A Practical Approach to Timeseries Forecasting using Python
Module #4

- Downloading the Dataset
- Manipulation in the Dataset
- Data Preprocessing
- RVT in Time Series in Python
- Feature Engineering and Stationarity in Time Series

Shahzaib Hamid
AI Sciences Instructor

Importance of Dataset

- The lack of quality and quantitative datasets are a cause of concern.
- Strong Dataset offer robust operations, evaluations, testing and trainings.

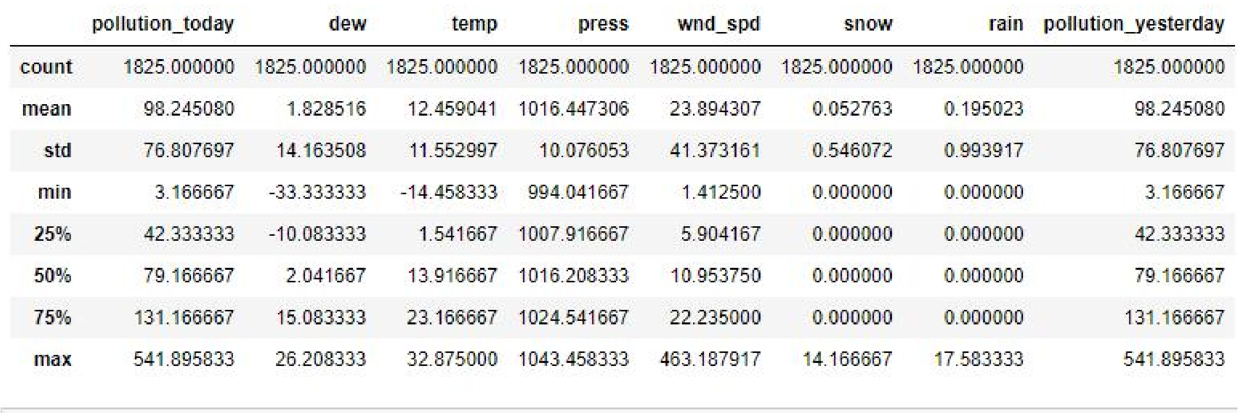
Following are the famous platforms for authenticated datasets

- 2. Famous platforms for authenticated datasets
- https://www.kaggle.com/datasets
- https://github.com/A-I-Studio/Datasets
- https://huggingface.co/datasets
- https://www.worlddata.info

