Weather Prediction Using ARIMA And SARIMA Models

Sumit 19312

DSE, IISER Bhopal

**INTRODUCTION**

Weather prediction is a real-time challenging issue witnessed by the world in the past few decades. This indirectly had an impact on the effective prediction of the weather data. Due to the latest technological updates, the capabilities of retrieving and storing have increased; resulting in the availability of massive meteorology data in different formats. In this project, one of the famous models used in time-series forecasting, ARIMA model-based prediction is discussed.

ARIMA, ‘Auto-Regressive Integrated Moving Average’ is a class of models that explains a given time series based on its own past values, that is, its own lags and the lagged forecast errors, so that equation can be used to forecast future values. Any non-seasonal time series that exhibits patterns and is not a random white noise can be modeled with ARIMA models

Seasonal Autoregressive Integrated Moving Average, SARIMA or Seasonal ARIMA, is an extension of ARIMA that explicitly supports univariate time series data with a seasonal component.

**PROJECT DESCRIPTION and SCOPE**

This project shows the use of ARIMA and SARIMA models for predicting monthly average temperatures of the city ‘Delhi’ for many months forward. Manual comparing of the models and automatic model generation methods are discussed and analysed in depth.

The scope of this project was to learn the use of the traditional forecasting models, finding their limitations and scope in order to get closer to more effective predictions.

**CODE WALK-THROUGH**

* The code starts with reading the CSV file using Pandas. After cleaning the data and extracting the chosen series (22 years of data showing the average temperatures in DELHI), the series is analysed using the *seasonal\_decompose* method from the *statsmodels* module. From the plot, we can notice seasonality in our data (which would obvious when dealing with weather data). There is not much we can infer from the trend plot but the ARIMA models might!
* Next part is where we check the stationarity of the data. Augmented Dickey-Fuller (ADF) test and Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test are the most common tests for checking stationarity. The results show that the data is pretty stationary and we can proceed to use the ARIMA models without differencing the series.
* The ARIMA models have the parameters p (AR term), d (order of differencing), and q (ME term) which can be obtained by analysing the Auto-Correlation and Partial Auto-Correlation plots. (ACF and PACF)

d=0 as we don’t need to difference the series.

q=2 from the ACF plot.

The p-value is not very clear from the PACF plot, so we take p=1,2,3 and compare all the models.

We find the second model (p=2, d=0, q=1) to be the best and can be used for prediction. The model is fitted on the training data (20 years) and forecasted for the next 23 months. We then plot the prediction along with the actual data. Root Mean Square Error is the most common error test for such predictions. Here, I defined the accuracy as the difference of actual values and the RMSE scaled to 100. (There can be other ways to define accuracy in such cases)

Result: 86%

* The above model is not the best model because it does not consider seasonality. One other variation of the ARIMA models is the SARIMA models (Seasonal ARIMA). These models use seasonal differencing instead of normal differencing. We can see in the plot that seasonal differencing is more effective in removing stationarity from the series.
* To build a SARIMA model, I am using the *auto\_arima* method from the *pmdarima* module. This method compares multiple models and gives the best model on its own. We then fit this model and forecast the results. Accuracy, in this case, is higher than in our previous case. (Obvious because we considered seasonal factor)

**CONCLUSION**

The project has successfully depicted fairly accurate predictions from the chosen models. I have learned a lot in this project about how different models are more suitable in different situations of time-series forecasting.

SARIMA is better than the normal ARIMA model because it considers the seasonal factor which is necessary when dealing with forecasting.

The *auto-arima* saves a lot of time that goes on in finding the best model to fit our data and gives great results. Also, while comparing models manually someone might not be able to come up with the best model as reading the correlation plots can be tricky.

I also noticed that the ADF and KPSS tests ignore seasonality while checking stationarity so it is always better to also plot the correlation plots or just use the *auto-arima* method.

**FUTURE RECOMMENDATIONS**

I would like to extend my project to analyse even more models and methods to improve the results and use them on some other weather features than temperature like wind speed, humidity, etc. One method which could improve the accuracy is the SARIMAX (SARIMA+X) model which considers an external factor that affects our variable along with the SARIMA model. I can also add the use of neural networks in the future to take this project even further.

**REFERENCES**

I took inspirations from the following sources:

* [Weather forecasting with Machine Learning, using Python | by Piero Paialunga | Towards Data Science](https://towardsdatascience.com/weather-forecasting-with-machine-learning-using-python-55e90c346647)
* [Time series analysis of climate variables using seasonal ARIMA approach (ias.ac.in)](https://www.ias.ac.in/article/fulltext/jess/129/0149)