sample

March 14, 2024

1 Loading Libraries

[1]:
```
"""
Converted on Fri Jul 27 10:37:32 2018

@author: Atul
"""
import numpy as np
import pandas as pd
import sklearn
from sklearn import linear_model
from sklearn import metrics
from sklearn import preprocessing
from load_data import *
import rpy2
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
plt.style.use('ggplot')
import statsmodels.api as sm
```

2 Loading Training and Test Data Sets from csv to dataFrame

[2]:
```
train_data=pd.read_csv("./A1Benchmark/real_23.csv") #training data set1
train_data1=pd.read_csv("./A1Benchmark/real_24.csv")  #training data set2

test_data1=pd.read_csv("./A1Benchmark/real_18.csv") #Test data set1
```

3 Selecting required columns

[3]:
```
x_train=train_data.iloc[:,(1)].values # selecting data values for training set1
y_train=train_data.iloc[:,2].values # selecting target class for training set1
```

```
x_train1=train_data1.iloc[:,(1)].values # selecting data values for training␣
 ↪set2
y_train1=train_data1.iloc[:,2].values #selecting target class for training set2

x_test1=test_data1.iloc[:,(1)].values # selecting data values for test set1
y_test1=test_data1.iloc[:,2].values #selecting target class for test set1
```

4 Reshaping selected dataframes

[4]:
```
x_train=x_train.reshape(-1,1) #reshaping training set 1 because it contains␣
 ↪onyl single feature
x_train1=x_train1.reshape(-1,1)# reshaping training set 2
x_test1=x_test1.reshape(-1,1) #reshaping test set 1
```

5 Normalizing the Test and Training data between 0 and 1 // If necessary ////No need now

[6]:
```
#min_max_scaler = preprocessing.MinMaxScaler()
#x_train1 = min_max_scaler.fit_transform(x_train1) # normalizing training set2
#x_test1 = min_max_scaler.fit_transform(x_test1) #normalizing test set 1
```

6 Creating model for linear regression

[5]:
```
clf=linear_model.LinearRegression()
clf.fit(x_train,y_train) #training model over training set1
```

```
/anaconda3/lib/python3.6/site-packages/sklearn/linear_model/base.py:509:
RuntimeWarning: internal gelsd driver lwork query error, required iwork
dimension not returned. This is likely the result of LAPACK bug 0038, fixed in
LAPACK 3.2.2 (released July 21, 2010). Falling back to 'gelss' driver.
  linalg.lstsq(X, y)
```

[5]:
```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=1, normalize=False)
```

7 Re-Training the generated Model

[6]:
```
import pickle
s = pickle.dumps(clf) #dumping the trained model over set1 into s
clf2 = pickle.loads(s) #This model will contain trained data and wil continue␣
 ↪the same.
clf2.fit(x_train1,y_train1) #Training the Previously generated model over set2
```

[6]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=1, normalize=False)

[7]:
```
y_expect=y_test1
y_pred=clf2.predict(x_test1) #calculating predicted value
```

8 Defining a Function for Rounding off the Predicted Values to 0 or 1

[8]:
```
def r(i):
    if (i > 1.309 and i < 1.5): # Limits set for i after analyzing the␣
 ↪Predicted values(y_pred) // for csv 18 and 49
        i =1
    elif (i < -0.19 or (i > 0.30 and i < 1)): # for csv 53
        i=1
    else:
         i=0
    return i
```

[9]:
```
y_pred_new=np.array([]) # initiating an empty numpy array for storing round off␣
 ↪values from y_pred
for i in y_pred:
    y_pred_new=np.append(y_pred_new,(r(i)))

#for i in y_pred_new: #Uncomment this line to see predicted values
#    print(i)
```

9 Importing Required libraries for Generating confusion Matrix

[10]:
```
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score
from sklearn.metrics import classification_report
```

[11]:
```
results=confusion_matrix(y_expect,y_pred_new)
print("confusion_matrix")
print("\n",results)
print("\n\n accuracy_score:",accuracy_score(y_expect,y_pred_new)) # for␣
 ↪printing accuracy
print('\n\n classification report\n\n',metrics.
 ↪classification_report(y_expect,y_pred_new)) # for printing confusion matrix
```

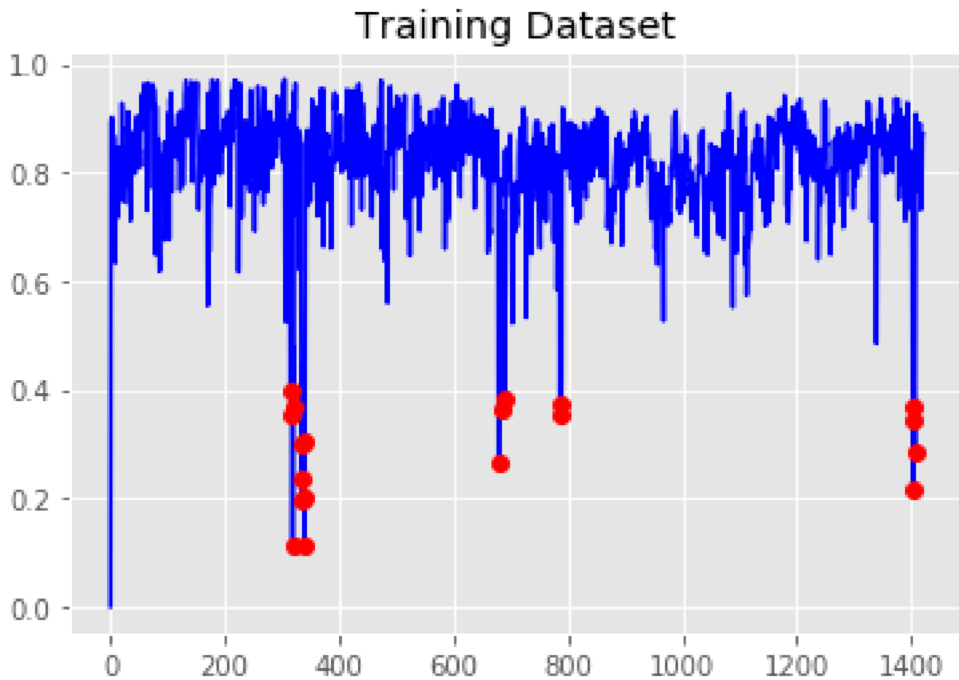
```
confusion_matrix

 [[1458    0]
 [   0    3]]
```

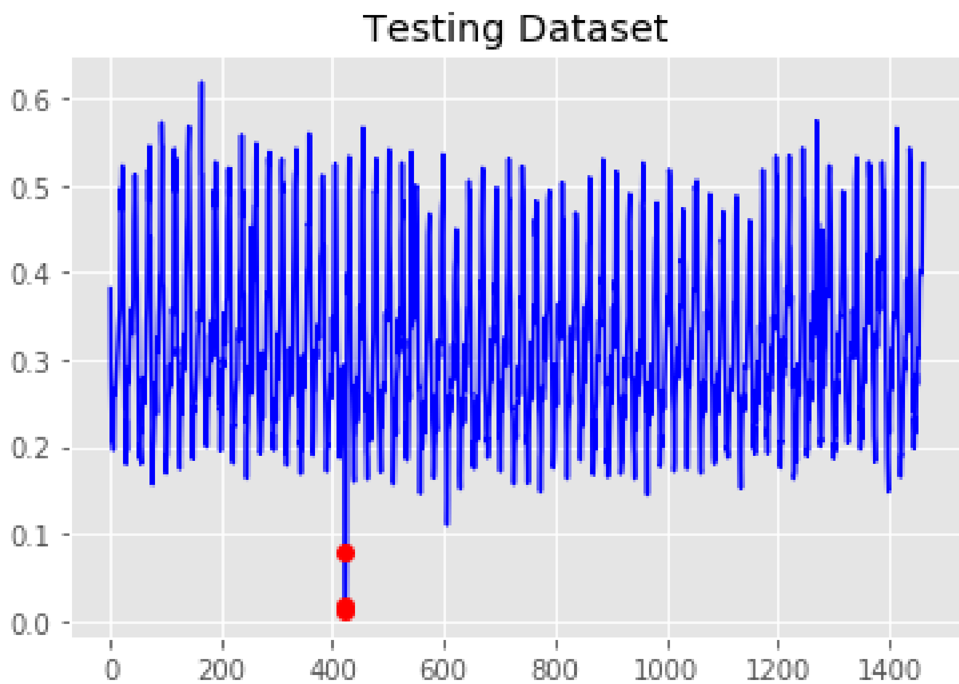
```
accuracy_score: 1.0


classification report
             precision    recall  f1-score   support

          0       1.00      1.00      1.00      1458
          1       1.00      1.00      1.00         3

avg / total       1.00      1.00      1.00      1461
```

10 Plotting of Time-Series Data along with anomalies

[12]:
```python
import pandas as pd
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline

train_data_anomaly_positions = train_data.loc[train_data['is_anomaly'] == 1, ␣
 ↪('timestamp','value')]
test_data_anomaly_positions = test_data1.loc[test_data1['is_anomaly'] == 1, ␣
 ↪('timestamp','value')]
fig = plt.figure(1)
plt.plot(train_data['timestamp'], train_data['value'], color="blue")
plt.title("Training Dataset")
plt.plot(train_data_anomaly_positions['timestamp'],␣
 ↪train_data_anomaly_positions['value'], 'ro', color="red")

fig = plt.figure(2)
plt.plot(test_data1['timestamp'], test_data1['value'], color="blue")
plt.title("Testing Dataset")
plt.plot(test_data_anomaly_positions['timestamp'],␣
 ↪test_data_anomaly_positions['value'], 'ro', color="red")
```

[12]:
```
[<matplotlib.lines.Line2D at 0x1105466a0>]
```


Training Dataset
1.0
0.8
0.6
0.4
0.2
0.0
0
200
400
600
800
1000
1200
1400


Testing Dataset
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0
200
400
600
800
1000
1200
1400

11 Generating Model for Linear Regression

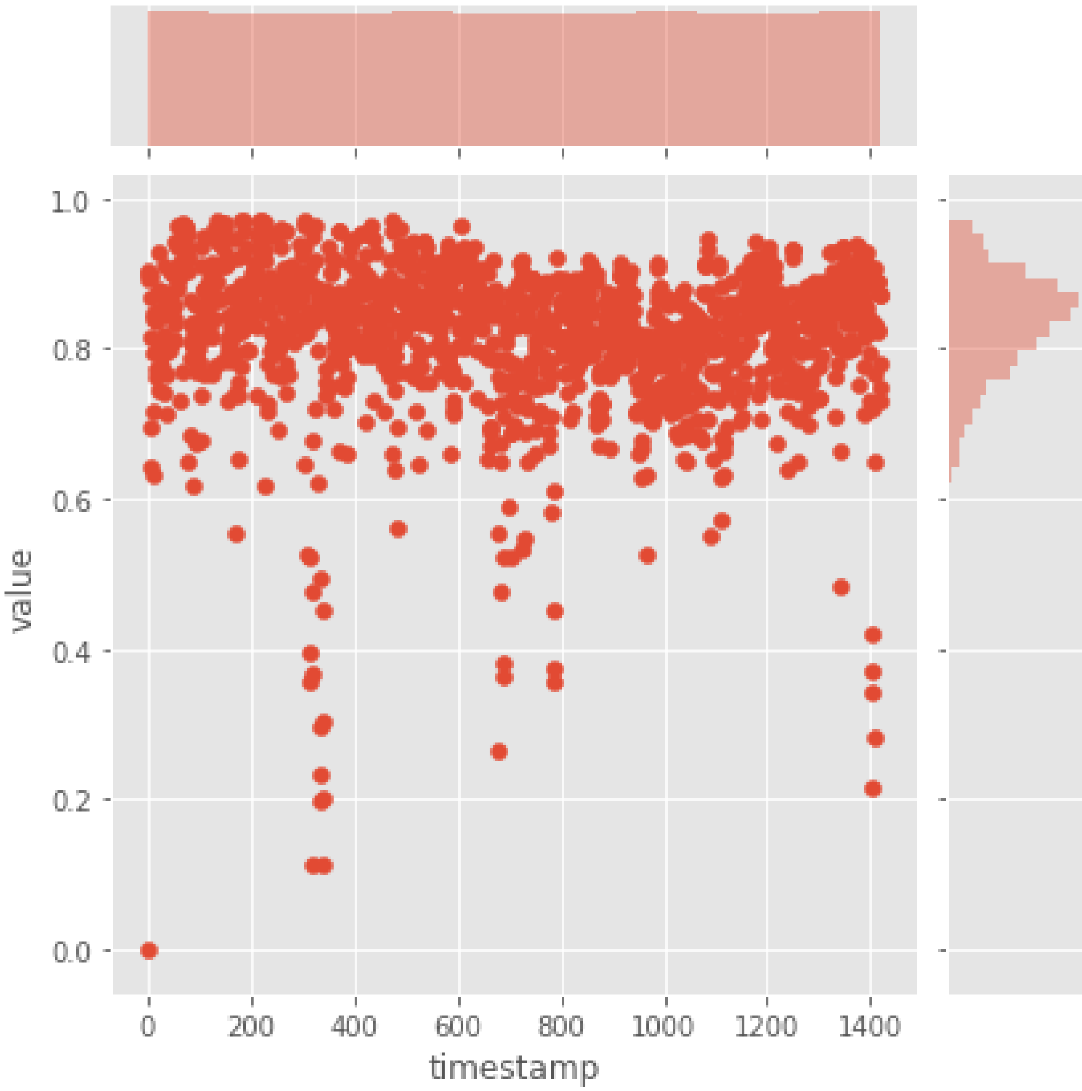
[13]:

```python
import seaborn as sns
from sklearn.linear_model import LinearRegression

"""
    Created on Sep 4
    @author : WaVeRiDeR(Atul)
"""

print(train_data.head()) #Returns first 5 rows of Train_data
print(test_data1.head())  #Returns first 5 rows of Test_data

#Returns Descriptive Statistics that summarizes the central Tendency
print(train_data.describe())
print(test_data1.describe())

#Information of a DataFrames
print(train_data.info())
print(test_data1.info())

#Prints the Shape of a DataFrames
print(test_data1.shape)
print(train_data.shape)

#Droping of Missing Data
#test_data = test_data.dropna()
#train_data = train_data.dropna()

#Prints the Shape of a DataFrames after droping
print(test_data1.shape)
print(train_data.shape)

#Visualizing Train_data and Test_data
sns.jointplot(x = 'timestamp', y = 'value', data = train_data)
sns.jointplot(x = 'timestamp', y = 'value', data = test_data1)

#Creation of Linear Model Object
lm = LinearRegression()

#Slicing of Datasets
x_train = pd.DataFrame(train_data.iloc[:,0].values)
y_train = pd.DataFrame(train_data.iloc[:,1].values)

x_test = pd.DataFrame(test_data1.iloc[:,0].values)
y_test = pd.DataFrame(test_data1.iloc[:,1].values)
```