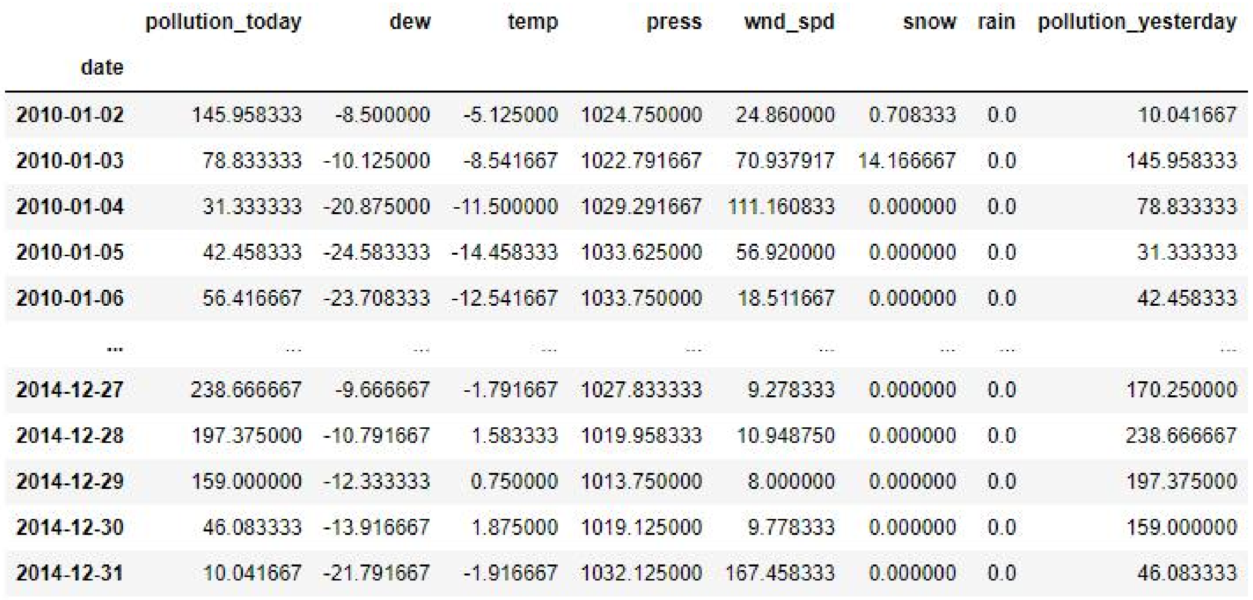
Data Manipulation

Basic Commands:

- air_pollution.columns
- air_pollution.describe()
- pd.DataFrame(air_pollution)



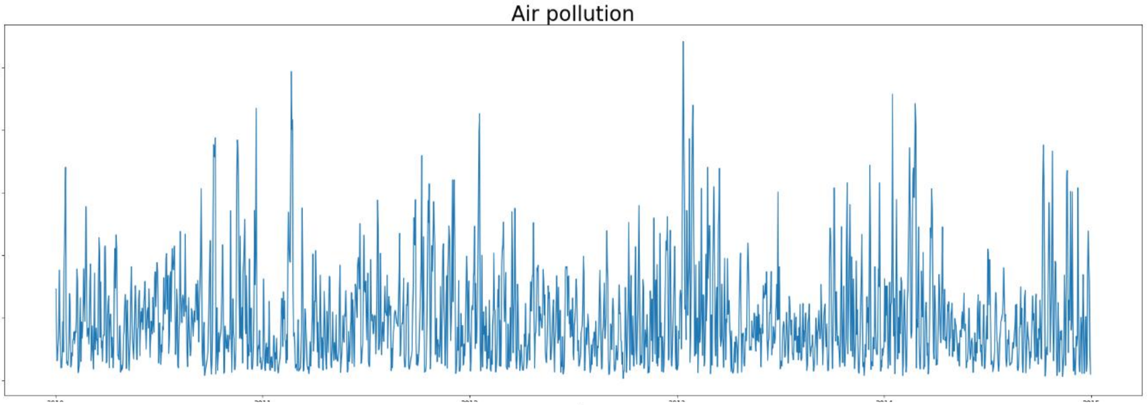
	pollution_today	dew	temp	press	wnd_spd	snow	rain	pollution_yesterday
count	1825.000000	1825.000000	1825.000000	1825.000000	1825.000000	1825.000000	1825.000000	1825.000000
mean	98.245080	1.828516	12.459041	1016.447306	23.894307	0.052763	0.195023	98.245080
std	76.807697	14.163508	11.552997	10.076053	41.373161	0.546072	0.993917	76.807697
min	3.166667	-33.333333	-14.458333	994.041667	1.412500	0.000000	0.000000	3.166667
25%	42.333333	-10.083333	1.541667	1007.916667	5.904167	0.000000	0.000000	42.333333
50%	79.166667	2.041667	13.916667	1016.208333	10.953750	0.000000	0.000000	79.166667
75%	131.166667	15.083333	23.166667	1024.541667	22.235000	0.000000	0.000000	131.166667
max	541.895833	26.208333	32.875000	1043.458333	463.187917	14.166667	17.583333	541.895833

