

M01_Xarxa_walkForward2

December 21, 2019

1 Xarxa neuronal

```
In [1]: import pandas as pd
import numpy as np
from pandas import datetime
from matplotlib import pyplot as plt

import keras
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import LSTM

from keras.optimizers import SGD
from sklearn.model_selection import StratifiedKFold
from scipy.stats import uniform as sp_rand
from scipy.stats import randint
from time import time
from sklearn import preprocessing

import math
from sklearn.metrics import mean_squared_error
```

Using TensorFlow backend.

1.1 Consum diari total univariate one-step

```
In [2]: daily=pd.read_csv('C:/Users/Laura/Desktop/Smart meters London/workspace R/Dades netes/1')
daily.head(5)
```

```
Out[2]:
```

	date	apparentTemperatureMax	sunsetTimeHour	weekday	season	\
0	2013-01-16	-0.15	16	3	winter	
1	2013-01-20	-0.46	16	7	winter	
2	2013-01-10	2.36	16	4	winter	
3	2013-01-06	6.98	16	7	winter	
4	2012-01-31	1.13	16	2	winter	

cloudCover humidity visibility month energy_sum

0	0.48	0.91	4.12	1	13.147536
1	0.85	0.91	5.10	1	15.021900
2	0.70	0.94	5.21	1	12.066789
3	0.67	0.96	5.50	1	12.422263
4	0.55	0.84	5.62	1	13.890518

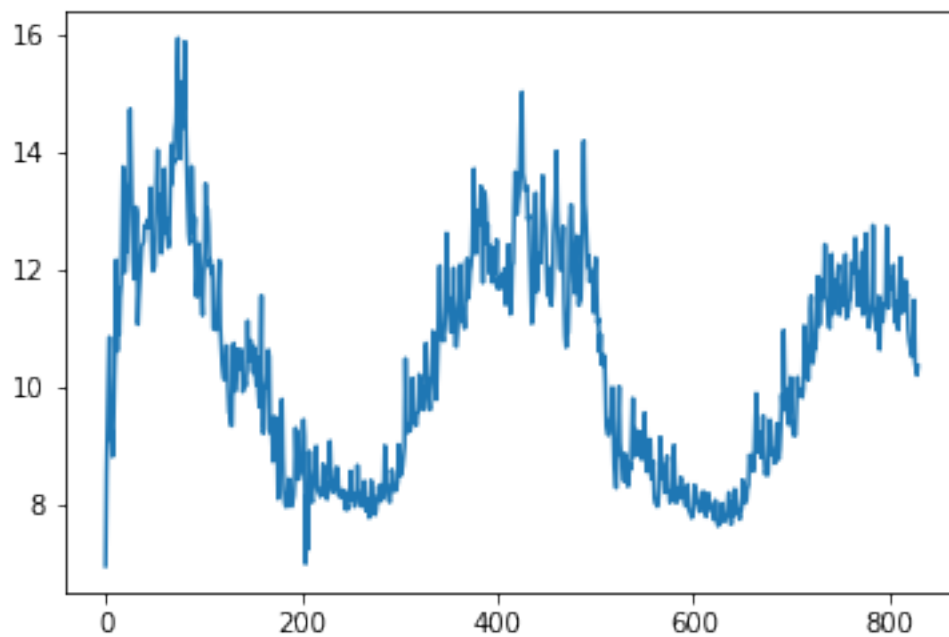
```
In [3]: #Ens quedem amb date i energy_sum, ordenem valors per data i resetejem index
daily_dia=daily[['date','energy_sum']].sort_values(by=['date']).reset_index()
daily_dia.head(5)
```

```
Out[3]:
```

	index	date	energy_sum
0	677	2011-11-23	6.952692
1	691	2011-11-24	8.536480
2	713	2011-11-25	9.499781
3	728	2011-11-26	10.267707
4	729	2011-11-27	10.850805

```
In [16]: plt.plot(daily_dia )
```

```
Out[16]: [<matplotlib.lines.Line2D at 0x24f9e752240>]
```



```
In [4]: daily_dia['t-1']=daily_dia['energy_sum'].shift(1)
daily_dia['t-2']=daily_dia['energy_sum'].shift(2)
daily_dia['t-3']=daily_dia['energy_sum'].shift(3)
daily_dia['t-4']=daily_dia['energy_sum'].shift(4)
daily_dia['t-5']=daily_dia['energy_sum'].shift(5)
```

```
daily_dia['t-6']=daily_dia['energy_sum'].shift(6)
daily_dia['t-7']=daily_dia['energy_sum'].shift(7)
```

```
daily_dia
```

```
Out[4]:
```

