08-Weather-conditions

17th October 2020

HAMOYE

Quadri Bello 11a529e8ac1f000

Role: Data Scientist
Project Lead

Iheagwara Ifeanyi fba56f

Role: Data Scientist
Query Analyst

Tomiwa Obanla

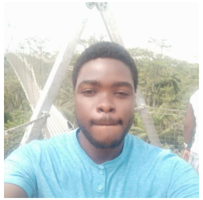
Role: Data Scientist

Moses Otu

Role: Data Scientist

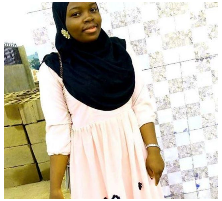
Habeebullah Agbaje eb6981

Role: Data Scientist
Assistant Project Lead



Chizurum Olorondu

Role: Data Engineer



Lateefah Bello

Role: Data StoryTeller

Ezeh Jane

Role: Data StoryTeller

Problem Statement

- To predict the temperature of any given city across a specific time period.

Existing solutions

- Using simple univariate forecasting methods like AR
- Another simple solution is to forecast values for each series individually using the techniques we already know

Our approach

- We used Multivariate forecasting methods, **our approach was able to understand and use the relationship between several variables**. This is useful for describing the dynamic behavior of the data and also provides better forecasting results.

Dataset description

The dataset contains information on weather conditions recorded on each day at various weather stations around the world.

- Information includes precipitation, snowfall, temperatures, wind speed and whether the day included thunder storms or other poor weather conditions.
- Data source: kaggle.com/smid80/weatherww2/data
- Data source origin: ncdc.noaa.gov/data-access/land-based-station-data/land-based-datasets/world-war-ii-era-data
- Two csv files: weather_condition and Weather stations locations

Dataset description

- <u>Weather stations locations;</u>

Column name	Description
WBAN	Weather station number
NAME	weather station name
STATE/COUNTRY ID	acronym of countries
Latitude	Latitude of weather station
Longitude	Longitude of weather station
Elev	Elevation

0 nans values
Shape: 161 x 8

```
locationdf.head(3)
```

	WBAN	NAME	STATE/COUNTRY ID	LAT	LON	ELEV	Latitude	Longitude
0	33013	AIN EL	AL	3623N	00637E	611	36.383333	6.650000
1	33031	LA SENIA	AL	3537N	00037E	88	35.616667	0.583333
2	33023	MAISON BLANCHE	AL	3643N	00314E	23	36.716667	3.216667

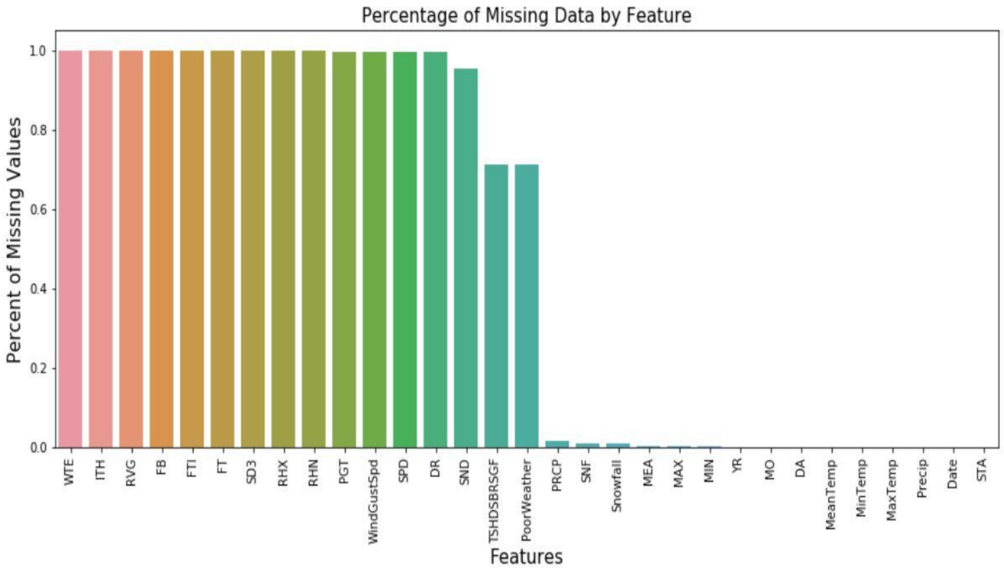
```
locationdf.isna().sum()
```

```
WBAN                0
NAME                0
STATE/COUNTRY ID    0
LAT                 0
LON                 0
ELEV                0
Latitude            0
Longitude           0
dtype: int64
```