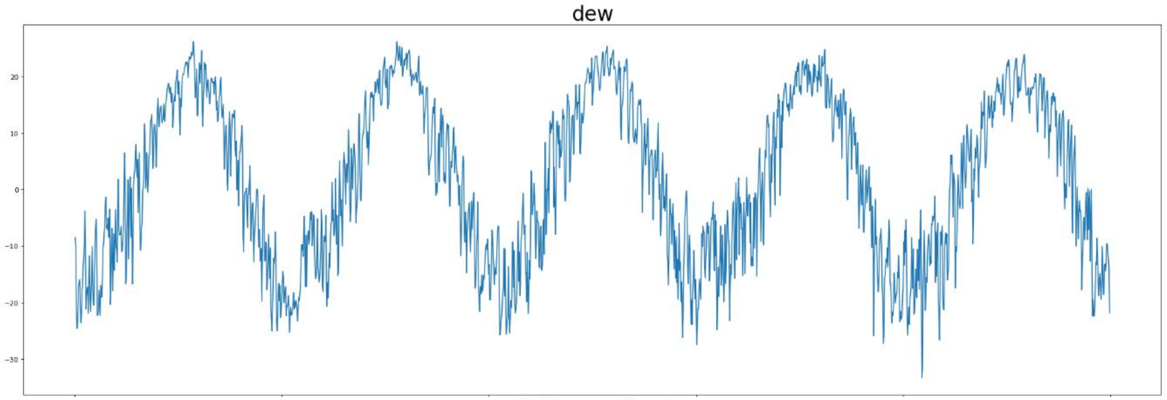


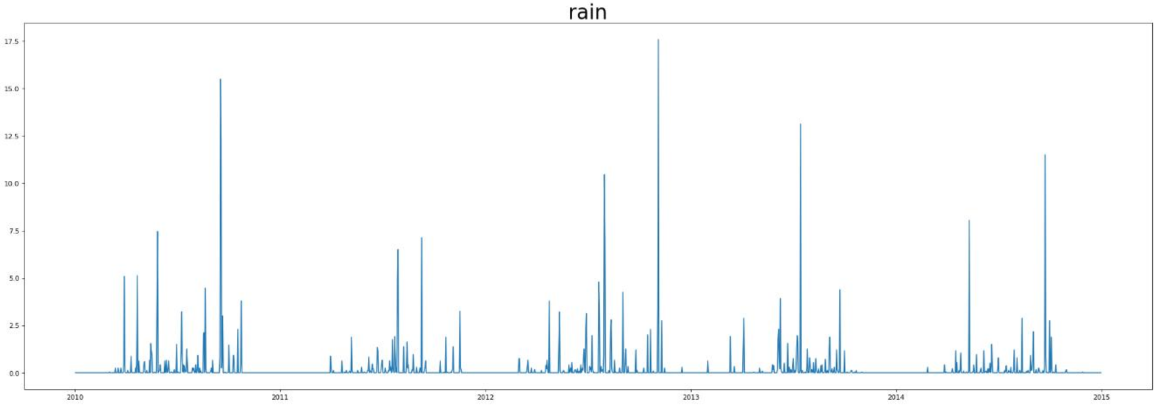
	pollution_today	dew	temp	press	wnd_spd	snow	rain	pollution_yesterday
date								
2010-01-02	145.958333	-8.500000	-5.125000	1024.750000	24.860000	0.708333	0.0	10.041667
2010-01-03	78.833333	-10.125000	-8.541667	1022.791667	70.937917	14.166667	0.0	145.958333
2010-01-04	31.333333	-20.875000	-11.500000	1029.291667	111.160833	0.000000	0.0	78.833333
2010-01-05	42.458333	-24.583333	-14.458333	1033.625000	56.920000	0.000000	0.0	31.333333
2010-01-06	56.416667	-23.708333	-12.541667	1033.750000	18.511667	0.000000	0.0	42.458333
...	...	...	...	...	...	...	...	...
2014-12-27	238.666667	-9.666667	-1.791667	1027.833333	9.278333	0.000000	0.0	170.250000
2014-12-28	197.375000	-10.791667	1.583333	1019.958333	10.948750	0.000000	0.0	238.666667
2014-12-29	159.000000	-12.333333	0.750000	1013.750000	8.000000	0.000000	0.0	197.375000
2014-12-30	46.083333	-13.916667	1.875000	1019.125000	9.778333	0.000000	0.0	159.000000
2014-12-31	10.041667	-21.791667	-1.916667	1032.125000	167.458333	0.000000	0.0	46.083333

