finn

September 4, 2018

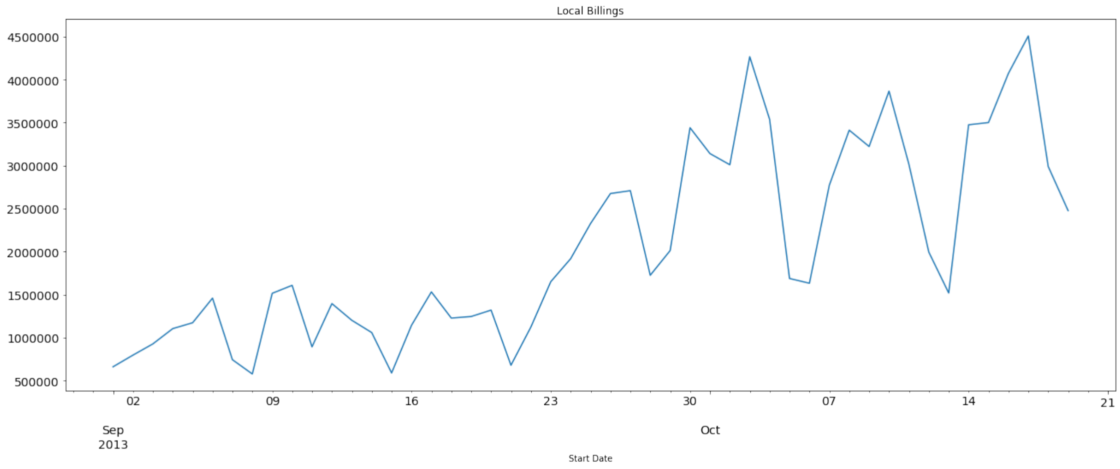
```
In [1]: import pandas as pd
        df_raw = pd.read_excel('Q4_2013_Groupon_North_America_Data_XLSX.xlsx', 'Q4 2013 Raw Da
        df_Local = df_raw[df_raw['Segment'] == 'Local']
        df_Goods = df_raw[df_raw['Segment'] == 'Goods']
        df_Travel = df_raw[df_raw['Segment'] == 'Travel']
```

```
In [2]: %matplotlib inline
        import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import datetime
        from dateutil.relativedelta import relativedelta
        import seaborn as sns
        import statsmodels.api as sm
        from statsmodels.tsa.stattools import acf
        from statsmodels.tsa.stattools import pacf
        from statsmodels.tsa.seasonal import seasonal_decompose
```

```
In [3]: Local_Billings = df_Local.groupby('Start Date')['Start Date', 'Billings'].sum()
```

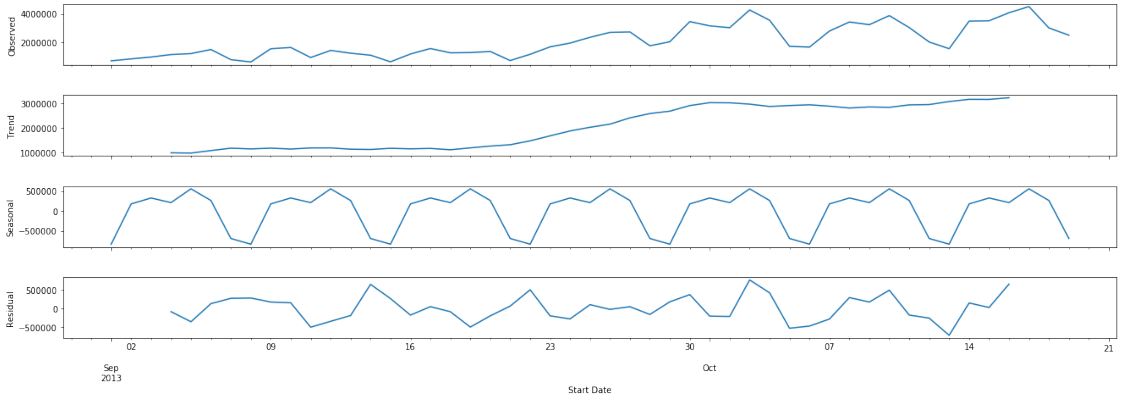
```
In [4]: data = Local_Billings['2013-09-01':'2013-10-19']
```

```
In [5]: data.Billings.plot(figsize=(22,8), title= 'Local Billings', fontsize=14)
        plt.savefig('Local_Billings.png', bbox_inches='tight')
```



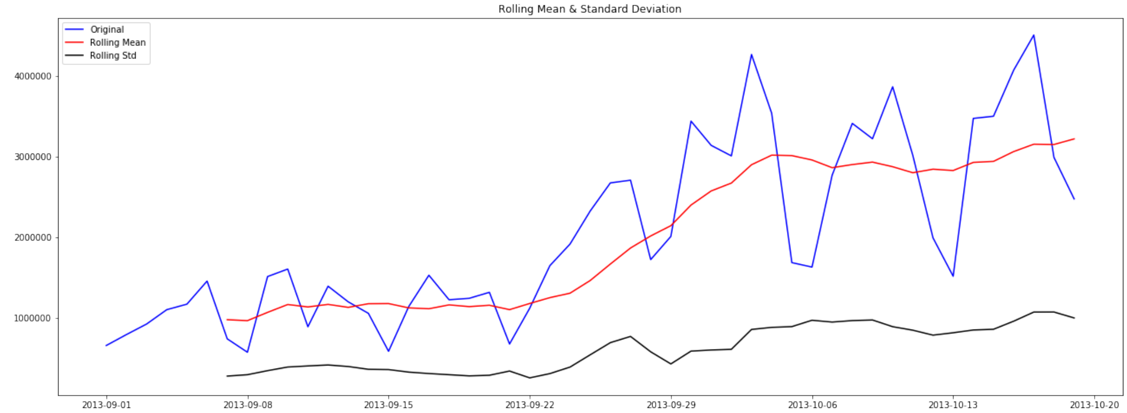
```
In [6]: decomposition = seasonal_decompose(data.Billings, freq=7)
        fig = plt.figure()
        fig = decomposition.plot()
        fig.set_size_inches(22, 8)
```

```
<Figure size 432x288 with 0 Axes>
```