Dataset description

- <u>Weather_conditions;</u>

Shape: 119040 x 31

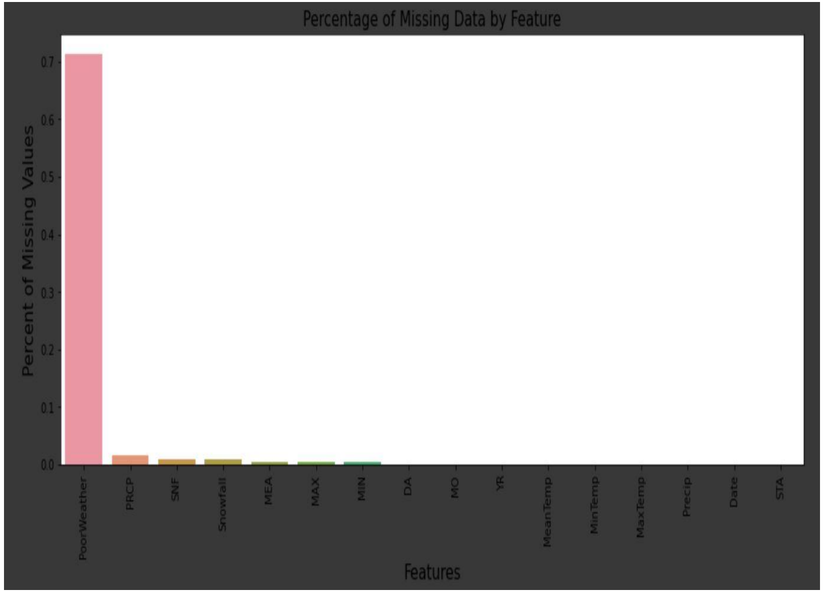


Column name	Description	
STA	STATION NUMBER	
YR	YEAR	
MO	MONTH	
DA	DAY	
PRCP	24-HOUR PRECIPITATION	INCHES & HUNDREDTHS
DR	PEAK WIND GUST DIRECTION	TENS OF DEGREES
SPD	PEAK WIND GUST SPEED	KNOTS
MAX	MAXIMUM TEMPERATURE	FAHRENHEIT
MIN	MINIMUM TEMPERATURE	FAHRENHEIT
MEA	MEAN TEMPERATURE	FAHRENHEIT
SNF	SNOWFALL	
SND	SNOW DEPTH	
FT	FROZEN GROUND TOP	DEPTH IN INCHES
FB	FROZEN GROUND BASE	DEPTH IN INCHES
FTI	FROZEN GROUND THICKNESS	THICKNESS IN INCHES
ITH	ICE THICKNESS ON WATER	INCHES & TENTHS
PGT	PEAK WIND GUST TIME	
TSHDSBRSGF	(days with THUNDER SLEET HAIL DUST OR SAND SMOKE OR HAZE BLOWING SNOW RAIN SNOW GLAZE FOG) 0 = NO, 1 = YES	
SD3	SNOW DEPTH	
RHX	RELATIVE WHOLE % HUMIDITY	
RHN	RELATIVE WHOLE % HUMIDITY	
RVG	RIVER GUAGE	
WTE	WATER EQUIVALENT OF SNOW/ICE ON GROUND	

Data Wrangling, Processing and Exploration Workflow

Weather conditions Summary:

- *Task: Handling missing values;*
- Drop columns with 80% nan.
- Drop TSHDSBRSGF column since it is the same as the poor weather from description.
- Drop MIN, MAX MEA (Fahrenheit) because it is similar to min temp, max temp and mean temp (Celsius). Only difference is the measurement unit.
- PRCP, SNF, snowfall (10% each) and poor weather (70%) still left with nan.



