A REPORT

ON

"Statistical Analysis and Forecasting of Wind Energy (Intra-State)"

By:

Group 2- (Bernoulli Group)

Somil Singla (2018B4A70854P)

Sanskar Jhajharia (2019A7PS0148P)

Deepshikha Sharma (2018B2A30595P)

Nihir Agarwal (2018B4A70701P)

Nandan Parikh (2019A7PS0097P)

Mukund Choudhary (2018B4A30878P)

Krupa Bhayani (2018B4A70844P)

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

7th December, 2021

Table Of Contents

INTRODUCTION

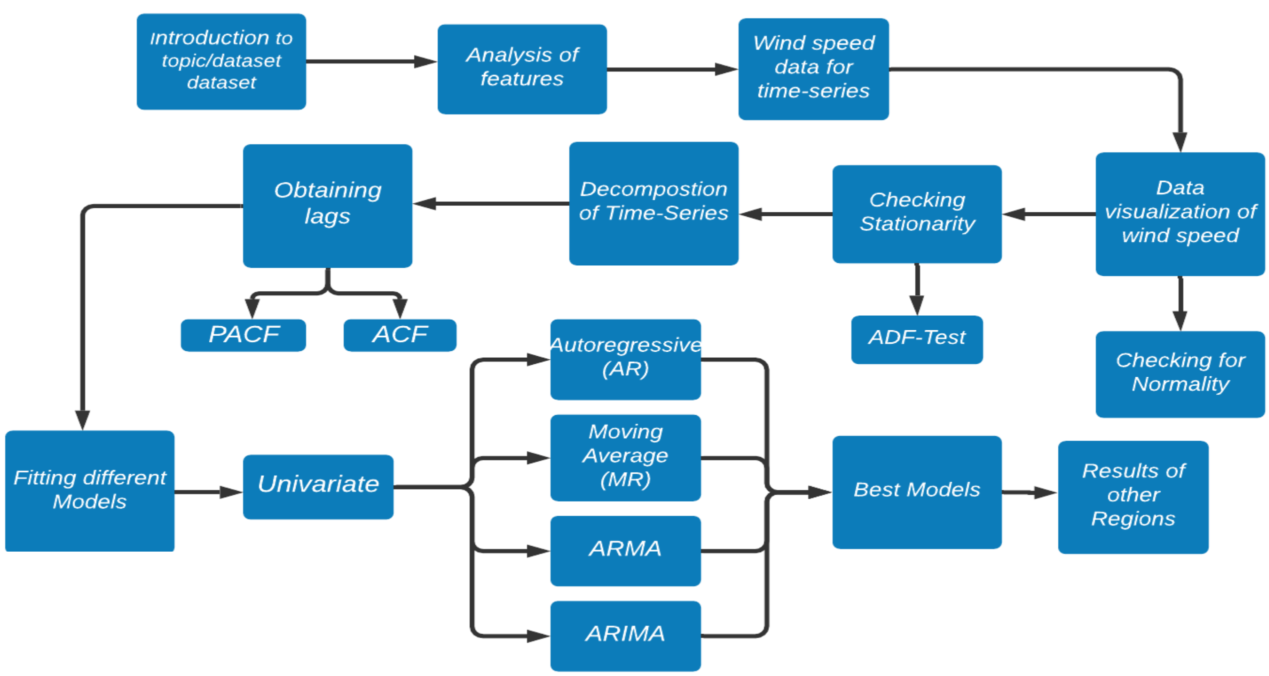
The use of wind turbines to generate electricity is known as wind energy. Wind energy is a popular, renewable energy source that has a far lower environmental impact than burning fossil fuels. Many individual wind turbines are connected to the electric power transmission network to form wind farms. This scientific process is highly volatile and varying. Because the quantity of energy created is dependent on wind speed, which is dependent on various geographic elements such as location, temperature, humidity, pressure, dew point, and so on, it necessitates a great deal of forecast and analysis of current data.

Parameters:-

1. *Direct Normal Irradiance (DNI)*: The quantity of solar radiation received per unit area by a surface that is always maintained perpendicular (or normal) to the rays that arrive in a straight line from the direction of the sun at its present location in the sky is known as direct normal irradiation (DNI).
2. *Diffuse Horizontal Irradiance (DHI)*: It is the quantity of energy received per unit area by a surface that has been dispersed by molecules and particles in the atmosphere rather than arriving on a direct path from the sun. Essentially, it is the light emitted by clouds and the blue sky.
3. *Global Horizontal Irradiance (GHI)*: The radiation that reaches the surface of the planet can be depicted in a variety of ways. The entire quantity of shortwave radiation received by a surface horizontal to the ground is referred to as global horizontal irradiance (GHI). This figure, which combines both Direct Normal Irradiance (DNI) and Diffuse Horizontal Irradiance (DHI), is particularly important for solar systems.

Temperature, pressure, dew point and humidity (relative and absolute) affect the wind speed, i.e. a higher pressure gradient leads to higher wind speed and this pressure gradient is related to the temperature and humidity.