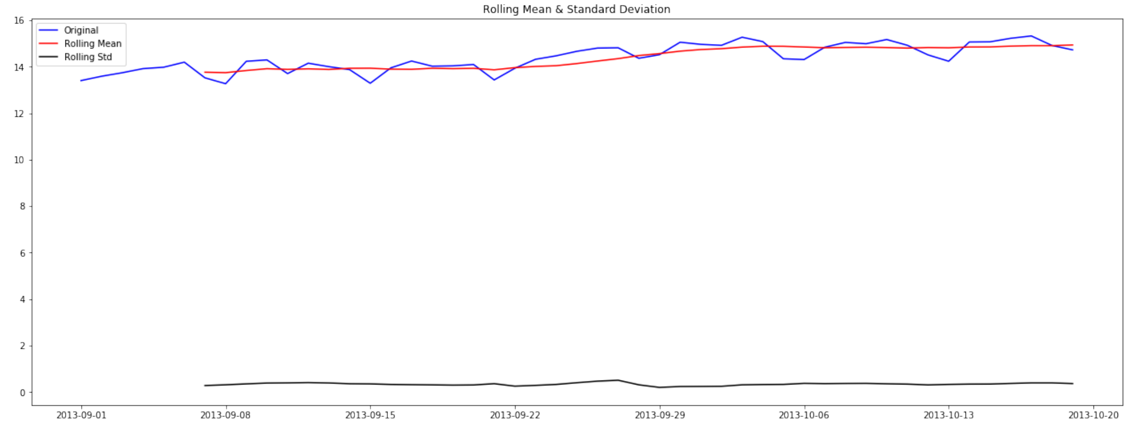
```
In [7]: from statsmodels.tsa.stattools import adfuller
        def test_stationarity(timeseries):

            #Determing rolling statistics
            rolmean = timeseries.rolling(window=7).mean()
            rolstd = timeseries.rolling(window=7).std()

            #Plot rolling statistics:
            fig = plt.figure(figsize=(22, 8))
            orig = plt.plot(timeseries, color='blue',label='Original')
            mean = plt.plot(rolmean, color='red', label='Rolling Mean')
            std = plt.plot(rolstd, color='black', label = 'Rolling Std')
            plt.legend(loc='best')
            plt.title('Rolling Mean & Standard Deviation')
            plt.show()

            #Perform Dickey-Fuller test:
            print ('Results of Dickey-Fuller Test:')
            dftest = adfuller(timeseries, autolag='AIC')
            dfoutput = pd.Series(dftest[0:4], index=['Test Statistic','p-value','#Lags Used','
            for key,value in dftest[4].items():
                dfoutput['Critical Value (%s)'%key] = value
            print (dfoutput)
```

```
In [8]: test_stationarity(data.Billings)
```



```
Results of Dickey-Fuller Test:
Test Statistic                 -0.493765
p-value                         0.893227
#Lags Used                      8.000000
Number of Observations Used    40.000000
Critical Value (1%)            -3.605565
Critical Value (5%)            -2.937069
Critical Value (10%)           -2.606986
dtype: float64
```

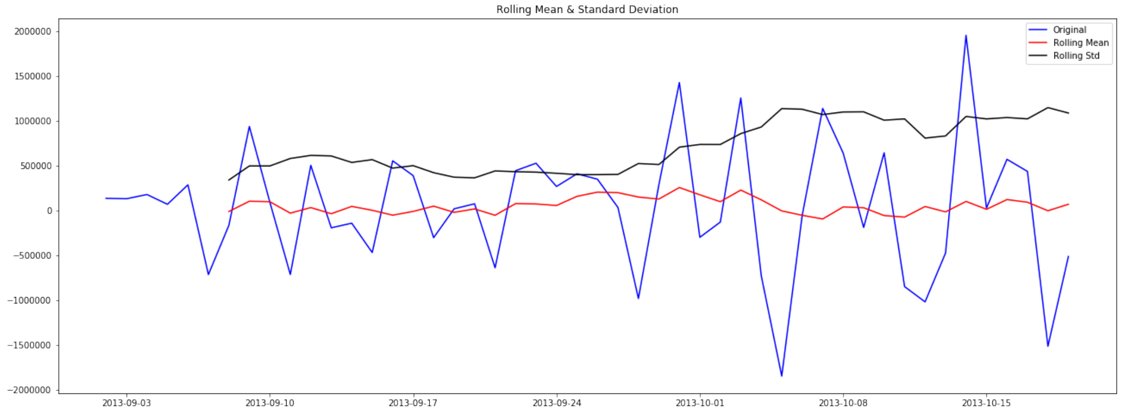
```
In [9]: data.Billings_log = data.Billings.apply(lambda x: np.log(x))
        test_stationarity(data.Billings_log)

/Users/vicky/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:1: UserWarning: Panda
  """Entry point for launching an IPython kernel.
```



```
Results of Dickey-Fuller Test:
Test Statistic                 -0.676594
p-value                         0.852720
#Lags Used                      8.000000
Number of Observations Used    40.000000
Critical Value (1%)            -3.605565
Critical Value (5%)            -2.937069
Critical Value (10%)           -2.606986
dtype: float64
```

```
In [10]: data['first_difference'] = data.Billings - data.Billings.shift(1)
         test_stationarity(data.first_difference.dropna(inplace=False))
```