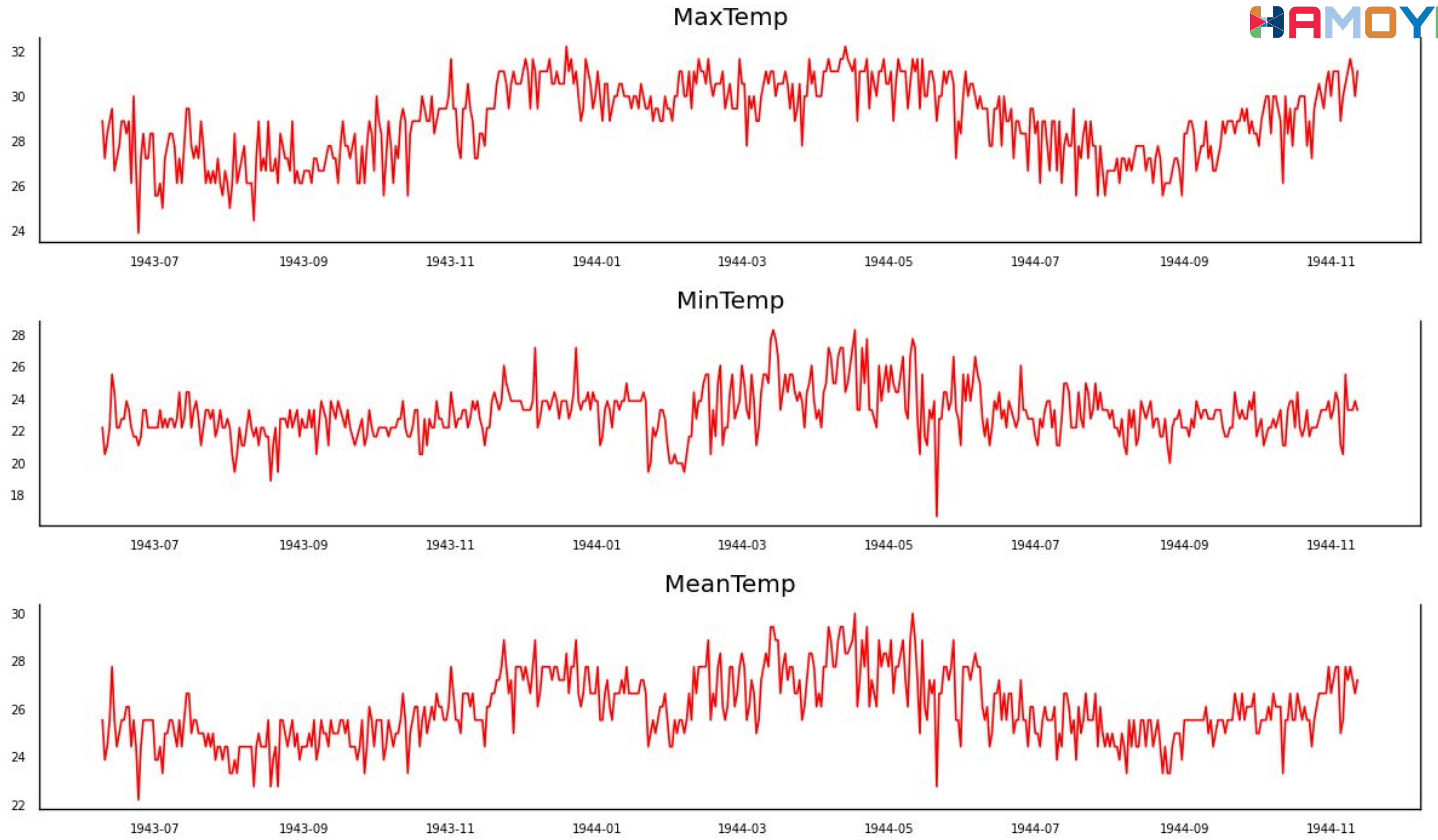
Data Wrangling, Processing and Exploration Workflow

Weather conditions Summary:

- We do not advise dropping nan since this can cause a gap in time.
- Drop PRCP and SNF since there are replicates of precip and snowfall column.
- We used bfill to fill up the snowfall since the weather in question is logically related to the day before.
- The poor weather column is related to the snowfall. According to the description, it is considered a poor weather if there is snowfall, hail or thunder. Poor weather is 0 if there is no snowfall.
- Finally because our dataset contain weather reports from various weather stations from various cities in the world, we decided to pick one.
- Apapa, lagos, Nigeria: Station number “30001”. Thus data from STA 30001 was extracted into a new csv for our model prediction.

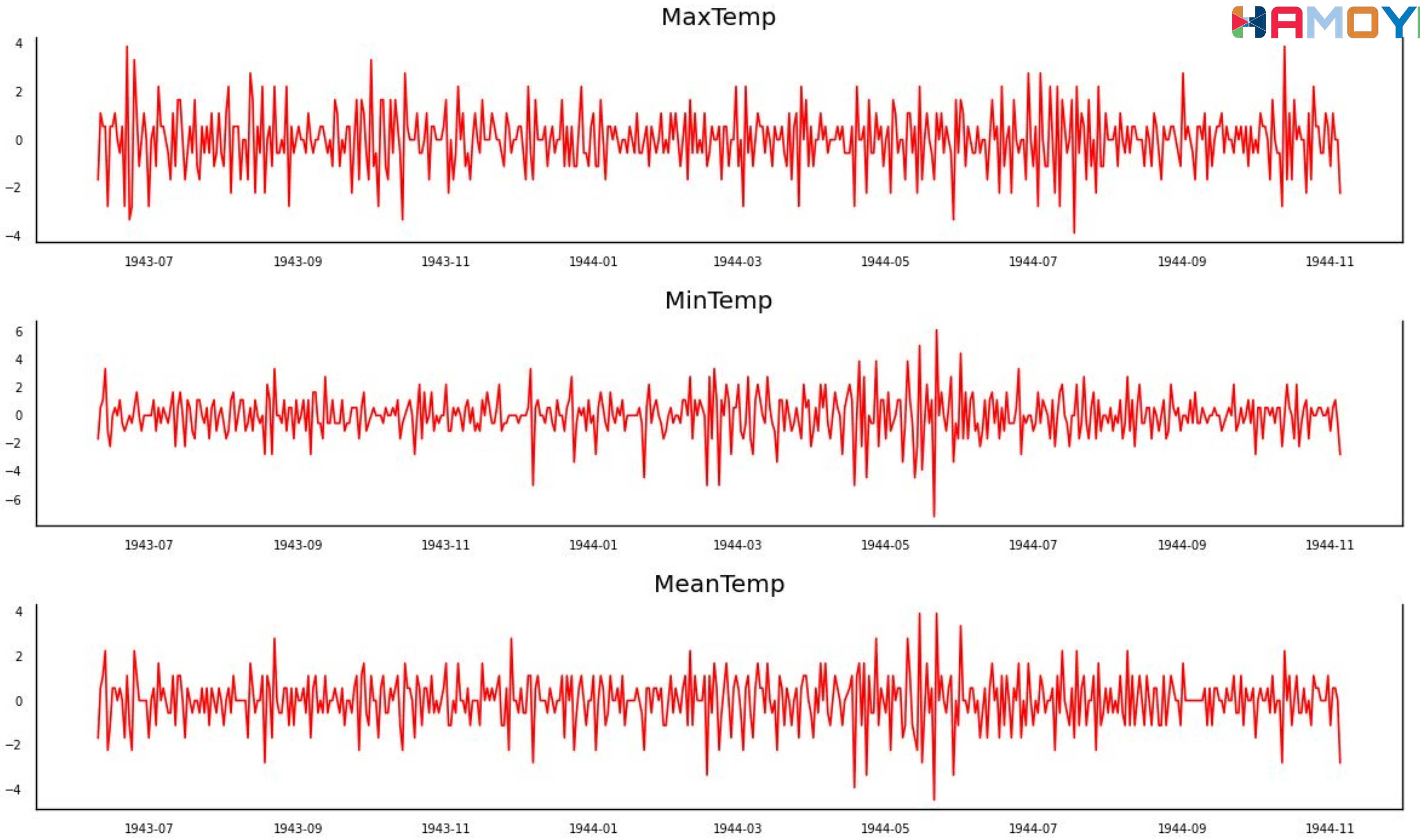
Model

- After importing data we went through the usual data wrangling ritual (selecting columns of interest, summary statistics etc.).
- we visualize the data to give us the necessary intuition needed for model evaluation.


MaxTemp
32
30
28
26
24
1943-07
1943-09
1943-11
1944-01
1944-03
1944-05
1944-07
1944-09
1944-11
MinTemp
28
26
24
22
20
18
1943-07
1943-09
1943-11
1944-01
1944-03
1944-05
1944-07
1944-09
1944-11
MeanTemp
30
28
26
24
22
1943-07
1943-09
1943-11
1944-01
1944-03
1944-05
1944-07
1944-09
1944-11