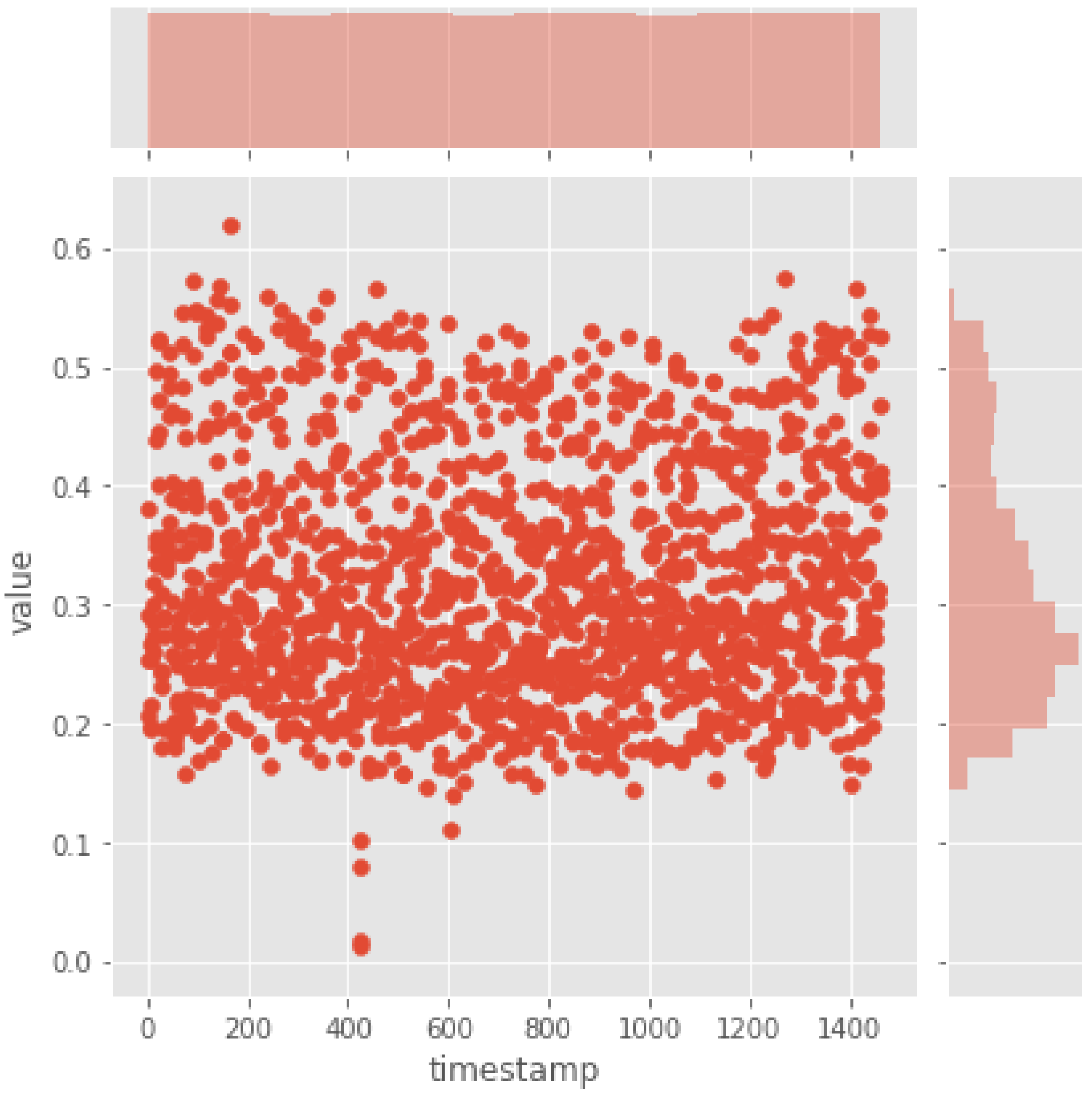
```python
#Training the Model by training dataset
lm.fit(x_train,y_train)

#Prints the Accuracy of Model
accuracy = round(lm.score(x_train,y_train) *100,2)
print('Accuracy:', accuracy)

#Prints the Coefficients
print('Coefficients', lm.coef_)

#Estimated prediction of y_test values based on trained model
predictions = lm.predict(x_test)
```

```
   timestamp     value  is_anomaly
0          1  0.000000           0
1          2  0.892033           0
2          3  0.901426           0
3          4  0.902496           0
4          5  0.897662           0
   timestamp     value  is_anomaly
0          1  0.381389           0
1          2  0.290556           0
2          3  0.252778           0
3          4  0.206111           0
4          5  0.216111           0
         timestamp        value   is_anomaly
count  1420.000000  1420.000000  1420.000000
mean    710.500000     0.819815     0.013380
std     410.063003     0.103296     0.114937
min       1.000000     0.000000     0.000000
25%     355.750000     0.783908     0.000000
50%     710.500000     0.839665     0.000000
75%    1065.250000     0.879390     0.000000
max    1420.000000     0.972366     1.000000
         timestamp        value   is_anomaly
count  1461.000000  1461.000000  1461.000000
mean    731.000000     0.320808     0.002053
std     421.898685     0.102452     0.045283
min       1.000000     0.011667     0.000000
25%     366.000000     0.241944     0.000000
50%     731.000000     0.300833     0.000000
75%    1096.000000     0.393056     0.000000
max    1461.000000     0.618889     1.000000
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1420 entries, 0 to 1419
Data columns (total 3 columns):
timestamp     1420 non-null int64
```

```
value         1420 non-null float64
is_anomaly    1420 non-null int64
dtypes: float64(1), int64(2)
memory usage: 33.4 KB
None
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1461 entries, 0 to 1460
Data columns (total 3 columns):
timestamp     1461 non-null int64
value         1461 non-null float64
is_anomaly    1461 non-null int64
dtypes: float64(1), int64(2)
memory usage: 34.3 KB
None
(1461, 3)
(1420, 3)
(1461, 3)
(1420, 3)
Accuracy: 0.98
Coefficients [[-2.49216219e-05]]
```



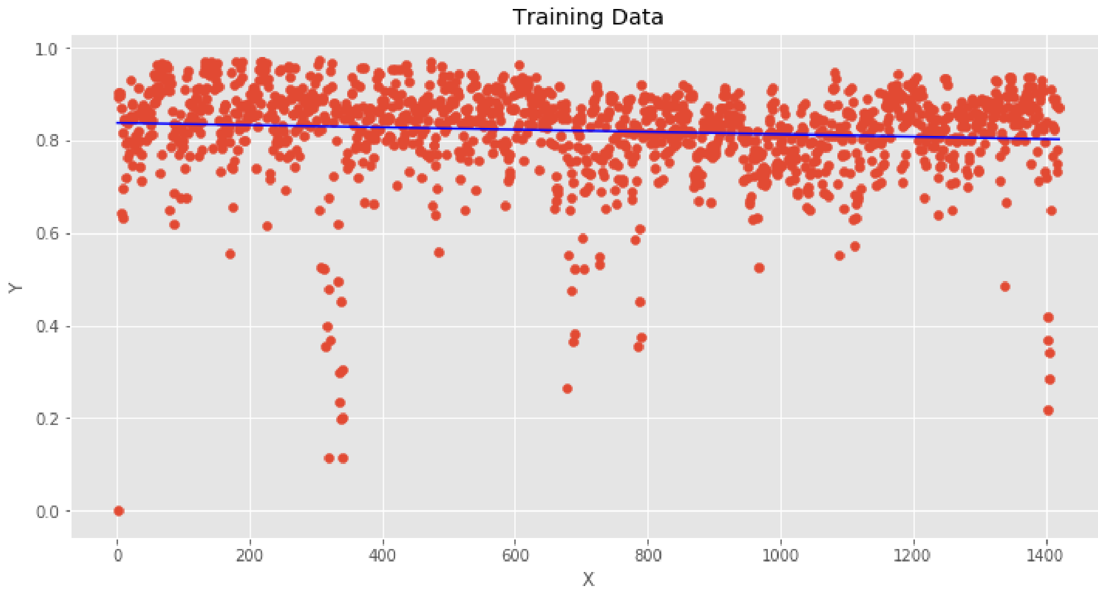
1.0
0.8
0.6
0.4
0.2
0.0
value
0
200
400
600
800
1000
1200
1400
timestamp



12 Plotting the Regression Line

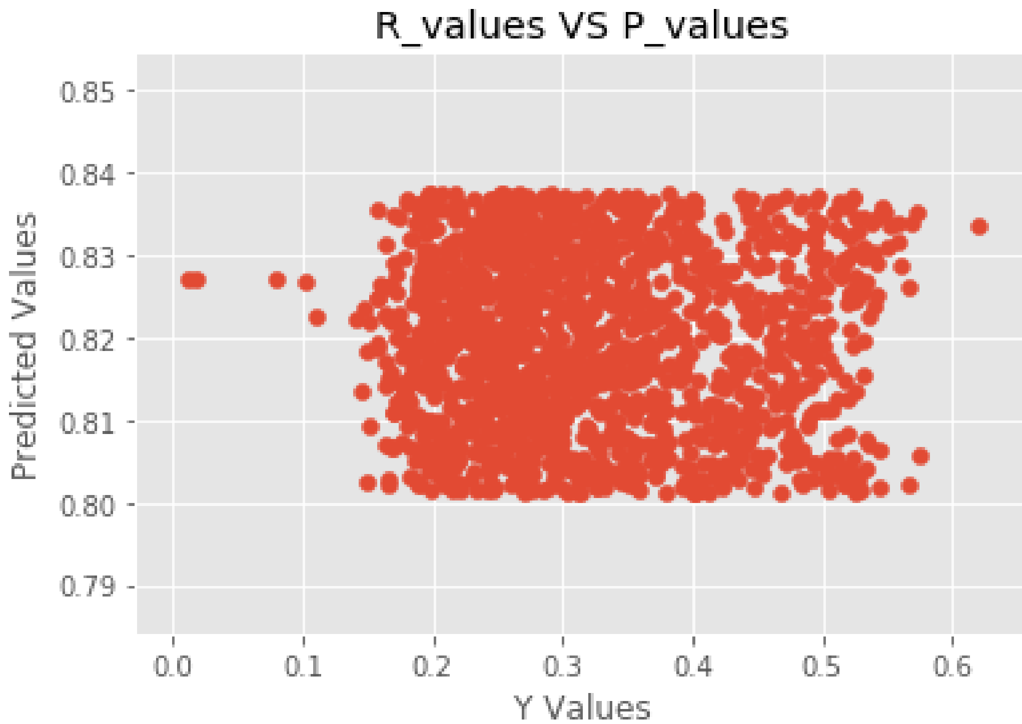
[15]:
```
#Visualizing the Training Dataset
plt.figure(figsize = (12,6))
plt.scatter(x_train,y_train)
plt.plot(x_train,lm.predict(x_train), color = 'blue')
#plt.xlim(5)
#plt.ylim(2)
plt.xlabel('X')
plt.ylabel('Y')
plt.title('Training Data')
```

```
# it is what gives the transparency to the points.
# if they suppose themselves, the colors are added.

plt.show()
```



[16]:
```
#Real Test Values Versus Predicted Test Values
plt.scatter(y_test,predictions)
plt.xlabel('Y Values')
plt.ylabel('Predicted Values')
plt.title('R_values VS P_values')
```

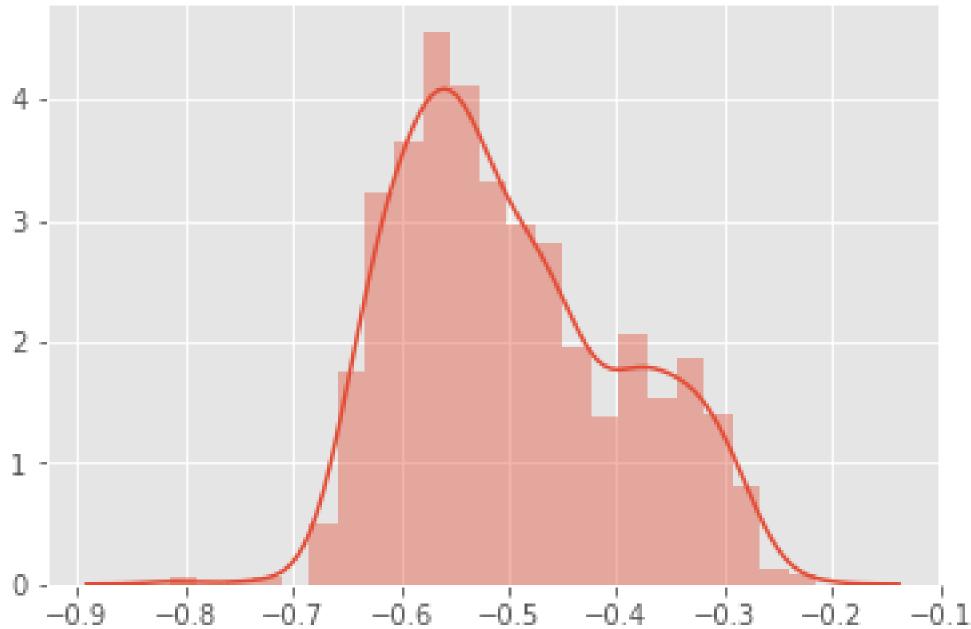
[16]: Text(0.5, 1.0, 'R_values VS P_values')