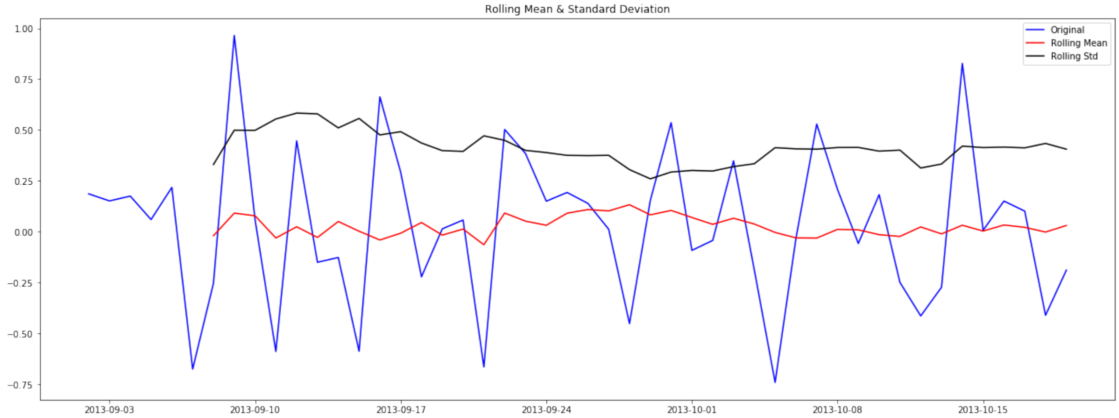
```
/Users/vicky/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:1: SettingWithCopyWarn
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html
  """Entry point for launching an IPython kernel.
```



```
Results of Dickey-Fuller Test:
Test Statistic                 -1.771613
p-value                         0.394643
#Lags Used                      7.000000
Number of Observations Used    40.000000
Critical Value (1%)            -3.605565
Critical Value (5%)            -2.937069
Critical Value (10%)           -2.606986
dtype: float64
```

```
In [11]: data['log_first_difference'] = data.Billings_log - data.Billings_log.shift(1)
         test_stationarity(data.log_first_difference.dropna(inplace=False))
```

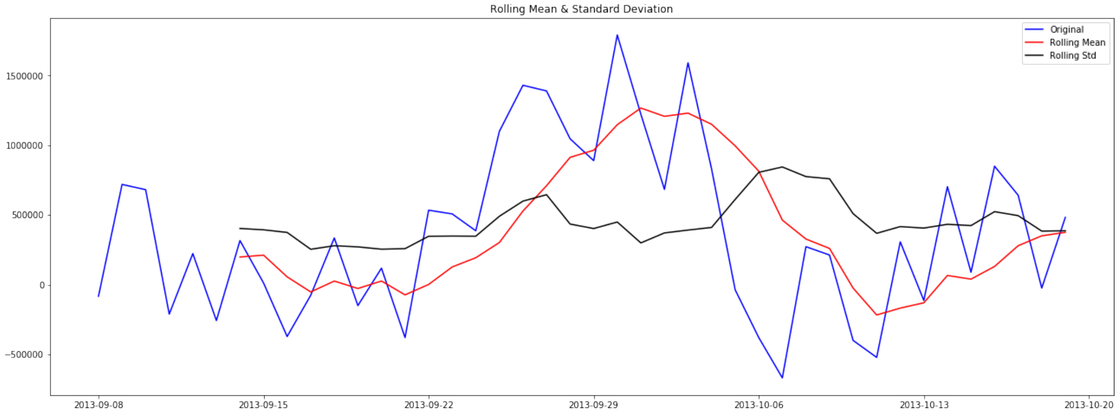
```
/Users/vicky/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:1: SettingWithCopyWar
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.htm
  """Entry point for launching an IPython kernel.
```



```
Results of Dickey-Fuller Test:
Test Statistic                  -1.831369
p-value                          0.365003
#Lags Used                       7.000000
Number of Observations Used     40.000000
Critical Value (1%)             -3.605565
Critical Value (5%)             -2.937069
Critical Value (10%)            -2.606986
dtype: float64
```

```
In [12]: data['seasonal_difference'] = data.Billings - data.Billings.shift(7)
         test_stationarity(data.seasonal_difference.dropna(inplace=False))
```