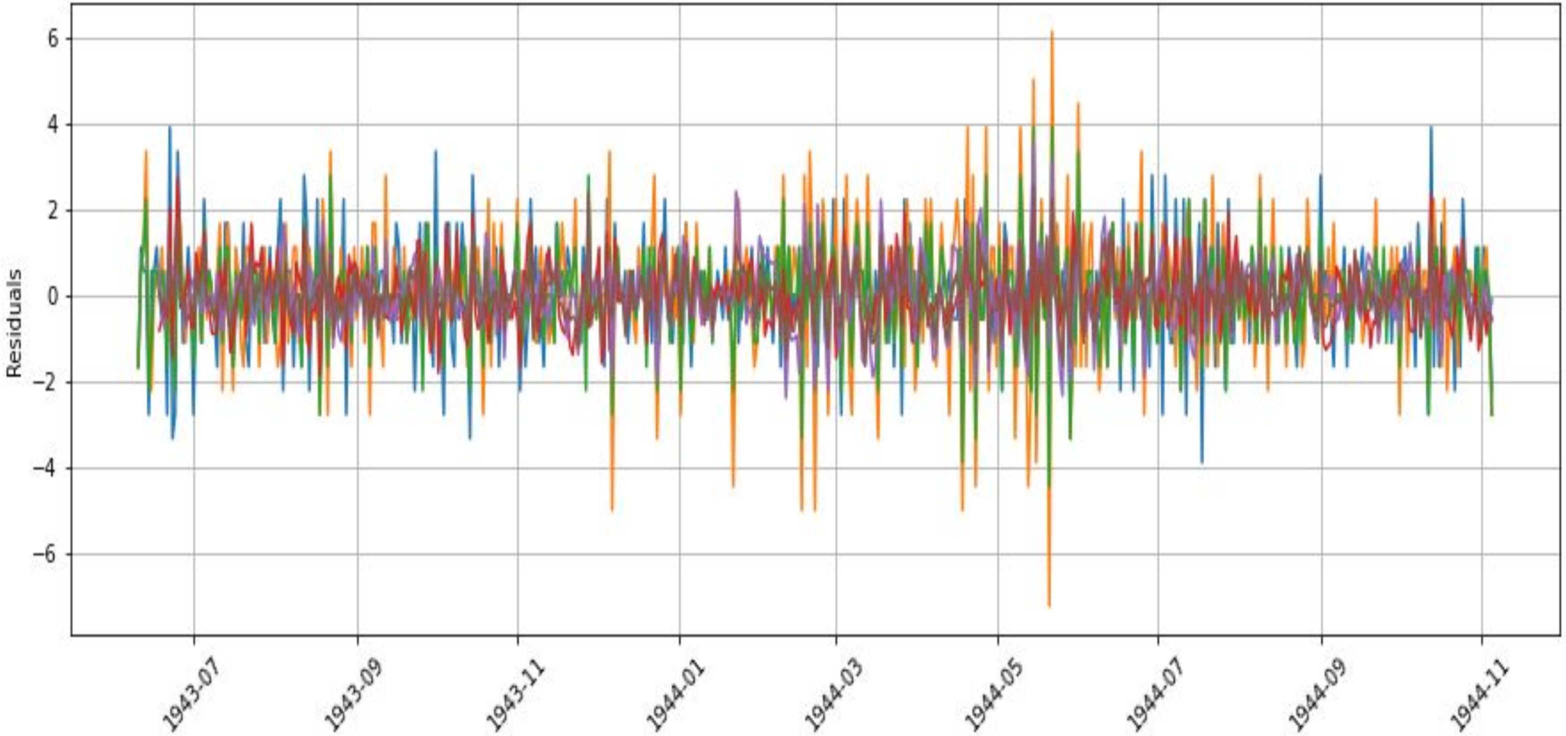
VAR Model

- We checked whether data is stationary. For that we run Augmented Dickey-Fuller (ADF) Test
- We performed series transformation to remove systematic structure from the Time series
- We checked if there's a correlation between the variables. For that we run Granger's causality test
- We split data into training and testing set


MaxTemp
4
2
0
−2
−4
1943-07
1943-09
1943-11
1944-01
1944-03
1944-05
1944-07
1944-09
1944-11
MinTemp