METHODOLOGY



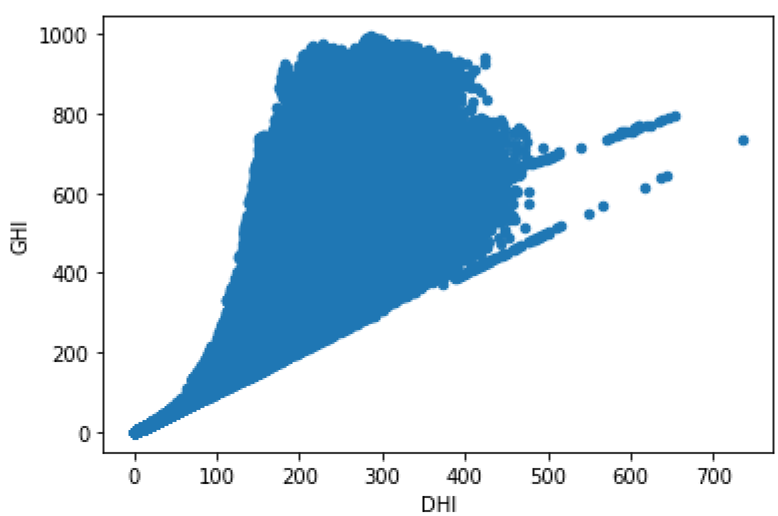
ANALYSIS OF PARAMETERS

Given below is a plot of the correlation values of the variables amongst themselves:

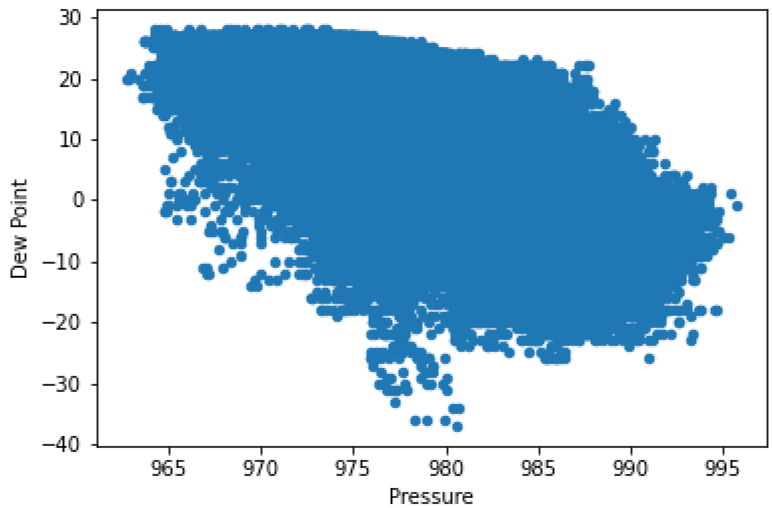
```
1 corr = df2.corr()
2 corr.style.background_gradient(cmap='Blues')
```

1	DHI	DNI	GHI	Dew Point	Temperature	Pressure	Relative Humidity	Wind Speed
1								
DHI	1.000000	0.810126	0.927168	0.136632	0.614919	-0.165429	-0.204736	-0.082755
DNI	0.810126	1.000000	0.940300	-0.056640	0.466896	0.079642	-0.316748	-0.196673
GHI	0.927168	0.940300	1.000000	0.063107	0.591803	-0.076907	-0.262982	-0.123800
Dew Point	0.136632	-0.056640	0.063107	1.000000	0.495483	-0.778792	0.805823	0.289728
Temperature	0.614919	0.466896	0.591803	0.495483	1.000000	-0.644223	-0.035681	0.108974
Pressure	-0.165429	0.079642	-0.076907	-0.778792	-0.644223	1.000000	-0.504076	-0.395237
Relative Humidity	-0.204736	-0.316748	-0.262982	0.805823	-0.035681	-0.504076	1.000000	0.223948
Wind Speed	-0.082755	-0.196673	-0.123800	0.289728	0.108974	-0.395237	0.223948	1.000000

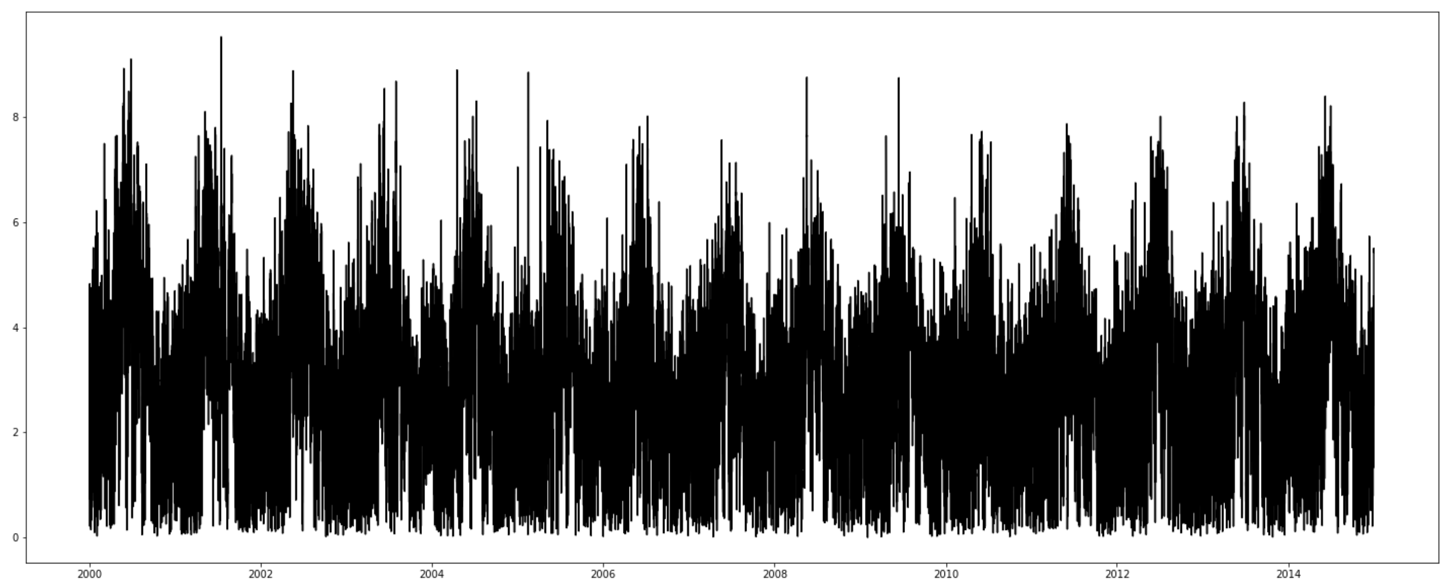
To analyse the correlation between the factors, we tried plotting the graphs.
The following Graph shows the case of a strong positive correlation between GHI and DHI (0.927168)



The following graph is a case of strong negative correlation between Dew Point and Pressure (-0.778792)



WIND IS SPEED DATA FOR TIME SERIES



Given is the graph for the timeseries data of the wind speed over the years mentioned.

DISTRIBUTION OF DATA

The distribution of wind speed data was estimated using the maximum likelihood method on several known distributions. The goodness of fit was then determined using the **Kolmogorov-Smirnov** test at 5% significance level.

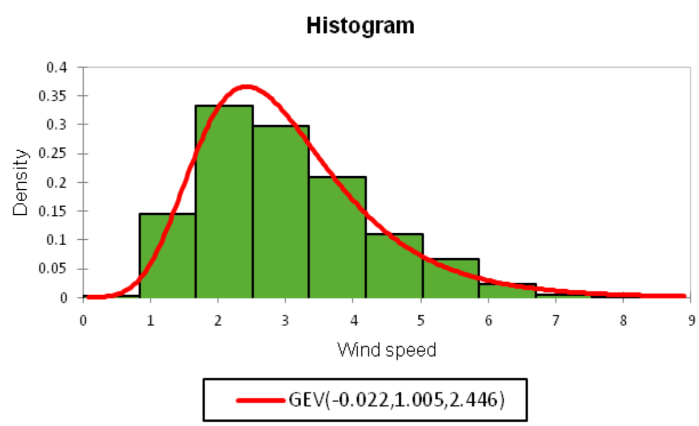
The hypothesis used for the **KS** test was:

H_0: The sample follows the tested distribution

H_a: The sample does not follows the tested distribution

The p-values of some of the prominent distributions are tabulated below along with the histogram of the best fit distribution(**GEV**). From the graph the data is right skewed with skewness of 0.87.

Distribution	p-value
Normal	0.000
Gamma	0.023
GEV	0.038
Exponential	0.000
Fisher-Tippett	0.022