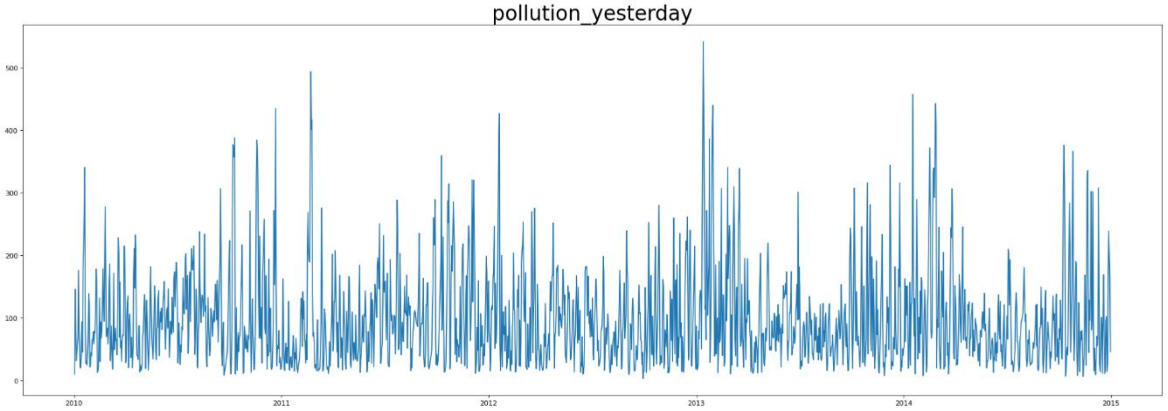
Data Preprocessing

1. Data Cleaning
2. Row and Column Handling
3. Data Visualization for the Dataset in Python









Resampling, Visualize and Transform (RVT)

1. Core Objective: To decompose our series
2. The parts we can divide a time series into are:
    1. Level
    2. Trend
    3. Seasonality and
    4. Noise

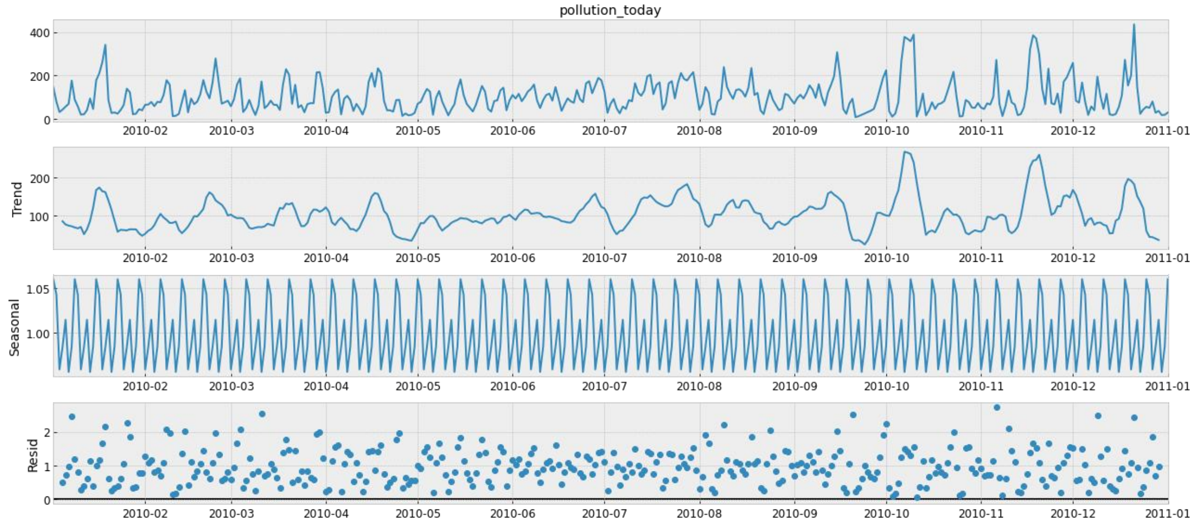
    This part combine either additively or multiplicatively into the time series.