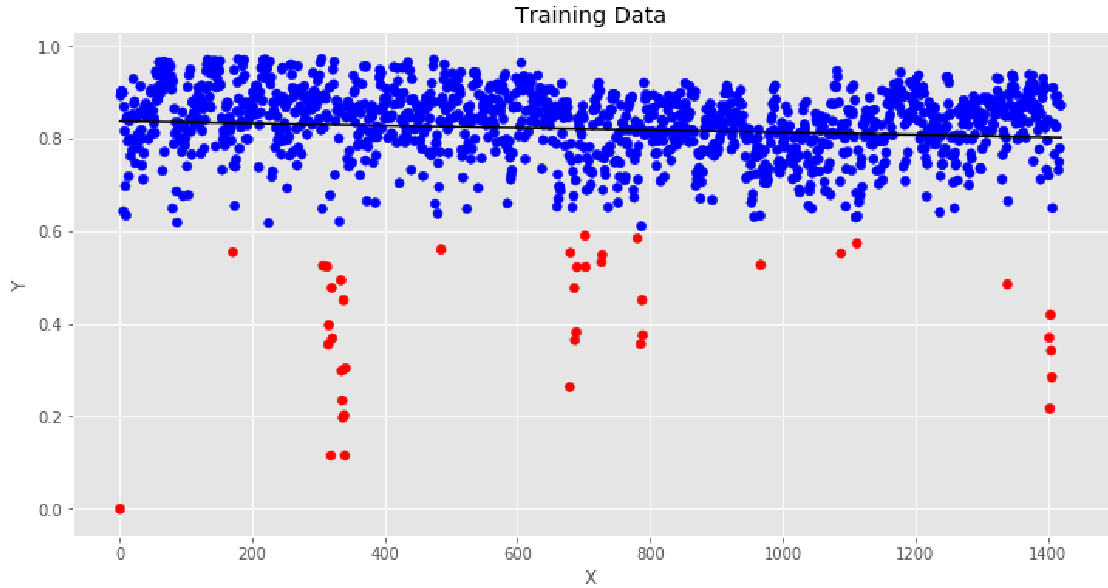
13 Lets check the distribution of our Dataset

[17]:
```
#Model was correct choice for data because of Normal distribution
sns.distplot((y_test-predictions))
```

[17]: <matplotlib.axes._subplots.AxesSubplot at 0x110b49588>



[18]:
```
#plt.plot(x_train,lm.predict(x_train), color = 'red')
plt.figure(figsize = (12,6))
cols = np.where(y_train[0]<=0.6,'r','b')
plt.xlabel('X')
plt.ylabel('Y')
plt.title('Training Data')
plt.scatter(x=x_train,y=y_train,c=cols) #Pass on the list created by the␣
↪function here
plt.plot(x_train,lm.predict(x_train), color = 'black')
plt.show()
plt.savefig("new.jpg")
```



<Figure size 432x288 with 0 Axes>

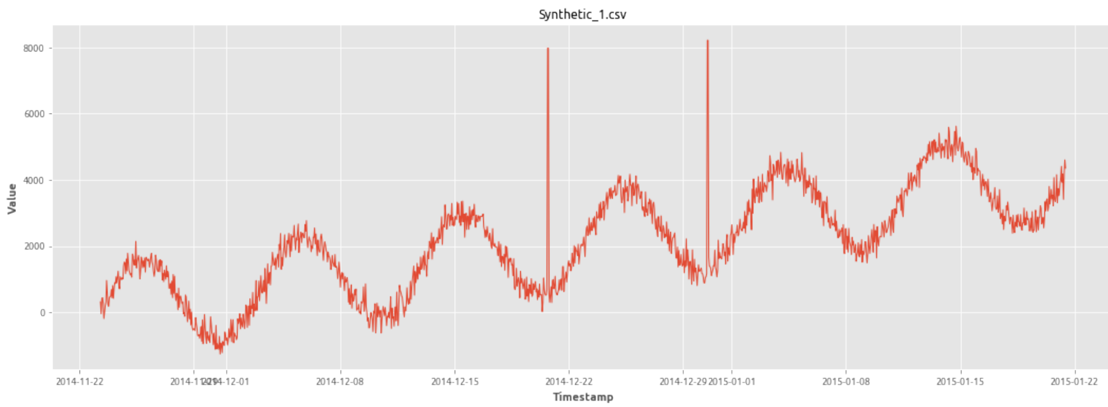
14 Visualizing the benchmark files (A2)

[19]:
```
def apply_styles():
    matplotlib.rcParams['font.size'] = 12
    matplotlib.rcParams['figure.figsize'] = (18, 6)
    matplotlib.rcParams['lines.linewidth'] = 1
    plt.rcParams['font.family'] = 'serif'
    plt.rcParams['font.serif'] = 'Ubuntu'
    plt.rcParams['font.monospace'] = 'Ubuntu Mono'
    plt.rcParams['font.size'] = 12
    plt.rcParams['axes.labelsize'] = 11
    plt.rcParams['axes.labelweight'] = 'bold'
    plt.rcParams['axes.titlesize'] = 12
    plt.rcParams['xtick.labelsize'] = 9
    plt.rcParams['ytick.labelsize'] = 9
    plt.rcParams['legend.fontsize'] = 11
    plt.rcParams['figure.titlesize'] = 13

apply_styles()

path='../A2Benchmark/*.csv'
Benchmark = []

data_load(path, Benchmark)
```

```python
check_null(Benchmark)

convert_to_date_time(Benchmark)

#set_index_df(Benchmark)

count_data_instance(Benchmark)

count_anomaly_instances(Benchmark)

labelled_anomaly_positions(Benchmark)

data_stat(Benchmark)

plt.xlabel('Timestamp')
plt.ylabel('Value')
plt.title("Synthetic_1.csv")
#Benchmark.plot(subplots=True, figsize=(10,12))
A2A = plt.plot(Benchmark[0]['timestamp'], Benchmark[0]['value'])
plt.savefig('synthetic_1.png')
plt.show(A2A)
```

```
Check for any NULL value -
timestamp     0
value         0
is_anomaly    0
dtype: int64
Total data instances in A2 Benchmark: 142100
Total anomaly instances in A2 Benchmark: 466
                        Max          Min          Mean           Std
synthetic_85    8210.758177 -1254.672335   2129.056749   1518.281126
synthetic_91  128420.057623   -93.303340  29558.600651  17617.490683
synthetic_46   12531.576868 -1231.851849   4883.689181   3003.235603
synthetic_52    8631.750691 -1820.080264   2685.056514   1851.799837
synthetic_53   22106.473333 -1266.218014   5334.901872   3433.340594
```

```
/anaconda3/lib/python3.6/site-packages/pandas/plotting/_converter.py:129:
FutureWarning: Using an implicitly registered datetime converter for a
matplotlib plotting method. The converter was registered by pandas on import.
Future versions of pandas will require you to explicitly register matplotlib
converters.

To register the converters:
        >>> from pandas.plotting import register_matplotlib_converters
        >>> register_matplotlib_converters()
  warnings.warn(msg, FutureWarning)
```



14.1 Working on ARIMA Model.

ARIMA Model on Yahoo Benchmark Dataset

14.1.1 Adding all the A2 Datasets in One Frame

[3]:
```
A2Apaths = ['./A2Benchmark/synthetic_1.csv', './A2Benchmark/synthetic_2.csv','./
↪A2Benchmark/synthetic_3.csv','./A2Benchmark/synthetic_4.csv','./A2Benchmark/
↪synthetic_5.csv'
,'./A2Benchmark/synthetic_6.csv','./A2Benchmark/synthetic_7.csv','./A2Benchmark/
↪synthetic_8.csv','./A2Benchmark/synthetic_9.csv','./A2Benchmark/synthetic_10.
↪csv'
          ,'./A2Benchmark/synthetic_10.csv','./A2Benchmark/synthetic_11.csv','./
↪A2Benchmark/synthetic_12.csv','./A2Benchmark/synthetic_13.csv'
          ,'./A2Benchmark/synthetic_14.csv','./A2Benchmark/synthetic_15.csv','./
↪A2Benchmark/synthetic_17.csv','./A2Benchmark/synthetic_18.csv','./
↪A2Benchmark/synthetic_19.csv', './A2Benchmark/synthetic_20.csv','./
↪A2Benchmark/synthetic_21.csv','./A2Benchmark/synthetic_22.csv','./
↪A2Benchmark/synthetic_23.csv'
,'./A2Benchmark/synthetic_24.csv','./A2Benchmark/synthetic_25.csv','./
↪A2Benchmark/synthetic_26.csv','./A2Benchmark/synthetic_27.csv','./
↪A2Benchmark/synthetic_28.csv','./A2Benchmark/synthetic_29.csv', './
↪A2Benchmark/synthetic_30.csv','./A2Benchmark/synthetic_31.csv','./
↪A2Benchmark/synthetic_32.csv','./A2Benchmark/synthetic_33.csv'
,'./A2Benchmark/synthetic_34.csv','./A2Benchmark/synthetic_35.csv','./
↪A2Benchmark/synthetic_36.csv','./A2Benchmark/synthetic_37.csv','./
↪A2Benchmark/synthetic_38.csv','./A2Benchmark/synthetic_39.csv','./
↪A2Benchmark/synthetic_40.csv','./A2Benchmark/synthetic_41.csv'
          ,'./A2Benchmark/synthetic_42.csv','./A2Benchmark/synthetic_43.csv','./
↪A2Benchmark/synthetic_44.csv','./A2Benchmark/synthetic_45.csv'
```

```
          ,'./A2Benchmark/synthetic_46.csv','./A2Benchmark/synthetic_47.csv','./
 ↪A2Benchmark/synthetic_48.csv', './A2Benchmark/synthetic_49.csv','./
 ↪A2Benchmark/synthetic_50.csv','./A2Benchmark/synthetic_51.csv','./
 ↪A2Benchmark/synthetic_52.csv'
,'./A2Benchmark/synthetic_53.csv','./A2Benchmark/synthetic_54.csv','./
 ↪A2Benchmark/synthetic_55.csv','./A2Benchmark/synthetic_56.csv','./
 ↪A2Benchmark/synthetic_57.csv'
          ,'./A2Benchmark/synthetic_58.csv','./A2Benchmark/synthetic_59.csv','./
 ↪A2Benchmark/synthetic_60.csv','./A2Benchmark/synthetic_61.csv'
          ,'./A2Benchmark/synthetic_62.csv','./A2Benchmark/synthetic_63.csv']
df = pd.concat(map(pd.read_csv, A2Apaths))
```

How many Anomalies are in total of A2 Benchmark?

[4]:
```
df.head()
df.is_anomaly.sum()
```

[4]: 294

Let's change the Datetime to readable format for eaasy visualization

[5]:
```
df['timestamp'] = pd.to_datetime(df['timestamp'], unit='s')
#A2df.timestamp = A2df.timestamp.dt.tz_localize('UTC')
```

[8]:
```
print(df['timestamp'].head(10))
```

```
0   2014-11-23 07:00:00
1   2014-11-23 08:00:00
2   2014-11-23 09:00:00
3   2014-11-23 10:00:00
4   2014-11-23 11:00:00
5   2014-11-23 12:00:00
6   2014-11-23 13:00:00
7   2014-11-23 14:00:00
8   2014-11-23 15:00:00
9   2014-11-23 16:00:00
Name: timestamp, dtype: datetime64[ns]
```

[9]:
```
df.head()
```

[9]:

	timestamp	value	is_anomaly
0	2014-11-23 07:00:00	13.894031	0
1	2014-11-23 08:00:00	33.578274	0
2	2014-11-23 09:00:00	88.933746	0
3	2014-11-23 10:00:00	125.389424	0
4	2014-11-23 11:00:00	152.962000	0

15 Importing required libraries and functions

[10]:
```
import warnings
import itertools
import numpy as np
import matplotlib.pyplot as plt
warnings.filterwarnings("ignore")
plt.style.use('fivethirtyeight')
import pandas as pd
import statsmodels.api as sm
import matplotlib
matplotlib.rcParams['axes.labelsize'] = 14
matplotlib.rcParams['xtick.labelsize'] = 12
matplotlib.rcParams['ytick.labelsize'] = 12
matplotlib.rcParams['text.color'] = 'k'
```

15.0.1 Creating a New Dataframe with values and one without values.

[11]:
```
#df.head()
newdf = df[['timestamp','value']]
newdf['timestamp'] = pd.to_datetime(newdf['timestamp'], unit='s')
#newdf.head()

#df with anomaly

dfano = df[['timestamp','is_anomaly']]
dfano['timestamp'] = pd.to_datetime(dfano['timestamp'],unit = 's')
dfano.head()
```

[11]:
```
            timestamp  is_anomaly
0 2014-11-23 07:00:00           0
1 2014-11-23 08:00:00           0
2 2014-11-23 09:00:00           0
3 2014-11-23 10:00:00           0
4 2014-11-23 11:00:00           0
```

15.0.2 Let's check the starting and ending dates available.

[12]:
```
newdf['timestamp'].min(), newdf['timestamp'].max()
```

[12]:
```
(Timestamp('2014-11-23 07:00:00'), Timestamp('2015-01-21 11:00:00'))
```

[13]:
```
newdf.isnull().sum()
```

[13]:
```
timestamp    0
value        0
```

```
dtype: int64
```

[14]:
```
dfano.sort_values('timestamp').head()
```

[14]:
```
            timestamp  is_anomaly
0 2014-11-23 07:00:00           0
0 2014-11-23 07:00:00           0
0 2014-11-23 07:00:00           0
0 2014-11-23 07:00:00           0
0 2014-11-23 07:00:00           0
```

[15]:
```
newdf = newdf.groupby('timestamp')['value'].sum().reset_index()
dfano = dfano.groupby('timestamp')['is_anomaly'].sum().reset_index()
```

[16]:
```
dfano.head()
#newdf.head()
```

[16]:
```
            timestamp  is_anomaly
0 2014-11-23 07:00:00           0
1 2014-11-23 08:00:00           0
2 2014-11-23 09:00:00           0
3 2014-11-23 10:00:00           0
4 2014-11-23 11:00:00           0
```

[17]:
```
newdf = newdf.set_index('timestamp')
newdf.index

dfano = dfano.set_index('timestamp')
dfano.index
```