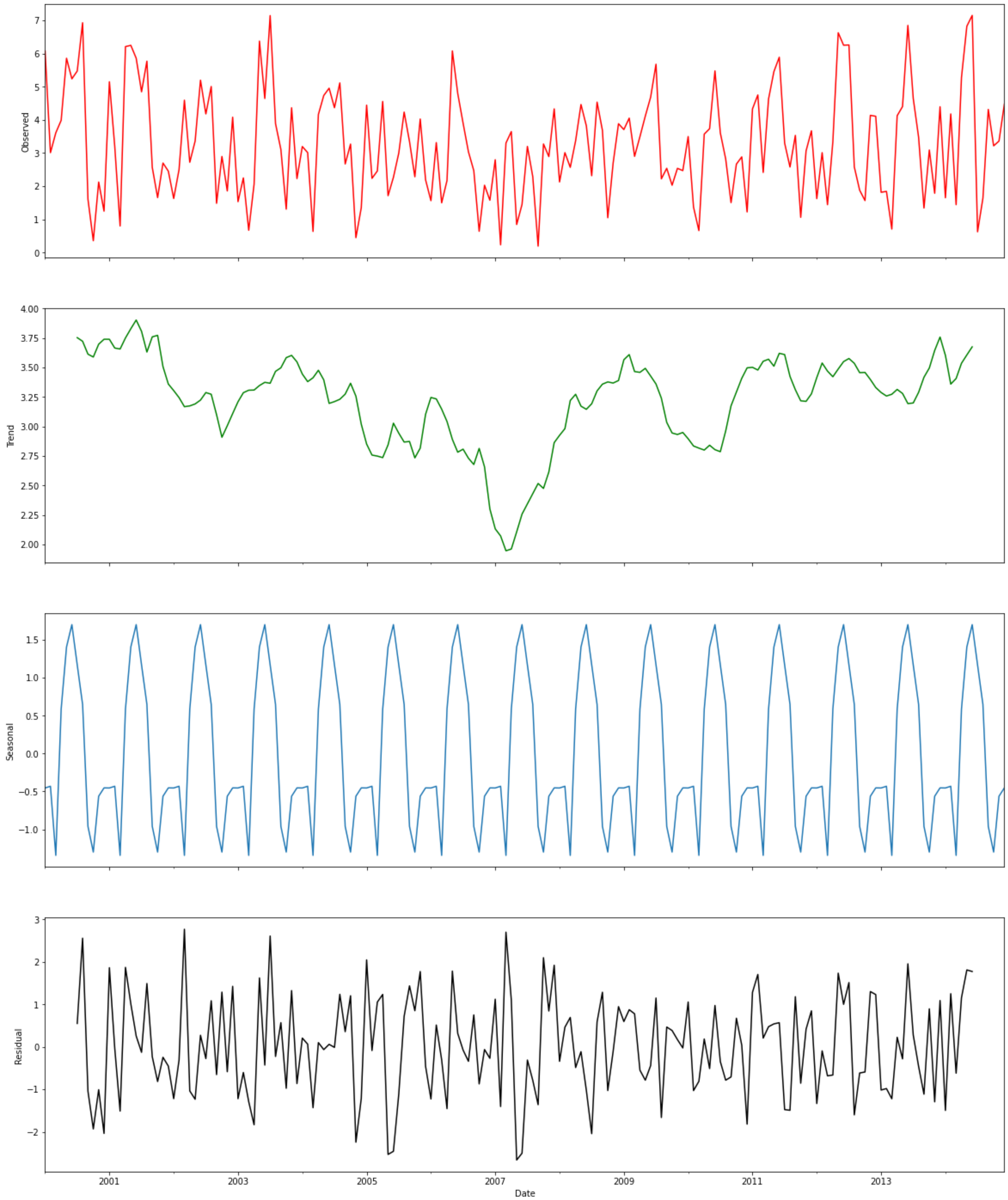


As apparent from the p-values and the frequency plot the distribution is not normal. The closest fit was obtained for the **GEV** (Generalised Extreme values) distribution and even that did not pass the Kolmogorov-Smirnov test. The parameters estimated using **MLE** for the **GEV** distribution and its **KS** test statistics are given below.

D	0.019
P-value (Two-tailed)	0.038
alpha	0.05

Parameter	Value	Standard Error
k	-0.022	0.012
beta	1.005	0.013
μ	2.446	0.012

SERIES DECOMPOSITION

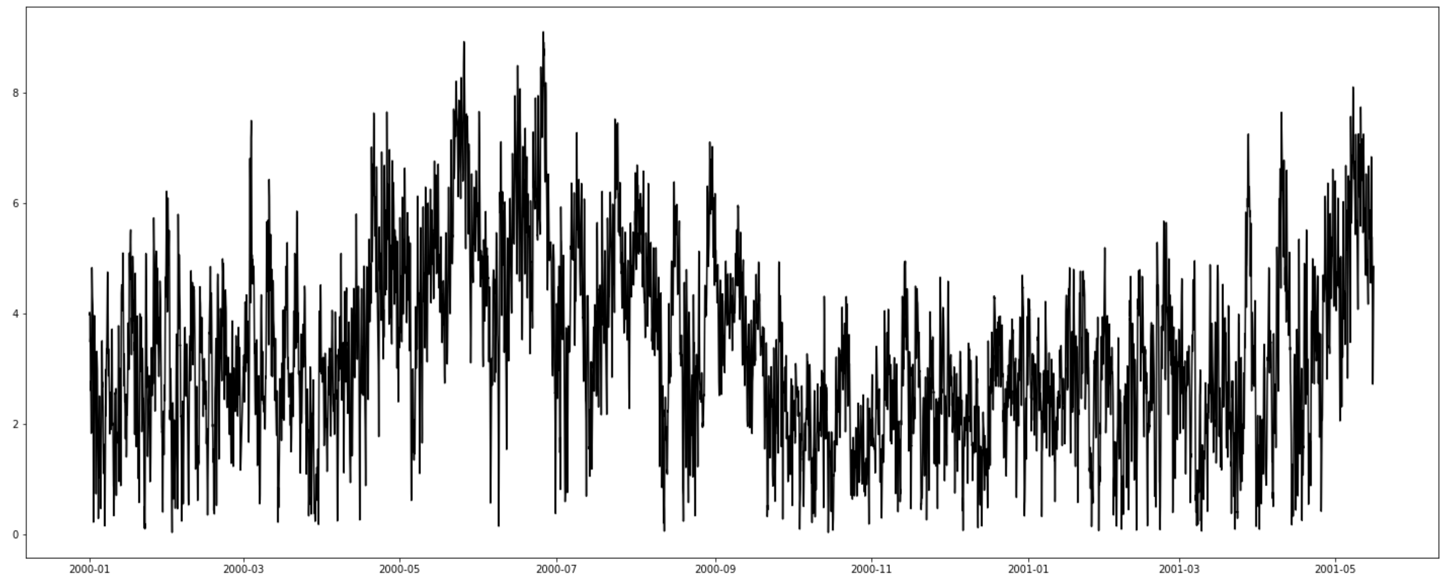


The series has been decomposed into various components such as trend, seasonality and residue, as seen from the figure above there is not much difference between the observed and the residue component. It can be interpreted that there is not much noise in the data. Since seasonal fluctuations don't vary with the level of the series so an additive model was preferred.