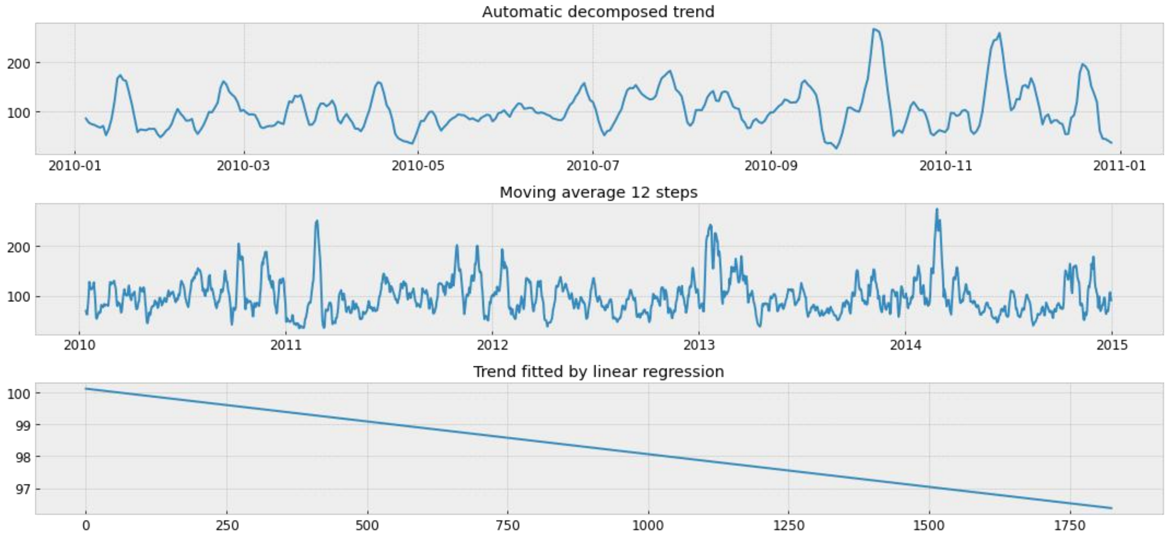
    Additive Model y(t) = Level + Trend + Seasonality + Noise

    Mupltiplicative model y(t) = Level * Trend * Seasonality * Noise

Automatic Time Series Decomposition

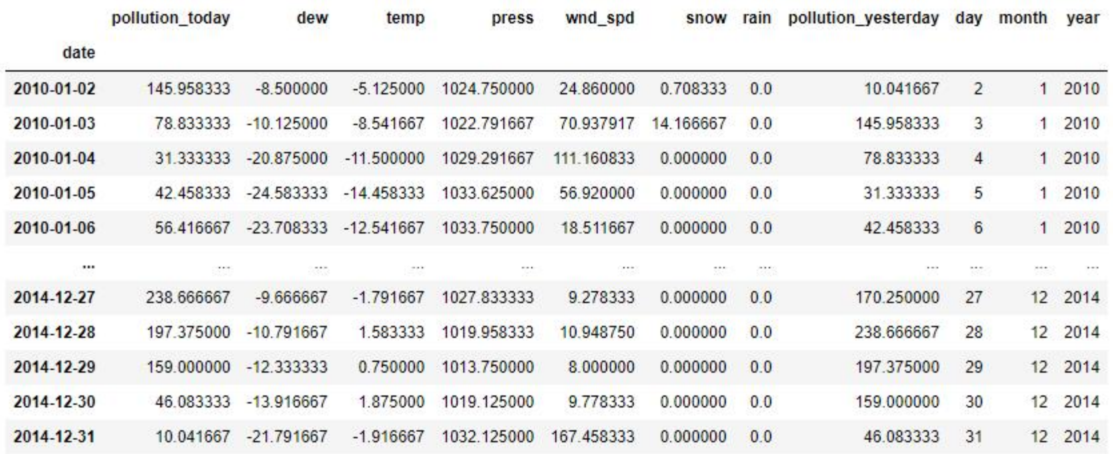
1. Here we will use Statsmodel.
2. Trend in Automatic Time Series Decomposition using Moving Average Filter
3. Seasonality
4. Noise in Automatic Time Series Decomposition



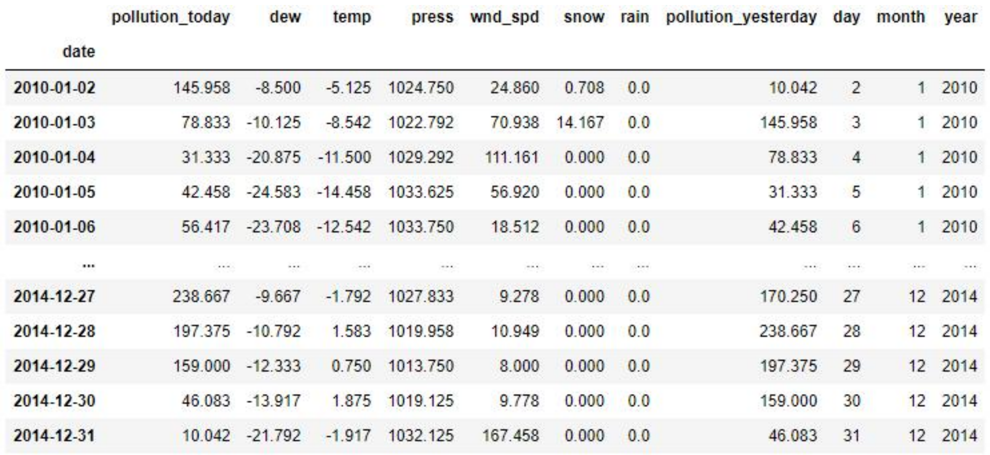


Feature Engineering

1. Handling missing values
2. Handling outliers
3. Combining numeric variable
4. Encoding categorical feature
5. Numerical transformations
6. Calling numerical features



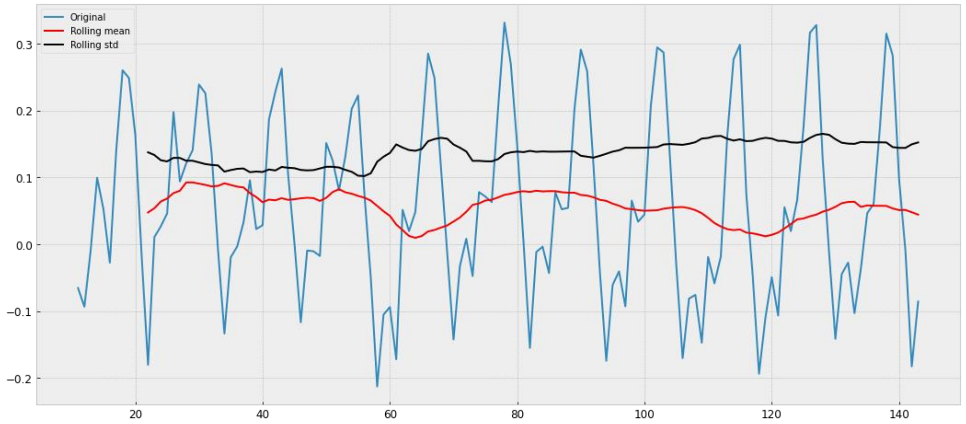
	pollution_today	dew	temp	press	wnd_spd	snow	rain	pollution_yesterday	day	month	year
date											
2010-01-02	145.958333	-8.500000	-5.125000	1024.750000	24.860000	0.708333	0.0	10.041667	2	1	2010
2010-01-03	78.833333	-10.125000	-8.541667	1022.791667	70.937917	14.166667	0.0	145.958333	3	1	2010
2010-01-04	31.333333	-20.875000	-11.500000	1029.291667	111.160833	0.000000	0.0	78.833333	4	1	2010
2010-01-05	42.458333	-24.583333	-14.458333	1033.625000	56.920000	0.000000	0.0	31.333333	5	1	2010
2010-01-06	56.416667	-23.708333	-12.541667	1033.750000	18.511667	0.000000	0.0	42.458333	6	1	2010
...	...	...	...	...	...	...	...	...	...	...	...
2014-12-27	238.666667	-9.666667	-1.791667	1027.833333	9.278333	0.000000	0.0	170.250000	27	12	2014
2014-12-28	197.375000	-10.791667	1.583333	1019.958333	10.948750	0.000000	0.0	238.666667	28	12	2014
2014-12-29	159.000000	-12.333333	0.750000	1013.750000	8.000000	0.000000	0.0	197.375000	29	12	2014
2014-12-30	46.083333	-13.916667	1.875000	1019.125000	9.778333	0.000000	0.0	159.000000	30	12	2014
2014-12-31	10.041667	-21.791667	-1.916667	1032.125000	167.458333	0.000000	0.0	46.083333	31	12	2014