6
4
2
0
−2
−4
−6
1943-07
1943-09
1943-11
1944-01
1944-03
1944-05
1944-07
1944-09
1944-11
MeanTemp
4
2
0
−2
−4
1943-07
1943-09
1943-11
1944-01
1944-03
1944-05
1944-07
1944-09
1944-11

Model

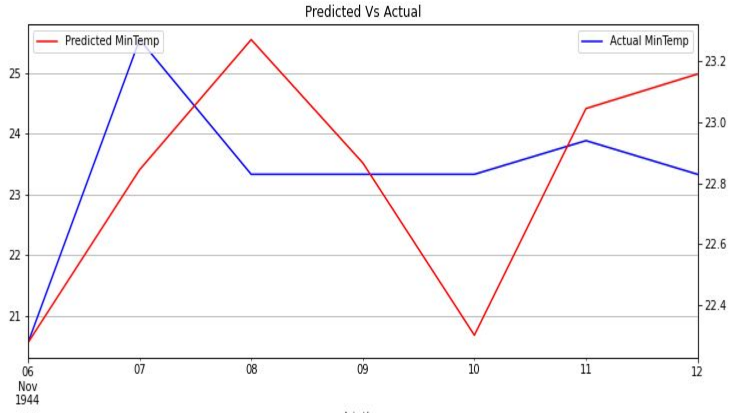
- We instantiate the model and then fit the model to first differenced data.
- Forecasting
- Invert transformation
- Plotting
- Evaluate the forecasts, we compute a comprehensive set of metrics, namely, the MAPE, ME, MAE, MPE, RMSE, corr and minmax.