[17]:
```
DatetimeIndex(['2014-11-23 07:00:00', '2014-11-23 08:00:00',
               '2014-11-23 09:00:00', '2014-11-23 10:00:00',
               '2014-11-23 11:00:00', '2014-11-23 12:00:00',
               '2014-11-23 13:00:00', '2014-11-23 14:00:00',
               '2014-11-23 15:00:00', '2014-11-23 16:00:00',
               …
               '2015-01-21 02:00:00', '2015-01-21 03:00:00',
               '2015-01-21 04:00:00', '2015-01-21 05:00:00',
               '2015-01-21 06:00:00', '2015-01-21 07:00:00',
               '2015-01-21 08:00:00', '2015-01-21 09:00:00',
               '2015-01-21 10:00:00', '2015-01-21 11:00:00'],
              dtype='datetime64[ns]', name='timestamp', length=1421, freq=None)
```

[18]:
```
y = newdf['value'].resample('H').median()
#y = dfano['is_anomaly'].resample('H').mean()
with pd.option_context('display.max_rows', None, 'display.max_columns', None):
    print(dfano)
```

```python
#print(dfano)
```

```
                     is_anomaly
timestamp
2014-11-23 07:00:00           0
2014-11-23 08:00:00           0
2014-11-23 09:00:00           0
2014-11-23 10:00:00           0
2014-11-23 11:00:00           0
2014-11-23 12:00:00           0
2014-11-23 13:00:00           0
2014-11-23 14:00:00           0
2014-11-23 15:00:00           0
2014-11-23 16:00:00           0
2014-11-23 17:00:00           0
2014-11-23 18:00:00           0
2014-11-23 19:00:00           0
2014-11-23 20:00:00           0
2014-11-23 21:00:00           0
2014-11-23 22:00:00           0
2014-11-23 23:00:00           0
2014-11-24 00:00:00           0
2014-11-24 01:00:00           0
2014-11-24 02:00:00           0
2014-11-24 03:00:00           0
2014-11-24 04:00:00           0
2014-11-24 05:00:00           0
2014-11-24 06:00:00           0
2014-11-24 07:00:00           0
2014-11-24 08:00:00           0
2014-11-24 09:00:00           0
2014-11-24 10:00:00           0
2014-11-24 11:00:00           0
2014-11-24 12:00:00           0
2014-11-24 13:00:00           0
2014-11-24 14:00:00           0
2014-11-24 15:00:00           0
2014-11-24 16:00:00           0
2014-11-24 17:00:00           0
2014-11-24 18:00:00           0
2014-11-24 19:00:00           0
2014-11-24 20:00:00           0
2014-11-24 21:00:00           0
2014-11-24 22:00:00           0
2014-11-24 23:00:00           0
2014-11-25 00:00:00           0
2014-11-25 01:00:00           0
```

2014-11-25 02:00:00	0
2014-11-25 03:00:00	0
2014-11-25 04:00:00	0
2014-11-25 05:00:00	0
2014-11-25 06:00:00	0
2014-11-25 07:00:00	0
2014-11-25 08:00:00	0
2014-11-25 09:00:00	0
2014-11-25 10:00:00	0
2014-11-25 11:00:00	0
2014-11-25 12:00:00	0
2014-11-25 13:00:00	0
2014-11-25 14:00:00	0
2014-11-25 15:00:00	0
2014-11-25 16:00:00	0
2014-11-25 17:00:00	0
2014-11-25 18:00:00	0
2014-11-25 19:00:00	0
2014-11-25 20:00:00	0
2014-11-25 21:00:00	0
2014-11-25 22:00:00	0
2014-11-25 23:00:00	0
2014-11-26 00:00:00	0
2014-11-26 01:00:00	0
2014-11-26 02:00:00	0
2014-11-26 03:00:00	0
2014-11-26 04:00:00	0
2014-11-26 05:00:00	0
2014-11-26 06:00:00	0
2014-11-26 07:00:00	0
2014-11-26 08:00:00	0
2014-11-26 09:00:00	0
2014-11-26 10:00:00	0
2014-11-26 11:00:00	0
2014-11-26 12:00:00	0
2014-11-26 13:00:00	0
2014-11-26 14:00:00	0
2014-11-26 15:00:00	0
2014-11-26 16:00:00	0
2014-11-26 17:00:00	0
2014-11-26 18:00:00	0
2014-11-26 19:00:00	0
2014-11-26 20:00:00	0
2014-11-26 21:00:00	0
2014-11-26 22:00:00	0
2014-11-26 23:00:00	0
2014-11-27 00:00:00	0
2014-11-27 01:00:00	0

2014-11-27 02:00:00	0
2014-11-27 03:00:00	0
2014-11-27 04:00:00	0
2014-11-27 05:00:00	0
2014-11-27 06:00:00	0
2014-11-27 07:00:00	0
2014-11-27 08:00:00	0
2014-11-27 09:00:00	0
2014-11-27 10:00:00	0
2014-11-27 11:00:00	0
2014-11-27 12:00:00	0
2014-11-27 13:00:00	0
2014-11-27 14:00:00	0
2014-11-27 15:00:00	0
2014-11-27 16:00:00	0
2014-11-27 17:00:00	0
2014-11-27 18:00:00	0
2014-11-27 19:00:00	0
2014-11-27 20:00:00	0
2014-11-27 21:00:00	0
2014-11-27 22:00:00	0
2014-11-27 23:00:00	0
2014-11-28 00:00:00	0
2014-11-28 01:00:00	0
2014-11-28 02:00:00	0
2014-11-28 03:00:00	0
2014-11-28 04:00:00	0
2014-11-28 05:00:00	0
2014-11-28 06:00:00	0
2014-11-28 07:00:00	0
2014-11-28 08:00:00	0
2014-11-28 09:00:00	0
2014-11-28 10:00:00	0
2014-11-28 11:00:00	0
2014-11-28 12:00:00	0
2014-11-28 13:00:00	0
2014-11-28 14:00:00	0
2014-11-28 15:00:00	0
2014-11-28 16:00:00	0
2014-11-28 17:00:00	0
2014-11-28 18:00:00	0
2014-11-28 19:00:00	0
2014-11-28 20:00:00	0
2014-11-28 21:00:00	0
2014-11-28 22:00:00	0
2014-11-28 23:00:00	0
2014-11-29 00:00:00	0
2014-11-29 01:00:00	0

2014-11-29 02:00:00	0
2014-11-29 03:00:00	0
2014-11-29 04:00:00	0
2014-11-29 05:00:00	0
2014-11-29 06:00:00	0
2014-11-29 07:00:00	0
2014-11-29 08:00:00	0
2014-11-29 09:00:00	0
2014-11-29 10:00:00	0
2014-11-29 11:00:00	0
2014-11-29 12:00:00	0
2014-11-29 13:00:00	0
2014-11-29 14:00:00	0
2014-11-29 15:00:00	0
2014-11-29 16:00:00	0
2014-11-29 17:00:00	0
2014-11-29 18:00:00	0
2014-11-29 19:00:00	0
2014-11-29 20:00:00	0
2014-11-29 21:00:00	0
2014-11-29 22:00:00	0
2014-11-29 23:00:00	0
2014-11-30 00:00:00	0
2014-11-30 01:00:00	0
2014-11-30 02:00:00	0
2014-11-30 03:00:00	0
2014-11-30 04:00:00	0
2014-11-30 05:00:00	0
2014-11-30 06:00:00	0
2014-11-30 07:00:00	0
2014-11-30 08:00:00	0
2014-11-30 09:00:00	0
2014-11-30 10:00:00	0
2014-11-30 11:00:00	0
2014-11-30 12:00:00	0
2014-11-30 13:00:00	0
2014-11-30 14:00:00	0
2014-11-30 15:00:00	0
2014-11-30 16:00:00	0
2014-11-30 17:00:00	0
2014-11-30 18:00:00	0
2014-11-30 19:00:00	0
2014-11-30 20:00:00	0
2014-11-30 21:00:00	0
2014-11-30 22:00:00	0
2014-11-30 23:00:00	0
2014-12-01 00:00:00	0
2014-12-01 01:00:00	0

2014-12-01 02:00:00	0
2014-12-01 03:00:00	0
2014-12-01 04:00:00	0
2014-12-01 05:00:00	0
2014-12-01 06:00:00	0
2014-12-01 07:00:00	0
2014-12-01 08:00:00	0
2014-12-01 09:00:00	0
2014-12-01 10:00:00	0
2014-12-01 11:00:00	0
2014-12-01 12:00:00	0
2014-12-01 13:00:00	0
2014-12-01 14:00:00	0
2014-12-01 15:00:00	0
2014-12-01 16:00:00	0
2014-12-01 17:00:00	0
2014-12-01 18:00:00	0
2014-12-01 19:00:00	0
2014-12-01 20:00:00	0
2014-12-01 21:00:00	0
2014-12-01 22:00:00	0
2014-12-01 23:00:00	0
2014-12-02 00:00:00	0
2014-12-02 01:00:00	0
2014-12-02 02:00:00	0
2014-12-02 03:00:00	0
2014-12-02 04:00:00	0
2014-12-02 05:00:00	0
2014-12-02 06:00:00	0
2014-12-02 07:00:00	0
2014-12-02 08:00:00	0
2014-12-02 09:00:00	0
2014-12-02 10:00:00	0
2014-12-02 11:00:00	0
2014-12-02 12:00:00	0
2014-12-02 13:00:00	0
2014-12-02 14:00:00	0
2014-12-02 15:00:00	0
2014-12-02 16:00:00	0
2014-12-02 17:00:00	0
2014-12-02 18:00:00	0
2014-12-02 19:00:00	0
2014-12-02 20:00:00	0
2014-12-02 21:00:00	0
2014-12-02 22:00:00	0
2014-12-02 23:00:00	0
2014-12-03 00:00:00	0
2014-12-03 01:00:00	0

2014-12-03 02:00:00	0
2014-12-03 03:00:00	0
2014-12-03 04:00:00	0
2014-12-03 05:00:00	0
2014-12-03 06:00:00	0
2014-12-03 07:00:00	0
2014-12-03 08:00:00	0
2014-12-03 09:00:00	0
2014-12-03 10:00:00	0
2014-12-03 11:00:00	0
2014-12-03 12:00:00	0
2014-12-03 13:00:00	0
2014-12-03 14:00:00	0
2014-12-03 15:00:00	0
2014-12-03 16:00:00	0
2014-12-03 17:00:00	0
2014-12-03 18:00:00	0
2014-12-03 19:00:00	0
2014-12-03 20:00:00	0
2014-12-03 21:00:00	0
2014-12-03 22:00:00	0
2014-12-03 23:00:00	0
2014-12-04 00:00:00	0
2014-12-04 01:00:00	0
2014-12-04 02:00:00	0
2014-12-04 03:00:00	0
2014-12-04 04:00:00	0
2014-12-04 05:00:00	0
2014-12-04 06:00:00	0
2014-12-04 07:00:00	0
2014-12-04 08:00:00	0
2014-12-04 09:00:00	0
2014-12-04 10:00:00	0
2014-12-04 11:00:00	0
2014-12-04 12:00:00	0
2014-12-04 13:00:00	0
2014-12-04 14:00:00	0
2014-12-04 15:00:00	0
2014-12-04 16:00:00	0
2014-12-04 17:00:00	0
2014-12-04 18:00:00	0
2014-12-04 19:00:00	0
2014-12-04 20:00:00	0
2014-12-04 21:00:00	0
2014-12-04 22:00:00	0
2014-12-04 23:00:00	0
2014-12-05 00:00:00	0
2014-12-05 01:00:00	0

2014-12-05 02:00:00	0
2014-12-05 03:00:00	0
2014-12-05 04:00:00	0
2014-12-05 05:00:00	0
2014-12-05 06:00:00	0
2014-12-05 07:00:00	0
2014-12-05 08:00:00	0
2014-12-05 09:00:00	0
2014-12-05 10:00:00	0
2014-12-05 11:00:00	0
2014-12-05 12:00:00	0
2014-12-05 13:00:00	0
2014-12-05 14:00:00	0
2014-12-05 15:00:00	0
2014-12-05 16:00:00	0
2014-12-05 17:00:00	0
2014-12-05 18:00:00	0
2014-12-05 19:00:00	0
2014-12-05 20:00:00	0
2014-12-05 21:00:00	0
2014-12-05 22:00:00	0
2014-12-05 23:00:00	0
2014-12-06 00:00:00	0
2014-12-06 01:00:00	0
2014-12-06 02:00:00	0
2014-12-06 03:00:00	0
2014-12-06 04:00:00	0
2014-12-06 05:00:00	0
2014-12-06 06:00:00	0
2014-12-06 07:00:00	0
2014-12-06 08:00:00	0
2014-12-06 09:00:00	0
2014-12-06 10:00:00	0
2014-12-06 11:00:00	0
2014-12-06 12:00:00	0
2014-12-06 13:00:00	0
2014-12-06 14:00:00	0
2014-12-06 15:00:00	0
2014-12-06 16:00:00	0
2014-12-06 17:00:00	0
2014-12-06 18:00:00	0
2014-12-06 19:00:00	0
2014-12-06 20:00:00	0
2014-12-06 21:00:00	0
2014-12-06 22:00:00	0
2014-12-06 23:00:00	0
2014-12-07 00:00:00	0
2014-12-07 01:00:00	0

2014-12-07 02:00:00	0
2014-12-07 03:00:00	0
2014-12-07 04:00:00	0
2014-12-07 05:00:00	0
2014-12-07 06:00:00	0
2014-12-07 07:00:00	0
2014-12-07 08:00:00	0
2014-12-07 09:00:00	0
2014-12-07 10:00:00	0
2014-12-07 11:00:00	0
2014-12-07 12:00:00	0
2014-12-07 13:00:00	0
2014-12-07 14:00:00	0
2014-12-07 15:00:00	0
2014-12-07 16:00:00	0
2014-12-07 17:00:00	0
2014-12-07 18:00:00	0
2014-12-07 19:00:00	0
2014-12-07 20:00:00	0
2014-12-07 21:00:00	0
2014-12-07 22:00:00	0
2014-12-07 23:00:00	0
2014-12-08 00:00:00	0
2014-12-08 01:00:00	0
2014-12-08 02:00:00	0
2014-12-08 03:00:00	0
2014-12-08 04:00:00	0
2014-12-08 05:00:00	0
2014-12-08 06:00:00	0
2014-12-08 07:00:00	0
2014-12-08 08:00:00	0
2014-12-08 09:00:00	0
2014-12-08 10:00:00	0
2014-12-08 11:00:00	0
2014-12-08 12:00:00	0
2014-12-08 13:00:00	0
2014-12-08 14:00:00	0
2014-12-08 15:00:00	0
2014-12-08 16:00:00	0
2014-12-08 17:00:00	0
2014-12-08 18:00:00	0
2014-12-08 19:00:00	0
2014-12-08 20:00:00	0
2014-12-08 21:00:00	0
2014-12-08 22:00:00	0
2014-12-08 23:00:00	0
2014-12-09 00:00:00	0
2014-12-09 01:00:00	0

