

```
In [2]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
In [3]: df=pd.read_csv("C:\\Users\\Lenovo\\OneDrive\\Desktop\\Timeseries_lab\\lab
```

Out[3]:

	Date	Beer_Sale
0	01-01-1963	591
1	02-01-1963	464
2	03-01-1963	461
3	04-01-1963	605
4	05-01-1963	586
...	...	...
703	08-01-2021	668
704	09-01-2021	725
705	10-01-2021	649
706	11-01-2021	725
707	12-01-2021	811

708 rows × 2 columns

```
In [4]:
```

Out[4]:

	Date	Beer_Sale
0	01-01-1963	591
1	02-01-1963	464
2	03-01-1963	461
3	04-01-1963	605
4	05-01-1963	586
703	08-01-2021	668
704	09-01-2021	725
705	10-01-2021	649
706	11-01-2021	725
707	12-01-2021	811

```
In [5]: series=pd.read_csv("C:\\Users\\Lenovo\\OneDrive\\Desktop\\Timeseries_lab\\
```

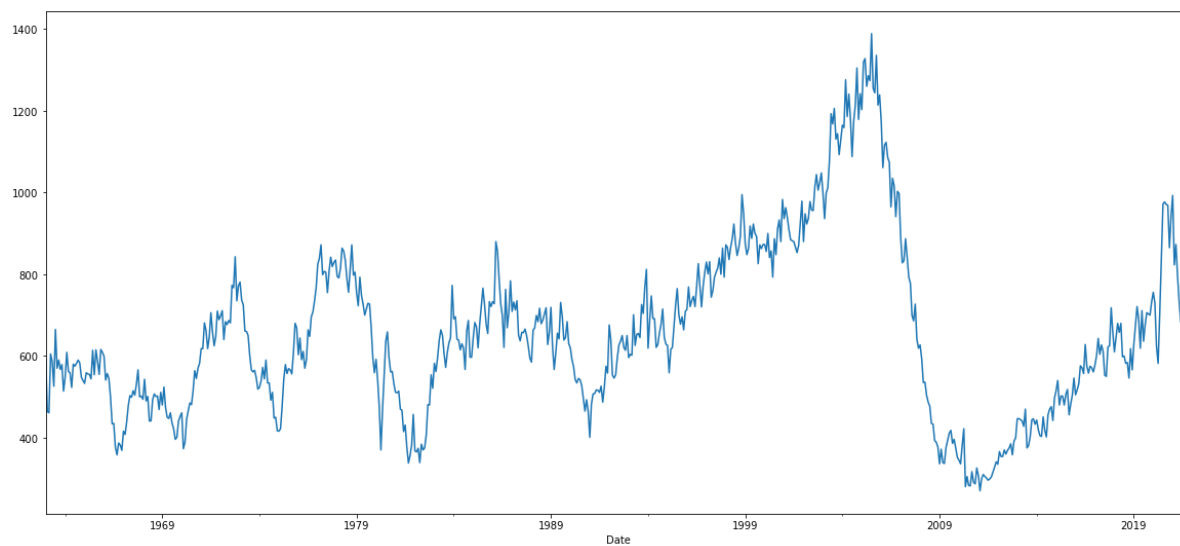
Out[5]:

Beer_Sale
Date

Beer_Sale	
Date	
01-01-1963	591
02-01-1963	464
03-01-1963	461

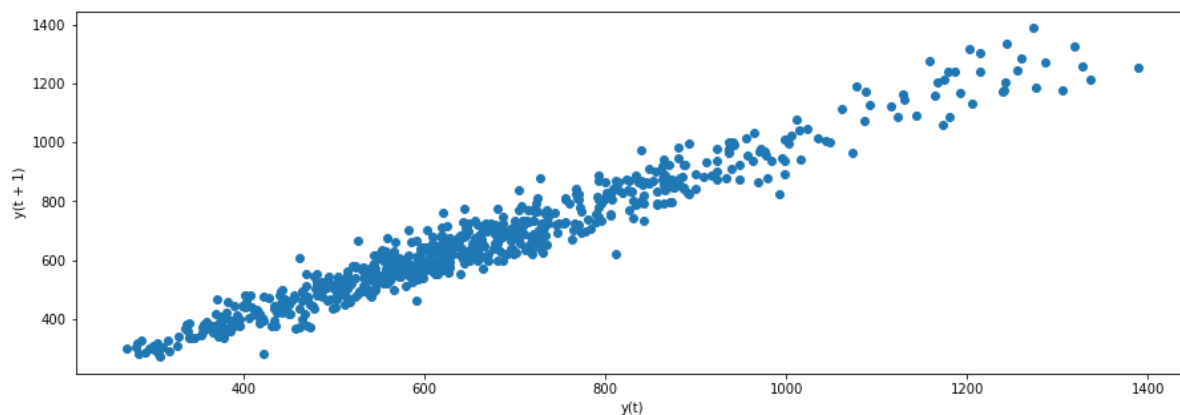
```
In [18]: plt.rcParams["figure.figsize"] = [20,9]
```

```
Out[18]: <AxesSubplot:xlabel='Date'>
```



```
In [20]: from pandas.plotting import lag_plot
plt.rcParams["figure.figsize"] = [15,5]
```

```
Out[20]: <AxesSubplot:xlabel='y(t)', ylabel='y(t + 1)'>
```



```
In [8]: from pandas import read_csv
from pandas import DataFrame
from pandas import concat
```

```
In [9]: series=pd.read_csv("C:\\Users\\Lenovo\\OneDrive\\Desktop\\Timeseries_lab\\
values=DataFrame(series.values)
dataframe=concat([values.shift(1),values],axis=1)
dataframe.columns=['t-1','t+1']
```

```
results=dataframe.corr()
```

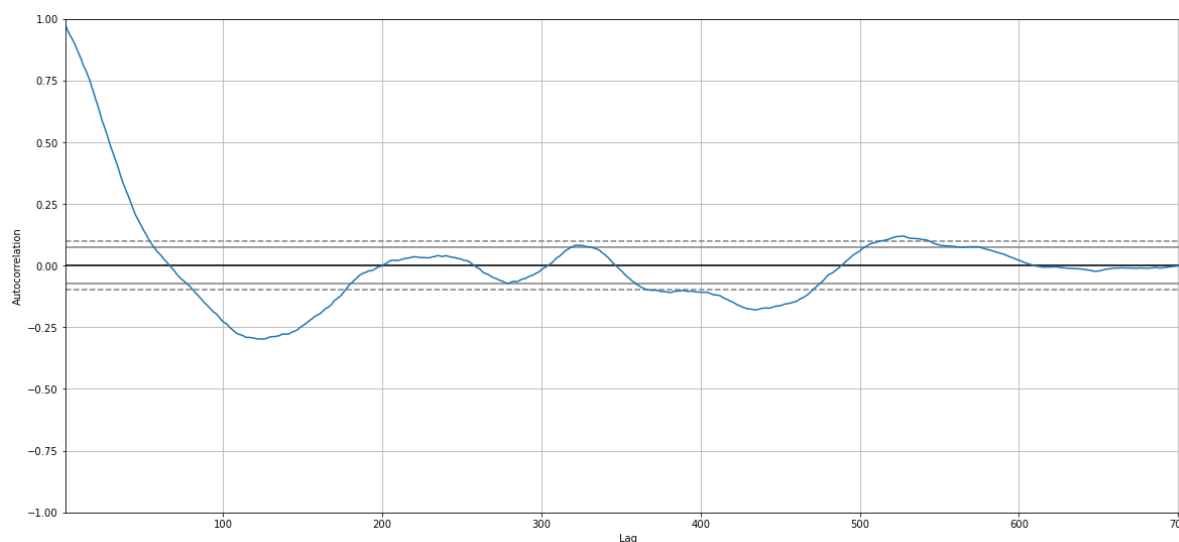
Out[9]:

	t-1	t+1
t-1	1.000000	0.974116
t+1	0.974116	1.000000

```
In [10]: from pandas.plotting import autocorrelation_plot
```

```
In [11]: series=pd.read_csv("C:\\Users\\Lenovo\\OneDrive\\Desktop\\Timeseries_lab\\  
autocorrelation_plot(series)
```

Out[11]: &lt;AxesSubplot:xlabel='Lag', ylabel='Autocorrelation'&gt;



In [ ]:

```
In [12]: ### # create and evaluate a static autoregressive model  
from pandas import read_csv  
from matplotlib import pyplot  
from statsmodels.tsa.ar_model import AutoReg  
from sklearn.metrics import mean_squared_error
```

```
In [13]: series=pd.read_csv("C:\\Users\\Lenovo\\OneDrive\\Desktop\\Timeseries_lab\\  
x=series.values  
x
```