DATA STATIONARITY

In stationary time series, the statistical properties like mean, variance and covariance do not depend on time. We require the time series to be stationary for effective and precise predictions using various statistical models.

On visualisation of our data and its decomposition plot , we inferred that there was no major trend and data was distributed around some mean value. Though, there is a clear seasonality on observing the whole data but this is the compressed form for 14 years, If we observe a part of data (say for initial 400 days), we will get the following plot -



Clearly, there is no seasonality factor in the above plot. These observations give us indication for stationarity in our plot but for a conclusive result we need a mathematical tool to confirm stationarity.

AUGMENTED DICKEY–FULLER TEST

This is the mathematical tool or test used for checking stationarity in time series. It determines the presence of unit root in the series. A unit root is a stochastic trend in a time series. If a time series has a unit root then it shows a systematic pattern that is unpredictable. The null and alternative hypotheses for this test are :

Null hypothesis (H_o) : The series has a unit root (or the series is non stationary)
Alternate hypothesis (H_1) : The series has no unit root (or the series is stationary)

On application of AD Fuller test, we obtain the following results -

```
from statsmodels.tsa.stattools import adfuller
series = df['Wind Speed']
X = series.values
result = adfuller(X[:1000])
print('ADF Statistic: %f' % result[0])
print('p-value:', result[1])
print('No of observations used:',result[3])
```

```
ADF Statistic: -6.440515
p-value: 1.612483425216694e-08
No of observations used: 995
```

```
ADF Test Statistic : -6.3795250243283546
p-value : 2.240211241769144e-08
#Lags Used : 31
Number of Observations Used : 5447
strong evidence against the null hypothesis(Ho), reject the null hypothesis. Data has no unit root and is stationary
```

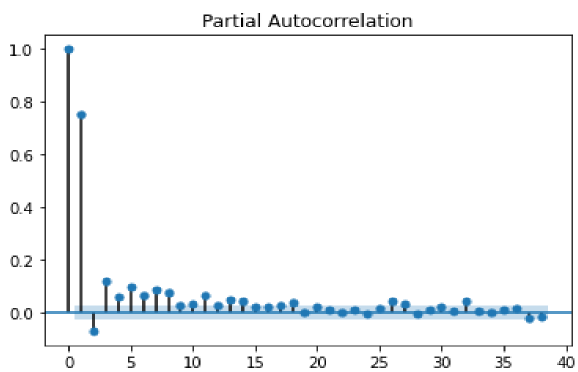
On testing our hypothesis on 1% (or 5%) confidence interval, we observe that our p-value is much lesser than the critical p-value of 0.01. So, the null hypothesis is rejected and we conclude that there is no unit root and our **time series is stationary.**

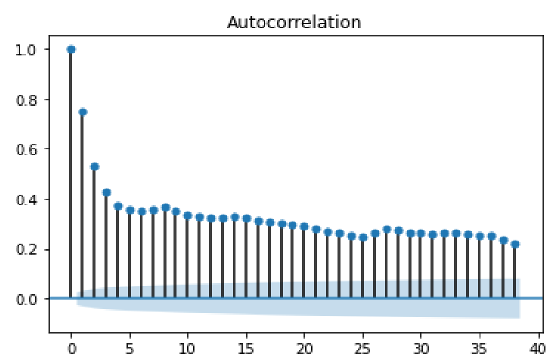
LAG PLOTS

Now, before proceeding for implementing various statistical methods on our statistical time series, we must obtain the value of model lags and moving average lags which are necessary for our modelling. To obtain these values, we have two lag plots -

- **ACF plot** : This autocorrelation function plot is the plot between correlation coefficients (pearson) between time series and lags of itself. It takes both the direct and indirect effects of lags on the correlation coefficient. Using ACF plot, we obtain moving average lags for our time series.
- **PACF plot** : This plots the partial autocorrelation coefficients which are obtained by considering only the direct impacts of the previous lags. Using PACF plot, we obtain model lags for modelling of our time series.

For our Wind Speed time series data, we obtain the following ACF and PACF plots -





From the PACF plot, we observe that there is only one significant correlation coefficient for the first lag. So, the model lags (p) for time series is 1. From the ACF plot, we observe that after three lags, the slope of the plot almost tapers to 0, so our moving average lags (q) are 3.

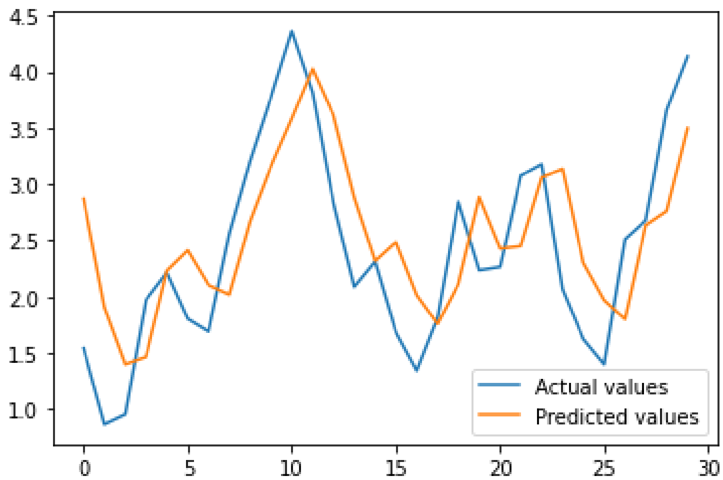
STATISTICAL METHODS AND PREDICTIONS

1. AutoRegression (AR) model :

Autoregression models predict the future values of time series using its own past values. From our PACF plot , we obtained number of lags for our model as 1 , so our model is first order autoregressive model and the corresponding equation will be -

$$Y_t = C_0 + C_1 y_{t-1}$$

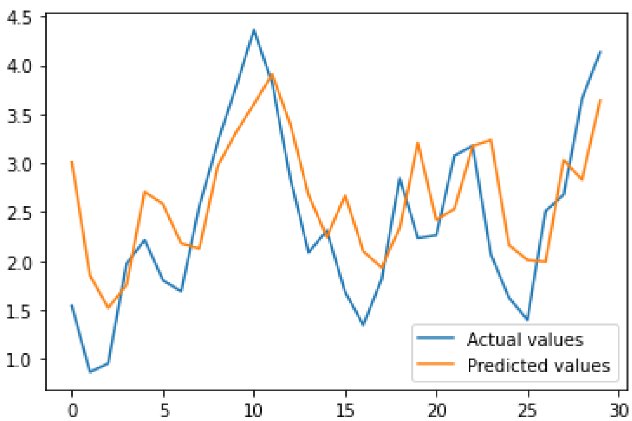
Here, y_{t-1} is the wind speed at first time lag. C_ks are the constant value and Y_t is the predicted value of wind speed at 't' th time step.



After fitting the AR model (p=1) on our data, given is the plotted predictions of wind speed for the last 30 days of the time series.

The MAE score obtained was **0.401**.

2. Moving Average (MA) model :



A moving average is a technique to get an overall idea of the trends in a data set. This model predicts the future values of time series using the past errors. From our ACF plot , we obtained number of moving average lags for our model as 3 , so our model is third order moving average model and the corresponding equation will be -

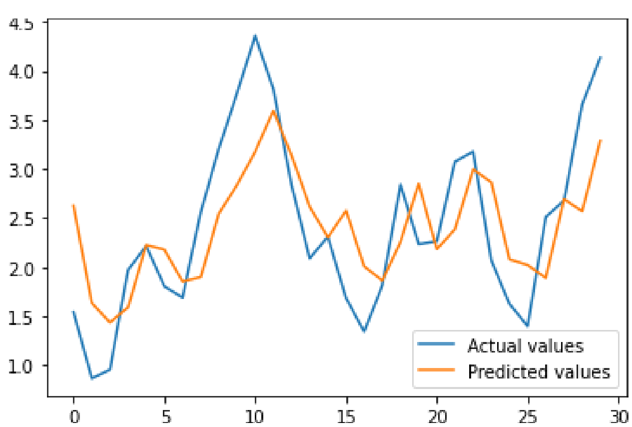
$$Y_t = C_1 + C_2\varepsilon_{t-1} + C_3\varepsilon_{t-2} + C_4\varepsilon_{t-3}$$

Here, ε_k is the error at 'k'th time lag. C_ks are the constant value and Y_t is the predicted value of wind speed at 't' th time step.

After fitting the MA model (q=3) on data, given is the plotted predictions of wind speed for the last 30 days of the time series.

The MAE score obtained was **0.417.**

3. Autoregressive Moving Average (ARMA) model :



An ARMA model, or Autoregressive Moving Average model, is the combination of two polynomials, one for autoregression and the other for moving average. As our model lags and moving average lags are 1 and 3 respectively, we will be fitting ARMA(1,3) model on our data.