2014-12-09 02:00:00	0
2014-12-09 03:00:00	0
2014-12-09 04:00:00	0
2014-12-09 05:00:00	0
2014-12-09 06:00:00	0
2014-12-09 07:00:00	0
2014-12-09 08:00:00	0
2014-12-09 09:00:00	0
2014-12-09 10:00:00	0
2014-12-09 11:00:00	0
2014-12-09 12:00:00	0
2014-12-09 13:00:00	0
2014-12-09 14:00:00	0
2014-12-09 15:00:00	0
2014-12-09 16:00:00	0
2014-12-09 17:00:00	0
2014-12-09 18:00:00	0
2014-12-09 19:00:00	0
2014-12-09 20:00:00	0
2014-12-09 21:00:00	0
2014-12-09 22:00:00	0
2014-12-09 23:00:00	0
2014-12-10 00:00:00	0
2014-12-10 01:00:00	0
2014-12-10 02:00:00	0
2014-12-10 03:00:00	0
2014-12-10 04:00:00	0
2014-12-10 05:00:00	0
2014-12-10 06:00:00	0
2014-12-10 07:00:00	0
2014-12-10 08:00:00	0
2014-12-10 09:00:00	0
2014-12-10 10:00:00	0
2014-12-10 11:00:00	0
2014-12-10 12:00:00	0
2014-12-10 13:00:00	0
2014-12-10 14:00:00	0
2014-12-10 15:00:00	0
2014-12-10 16:00:00	0
2014-12-10 17:00:00	0
2014-12-10 18:00:00	0
2014-12-10 19:00:00	0
2014-12-10 20:00:00	0
2014-12-10 21:00:00	0
2014-12-10 22:00:00	0
2014-12-10 23:00:00	0
2014-12-11 00:00:00	0
2014-12-11 01:00:00	0

2014-12-11 02:00:00	0
2014-12-11 03:00:00	0
2014-12-11 04:00:00	0
2014-12-11 05:00:00	0
2014-12-11 06:00:00	0
2014-12-11 07:00:00	0
2014-12-11 08:00:00	0
2014-12-11 09:00:00	0
2014-12-11 10:00:00	0
2014-12-11 11:00:00	0
2014-12-11 12:00:00	0
2014-12-11 13:00:00	0
2014-12-11 14:00:00	0
2014-12-11 15:00:00	0
2014-12-11 16:00:00	0
2014-12-11 17:00:00	0
2014-12-11 18:00:00	0
2014-12-11 19:00:00	0
2014-12-11 20:00:00	0
2014-12-11 21:00:00	0
2014-12-11 22:00:00	0
2014-12-11 23:00:00	0
2014-12-12 00:00:00	0
2014-12-12 01:00:00	0
2014-12-12 02:00:00	0
2014-12-12 03:00:00	0
2014-12-12 04:00:00	0
2014-12-12 05:00:00	0
2014-12-12 06:00:00	0
2014-12-12 07:00:00	0
2014-12-12 08:00:00	0
2014-12-12 09:00:00	0
2014-12-12 10:00:00	0
2014-12-12 11:00:00	0
2014-12-12 12:00:00	0
2014-12-12 13:00:00	0
2014-12-12 14:00:00	0
2014-12-12 15:00:00	0
2014-12-12 16:00:00	0
2014-12-12 17:00:00	0
2014-12-12 18:00:00	0
2014-12-12 19:00:00	0
2014-12-12 20:00:00	0
2014-12-12 21:00:00	0
2014-12-12 22:00:00	0
2014-12-12 23:00:00	0
2014-12-13 00:00:00	0
2014-12-13 01:00:00	0

2014-12-13 02:00:00	0
2014-12-13 03:00:00	0
2014-12-13 04:00:00	0
2014-12-13 05:00:00	0
2014-12-13 06:00:00	0
2014-12-13 07:00:00	0
2014-12-13 08:00:00	0
2014-12-13 09:00:00	0
2014-12-13 10:00:00	0
2014-12-13 11:00:00	0
2014-12-13 12:00:00	0
2014-12-13 13:00:00	0
2014-12-13 14:00:00	0
2014-12-13 15:00:00	0
2014-12-13 16:00:00	0
2014-12-13 17:00:00	0
2014-12-13 18:00:00	0
2014-12-13 19:00:00	0
2014-12-13 20:00:00	0
2014-12-13 21:00:00	0
2014-12-13 22:00:00	0
2014-12-13 23:00:00	0
2014-12-14 00:00:00	0
2014-12-14 01:00:00	0
2014-12-14 02:00:00	0
2014-12-14 03:00:00	0
2014-12-14 04:00:00	0
2014-12-14 05:00:00	0
2014-12-14 06:00:00	0
2014-12-14 07:00:00	0
2014-12-14 08:00:00	0
2014-12-14 09:00:00	0
2014-12-14 10:00:00	0
2014-12-14 11:00:00	0
2014-12-14 12:00:00	0
2014-12-14 13:00:00	0
2014-12-14 14:00:00	0
2014-12-14 15:00:00	0
2014-12-14 16:00:00	0
2014-12-14 17:00:00	0
2014-12-14 18:00:00	0
2014-12-14 19:00:00	0
2014-12-14 20:00:00	0
2014-12-14 21:00:00	0
2014-12-14 22:00:00	0
2014-12-14 23:00:00	0
2014-12-15 00:00:00	0
2014-12-15 01:00:00	0

2014-12-15 02:00:00	0
2014-12-15 03:00:00	0
2014-12-15 04:00:00	0
2014-12-15 05:00:00	0
2014-12-15 06:00:00	0
2014-12-15 07:00:00	0
2014-12-15 08:00:00	0
2014-12-15 09:00:00	0
2014-12-15 10:00:00	0
2014-12-15 11:00:00	0
2014-12-15 12:00:00	0
2014-12-15 13:00:00	0
2014-12-15 14:00:00	0
2014-12-15 15:00:00	0
2014-12-15 16:00:00	0
2014-12-15 17:00:00	0
2014-12-15 18:00:00	0
2014-12-15 19:00:00	0
2014-12-15 20:00:00	0
2014-12-15 21:00:00	0
2014-12-15 22:00:00	0
2014-12-15 23:00:00	0
2014-12-16 00:00:00	0
2014-12-16 01:00:00	0
2014-12-16 02:00:00	0
2014-12-16 03:00:00	0
2014-12-16 04:00:00	0
2014-12-16 05:00:00	0
2014-12-16 06:00:00	0
2014-12-16 07:00:00	0
2014-12-16 08:00:00	0
2014-12-16 09:00:00	0
2014-12-16 10:00:00	0
2014-12-16 11:00:00	0
2014-12-16 12:00:00	0
2014-12-16 13:00:00	0
2014-12-16 14:00:00	0
2014-12-16 15:00:00	0
2014-12-16 16:00:00	0
2014-12-16 17:00:00	0
2014-12-16 18:00:00	0
2014-12-16 19:00:00	0
2014-12-16 20:00:00	0
2014-12-16 21:00:00	0
2014-12-16 22:00:00	0
2014-12-16 23:00:00	0
2014-12-17 00:00:00	0
2014-12-17 01:00:00	0

2014-12-17 02:00:00	0
2014-12-17 03:00:00	0
2014-12-17 04:00:00	0
2014-12-17 05:00:00	0
2014-12-17 06:00:00	0
2014-12-17 07:00:00	0
2014-12-17 08:00:00	0
2014-12-17 09:00:00	0
2014-12-17 10:00:00	0
2014-12-17 11:00:00	0
2014-12-17 12:00:00	0
2014-12-17 13:00:00	0
2014-12-17 14:00:00	0
2014-12-17 15:00:00	0
2014-12-17 16:00:00	0
2014-12-17 17:00:00	0
2014-12-17 18:00:00	0
2014-12-17 19:00:00	0
2014-12-17 20:00:00	0
2014-12-17 21:00:00	0
2014-12-17 22:00:00	0
2014-12-17 23:00:00	0
2014-12-18 00:00:00	0
2014-12-18 01:00:00	0
2014-12-18 02:00:00	0
2014-12-18 03:00:00	0
2014-12-18 04:00:00	0
2014-12-18 05:00:00	0
2014-12-18 06:00:00	0
2014-12-18 07:00:00	0
2014-12-18 08:00:00	0
2014-12-18 09:00:00	0
2014-12-18 10:00:00	0
2014-12-18 11:00:00	0
2014-12-18 12:00:00	0
2014-12-18 13:00:00	0
2014-12-18 14:00:00	0
2014-12-18 15:00:00	0
2014-12-18 16:00:00	0
2014-12-18 17:00:00	0
2014-12-18 18:00:00	0
2014-12-18 19:00:00	0
2014-12-18 20:00:00	0
2014-12-18 21:00:00	0
2014-12-18 22:00:00	0
2014-12-18 23:00:00	0
2014-12-19 00:00:00	0
2014-12-19 01:00:00	0

2014-12-19 02:00:00	0
2014-12-19 03:00:00	0
2014-12-19 04:00:00	0
2014-12-19 05:00:00	0
2014-12-19 06:00:00	21
2014-12-19 07:00:00	21
2014-12-19 08:00:00	21
2014-12-19 09:00:00	0
2014-12-19 10:00:00	0
2014-12-19 11:00:00	0
2014-12-19 12:00:00	0
2014-12-19 13:00:00	0
2014-12-19 14:00:00	0
2014-12-19 15:00:00	0
2014-12-19 16:00:00	0
2014-12-19 17:00:00	0
2014-12-19 18:00:00	0
2014-12-19 19:00:00	0
2014-12-19 20:00:00	0
2014-12-19 21:00:00	0
2014-12-19 22:00:00	0
2014-12-19 23:00:00	0
2014-12-20 00:00:00	0
2014-12-20 01:00:00	0
2014-12-20 02:00:00	0
2014-12-20 03:00:00	0
2014-12-20 04:00:00	0
2014-12-20 05:00:00	0
2014-12-20 06:00:00	0
2014-12-20 07:00:00	0
2014-12-20 08:00:00	0
2014-12-20 09:00:00	0
2014-12-20 10:00:00	0
2014-12-20 11:00:00	0
2014-12-20 12:00:00	0
2014-12-20 13:00:00	0
2014-12-20 14:00:00	0
2014-12-20 15:00:00	0
2014-12-20 16:00:00	0
2014-12-20 17:00:00	42
2014-12-20 18:00:00	42
2014-12-20 19:00:00	21
2014-12-20 20:00:00	0
2014-12-20 21:00:00	0
2014-12-20 22:00:00	0
2014-12-20 23:00:00	0
2014-12-21 00:00:00	0
2014-12-21 01:00:00	0

2014-12-21 02:00:00	0
2014-12-21 03:00:00	0
2014-12-21 04:00:00	0
2014-12-21 05:00:00	0
2014-12-21 06:00:00	0
2014-12-21 07:00:00	0
2014-12-21 08:00:00	0
2014-12-21 09:00:00	0
2014-12-21 10:00:00	0
2014-12-21 11:00:00	0
2014-12-21 12:00:00	0
2014-12-21 13:00:00	0
2014-12-21 14:00:00	0
2014-12-21 15:00:00	0
2014-12-21 16:00:00	0
2014-12-21 17:00:00	0
2014-12-21 18:00:00	0
2014-12-21 19:00:00	0
2014-12-21 20:00:00	0
2014-12-21 21:00:00	0
2014-12-21 22:00:00	0
2014-12-21 23:00:00	0
2014-12-22 00:00:00	0
2014-12-22 01:00:00	0
2014-12-22 02:00:00	0
2014-12-22 03:00:00	0
2014-12-22 04:00:00	0
2014-12-22 05:00:00	0
2014-12-22 06:00:00	0
2014-12-22 07:00:00	0
2014-12-22 08:00:00	0
2014-12-22 09:00:00	0
2014-12-22 10:00:00	0
2014-12-22 11:00:00	0
2014-12-22 12:00:00	0
2014-12-22 13:00:00	0
2014-12-22 14:00:00	0
2014-12-22 15:00:00	0
2014-12-22 16:00:00	0
2014-12-22 17:00:00	0
2014-12-22 18:00:00	0
2014-12-22 19:00:00	0
2014-12-22 20:00:00	0
2014-12-22 21:00:00	0
2014-12-22 22:00:00	0
2014-12-22 23:00:00	0
2014-12-23 00:00:00	0
2014-12-23 01:00:00	0

2014-12-23 02:00:00	0
2014-12-23 03:00:00	0
2014-12-23 04:00:00	0
2014-12-23 05:00:00	0
2014-12-23 06:00:00	0
2014-12-23 07:00:00	0
2014-12-23 08:00:00	0
2014-12-23 09:00:00	0
2014-12-23 10:00:00	0
2014-12-23 11:00:00	0
2014-12-23 12:00:00	0
2014-12-23 13:00:00	0
2014-12-23 14:00:00	0
2014-12-23 15:00:00	0
2014-12-23 16:00:00	0
2014-12-23 17:00:00	0
2014-12-23 18:00:00	0
2014-12-23 19:00:00	0
2014-12-23 20:00:00	0
2014-12-23 21:00:00	0
2014-12-23 22:00:00	0
2014-12-23 23:00:00	0
2014-12-24 00:00:00	0
2014-12-24 01:00:00	0
2014-12-24 02:00:00	0
2014-12-24 03:00:00	0
2014-12-24 04:00:00	0
2014-12-24 05:00:00	0
2014-12-24 06:00:00	0
2014-12-24 07:00:00	0
2014-12-24 08:00:00	0
2014-12-24 09:00:00	0
2014-12-24 10:00:00	0
2014-12-24 11:00:00	0
2014-12-24 12:00:00	0
2014-12-24 13:00:00	0
2014-12-24 14:00:00	0
2014-12-24 15:00:00	0
2014-12-24 16:00:00	0
2014-12-24 17:00:00	0
2014-12-24 18:00:00	0
2014-12-24 19:00:00	0
2014-12-24 20:00:00	0
2014-12-24 21:00:00	0
2014-12-24 22:00:00	0
2014-12-24 23:00:00	0
2014-12-25 00:00:00	0
2014-12-25 01:00:00	0

2014-12-25 02:00:00	0
2014-12-25 03:00:00	0
2014-12-25 04:00:00	0
2014-12-25 05:00:00	0
2014-12-25 06:00:00	0
2014-12-25 07:00:00	0
2014-12-25 08:00:00	0
2014-12-25 09:00:00	0
2014-12-25 10:00:00	0
2014-12-25 11:00:00	0
2014-12-25 12:00:00	0
2014-12-25 13:00:00	0
2014-12-25 14:00:00	0
2014-12-25 15:00:00	0
2014-12-25 16:00:00	0
2014-12-25 17:00:00	0
2014-12-25 18:00:00	0
2014-12-25 19:00:00	0
2014-12-25 20:00:00	0
2014-12-25 21:00:00	0
2014-12-25 22:00:00	0
2014-12-25 23:00:00	0
2014-12-26 00:00:00	0
2014-12-26 01:00:00	0
2014-12-26 02:00:00	0
2014-12-26 03:00:00	0
2014-12-26 04:00:00	0
2014-12-26 05:00:00	0
2014-12-26 06:00:00	0
2014-12-26 07:00:00	0
2014-12-26 08:00:00	0
2014-12-26 09:00:00	0
2014-12-26 10:00:00	0
2014-12-26 11:00:00	0
2014-12-26 12:00:00	0
2014-12-26 13:00:00	0
2014-12-26 14:00:00	0
2014-12-26 15:00:00	0
2014-12-26 16:00:00	0
2014-12-26 17:00:00	0
2014-12-26 18:00:00	0
2014-12-26 19:00:00	0
2014-12-26 20:00:00	0
2014-12-26 21:00:00	0
2014-12-26 22:00:00	0
2014-12-26 23:00:00	0
2014-12-27 00:00:00	0
2014-12-27 01:00:00	0

2014-12-27 02:00:00	0
2014-12-27 03:00:00	0
2014-12-27 04:00:00	0
2014-12-27 05:00:00	0
2014-12-27 06:00:00	0
2014-12-27 07:00:00	0
2014-12-27 08:00:00	0
2014-12-27 09:00:00	0
2014-12-27 10:00:00	0
2014-12-27 11:00:00	0
2014-12-27 12:00:00	0
2014-12-27 13:00:00	0
2014-12-27 14:00:00	0
2014-12-27 15:00:00	0
2014-12-27 16:00:00	0
2014-12-27 17:00:00	0
2014-12-27 18:00:00	0
2014-12-27 19:00:00	0
2014-12-27 20:00:00	0
2014-12-27 21:00:00	0
2014-12-27 22:00:00	0
2014-12-27 23:00:00	0
2014-12-28 00:00:00	0
2014-12-28 01:00:00	0
2014-12-28 02:00:00	0
2014-12-28 03:00:00	0
2014-12-28 04:00:00	0
2014-12-28 05:00:00	0
2014-12-28 06:00:00	0
2014-12-28 07:00:00	0
2014-12-28 08:00:00	0
2014-12-28 09:00:00	0
2014-12-28 10:00:00	0
2014-12-28 11:00:00	0
2014-12-28 12:00:00	0
2014-12-28 13:00:00	0
2014-12-28 14:00:00	0
2014-12-28 15:00:00	0
2014-12-28 16:00:00	0
2014-12-28 17:00:00	0
2014-12-28 18:00:00	0
2014-12-28 19:00:00	0
2014-12-28 20:00:00	0
2014-12-28 21:00:00	0
2014-12-28 22:00:00	0
2014-12-28 23:00:00	0
2014-12-29 00:00:00	0
2014-12-29 01:00:00	0

2014-12-29 02:00:00	0
2014-12-29 03:00:00	0
2014-12-29 04:00:00	0
2014-12-29 05:00:00	0
2014-12-29 06:00:00	0
2014-12-29 07:00:00	0
2014-12-29 08:00:00	0
2014-12-29 09:00:00	0
2014-12-29 10:00:00	0
2014-12-29 11:00:00	0
2014-12-29 12:00:00	0
2014-12-29 13:00:00	0
2014-12-29 14:00:00	0
2014-12-29 15:00:00	0
2014-12-29 16:00:00	0
2014-12-29 17:00:00	0
2014-12-29 18:00:00	0
2014-12-29 19:00:00	0
2014-12-29 20:00:00	0
2014-12-29 21:00:00	0
2014-12-29 22:00:00	0
2014-12-29 23:00:00	0
2014-12-30 00:00:00	0
2014-12-30 01:00:00	0
2014-12-30 02:00:00	0
2014-12-30 03:00:00	0
2014-12-30 04:00:00	0
2014-12-30 05:00:00	0
2014-12-30 06:00:00	0
2014-12-30 07:00:00	0
2014-12-30 08:00:00	0
2014-12-30 09:00:00	0
2014-12-30 10:00:00	0
2014-12-30 11:00:00	0
2014-12-30 12:00:00	63
2014-12-30 13:00:00	42
2014-12-30 14:00:00	21
2014-12-30 15:00:00	0
2014-12-30 16:00:00	0
2014-12-30 17:00:00	0
2014-12-30 18:00:00	0
2014-12-30 19:00:00	0
2014-12-30 20:00:00	0
2014-12-30 21:00:00	0
2014-12-30 22:00:00	0
2014-12-30 23:00:00	0
2014-12-31 00:00:00	0
2014-12-31 01:00:00	0

2014-12-31 02:00:00	0
2014-12-31 03:00:00	0
2014-12-31 04:00:00	0
2014-12-31 05:00:00	0
2014-12-31 06:00:00	0
2014-12-31 07:00:00	0
2014-12-31 08:00:00	0
2014-12-31 09:00:00	0
2014-12-31 10:00:00	0
2014-12-31 11:00:00	0
2014-12-31 12:00:00	0
2014-12-31 13:00:00	0
2014-12-31 14:00:00	0
2014-12-31 15:00:00	0
2014-12-31 16:00:00	0
2014-12-31 17:00:00	0
2014-12-31 18:00:00	0
2014-12-31 19:00:00	0
2014-12-31 20:00:00	0
2014-12-31 21:00:00	0
2014-12-31 22:00:00	0
2014-12-31 23:00:00	0
2015-01-01 00:00:00	0
2015-01-01 01:00:00	0
2015-01-01 02:00:00	0
2015-01-01 03:00:00	0
2015-01-01 04:00:00	0
2015-01-01 05:00:00	0
2015-01-01 06:00:00	0
2015-01-01 07:00:00	0
2015-01-01 08:00:00	0
2015-01-01 09:00:00	0
2015-01-01 10:00:00	0
2015-01-01 11:00:00	0
2015-01-01 12:00:00	0
2015-01-01 13:00:00	0
2015-01-01 14:00:00	0
2015-01-01 15:00:00	0
2015-01-01 16:00:00	0
2015-01-01 17:00:00	0
2015-01-01 18:00:00	0
2015-01-01 19:00:00	0
2015-01-01 20:00:00	0
2015-01-01 21:00:00	0
2015-01-01 22:00:00	0
2015-01-01 23:00:00	0
2015-01-02 00:00:00	0
2015-01-02 01:00:00	0

2015-01-02 02:00:00	0
2015-01-02 03:00:00	0
2015-01-02 04:00:00	0
2015-01-02 05:00:00	0
2015-01-02 06:00:00	0
2015-01-02 07:00:00	0
2015-01-02 08:00:00	0
2015-01-02 09:00:00	0
2015-01-02 10:00:00	0
2015-01-02 11:00:00	0
2015-01-02 12:00:00	0
2015-01-02 13:00:00	0
2015-01-02 14:00:00	0
2015-01-02 15:00:00	0
2015-01-02 16:00:00	0
2015-01-02 17:00:00	0
2015-01-02 18:00:00	0
2015-01-02 19:00:00	0
2015-01-02 20:00:00	0
2015-01-02 21:00:00	0
2015-01-02 22:00:00	0
2015-01-02 23:00:00	0
2015-01-03 00:00:00	0
2015-01-03 01:00:00	0
2015-01-03 02:00:00	0
2015-01-03 03:00:00	0
2015-01-03 04:00:00	0
2015-01-03 05:00:00	0
2015-01-03 06:00:00	0
2015-01-03 07:00:00	0
2015-01-03 08:00:00	0
2015-01-03 09:00:00	0
2015-01-03 10:00:00	0
2015-01-03 11:00:00	0
2015-01-03 12:00:00	0
2015-01-03 13:00:00	0
2015-01-03 14:00:00	0
2015-01-03 15:00:00	0
2015-01-03 16:00:00	0
2015-01-03 17:00:00	0
2015-01-03 18:00:00	0
2015-01-03 19:00:00	0
2015-01-03 20:00:00	0
2015-01-03 21:00:00	0
2015-01-03 22:00:00	0
2015-01-03 23:00:00	0
2015-01-04 00:00:00	0
2015-01-04 01:00:00	0

2015-01-04 02:00:00	0
2015-01-04 03:00:00	0
2015-01-04 04:00:00	0
2015-01-04 05:00:00	0
2015-01-04 06:00:00	0
2015-01-04 07:00:00	0
2015-01-04 08:00:00	0
2015-01-04 09:00:00	0
2015-01-04 10:00:00	0
2015-01-04 11:00:00	0
2015-01-04 12:00:00	0
2015-01-04 13:00:00	0
2015-01-04 14:00:00	0
2015-01-04 15:00:00	0
2015-01-04 16:00:00	0
2015-01-04 17:00:00	0
2015-01-04 18:00:00	0
2015-01-04 19:00:00	0
2015-01-04 20:00:00	0
2015-01-04 21:00:00	0
2015-01-04 22:00:00	0
2015-01-04 23:00:00	0
2015-01-05 00:00:00	0
2015-01-05 01:00:00	0
2015-01-05 02:00:00	0
2015-01-05 03:00:00	0
2015-01-05 04:00:00	0
2015-01-05 05:00:00	0
2015-01-05 06:00:00	0
2015-01-05 07:00:00	0
2015-01-05 08:00:00	0
2015-01-05 09:00:00	0
2015-01-05 10:00:00	0
2015-01-05 11:00:00	0
2015-01-05 12:00:00	0
2015-01-05 13:00:00	0
2015-01-05 14:00:00	0
2015-01-05 15:00:00	0
2015-01-05 16:00:00	0
2015-01-05 17:00:00	0
2015-01-05 18:00:00	0
2015-01-05 19:00:00	0
2015-01-05 20:00:00	0
2015-01-05 21:00:00	0
2015-01-05 22:00:00	0
2015-01-05 23:00:00	0
2015-01-06 00:00:00	0
2015-01-06 01:00:00	0

2015-01-06 02:00:00	0
2015-01-06 03:00:00	0
2015-01-06 04:00:00	0
2015-01-06 05:00:00	0
2015-01-06 06:00:00	0
2015-01-06 07:00:00	0
2015-01-06 08:00:00	0
2015-01-06 09:00:00	0
2015-01-06 10:00:00	0
2015-01-06 11:00:00	0
2015-01-06 12:00:00	0
2015-01-06 13:00:00	0
2015-01-06 14:00:00	0
2015-01-06 15:00:00	0
2015-01-06 16:00:00	0
2015-01-06 17:00:00	0
2015-01-06 18:00:00	0
2015-01-06 19:00:00	0
2015-01-06 20:00:00	0
2015-01-06 21:00:00	0
2015-01-06 22:00:00	0
2015-01-06 23:00:00	0
2015-01-07 00:00:00	0
2015-01-07 01:00:00	0
2015-01-07 02:00:00	0
2015-01-07 03:00:00	0
2015-01-07 04:00:00	0
2015-01-07 05:00:00	0
2015-01-07 06:00:00	0
2015-01-07 07:00:00	0
2015-01-07 08:00:00	0
2015-01-07 09:00:00	0
2015-01-07 10:00:00	0
2015-01-07 11:00:00	0
2015-01-07 12:00:00	0
2015-01-07 13:00:00	0
2015-01-07 14:00:00	0
2015-01-07 15:00:00	0
2015-01-07 16:00:00	0
2015-01-07 17:00:00	0
2015-01-07 18:00:00	0
2015-01-07 19:00:00	0
2015-01-07 20:00:00	0
2015-01-07 21:00:00	0
2015-01-07 22:00:00	0
2015-01-07 23:00:00	0
2015-01-08 00:00:00	0
2015-01-08 01:00:00	0

2015-01-08 02:00:00	0
2015-01-08 03:00:00	0
2015-01-08 04:00:00	0
2015-01-08 05:00:00	0
2015-01-08 06:00:00	0
2015-01-08 07:00:00	0
2015-01-08 08:00:00	0
2015-01-08 09:00:00	0
2015-01-08 10:00:00	0
2015-01-08 11:00:00	0
2015-01-08 12:00:00	0
2015-01-08 13:00:00	0
2015-01-08 14:00:00	0
2015-01-08 15:00:00	0
2015-01-08 16:00:00	0
2015-01-08 17:00:00	0
2015-01-08 18:00:00	0
2015-01-08 19:00:00	0
2015-01-08 20:00:00	0
2015-01-08 21:00:00	0
2015-01-08 22:00:00	0
2015-01-08 23:00:00	0
2015-01-09 00:00:00	0
2015-01-09 01:00:00	0
2015-01-09 02:00:00	0
2015-01-09 03:00:00	0
2015-01-09 04:00:00	0
2015-01-09 05:00:00	0
2015-01-09 06:00:00	0
2015-01-09 07:00:00	0
2015-01-09 08:00:00	0
2015-01-09 09:00:00	0
2015-01-09 10:00:00	0
2015-01-09 11:00:00	0
2015-01-09 12:00:00	0
2015-01-09 13:00:00	0
2015-01-09 14:00:00	0
2015-01-09 15:00:00	0
2015-01-09 16:00:00	0
2015-01-09 17:00:00	0
2015-01-09 18:00:00	0
2015-01-09 19:00:00	0
2015-01-09 20:00:00	0
2015-01-09 21:00:00	0
2015-01-09 22:00:00	0
2015-01-09 23:00:00	0
2015-01-10 00:00:00	0
2015-01-10 01:00:00	0