Out[13]:

```
array([ 591,  464,  461,  605,  586,  526,  665,  570,  590,  567,  57
9,
       514,  549,  609,  562,  559,  523,  580,  575,  582,  590,  58
3,
       548,  540,  533,  559,  556,  555,  544,  614,  554,  615,  58
7,
       555,  616,  609,  599,  541,  557,  545,  499,  434,  435,  37
7,
       358,  387,  382,  369,  416,  408,  439,  479,  503,  499,  51
5,
       504,  531,  566,  500,  502,  494,  543,  490,  501,  441,  44
1,
       493,  507,  501,  502,  469,  511,  480,  524,  474,  450,  44
7,
       461,  436,  422,  396,  401,  441,  452,  461,  373,  389,  44
5,
       466,  485,  481,  515,  564,  545,  570,  582,  618,  618,  68
1,
       662,  618,  646,  706,  659,  625,  647,  710,  689,  698,  71
1,
       640,  684,  677,  687,  681,  773,  767,  843,  735,  772,  78
1,
       737,  725,  661,  660,  650,  601,  566,  561,  565,  547,  51
9,
       523,  539,  572,  544,  590,  534,  534,  492,  511,  448,  45
0,
       417,  416,  422,  477,  543,  579,  557,  569,  566,  556,  60
9,
       680,  669,  603,  644,  591,  611,  570,  591,  664,  648,  69
6,
       708,  735,  767,  825,  839,  872,  799,  807,  805,  755,  80
8,
       842,  819,  829,  835,  795,  791,  814,  864,  857,  834,  78
9,
       756,  812,  872,  798,  805,  754,  723,  793,  748,  727,  70
0,
       715,  729,  727,  670,  597,  559,  592,  541,  474,  370,  46
9,
       552,  636,  659,  596,  561,  562,  532,  511,  510,  514,  47
0,
       467,  415,  431,  378,  338,  356,  382,  457,  368,  365,  37
4,
       339,  384,  370,  375,  407,  481,  480,  554,  521,  582,  56
2,
       596,  638,  664,  651,  606,  572,  608,  632,  644,  773,  69
1,
       696,  641,  639,  615,  630,  619,  567,  662,  687,  597,  59
7,
       645,  682,  671,  620,  678,  722,  766,  726,  678,  655,  73
3,
       721,  733,  728,  880,  857,  789,  728,  698,  621,  763,  66
9,
       707,  784,  709,  732,  713,  735,  651,  637,  658,  657,  66
6,
       650,  625,  595,  585,  663,  669,  699,  684,  717,  679,  68
8,
```