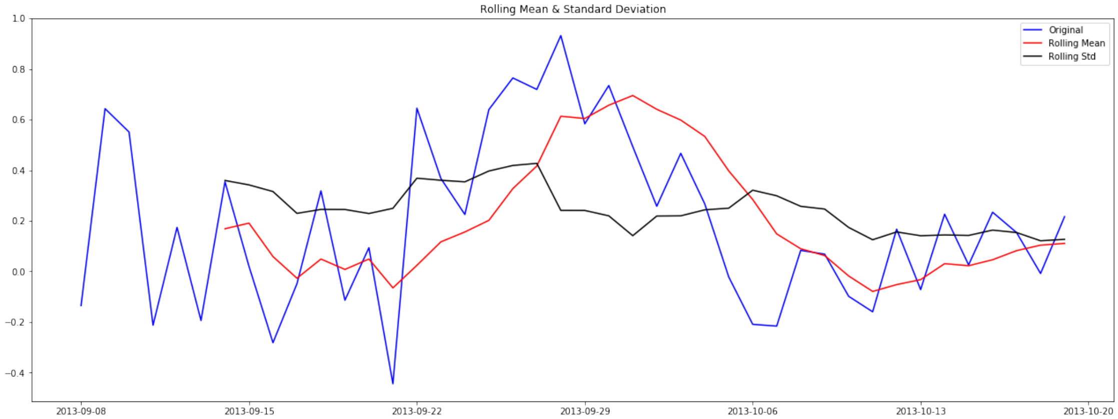
```
/Users/vicky/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:1: SettingWithCopyWar
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.htm
  """Entry point for launching an IPython kernel.
```



```
Results of Dickey-Fuller Test:
Test Statistic                 -3.247368
p-value                         0.017401
#Lags Used                      8.000000
Number of Observations Used    33.000000
Critical Value (1%)            -3.646135
Critical Value (5%)            -2.954127
Critical Value (10%)           -2.615968
dtype: float64
```

```
In [13]: data['log_seasonal_difference'] = data.Billings_log - data.Billings_log.shift(7)
         test_stationarity(data.log_seasonal_difference.dropna(inplace=False))
```

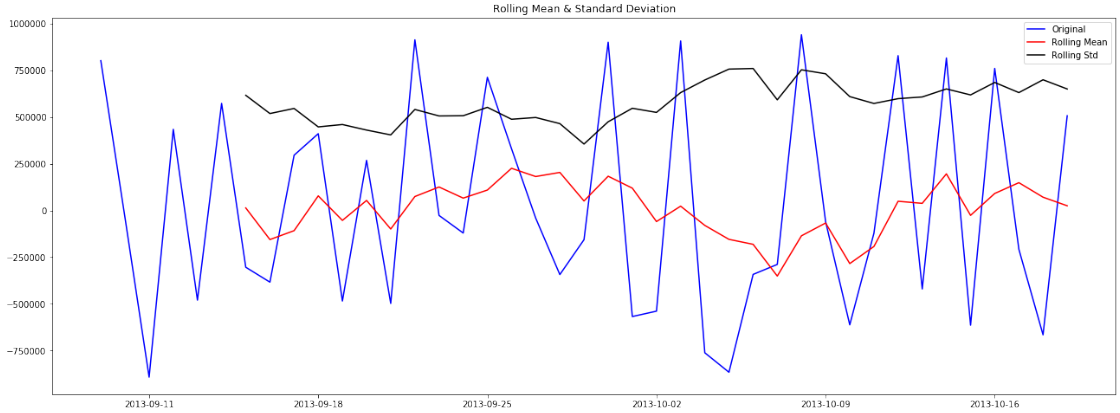
```
/Users/vicky/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:1: SettingWithCopyWarn
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html
  """Entry point for launching an IPython kernel.
```



```
Results of Dickey-Fuller Test:
Test Statistic                 -2.713955
p-value                         0.071654
#Lags Used                      8.000000
Number of Observations Used    33.000000
Critical Value (1%)            -3.646135
Critical Value (5%)            -2.954127
Critical Value (10%)           -2.615968
dtype: float64
```

```
In [14]: data['seasonal_first_difference'] = data.first_difference - data.first_difference.shi
         test_stationarity(data.seasonal_first_difference.dropna(inplace=False))
```

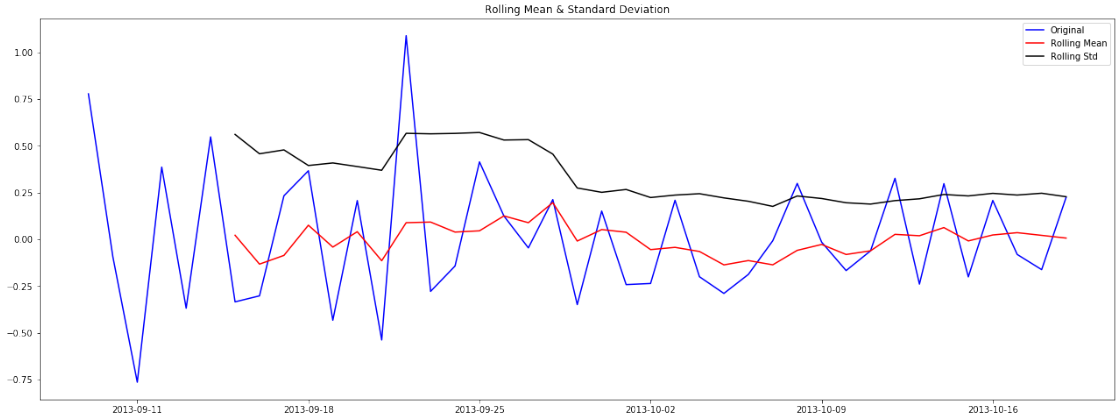
```
/Users/vicky/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:1: SettingWithCopyWarn
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html
  """Entry point for launching an IPython kernel.
```



```
Results of Dickey-Fuller Test:
Test Statistic                -8.999948e+00
p-value                        6.538471e-15
#Lags Used                     0.000000e+00
Number of Observations Used    4.000000e+01
Critical Value (1%)           -3.605565e+00
Critical Value (5%)           -2.937069e+00
Critical Value (10%)          -2.606986e+00
dtype: float64
```

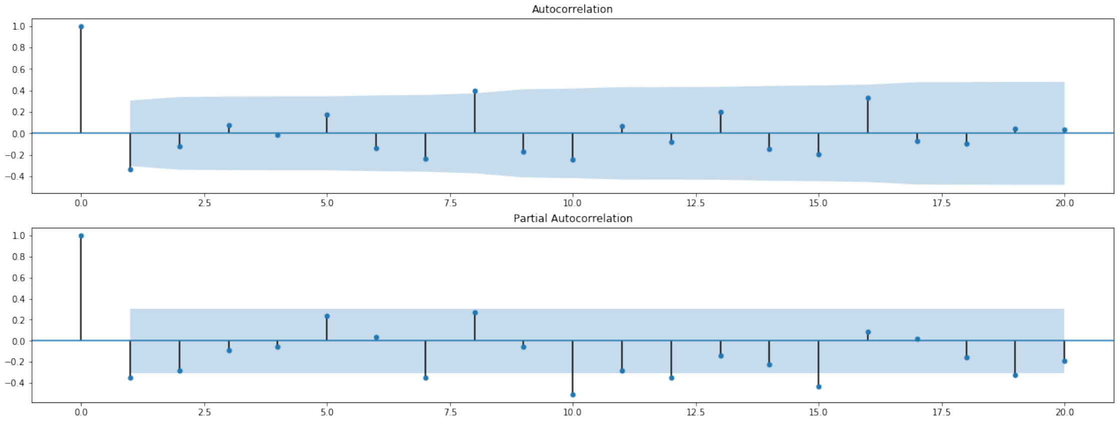
```
In [15]: data['log_seasonal_first_difference'] = data.log_first_difference - data.log_first_di
         test_stationarity(data.log_seasonal_first_difference.dropna(inplace=False))
```

```
/Users/vicky/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:1: SettingWithCopyWarn
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html
  """Entry point for launching an IPython kernel.
```



```
Results of Dickey-Fuller Test:
Test Statistic                 -2.755552
p-value                         0.064909
#Lags Used                      9.000000
Number of Observations Used    31.000000
Critical Value (1%)            -3.661429
Critical Value (5%)            -2.960525
Critical Value (10%)           -2.619319
dtype: float64
```

```
In [16]: fig = plt.figure(figsize=(22,8))
         ax1 = fig.add_subplot(211)
         fig = sm.graphics.tsa.plot_acf(data.seasonal_first_difference.iloc[8:], lags=20, ax=a
         ax2 = fig.add_subplot(212)
         fig = sm.graphics.tsa.plot_pacf(data.seasonal_first_difference.iloc[8:], lags=20, ax=a
```



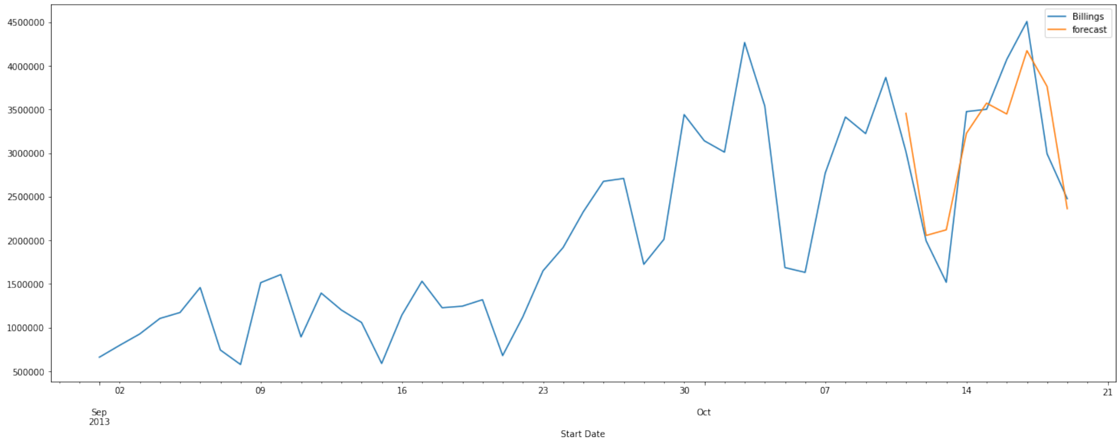
```
In [17]: mod = sm.tsa.statespace.SARIMAX(data.Billings, trend='n', order=(0,1,0), seasonal_orde
         results = mod.fit()
         print (results.summary())
```

```
                                 Statespace Model Results
=========================================================================================
Dep. Variable:                          Billings   No. Observations:                   49
Model:             SARIMAX(0, 1, 0)x(0, 1, 1, 7)   Log Likelihood                -599.473
Date:                           Tue, 04 Sep 2018   AIC                           1202.946
Time:                                   21:39:39   BIC                           1206.373
Sample:                               09-01-2013   HQIC                          1204.194
                                    - 10-19-2013
Covariance Type:                             opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ma.S.L7       -0.4188      0.265     -1.581      0.114      -0.938       0.101
sigma2      3.789e+11    1.9e-13      2e+24      0.000    3.79e+11    3.79e+11
===================================================================================
Ljung-Box (Q):                       69.66   Jarque-Bera (JB):                 1.29
Prob(Q):                              0.00   Prob(JB):                         0.52
Heteroskedasticity (H):               1.32   Skew:                             0.11
Prob(H) (two-sided):                  0.62   Kurtosis:                         2.16
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
[2] Covariance matrix is singular or near-singular, with condition number 2.76e+40. Standard e


/Users/vicky/anaconda3/lib/python3.6/site-packages/statsmodels/tsa/base/tsa_model.py:171: Valu
  % freq, ValueWarning)
```

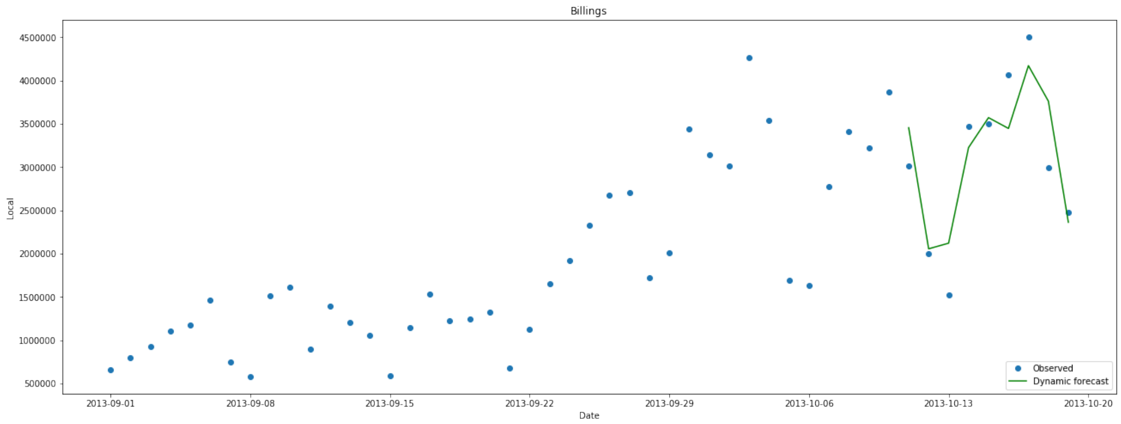
```
In [18]: data['forecast'] = results.predict(start = 40, end= 49, dynamic= True)
         data[['Billings', 'forecast']].plot(figsize=(22, 8))
```

```
/Users/vicky/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:1: SettingWithCopyWar
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.htm
  """Entry point for launching an IPython kernel.
```

```
Out[18]: <matplotlib.axes._subplots.AxesSubplot at 0x10e279a58>
```



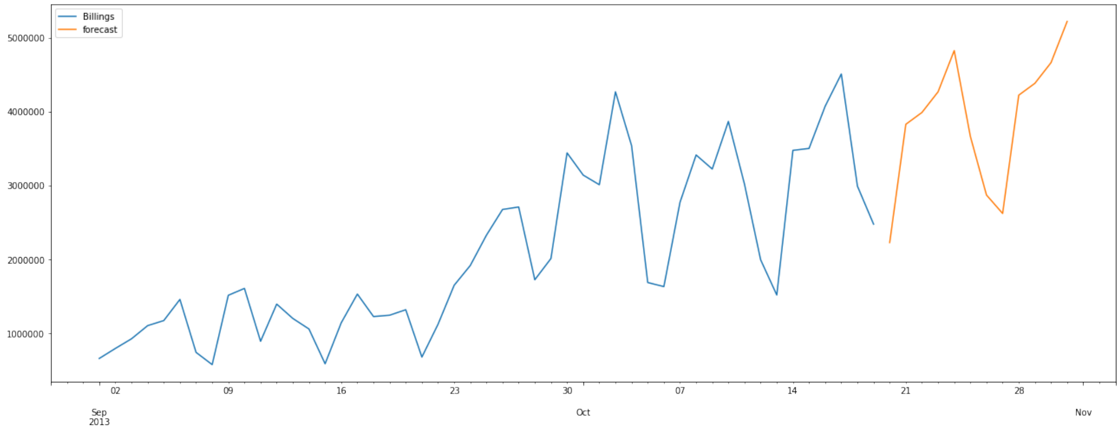
```
In [19]: npredict = data.Billings['2013-09-01':].shape[0]
         fig, ax = plt.subplots(figsize=(22,8))
         npre = 7
         ax.set(title='Billings', xlabel='Date', ylabel='Local')
         ax.plot(data.index[-npredict-npre+1:], data.iloc[-npredict-npre+1:]['Billings'], 'o',
         ax.plot(data.index[-npredict-npre+1:], data.iloc[-npredict-npre+1:]['forecast'], 'g',
         legend = ax.legend(loc='lower right')
         legend.get_frame().set_facecolor('w')
         #plt.savefig('ts_predict_compare.png', bbox_inches='tight')
```



```
In [20]: start  =  datetime.datetime.strptime("2013-10-20", "%Y-%m-%d")
         date_list = [start + relativedelta(days=x) for x in range(0,20)]
         future = pd.DataFrame(index=date_list, columns= data.columns)
         data = pd.concat([data, future])
```

```
In [21]: data['forecast'] = results.predict(start = 49, end = 60, dynamic= True)
         data[['Billings', 'forecast']].plot(figsize=(22, 8))
         #plt.savefig('ts_predict_future.png', bbox_inches='tight')
```

```
Out[21]: <matplotlib.axes._subplots.AxesSubplot at 0x10f722a20>
```



```
In [22]: data
```

```
Out[22]:                Billings  first_difference  log_first_difference  \
         2013-09-01  6.607904e+05               NaN                   NaN
         2013-09-02  7.963350e+05      1.355445e+05              0.186583
         2013-09-03  9.268435e+05      1.305085e+05              0.151765
```

```
2013-09-04  1.104587e+06      1.777438e+05              0.175442
2013-09-05  1.173147e+06      6.855923e+04              0.060218
2013-09-06  1.459212e+06      2.860653e+05              0.218207
2013-09-07  7.433991e+05     -7.158128e+05             -0.674419
2013-09-08  5.773694e+05     -1.660297e+05             -0.252751
2013-09-09  1.515168e+06      9.377988e+05              0.964799
2013-09-10  1.607966e+06      9.279752e+04              0.059443
2013-09-11  8.929169e+05     -7.150488e+05             -0.588232
2013-09-12  1.395849e+06      5.029323e+05              0.446765
2013-09-13  1.201757e+06     -1.940918e+05             -0.149718
2013-09-14  1.059236e+06     -1.425211e+05             -0.126237
2013-09-15  5.890799e+05     -4.701564e+05             -0.586742
2013-09-16  1.143040e+06      5.539596e+05              0.662884
2013-09-17  1.531653e+06      3.886131e+05              0.292656
2013-09-18  1.227678e+06     -3.039750e+05             -0.221223
2013-09-19  1.245592e+06      1.791426e+04              0.014487
2013-09-20  1.319770e+06      7.417796e+04              0.057847
2013-09-21  6.793485e+05     -6.404214e+05             -0.664078
2013-09-22  1.123031e+06      4.436824e+05              0.502652
2013-09-23  1.650251e+06      5.272196e+05              0.384896
2013-09-24  1.918048e+06      2.677974e+05              0.150381
2013-09-25  2.327159e+06      4.091115e+05              0.193340
2013-09-26  2.675751e+06      3.485919e+05              0.139582
2013-09-27  2.709123e+06      3.337194e+04              0.012395
2013-09-28  1.725309e+06     -9.838143e+05             -0.451219
2013-09-29  2.012469e+06      2.871604e+05              0.153956
2013-09-30  3.440883e+06      1.428414e+06              0.536366
...                  ...               ...                   ...
2013-10-10  3.866735e+06      6.439296e+05              0.182158
2013-10-11  3.015991e+06     -8.507440e+05             -0.248482
2013-10-12  1.994842e+06     -1.021149e+06             -0.413364
2013-10-13  1.519141e+06     -4.757016e+05             -0.272420
2013-10-14  3.475041e+06      1.955900e+06              0.827461
2013-10-15  3.501416e+06      2.637561e+04              0.007561
2013-10-16  4.072212e+06      5.707955e+05              0.151019
2013-10-17  4.507996e+06      4.357840e+05              0.101666
2013-10-18  2.991175e+06     -1.516821e+06             -0.410186
2013-10-19  2.477089e+06     -5.140854e+05             -0.188582
2013-10-20           NaN               NaN                   NaN
2013-10-21           NaN               NaN                   NaN
2013-10-22           NaN               NaN                   NaN
2013-10-23           NaN               NaN                   NaN
2013-10-24           NaN               NaN                   NaN
2013-10-25           NaN               NaN                   NaN
2013-10-26           NaN               NaN                   NaN
2013-10-27           NaN               NaN                   NaN
2013-10-28           NaN               NaN                   NaN
2013-10-29           NaN               NaN                   NaN
```

```
2013-10-30           NaN               NaN                   NaN
2013-10-31           NaN               NaN                   NaN
2013-11-01           NaN               NaN                   NaN
2013-11-02           NaN               NaN                   NaN
2013-11-03           NaN               NaN                   NaN
2013-11-04           NaN               NaN                   NaN
2013-11-05           NaN               NaN                   NaN
2013-11-06           NaN               NaN                   NaN
2013-11-07           NaN               NaN                   NaN
2013-11-08           NaN               NaN                   NaN

            seasonal_difference  log_seasonal_difference  \
2013-09-01                  NaN                      NaN
2013-09-02                  NaN                      NaN
2013-09-03                  NaN                      NaN
2013-09-04                  NaN                      NaN
2013-09-05                  NaN                      NaN
2013-09-06                  NaN                      NaN
2013-09-07                  NaN                      NaN
2013-09-08        -8.342101e+04                -0.134954
2013-09-09         7.188332e+05                 0.643262
2013-09-10         6.811222e+05                 0.550940
2013-09-11        -2.116704e+05                -0.212734
2013-09-12         2.227027e+05                 0.173813
2013-09-13        -2.574545e+05                -0.194111
2013-09-14         3.158372e+05                 0.354070
2013-09-15         1.171051e+04                 0.020080
2013-09-16        -3.721286e+05                -0.281835
2013-09-17        -7.631301e+04                -0.048622
2013-09-18         3.347608e+05                 0.318386
2013-09-19        -1.502573e+05                -0.113892
2013-09-20         1.180125e+05                 0.093672
2013-09-21        -3.798877e+05                -0.444169
2013-09-22         5.339511e+05                 0.645225
2013-09-23         5.072110e+05                 0.367236
2013-09-24         3.863953e+05                 0.224961
2013-09-25         1.099482e+06                 0.639524
2013-09-26         1.430159e+06                 0.764619
2013-09-27         1.389353e+06                 0.719168
2013-09-28         1.045960e+06                 0.932027
2013-09-29         8.894385e+05                 0.583331
2013-09-30         1.790632e+06                 0.734801
...                         ...                      ...
2013-10-10        -4.001325e+05                -0.098469
2013-10-11        -5.220921e+05                -0.159657
2013-10-12         3.069616e+05                 0.167091
2013-10-13        -1.134224e+05                -0.072006
2013-10-14         7.030876e+05                 0.226054
```

```
2013-10-15         8.868560e+04                 0.025655
2013-10-16         8.494065e+05                 0.233934
2013-10-17         6.412609e+05                 0.153442
2013-10-18        -2.481629e+04                -0.008262
2013-10-19         4.822471e+05                 0.216519
2013-10-20                  NaN                      NaN
2013-10-21                  NaN                      NaN
2013-10-22                  NaN                      NaN
2013-10-23                  NaN                      NaN
2013-10-24                  NaN                      NaN
2013-10-25                  NaN                      NaN
2013-10-26                  NaN                      NaN
2013-10-27                  NaN                      NaN
2013-10-28                  NaN                      NaN
2013-10-29                  NaN                      NaN
2013-10-30                  NaN                      NaN
2013-10-31                  NaN                      NaN
2013-11-01                  NaN                      NaN
2013-11-02                  NaN                      NaN
2013-11-03                  NaN                      NaN
2013-11-04                  NaN                      NaN
2013-11-05                  NaN                      NaN
2013-11-06                  NaN                      NaN
2013-11-07                  NaN                      NaN
2013-11-08                  NaN                      NaN

            seasonal_first_difference  log_seasonal_first_difference  \
2013-09-01                        NaN                            NaN
2013-09-02                        NaN                            NaN
2013-09-03                        NaN                            NaN
2013-09-04                        NaN                            NaN
2013-09-05                        NaN                            NaN
2013-09-06                        NaN                            NaN
2013-09-07                        NaN                            NaN
2013-09-08                        NaN                            NaN
2013-09-09                802254.2235                       0.778216
2013-09-10                -37711.0285                      -0.092321
2013-09-11               -892792.6300                      -0.763674
2013-09-12                434373.1045                       0.386547
2013-09-13               -480157.1195                      -0.367925
2013-09-14                573291.6745                       0.548182
2013-09-15               -304126.7055                      -0.333991
2013-09-16               -383839.1325                      -0.301915
2013-09-17                295815.6160                       0.233213
2013-09-18                411073.8190                       0.367009
2013-09-19               -485018.0770                      -0.432278
2013-09-20                268269.7655                       0.207565
2013-09-21               -497900.2370                      -0.537842
```

```
2013-09-22                913838.8050                       1.089394
2013-09-23                -26740.0170                      -0.277989
2013-09-24               -120815.7665                      -0.142276
2013-09-25                713086.4460                       0.414563
2013-09-26                330677.6475                       0.125095
2013-09-27                -40806.0180                      -0.045452
2013-09-28               -343392.8820                       0.212860
2013-09-29               -156521.9950                      -0.348696
2013-09-30                901194.0185                       0.151470
...                               ...                            ...
2013-10-10               -612539.6850                      -0.166649
2013-10-11               -121959.6600                      -0.061187
2013-10-12                829053.7765                       0.326748
2013-10-13               -420384.0220                      -0.239098
2013-10-14                816509.9795                       0.298060
2013-10-15               -614401.9965                      -0.200399
2013-10-16                760720.9340                       0.208279
2013-10-17               -208145.6090                      -0.080492
2013-10-18               -666077.2180                      -0.161704
2013-10-19                507063.3720                       0.224782
2013-10-20                        NaN                            NaN
2013-10-21                        NaN                            NaN
2013-10-22                        NaN                            NaN
2013-10-23                        NaN                            NaN
2013-10-24                        NaN                            NaN
2013-10-25                        NaN                            NaN
2013-10-26                        NaN                            NaN
2013-10-27                        NaN                            NaN
2013-10-28                        NaN                            NaN
2013-10-29                        NaN                            NaN
2013-10-30                        NaN                            NaN
2013-10-31                        NaN                            NaN
2013-11-01                        NaN                            NaN
2013-11-02                        NaN                            NaN
2013-11-03                        NaN                            NaN
2013-11-04                        NaN                            NaN
2013-11-05                        NaN                            NaN
2013-11-06                        NaN                            NaN
2013-11-07                        NaN                            NaN
2013-11-08                        NaN                            NaN

                forecast
2013-09-01           NaN
2013-09-02           NaN
2013-09-03           NaN
2013-09-04           NaN
2013-09-05           NaN
2013-09-06           NaN
```

```
2013-09-07           NaN
2013-09-08           NaN
2013-09-09           NaN
2013-09-10           NaN
2013-09-11           NaN
2013-09-12           NaN
2013-09-13           NaN
2013-09-14           NaN
2013-09-15           NaN
2013-09-16           NaN
2013-09-17           NaN
2013-09-18           NaN
2013-09-19           NaN
2013-09-20           NaN
2013-09-21           NaN
2013-09-22           NaN
2013-09-23           NaN
2013-09-24           NaN
2013-09-25           NaN
2013-09-26           NaN
2013-09-27           NaN
2013-09-28           NaN
2013-09-29           NaN
2013-09-30           NaN
...                  ...
2013-10-10           NaN
2013-10-11           NaN
2013-10-12           NaN
2013-10-13           NaN
2013-10-14           NaN
2013-10-15           NaN
2013-10-16           NaN
2013-10-17           NaN
2013-10-18           NaN
2013-10-19           NaN
2013-10-20  2.227903e+06
2013-10-21  3.827660e+06
2013-10-22  3.987888e+06
2013-10-23  4.267343e+06
2013-10-24  4.824704e+06
2013-10-25  3.663981e+06
2013-10-26  2.871038e+06
2013-10-27  2.621852e+06
2013-10-28  4.221609e+06
2013-10-29  4.381838e+06
2013-10-30  4.661292e+06
2013-10-31  5.218653e+06
2013-11-01           NaN
```

```
2013-11-02           NaN
2013-11-03           NaN
2013-11-04           NaN
2013-11-05           NaN
2013-11-06           NaN
2013-11-07           NaN
2013-11-08           NaN

[69 rows x 8 columns]
```

```
In [23]: data['forecast']['2013-10-20':'2013-10-30'].sum()
```

```
Out[23]: 41557108.592442185
```