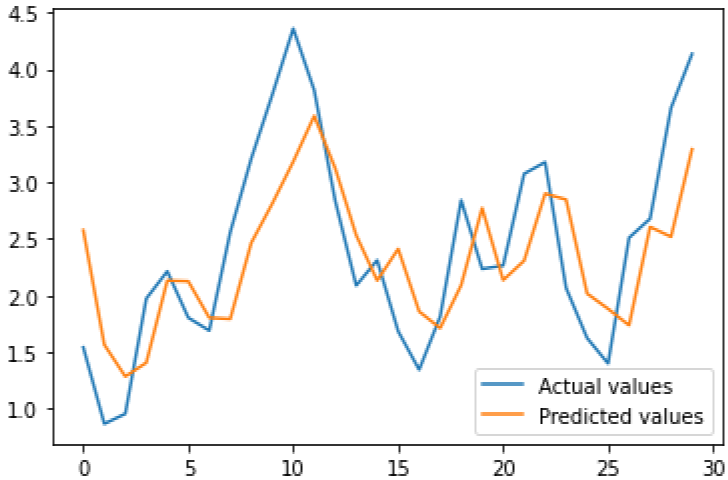
After fitting the ARMA (1,3) model on the data, given is the plotted predictions of wind speed for the last 30 days of the time series.

The MAE score obtained was **0.394**.

4. Autoregressive Integrated Moving Average (ARIMA) model :

ARIMA or 'Auto Regressive Integrated Moving Average' model has the "integrated" part added to the ARMA model. The integrated part refers to the differencing (d) which is necessary to make a time series stationary . As wind speed data was already stationary, so differencing will be 0 and hence ARMA and ARIMA models will be the same.

5. Seasonal Autoregressive Integrated Moving Average (SARIMA) model :

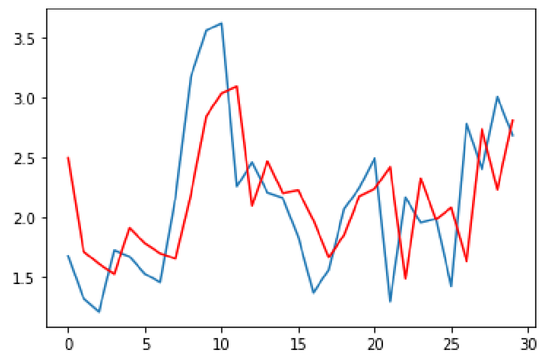


SARIMA model combines the ARIMA model with the ability to perform the same autoregression , differencing, and moving average modelling at seasonal level. We tried a grid search on the parameters of the order and the seasonal order. For us (2, 1, 2) (0, 0, 0) (0) was the best fit for the data. With more computational power we would have tried to do the same procedure with 365 as the seasonal variable but unfortunately it was too computational intensive.

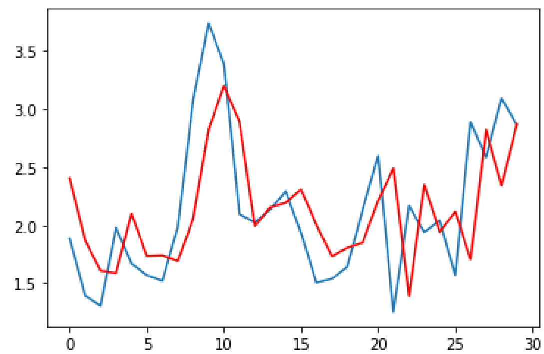
We obtained a MAE score of 0.393

SPATIAL ANALYSIS

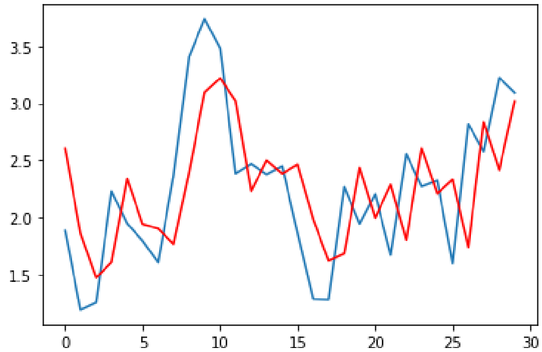
The forecasting is done for inter-state regions of Rajasthan, the above given results are obtained for Region-1. To carry out forecasting of other 4 regions, among the above given statistical methods, ARMA seems to work better as it gives 74.3% accuracy and a MSE value of 0.396. Thus we forecasted the wind speed for the other regions on the ARMA(1,3) model with the same parameters. Given below are the graphs of forecasting with the MSE scores. To our surprise, the results obtained with the same model on the other 4 regions are almost similar(or better than) to region-1. This can be because the regions are of the same state and thus will have almost the similar wind speed daily.



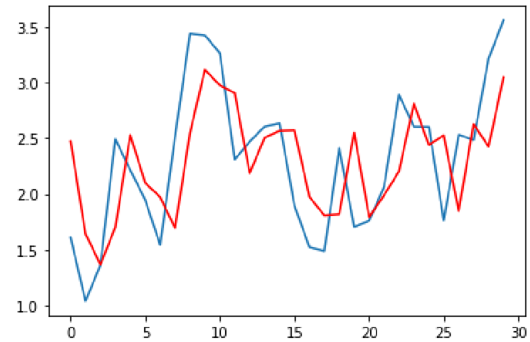
Region-2, MSE = 0.308



Region-3, MSE = 0.299



Region4, MSE = 0.302



Region5, MSE = 0.284

Models comparison:

	AR model	MA model	ARMA/ARIMA model	SARIMA Model
MAE Score	**0.401**	**0.417**	**0.394**	**0.393**

In comparison for all the models, the SARIMA model performed marginally well giving the least MAE score but was at the same time computationally heavy.