1,	703,	718,	628,	658,	719,	622,	567,	608,	656,	642,	73
5,	697,	639,	645,	684,	630,	620,	591,	574,	542,	534,	54
7,	542,	528,	496,	465,	493,	464,	401,	482,	507,	508,	51
6,	516,	511,	526,	487,	524,	575,	558,	676,	639,	553,	54
2,	554,	596,	627,	636,	650,	621,	614,	650,	596,	604,	60
6,	701,	626,	653,	655,	645,	726,	704,	769,	812,	619,	68
6,	747,	692,	691,	621,	628,	656,	677,	715,	646,	629,	62
9,	559,	616,	621,	674,	725,	765,	701,	678,	696,	664,	70
1,	714,	769,	721,	736,	746,	721,	770,	826,	770,	720,	77
0,	805,	830,	801,	831,	744,	760,	793,	805,	815,	840,	80
4,	864,	793,	872,	866,	836,	866,	887,	923,	876,	846,	86
3,	893,	995,	949,	875,	848,	863,	918,	888,	923,	900,	89
7,	826,	872,	863,	873,	873,	856,	900,	841,	857,	793,	88
2,	848,	912,	933,	880,	983,	936,	963,	939,	909,	885,	88
8,	880,	866,	853,	871,	924,	979,	880,	948,	923,	936,	97
2,	957,	956,	1014,	1044,	1006,	1024,	1048,	999,	936,	999,	101
6,	1078,	1193,	1168,	1206,	1131,	1144,	1093,	1129,	1165,	1159,	127
9,	1186,	1241,	1180,	1088,	1175,	1214,	1305,	1179,	1242,	1203,	131
4,	1328,	1260,	1286,	1274,	1389,	1255,	1244,	1336,	1214,	1239,	117
8,	1061,	1116,	1123,	1086,	1074,	965,	1035,	1016,	941,	1003,	99
1,	891,	828,	833,	887,	842,	793,	778,	699,	686,	727,	64
3,	619,	627,	593,	535,	536,	504,	487,	477,	435,	433,	39
6,	389,	377,	336,	372,	339,	337,	376,	393,	411,	418,	38
2,	396,	375,	352,	345,	336,	381,	422,	280,	305,	283,	28
6,	317,	291,	287,	326,	307,	270,	300,	310,	305,	301,	29
0,	299,	304,	316,	328,	341,	335,	366,	354,	354,	370,	36
8,	369,	375,	385,	358,	392,	399,	446,	447,	444,	441,	42
3,	470,	375,	381,	403,	444,	446,	433,	443,	420,	405,	40

```
0,      451,  418,  402,  456,  470,  476,  442,  497,  515,  540,  48
7,      502,  502,  480,  506,  518,  456,  482,  504,  546,  505,  51
      522,  576,  571,  557,  628,  575,  558,  575,  571,  561,  57
```

```
In [14]: train,test=x[1:len(x)-7],x[len(x)-7:]
```

```
In [15]:
```

```
Out[15]:
```

```
array([ 464,  461,  605,  586,  526,  665,  570,  590,  567,  579,  51
4,
        549,  609,  562,  559,  523,  580,  575,  582,  590,  583,  54
8,
        540,  533,  559,  556,  555,  544,  614,  554,  615,  587,  55
5,
        616,  609,  599,  541,  557,  545,  499,  434,  435,  377,  35
8,
        387,  382,  369,  416,  408,  439,  479,  503,  499,  515,  50
4,
        531,  566,  500,  502,  494,  543,  490,  501,  441,  441,  49
3,
        507,  501,  502,  469,  511,  480,  524,  474,  450,  447,  46
1,
        436,  422,  396,  401,  441,  452,  461,  373,  389,  445,  46
6,
        485,  481,  515,  564,  545,  570,  582,  618,  618,  681,  66
2,
        618,  646,  706,  659,  625,  647,  710,  689,  698,  711,  64
0,
        684,  677,  687,  681,  773,  767,  843,  735,  772,  781,  73
7,
        725,  661,  660,  650,  601,  566,  561,  565,  547,  519,  52
3,
        539,  572,  544,  590,  534,  534,  492,  511,  448,  450,  41
7,
        416,  422,  477,  543,  579,  557,  569,  566,  556,  609,  68
0,
        669,  603,  644,  591,  611,  570,  591,  664,  648,  696,  70
8,
        735,  767,  825,  839,  872,  799,  807,  805,  755,  808,  84
2,
        819,  829,  835,  795,  791,  814,  864,  857,  834,  789,  75
6,
        812,  872,  798,  805,  754,  723,  793,  748,  727,  700,  71
5,
        729,  727,  670,  597,  559,  592,  541,  474,  370,  469,  55
2,
        636,  659,  596,  561,  562,  532,  511,  510,  514,  470,  46
7,
        415,  431,  378,  338,  356,  382,  457,  368,  365,  374,  33
9,
        384,  370,  375,  407,  481,  480,  554,  521,  582,  562,  59
6,
        638,  664,  651,  606,  572,  608,  632,  644,  773,  691,  69
6,
        641,  639,  615,  630,  619,  567,  662,  687,  597,  597,  64
5,
        682,  671,  620,  678,  722,  766,  726,  678,  655,  733,  72
1,
        733,  728,  880,  857,  789,  728,  698,  621,  763,  669,  70
7,
        784,  709,  732,  713,  735,  651,  637,  658,  657,  666,  65
0,
        625,  595,  585,  663,  669,  699,  684,  717,  679,  688,  70
3,
```

7,	718,	628,	658,	719,	622,	567,	608,	656,	642,	731,	69
2,	639,	645,	684,	630,	620,	591,	574,	542,	534,	545,	54
6,	528,	496,	465,	493,	464,	401,	482,	507,	508,	517,	51
4,	511,	526,	487,	524,	575,	558,	676,	639,	553,	546,	55
1,	596,	627,	636,	650,	621,	614,	650,	596,	604,	602,	70
7,	626,	653,	655,	645,	726,	704,	769,	812,	619,	686,	74
9,	692,	691,	621,	628,	656,	677,	715,	646,	629,	626,	55
4,	616,	621,	674,	725,	765,	701,	678,	696,	664,	709,	71
5,	769,	721,	736,	746,	721,	770,	826,	770,	720,	771,	80
4,	830,	801,	831,	744,	760,	793,	805,	815,	840,	800,	86
3,	793,	872,	866,	836,	866,	887,	923,	876,	846,	864,	89
6,	995,	949,	875,	848,	863,	918,	888,	923,	900,	893,	82