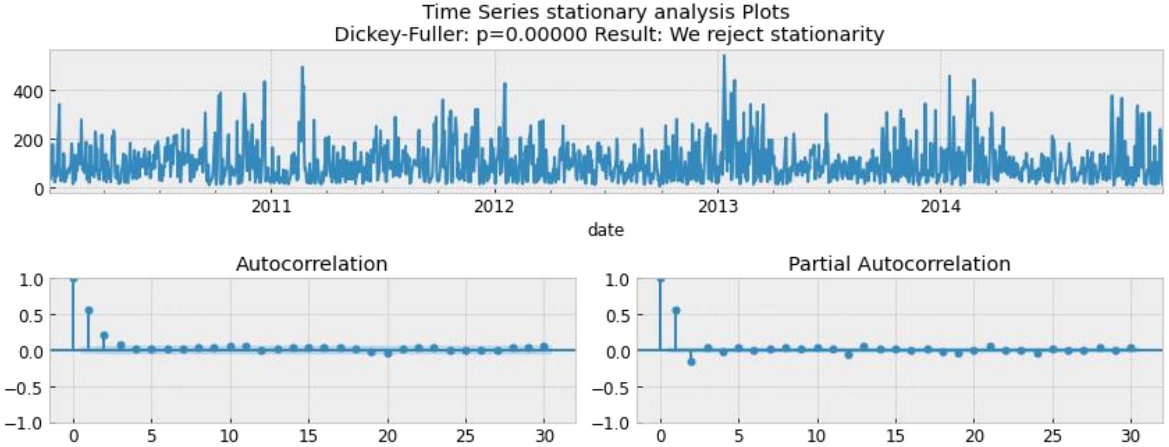


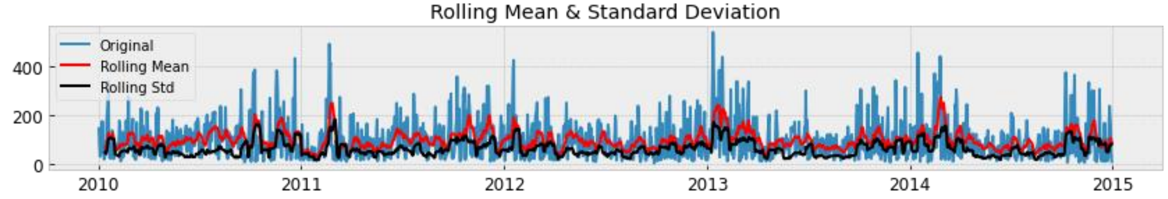
	pollution_today	dew	temp	press	wnd_spd	snow	rain	pollution_yesterday	day	month	year
date											
2010-01-02	145.958	-8.500	-5.125	1024.750	24.860	0.708	0.0	10.042	2	1	2010
2010-01-03	78.833	-10.125	-8.542	1022.792	70.938	14.167	0.0	145.958	3	1	2010
2010-01-04	31.333	-20.875	-11.500	1029.292	111.161	0.000	0.0	78.833	4	1	2010
2010-01-05	42.458	-24.583	-14.458	1033.625	56.920	0.000	0.0	31.333	5	1	2010
2010-01-06	56.417	-23.708	-12.542	1033.750	18.512	0.000	0.0	42.458	6	1	2010
...	...	...	...	...	...	...	...	...	...	...	...
2014-12-27	238.667	-9.667	-1.792	1027.833	9.278	0.000	0.0	170.250	27	12	2014
2014-12-28	197.375	-10.792	1.583	1019.958	10.949	0.000	0.0	238.667	28	12	2014
2014-12-29	159.000	-12.333	0.750	1013.750	8.000	0.000	0.0	197.375	29	12	2014
2014-12-30	46.083	-13.917	1.875	1019.125	9.778	0.000	0.0	159.000	30	12	2014