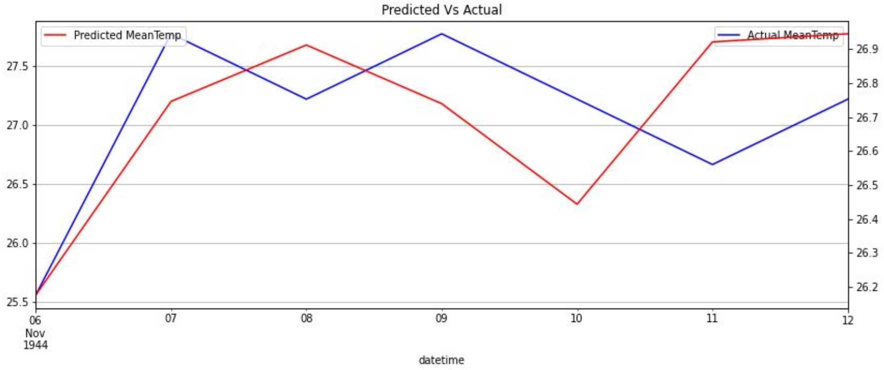


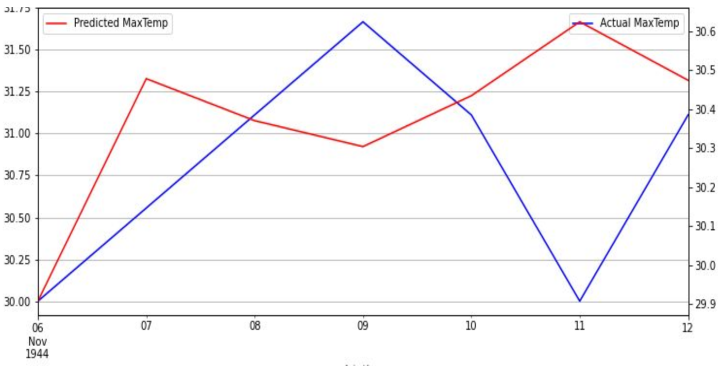
Residual plot looks normal with constant mean throughout apart from some large fluctuation

VAR MODEL

Predicted Vs Actual







LSTM Model

- The Long Short Term Memory(LSTM) is a gated recurrent neural network
- It takes two inputs, the state of the previous layer and that of the present layer
- To use LSTM for time series the data needs to be converted to supervised learning, while keeping intact the series order
- Previous observations are used as the predictors(features) and the next in the sequence is used as the predicted(target).

LSTM Model

These are the procedures taken to fit and use for prediction;

- Scaling the values using MinMax Scaler
- Transforming from a time series to a supervised learning format
- Splitting the data into train, val, test sets and into X and Y sets (i.e feature and target sets)
- Reshaping the input columns to 3D for input into the LSTM
- The model is defined using Keras Sequential model with a Bidirectional LSTM with 50 neurons, a Dropout rate of 0.5, and a Dense output of 3 since we forecasting 3 columns
- The model is compiled with the Adam optimizer and MAE as loss equation.
- The model is then fit with 50 epochs and a batch size of 30, with shuffle set to False

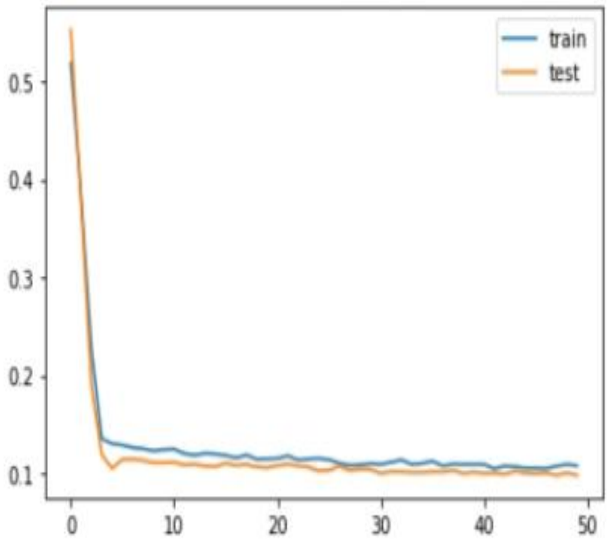
LSTM Model

Procedures taken for prediction and visualization of results

- Plotting the train and validation loss
- Using the predict function of the keras Sequential model on the test set of 7 days
- The test set is reshaped back to 2D vector
- The test set and the forecast are inverted back to the normal values
- The forecast and true values are plotted
- Performance metrics are evaluated on the forecast values i.e MAPE, ME, MAE, MPE, RMSE, CORR and MINMAX.