In [16]:

Out[16]: array([683, 704, 668, 725, 649, 725, 811], dtype=int64)

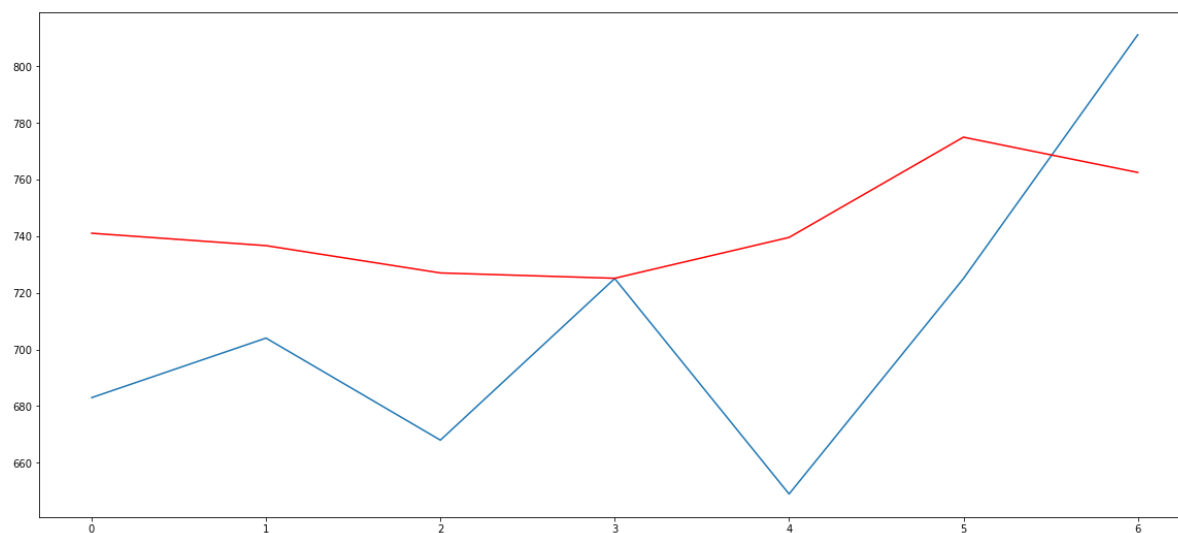
In [17]:

```
# load dataset
series = read_csv("C:\\Users\\Lenovo\\OneDrive\\Desktop\\Timeseries_lab\\
# split dataset
X = series.values
train, test = X[1:len(X)-7], X[len(X)-7:]
# train autoregression
model = AutoReg(train, lags=28)
model_fit = model.fit()
print('Coefficients: %s' % model_fit.params)
# make predictions
predictions = model_fit.predict(start=len(train), end=len(train)+len(test)
for i in range(len(predictions)):
    print('predicted=%f, expected=%f' % (predictions[i], test[i]))
rmse = sqrt(mean_squared_error(test, predictions))
print('Test RMSE: %.3f' % rmse)
# plot results
pyplot.plot(test)
pyplot.plot(predictions, color='red')
plt.show()
```



```
Coefficients: [ 1.86363499e+01  7.84853889e-01  1.31070123e-01  3.2246
5481e-02
-7.10689302e-02  1.60544294e-01 -1.70168438e-02  2.05970745e-02
-6.93842994e-02  3.85177345e-02  4.86869370e-02 -4.62294010e-02
-1.46583767e-01  1.07676179e-01  7.40961528e-02 -3.30393594e-02
 3.26543304e-02  1.11345956e-02 -1.37680457e-04  1.79638019e-02
-1.61303014e-01  1.65014263e-01 -3.72437611e-02 -7.73743407e-02
-1.61858493e-01  1.82472430e-01 -9.63568158e-03  4.85699094e-02
-5.32533808e-02]
```

```
predicted=740.989863, expected=683.000000
```



In [ ]: