new column a. If the date is in 2017, it’s a column data will be y, otherwise, it will be n. Then I can easy to split the data into training and testing data.

Graphical user interface, text, application, email

Description automatically generated

1. Here I’ll use three different time series models and compare them. The first model is the ARMA model.

Text

Description automatically generated with low confidence

Chart, line chart, histogram

Description automatically generated

We can see that the predicted data is very close to the test data and the RMSE is 9.2. I think this model already fits the data well. (We can see that the data start from 800 on the y-axis).

1. And my next model is the ARIMA model.

Table

Description automatically generated

Chart, line chart

Description automatically generated

This model also fits the data well and has a lower RMSE.

1. My third model is the MarkovRegression.

Chart, line chart, scatter chart

Description automatically generated

It's obvious that the MarkovRegression model doesn’t suit this case and it can’t predict well and get a large RMSE.

After comparing the three models’ RMSE, I decided to choose the ARIMA model as the best model in the python group. Let’s move to the DataRobots group.

DataRobot

Chart, histogram

Description automatically generated

Firstly, I chose ‘Close’ as my target variable.

Chart

Description automatically generated

Then I chose Date as the date feature.

Graphical user interface, website

Description automatically generated

Because the other two time series model function is closed, so I just can do a machine learning model.A screenshot of a computer

Description automatically generated

I set the holdout data date from Jan 1, 2017, to Dec 20, 2017.

Graphical user interface

Description automatically generated with medium confidence

Then I used RMSE as the indicator to compare the model.

Graphical user interface, text, application

Description automatically generated

Here we can see some models can’t calculate RMSE on the holdout dataset. I just need to choose the model which has the lowest RMSE.

Graphical user interface, text, application

Description automatically generated

These three models have the low RMSA in this case on DataRobot.

Graphical user interface, chart

Description automatically generated

But this one seems like just a linear regression and doesn’t show any time series feature. So I drop this one and chose the second model.

Graphical user interface, chart

Description automatically generated

In this model, we can see that it has a better performance but still doesn’t fit the data well.

A screenshot of a video game

Description automatically generated

But when I change the dataset to holdout dataset, I find it fits better than the training dataset.

Graphical user interface, chart

Description automatically generated

The third one looks similar to the second one but they are different. The third one is the predicted data above the actual but the first one is the predicted data under the actual data.

Graphical user interface, chart, line chart

Description automatically generated

When I look at its holdout dataset, I think maybe it's better than the second one.

Conclusion:

Chart, line chart, histogram

Description automatically generatedGraphical user interface, chart, line chart

Description automatically generatedGraphical user interface, application

Description automatically generated

When I compared these models in python and DataRobot, the ARIMA model has a better performance in RMSE and shows a better fit. But Robots show a good visualization. Just remind, if we run time series on DataRobot, I am sure it’ll have a better performance. But because I don't have access, so I just run a time-aware model. In the above three models on DataRobot, there are the neural network data but they also show a good time-series feature.

What’s more, it’s easier to run a model on DataRobot than python. For example, when we want to run time series on python, I needed to read the document on the website and understand the model, and know how to set up it. But on DataRobots, it can be easy to use all kinds of models which I even didn’t learn. This is very useful for those companies who want to quickly develop a model to use.