2015-01-10 02:00:00	0
2015-01-10 03:00:00	0
2015-01-10 04:00:00	0
2015-01-10 05:00:00	0
2015-01-10 06:00:00	0
2015-01-10 07:00:00	0
2015-01-10 08:00:00	0
2015-01-10 09:00:00	0
2015-01-10 10:00:00	0
2015-01-10 11:00:00	0
2015-01-10 12:00:00	0
2015-01-10 13:00:00	0
2015-01-10 14:00:00	0
2015-01-10 15:00:00	0
2015-01-10 16:00:00	0
2015-01-10 17:00:00	0
2015-01-10 18:00:00	0
2015-01-10 19:00:00	0
2015-01-10 20:00:00	0
2015-01-10 21:00:00	0
2015-01-10 22:00:00	0
2015-01-10 23:00:00	0
2015-01-11 00:00:00	0
2015-01-11 01:00:00	0
2015-01-11 02:00:00	0
2015-01-11 03:00:00	0
2015-01-11 04:00:00	0
2015-01-11 05:00:00	0
2015-01-11 06:00:00	0
2015-01-11 07:00:00	0
2015-01-11 08:00:00	0
2015-01-11 09:00:00	0
2015-01-11 10:00:00	0
2015-01-11 11:00:00	0
2015-01-11 12:00:00	0
2015-01-11 13:00:00	0
2015-01-11 14:00:00	0
2015-01-11 15:00:00	0
2015-01-11 16:00:00	0
2015-01-11 17:00:00	0
2015-01-11 18:00:00	0
2015-01-11 19:00:00	0
2015-01-11 20:00:00	0
2015-01-11 21:00:00	0
2015-01-11 22:00:00	0
2015-01-11 23:00:00	0
2015-01-12 00:00:00	0
2015-01-12 01:00:00	0

2015-01-12 02:00:00	0
2015-01-12 03:00:00	0
2015-01-12 04:00:00	0
2015-01-12 05:00:00	0
2015-01-12 06:00:00	0
2015-01-12 07:00:00	0
2015-01-12 08:00:00	0
2015-01-12 09:00:00	0
2015-01-12 10:00:00	0
2015-01-12 11:00:00	0
2015-01-12 12:00:00	0
2015-01-12 13:00:00	0
2015-01-12 14:00:00	0
2015-01-12 15:00:00	0
2015-01-12 16:00:00	0
2015-01-12 17:00:00	0
2015-01-12 18:00:00	0
2015-01-12 19:00:00	0
2015-01-12 20:00:00	0
2015-01-12 21:00:00	0
2015-01-12 22:00:00	0
2015-01-12 23:00:00	0
2015-01-13 00:00:00	0
2015-01-13 01:00:00	0
2015-01-13 02:00:00	0
2015-01-13 03:00:00	0
2015-01-13 04:00:00	0
2015-01-13 05:00:00	0
2015-01-13 06:00:00	0
2015-01-13 07:00:00	0
2015-01-13 08:00:00	0
2015-01-13 09:00:00	0
2015-01-13 10:00:00	0
2015-01-13 11:00:00	0
2015-01-13 12:00:00	0
2015-01-13 13:00:00	0
2015-01-13 14:00:00	0
2015-01-13 15:00:00	0
2015-01-13 16:00:00	0
2015-01-13 17:00:00	0
2015-01-13 18:00:00	0
2015-01-13 19:00:00	0
2015-01-13 20:00:00	0
2015-01-13 21:00:00	0
2015-01-13 22:00:00	0
2015-01-13 23:00:00	0
2015-01-14 00:00:00	0
2015-01-14 01:00:00	0

2015-01-14 02:00:00	0
2015-01-14 03:00:00	0
2015-01-14 04:00:00	0
2015-01-14 05:00:00	0
2015-01-14 06:00:00	0
2015-01-14 07:00:00	0
2015-01-14 08:00:00	0
2015-01-14 09:00:00	0
2015-01-14 10:00:00	0
2015-01-14 11:00:00	0
2015-01-14 12:00:00	0
2015-01-14 13:00:00	0
2015-01-14 14:00:00	0
2015-01-14 15:00:00	0
2015-01-14 16:00:00	0
2015-01-14 17:00:00	0
2015-01-14 18:00:00	0
2015-01-14 19:00:00	0
2015-01-14 20:00:00	0
2015-01-14 21:00:00	0
2015-01-14 22:00:00	0
2015-01-14 23:00:00	0
2015-01-15 00:00:00	0
2015-01-15 01:00:00	0
2015-01-15 02:00:00	0
2015-01-15 03:00:00	0
2015-01-15 04:00:00	0
2015-01-15 05:00:00	0
2015-01-15 06:00:00	0
2015-01-15 07:00:00	0
2015-01-15 08:00:00	0
2015-01-15 09:00:00	0
2015-01-15 10:00:00	0
2015-01-15 11:00:00	0
2015-01-15 12:00:00	0
2015-01-15 13:00:00	0
2015-01-15 14:00:00	0
2015-01-15 15:00:00	0
2015-01-15 16:00:00	0
2015-01-15 17:00:00	0
2015-01-15 18:00:00	0
2015-01-15 19:00:00	0
2015-01-15 20:00:00	0
2015-01-15 21:00:00	0
2015-01-15 22:00:00	0
2015-01-15 23:00:00	0
2015-01-16 00:00:00	0
2015-01-16 01:00:00	0

2015-01-16 02:00:00	0
2015-01-16 03:00:00	0
2015-01-16 04:00:00	0
2015-01-16 05:00:00	0
2015-01-16 06:00:00	0
2015-01-16 07:00:00	0
2015-01-16 08:00:00	0
2015-01-16 09:00:00	0
2015-01-16 10:00:00	0
2015-01-16 11:00:00	0
2015-01-16 12:00:00	0
2015-01-16 13:00:00	0
2015-01-16 14:00:00	0
2015-01-16 15:00:00	0
2015-01-16 16:00:00	0
2015-01-16 17:00:00	0
2015-01-16 18:00:00	0
2015-01-16 19:00:00	0
2015-01-16 20:00:00	0
2015-01-16 21:00:00	0
2015-01-16 22:00:00	0
2015-01-16 23:00:00	0
2015-01-17 00:00:00	0
2015-01-17 01:00:00	0
2015-01-17 02:00:00	0
2015-01-17 03:00:00	0
2015-01-17 04:00:00	0
2015-01-17 05:00:00	0
2015-01-17 06:00:00	0
2015-01-17 07:00:00	0
2015-01-17 08:00:00	0
2015-01-17 09:00:00	0
2015-01-17 10:00:00	0
2015-01-17 11:00:00	0
2015-01-17 12:00:00	0
2015-01-17 13:00:00	0
2015-01-17 14:00:00	0
2015-01-17 15:00:00	0
2015-01-17 16:00:00	0
2015-01-17 17:00:00	0
2015-01-17 18:00:00	0
2015-01-17 19:00:00	0
2015-01-17 20:00:00	0
2015-01-17 21:00:00	0
2015-01-17 22:00:00	0
2015-01-17 23:00:00	0
2015-01-18 00:00:00	0
2015-01-18 01:00:00	0

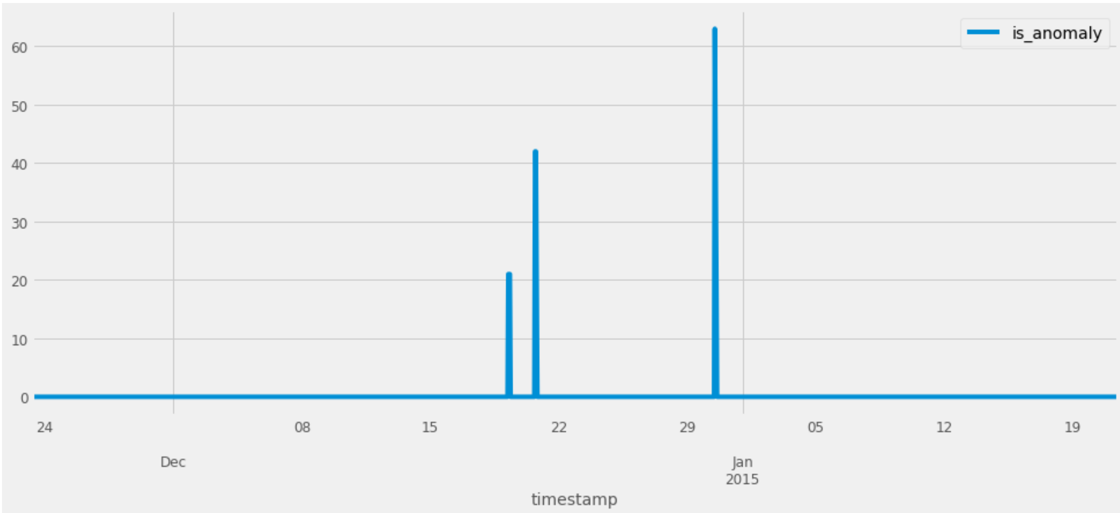
2015-01-18 02:00:00	0
2015-01-18 03:00:00	0
2015-01-18 04:00:00	0
2015-01-18 05:00:00	0
2015-01-18 06:00:00	0
2015-01-18 07:00:00	0
2015-01-18 08:00:00	0
2015-01-18 09:00:00	0
2015-01-18 10:00:00	0
2015-01-18 11:00:00	0
2015-01-18 12:00:00	0
2015-01-18 13:00:00	0
2015-01-18 14:00:00	0
2015-01-18 15:00:00	0
2015-01-18 16:00:00	0
2015-01-18 17:00:00	0
2015-01-18 18:00:00	0
2015-01-18 19:00:00	0
2015-01-18 20:00:00	0
2015-01-18 21:00:00	0
2015-01-18 22:00:00	0
2015-01-18 23:00:00	0
2015-01-19 00:00:00	0
2015-01-19 01:00:00	0
2015-01-19 02:00:00	0
2015-01-19 03:00:00	0
2015-01-19 04:00:00	0
2015-01-19 05:00:00	0
2015-01-19 06:00:00	0
2015-01-19 07:00:00	0
2015-01-19 08:00:00	0
2015-01-19 09:00:00	0
2015-01-19 10:00:00	0
2015-01-19 11:00:00	0
2015-01-19 12:00:00	0
2015-01-19 13:00:00	0
2015-01-19 14:00:00	0
2015-01-19 15:00:00	0
2015-01-19 16:00:00	0
2015-01-19 17:00:00	0
2015-01-19 18:00:00	0
2015-01-19 19:00:00	0
2015-01-19 20:00:00	0
2015-01-19 21:00:00	0
2015-01-19 22:00:00	0
2015-01-19 23:00:00	0
2015-01-20 00:00:00	0
2015-01-20 01:00:00	0

```
2015-01-20 02:00:00          0
2015-01-20 03:00:00          0
2015-01-20 04:00:00          0
2015-01-20 05:00:00          0
2015-01-20 06:00:00          0
2015-01-20 07:00:00          0
2015-01-20 08:00:00          0
2015-01-20 09:00:00          0
2015-01-20 10:00:00          0
2015-01-20 11:00:00          0
2015-01-20 12:00:00          0
2015-01-20 13:00:00          0
2015-01-20 14:00:00          0
2015-01-20 15:00:00          0
2015-01-20 16:00:00          0
2015-01-20 17:00:00          0
2015-01-20 18:00:00          0
2015-01-20 19:00:00          0
2015-01-20 20:00:00          0
2015-01-20 21:00:00          0
2015-01-20 22:00:00          0
2015-01-20 23:00:00          0
2015-01-21 00:00:00          0
2015-01-21 01:00:00          0
2015-01-21 02:00:00          0
2015-01-21 03:00:00          0
2015-01-21 04:00:00          0
2015-01-21 05:00:00          0
2015-01-21 06:00:00          0
2015-01-21 07:00:00          0
2015-01-21 08:00:00          0
2015-01-21 09:00:00          0
2015-01-21 10:00:00          0
2015-01-21 11:00:00          0
```

16 Anomalies Present during which time-frame?

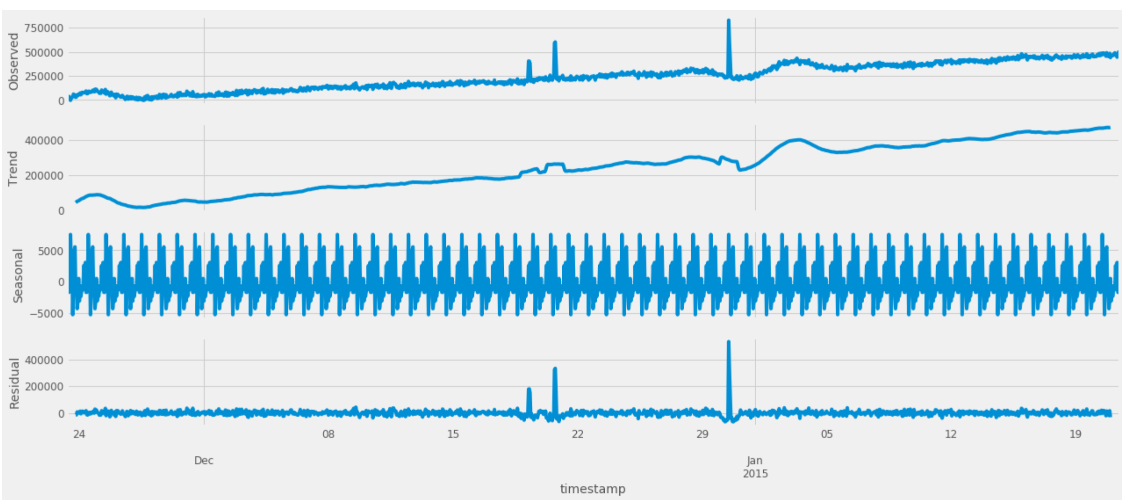
16.0.1 As we can see most of the anomalies are present during End of December 2014 and Starting of Jan 2015

[19]:
```
dfano.plot(figsize=(15, 6))
plt.show()
```



17 Plotting Trends and Seasonality of Data.

[20]:
```
from pylab import rcParams
rcParams['figure.figsize'] = 18, 8
decomposition = sm.tsa.seasonal_decompose(y, model='additive')
fig = decomposition.plot()
plt.show()
```



18 Examples of parameter combinations for Seasonal ARIMA…

[21]:
```
p = d = q = range(0, 2)
pdq = list(itertools.product(p, d, q))
seasonal_pdq = [(x[0], x[1], x[2], 12) for x in list(itertools.product(p, d,␣
↪q))]
print('Examples of parameter combinations for Seasonal ARIMA...')
print('SARIMAX: {} x {}'.format(pdq[1], seasonal_pdq[1]))
print('SARIMAX: {} x {}'.format(pdq[1], seasonal_pdq[2]))
print('SARIMAX: {} x {}'.format(pdq[2], seasonal_pdq[3]))
print('SARIMAX: {} x {}'.format(pdq[2], seasonal_pdq[4]))
```

```
Examples of parameter combinations for Seasonal ARIMA…
SARIMAX: (0, 0, 1) x (0, 0, 1, 12)
SARIMAX: (0, 0, 1) x (0, 1, 0, 12)
SARIMAX: (0, 1, 0) x (0, 1, 1, 12)
SARIMAX: (0, 1, 0) x (1, 0, 0, 12)
```

19 Finding the best parameters of ARIMA

19.0.1 The best param so far is ARIMA(1, 1, 1)x(1, 1, 1, 12)12 - AIC:32764.2410727543

So we will use this for the forecasting.