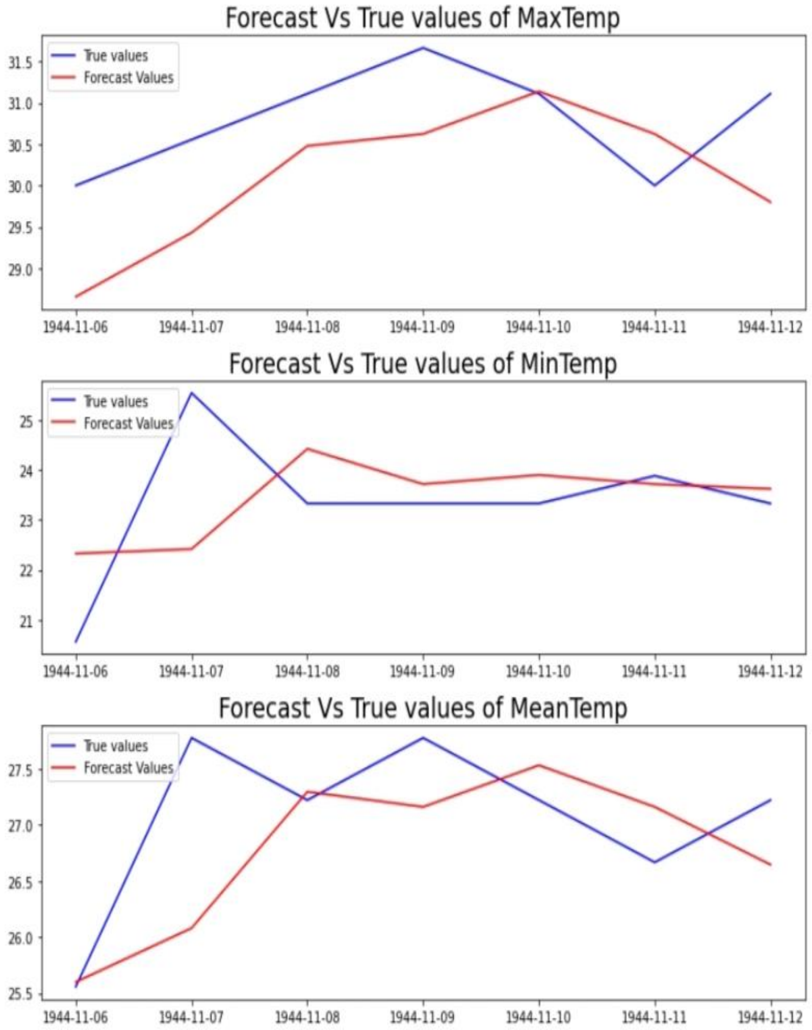
LSTM Model

Train/validation loss



LSTM Model

Forecast Vs
True values



Creating a Machine Learning Pipeline

We converted our LSTM model to a service by creating a machine learning pipeline which could be utilized by the open-source community.

The benefits of this process include:

- Portability
- Scalability
- Reproducibility
- Scheduling and Runtime Optimization
- Language and Framework Agonistic

Machine Learning Pipeline Components

The components of our pipeline include:

- Data Injections
- Data Transformation
- Model Building
- Model Packaging
- Model Validation

Some important components that could be added in the future include:

- Model Deployment
- Model Monitoring

Summary

- In this Project we covered VAR, VECM, LSTM from scratch beginning from the intuition behind it, causality tests, preparing the data for forecasting, build the model, inverting the transform to get the actual forecasts, plotting the results and computing the accuracy metrics.

THANK YOU!