APPENDIX: IMPLEMENTATION AND MODEL SUMMARY

All the implementations are done in python.

AR Model Implementation

```
from statsmodels.tsa.ar_model import AutoReg
model=AutoReg(df["Wind Speed"], lags=1).fit()
model.summary()
```

AutoReg Model Results

Dep. Variable:	Wind Speed	**No. Observations:**	5479
Model:	AutoReg(1)	**Log Likelihood**	-6690.484
Method:	Conditional MLE	**S.D. of innovations**	0.821
Date:	Fri, 27 Nov 2020	**AIC**	-0.394
Time:	14:23:43	**BIC**	-0.390
Sample:	01-02-2000	**HQIC**	-0.393
	- 12-31-2014		

	coef	std err	z	P>|z|	[0.025	0.975]
intercept	0.7497	0.029	25.797	0.000	0.693	0.807
Wind Speed.L1	0.7510	0.009	84.178	0.000	0.734	0.769

Roots

	Real	Imaginary	Modulus	Frequency
AR.1	1.3315	+0.0000j	1.3315	0.0000

MA Model Implementation

```
from statsmodels.tsa.arima_model import ARMA
model2 = ARMA(df["Wind Speed"], order=(0, 3)).fit()
model2.summary()
```

```
/usr/local/lib/python3.6/dist-packages/statsmodels/tsa/base/ts
  % freq, ValueWarning)
```

ARMA Model Results

Dep. Variable:	Wind Speed	**No. Observations:**	5479
Model:	ARMA(0, 3)	**Log Likelihood**	-6811.783
Method:	css-mle	**S.D. of innovations**	0.839
Date:	Fri, 27 Nov 2020	**AIC**	13633.567
Time:	17:09:03	**BIC**	13666.610
Sample:	01-01-2000	**HQIC**	13645.094
	- 12-31-2014		

	coef	std err	z	P>|z|	[0.025	0.975]
const	3.0113	0.029	105.523	0.000	2.955	3.067
ma.L1.Wind Speed	0.8268	0.013	62.099	0.000	0.801	0.853
ma.L2.Wind Speed	0.4733	0.014	32.914	0.000	0.445	0.501
ma.L3.Wind Speed	0.2186	0.012	18.460	0.000	0.195	0.242

Roots

	Real	Imaginary	Modulus	Frequency
MA.1	-1.5934	-0.0000j	1.5934	-0.5000
MA.2	-0.2860	-1.6702j	1.6945	-0.2770
MA.3	-0.2860	+1.6702j	1.6945	0.2770

ARMA Model Implementation

```python
from statsmodels.tsa.arima_model import ARMA
model2 = ARMA(df["Wind Speed"], order=(1, 3)).fit()
model2.summary()
```

```
/usr/local/lib/python3.6/dist-packages/statsmodels/tsa/
  % freq, ValueWarning)
```

ARMA Model Results

Dep. Variable:	Wind Speed	**No. Observations:**	5479
Model:	ARMA(1, 3)	**Log Likelihood**	-6550.122
Method:	css-mle	**S.D. of innovations**	0.800
Date:	Fri, 27 Nov 2020	**AIC**	13112.244
Time:	18:01:06	**BIC**	13151.896
Sample:	01-01-2000	**HQIC**	13126.077
	- 12-31-2014		

	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
const	3.0077	0.105	28.523	0.000	2.801	3.214
ar.L1.Wind Speed	0.9775	0.005	214.920	0.000	0.969	0.986
ma.L1.Wind Speed	-0.2284	0.014	-16.030	0.000	-0.256	-0.200
ma.L2.Wind Speed	-0.3708	0.015	-24.516	0.000	-0.400	-0.341
ma.L3.Wind Speed	-0.1794	0.013	-13.342	0.000	-0.206	-0.153

Roots

	Real	**Imaginary**	**Modulus**	**Frequency**
AR.1	1.0231	+0.0000j	1.0231	0.0000
MA.1	1.1355	-0.0000j	1.1355	-0.0000
MA.2	-1.6011	-1.5315j	2.2156	-0.3785
MA.3	-1.6011	+1.5315j	2.2156	0.3785

REFERENCES

- Andreson, D., Sweeny, & Camm, J. D. (2008). *Statistics for Business and Economics* (12th ed.). Cengage Learning.
- MachineLearningMastery Pvt. Ltd. (2017, February). *Machine Learning Mastery*. Introduction to Autocorrelation

  https://machinelearningmastery.com/gentle-introduction-autocorrelation-partial-autocorrelation/
- The Pennsylvania State University. (2019, September 19). *STAT 501*. 14.1 Auto-regressive Models.

  https://online.stat.psu.edu/stat501/lesson/14/14.1
- Selva Pvt. Ltd. (2017, November 1). *ML+*. ARIMA Model.

  https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/
- Statistics How To. (2013, September 24). *Statistics How To*. Moving Average.

  https://www.statisticshowto.com/moving-average
- Statistics How To. (2019, January 4). *Statistics How to*. ARMA model.

  https://www.statisticshowto.com/arma-model/