[22]:
```
for param in pdq:
    for param_seasonal in seasonal_pdq:
        try:
            mod = sm.tsa.statespace.SARIMAX(y, order=param,␣
↪seasonal_order=param_seasonal,enforce_stationarity=False,␣
↪enforce_invertibility=False)
            results = mod.fit()
            print('ARIMA{}x{}12 - AIC:{}'.format(param, param_seasonal, results.
↪aic))
        except:
            continue
```

```
ARIMA(0, 0, 0)x(0, 0, 0, 12)12 - AIC:39612.5152726619
ARIMA(0, 0, 0)x(0, 0, 1, 12)12 - AIC:38354.784205169606
ARIMA(0, 0, 0)x(0, 1, 0, 12)12 - AIC:34039.61026229999
ARIMA(0, 0, 0)x(0, 1, 1, 12)12 - AIC:33404.16622608983
ARIMA(0, 0, 0)x(1, 0, 0, 12)12 - AIC:34064.844434927276
ARIMA(0, 0, 0)x(1, 0, 1, 12)12 - AIC:33648.57707022065
ARIMA(0, 0, 0)x(1, 1, 0, 12)12 - AIC:33517.29611458641
ARIMA(0, 0, 0)x(1, 1, 1, 12)12 - AIC:33404.116881098125
ARIMA(0, 0, 1)x(0, 0, 0, 12)12 - AIC:38587.78425393491
ARIMA(0, 0, 1)x(0, 0, 1, 12)12 - AIC:38030.6316661478
ARIMA(0, 0, 1)x(0, 1, 0, 12)12 - AIC:33695.03100454179
```

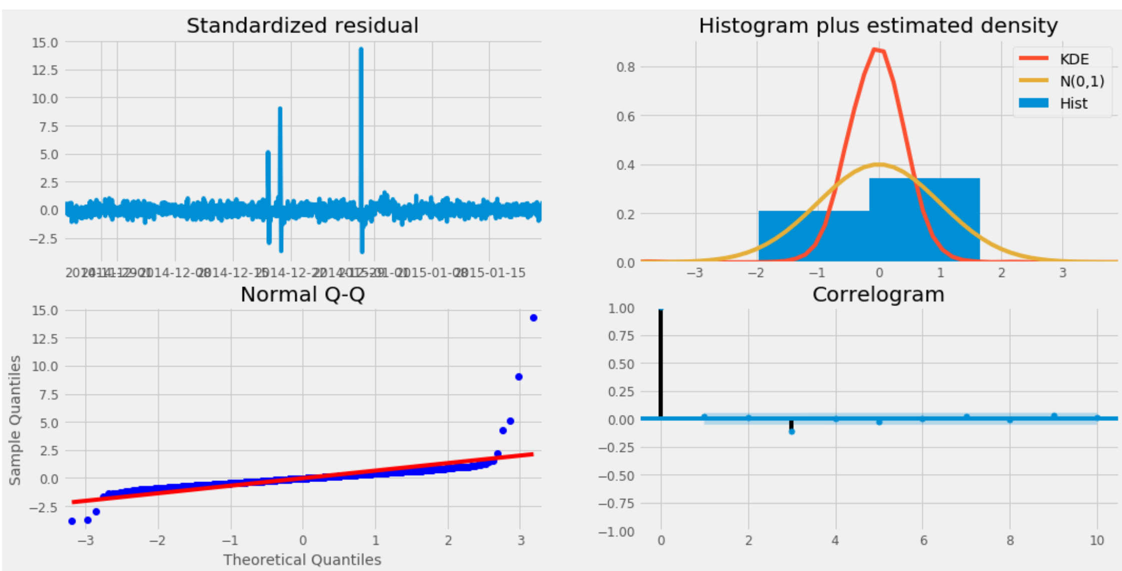
```
ARIMA(0, 0, 1)x(0, 1, 1, 12)12 - AIC:33017.59089490472
ARIMA(0, 0, 1)x(1, 0, 0, 12)12 - AIC:37922.07594313397
ARIMA(0, 0, 1)x(1, 0, 1, 12)12 - AIC:37861.67200363868
ARIMA(0, 0, 1)x(1, 1, 0, 12)12 - AIC:33185.440944489266
ARIMA(0, 0, 1)x(1, 1, 1, 12)12 - AIC:33019.05048900604
ARIMA(0, 1, 0)x(0, 0, 0, 12)12 - AIC:33276.411290365235
ARIMA(0, 1, 0)x(0, 0, 1, 12)12 - AIC:33005.11577277919
ARIMA(0, 1, 0)x(0, 1, 0, 12)12 - AIC:33996.05808845328
ARIMA(0, 1, 0)x(0, 1, 1, 12)12 - AIC:32818.74902801201
ARIMA(0, 1, 0)x(1, 0, 0, 12)12 - AIC:33027.46721478706
ARIMA(0, 1, 0)x(1, 0, 1, 12)12 - AIC:33006.92174647196
ARIMA(0, 1, 0)x(1, 1, 0, 12)12 - AIC:33352.1394531121
ARIMA(0, 1, 0)x(1, 1, 1, 12)12 - AIC:32810.143890973384
ARIMA(0, 1, 1)x(0, 0, 0, 12)12 - AIC:33130.62726999829
ARIMA(0, 1, 1)x(0, 0, 1, 12)12 - AIC:32859.39103415761
ARIMA(0, 1, 1)x(0, 1, 0, 12)12 - AIC:33838.79708978505
ARIMA(0, 1, 1)x(0, 1, 1, 12)12 - AIC:32868.93012424608
ARIMA(0, 1, 1)x(1, 0, 0, 12)12 - AIC:32904.68818270531
ARIMA(0, 1, 1)x(1, 0, 1, 12)12 - AIC:32861.34971795847
ARIMA(0, 1, 1)x(1, 1, 0, 12)12 - AIC:33272.831803409645
ARIMA(0, 1, 1)x(1, 1, 1, 12)12 - AIC:32870.876353016574
ARIMA(1, 0, 0)x(0, 0, 0, 12)12 - AIC:33298.20002867648
ARIMA(1, 0, 0)x(0, 0, 1, 12)12 - AIC:33026.81746057302
ARIMA(1, 0, 0)x(0, 1, 0, 12)12 - AIC:33623.20236938847
ARIMA(1, 0, 0)x(0, 1, 1, 12)12 - AIC:32809.12307499927
ARIMA(1, 0, 0)x(1, 0, 0, 12)12 - AIC:33026.82520356486
ARIMA(1, 0, 0)x(1, 0, 1, 12)12 - AIC:32894.62545758141
ARIMA(1, 0, 0)x(1, 1, 0, 12)12 - AIC:33036.26647557706
ARIMA(1, 0, 0)x(1, 1, 1, 12)12 - AIC:32810.505603139776
ARIMA(1, 0, 1)x(0, 0, 0, 12)12 - AIC:33155.44214440265
ARIMA(1, 0, 1)x(0, 0, 1, 12)12 - AIC:32884.43063269574
ARIMA(1, 0, 1)x(0, 1, 0, 12)12 - AIC:33601.764900618626
ARIMA(1, 0, 1)x(0, 1, 1, 12)12 - AIC:32771.03913499564
ARIMA(1, 0, 1)x(1, 0, 0, 12)12 - AIC:32906.822279669395
ARIMA(1, 0, 1)x(1, 0, 1, 12)12 - AIC:32884.958336444935
ARIMA(1, 0, 1)x(1, 1, 0, 12)12 - AIC:33038.71739879722
ARIMA(1, 0, 1)x(1, 1, 1, 12)12 - AIC:32771.999300899864
ARIMA(1, 1, 0)x(0, 0, 0, 12)12 - AIC:33205.200745675626
ARIMA(1, 1, 0)x(0, 0, 1, 12)12 - AIC:32934.01190252885
ARIMA(1, 1, 0)x(0, 1, 0, 12)12 - AIC:33917.1146429276
ARIMA(1, 1, 0)x(0, 1, 1, 12)12 - AIC:32938.3206752635
ARIMA(1, 1, 0)x(1, 0, 0, 12)12 - AIC:32934.016068631485
ARIMA(1, 1, 0)x(1, 0, 1, 12)12 - AIC:32935.913662771265
ARIMA(1, 1, 0)x(1, 1, 0, 12)12 - AIC:33302.57489093023
ARIMA(1, 1, 0)x(1, 1, 1, 12)12 - AIC:32940.23670961004
ARIMA(1, 1, 1)x(0, 0, 0, 12)12 - AIC:32954.377813125386
ARIMA(1, 1, 1)x(0, 0, 1, 12)12 - AIC:32684.02735905658
ARIMA(1, 1, 1)x(0, 1, 0, 12)12 - AIC:33598.57556913822
```

```
ARIMA(1, 1, 1)x(0, 1, 1, 12)12 - AIC:32762.245678863976
ARIMA(1, 1, 1)x(1, 0, 0, 12)12 - AIC:32707.111385336702
ARIMA(1, 1, 1)x(1, 0, 1, 12)12 - AIC:32684.992984635588
ARIMA(1, 1, 1)x(1, 1, 0, 12)12 - AIC:33079.379806898534
ARIMA(1, 1, 1)x(1, 1, 1, 12)12 - AIC:32764.2410727543
```

[25]:
```
mod = sm.tsa.statespace.SARIMAX(y,
                                order=(1, 1, 1),
                                seasonal_order=(1, 1, 1, 12),
                                enforce_stationarity=False,
                                enforce_invertibility=False)
results = mod.fit()
print(results.summary().tables[1])
```

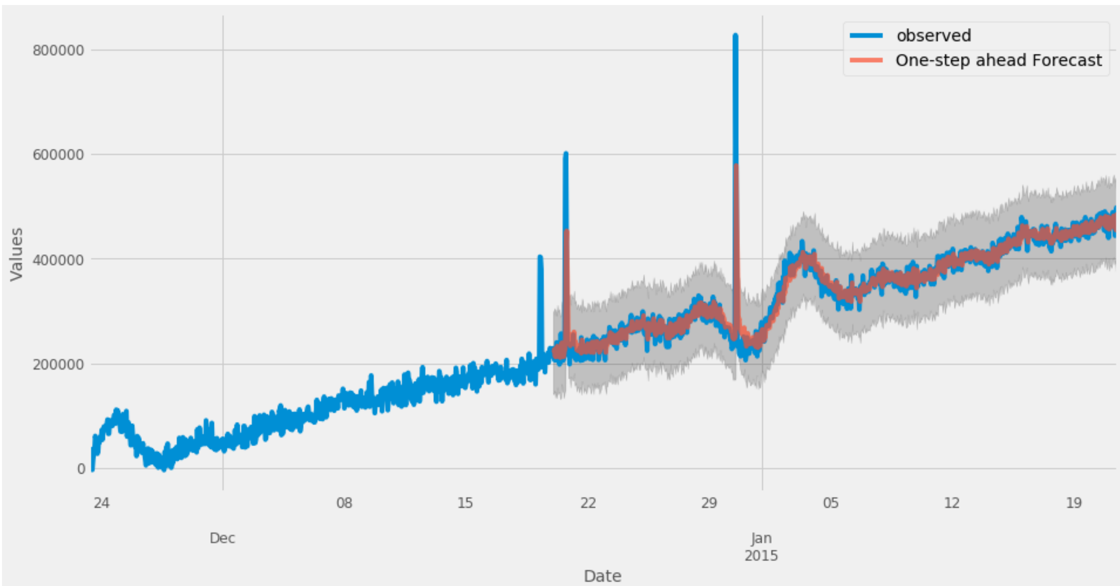
```
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.4729      0.032     15.011      0.000       0.411       0.535
ma.L1         -0.9146      0.017    -52.767      0.000      -0.949      -0.881
ar.S.L12      -0.0028      0.042     -0.065      0.948      -0.086       0.081
ma.S.L12      -0.9904      0.013    -74.710      0.000      -1.016      -0.964
sigma2       1.62e+09   4.15e-11     3.9e+19      0.000    1.62e+09    1.62e+09
==============================================================================
```

[26]:
```
results.plot_diagnostics(figsize=(16, 8))
plt.show()
```



[27]:
```
pred = results.get_prediction(start=pd.to_datetime('2014-12-20'), dynamic=False)
pred_ci = pred.conf_int()
```

```
ax = y['2014':].plot(label='observed')
pred.predicted_mean.plot(ax=ax, label='One-step ahead Forecast', alpha=.7,␣
 ↪figsize=(14, 7))
ax.fill_between(pred_ci.index,
                pred_ci.iloc[:, 0],
                pred_ci.iloc[:, 1], color='k', alpha=.2)
ax.set_xlabel('Date')
ax.set_ylabel('Values')
plt.legend()
plt.show()
```




```
[ ]:
```