2014-12-31	10.042	-21.792	-1.917	1032.125	167.458	0.000	0.0	46.083	31	12	2014

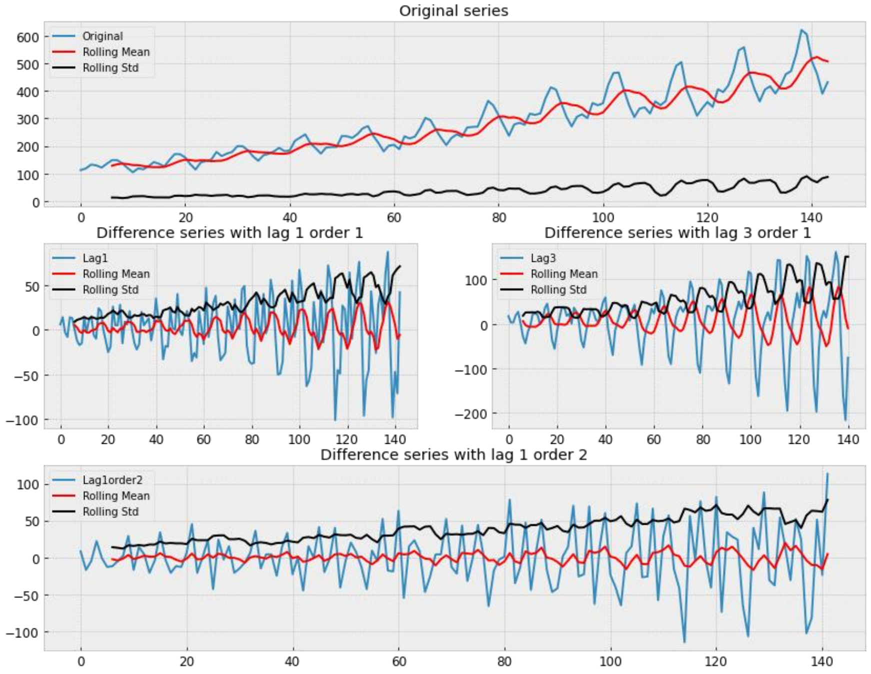
Stationarity in Time Series

1. Check Stationarity
2. Rolling means and standard deviation of our series
3. Augmented Dickey-Fuller test
4. Make any time series a Stationary Time Series
5. Log Scale Transformation
6. Smoothing


Original
Rolling mean
Rolling std


Time Series stationary analysis Plots
Dickey-Fuller: p=0.00000 Result: We reject stationarity
date
Autocorrelation
Partial Autocorrelation


Rolling Mean & Standard Deviation
Original
Rolling Mean
Rolling Std


Original series
Original
Rolling Mean
Rolling Std
Difference series with lag 1 order 1
Lag1
Rolling Mean
Rolling Std
Difference series with lag 3 order 1
Lag3
Rolling Mean
Rolling Std
Difference series with lag 1 order 2
Lag1order2
Rolling Mean
Rolling Std