	index	date	energy_sum	t-1	t-2	t-3	\
0	677	2011-11-23	6.952692	NaN	NaN	NaN	
1	691	2011-11-24	8.536480	6.952692	NaN	NaN	
2	713	2011-11-25	9.499781	8.536480	6.952692	NaN	
3	728	2011-11-26	10.267707	9.499781	8.536480	6.952692	
4	729	2011-11-27	10.850805	10.267707	9.499781	8.536480	
5	704	2011-11-28	9.103382	10.850805	10.267707	9.499781	
6	718	2011-11-29	9.274873	9.103382	10.850805	10.267707	
7	727	2011-11-30	8.813513	9.274873	9.103382	10.850805	
8	778	2011-12-01	9.227707	8.813513	9.274873	9.103382	
9	773	2011-12-02	10.145910	9.227707	8.813513	9.274873	
10	791	2011-12-03	10.780273	10.145910	9.227707	8.813513	
11	822	2011-12-04	12.163127	10.780273	10.145910	9.227707	
12	807	2011-12-05	10.609714	12.163127	10.780273	10.145910	
13	813	2011-12-06	11.673417	10.609714	12.163127	10.780273	
14	810	2011-12-07	10.889362	11.673417	10.609714	12.163127	
15	788	2011-12-08	11.525150	10.889362	11.673417	10.609714	
16	797	2011-12-09	11.759837	11.525150	10.889362	11.673417	
17	799	2011-12-10	12.633801	11.759837	11.525150	10.889362	
18	776	2011-12-11	13.749174	12.633801	11.759837	11.525150	
19	775	2011-12-12	11.951958	13.749174	12.633801	11.759837	
20	786	2011-12-13	11.957446	11.951958	13.749174	12.633801	
21	818	2011-12-14	12.392776	11.957446	11.951958	13.749174	
22	795	2011-12-15	12.307079	12.392776	11.957446	11.951958	
23	763	2011-12-16	13.376080	12.307079	12.392776	11.957446	
24	770	2011-12-17	13.511968	13.376080	12.307079	12.392776	
25	808	2011-12-18	14.732271	13.511968	13.376080	12.307079	
26	757	2011-12-19	13.774471	14.732271	13.511968	13.376080	
27	803	2011-12-20	12.709106	13.774471	14.732271	13.511968	
28	748	2011-12-21	12.148570	12.709106	13.774471	14.732271	
29	806	2011-12-22	11.839403	12.148570	12.709106	13.774471	
...	
800	21	2014-01-29	11.800777	11.344805	11.753871	12.729659	
801	10	2014-01-30	11.685169	11.800777	11.344805	11.753871	
802	12	2014-01-31	11.857957	11.685169	11.800777	11.344805	
803	129	2014-02-01	11.710582	11.857957	11.685169	11.800777	
804	155	2014-02-02	12.078164	11.710582	11.857957	11.685169	
805	145	2014-02-03	11.280011	12.078164	11.710582	11.857957	
806	134	2014-02-04	11.095584	11.280011	12.078164	11.710582	
807	123	2014-02-05	11.415105	11.095584	11.280011	12.078164	
808	118	2014-02-06	11.445403	11.415105	11.095584	11.280011	
809	122	2014-02-07	10.972318	11.445403	11.415105	11.095584	

810	126	2014-02-08	11.569300	10.972318	11.445403	11.415105
811	149	2014-02-09	12.202967	11.569300	10.972318	11.445403
812	132	2014-02-10	11.264175	12.202967	11.569300	10.972318
813	143	2014-02-11	11.452649	11.264175	12.202967	11.569300
814	131	2014-02-12	11.679099	11.452649	11.264175	12.202967
815	164	2014-02-13	11.285737	11.679099	11.452649	11.264175
816	125	2014-02-14	11.816914	11.285737	11.679099	11.452649
817	141	2014-02-15	11.490470	11.816914	11.285737	11.679099
818	151	2014-02-16	11.582159	11.490470	11.816914	11.285737
819	116	2014-02-17	10.979566	11.582159	11.490470	11.816914
820	128	2014-02-18	10.781898	10.979566	11.582159	11.490470
821	115	2014-02-19	10.674624	10.781898	10.979566	11.582159
822	121	2014-02-20	10.573835	10.674624	10.781898	10.979566
823	174	2014-02-21	10.518126	10.573835	10.674624	10.781898
824	167	2014-02-22	10.776242	10.518126	10.573835	10.674624
825	139	2014-02-23	11.480411	10.776242	10.518126	10.573835
826	162	2014-02-24	10.411403	11.480411	10.776242	10.518126
827	136	2014-02-25	10.294997	10.411403	11.480411	10.776242
828	161	2014-02-26	10.202945	10.294997	10.411403	11.480411
829	133	2014-02-27	10.356350	10.202945	10.294997	10.411403

	t-4	t-5	t-6	t-7
0	NaN	NaN	NaN	NaN
1	NaN	NaN	NaN	NaN
2	NaN	NaN	NaN	NaN
3	NaN	NaN	NaN	NaN
4	6.952692	NaN	NaN	NaN
5	8.536480	6.952692	NaN	NaN
6	9.499781	8.536480	6.952692	NaN
7	10.267707	9.499781	8.536480	6.952692
8	10.850805	10.267707	9.499781	8.536480
9	9.103382	10.850805	10.267707	9.499781
10	9.274873	9.103382	10.850805	10.267707
11	8.813513	9.274873	9.103382	10.850805
12	9.227707	8.813513	9.274873	9.103382
13	10.145910	9.227707	8.813513	9.274873
14	10.780273	10.145910	9.227707	8.813513
15	12.163127	10.780273	10.145910	9.227707
16	10.609714	12.163127	10.780273	10.145910
17	11.673417	10.609714	12.163127	10.780273
18	10.889362	11.673417	10.609714	12.163127
19	11.525150	10.889362	11.673417	10.609714
20	11.759837	11.525150	10.889362	11.673417
21	12.633801	11.759837	11.525150	10.889362
22	13.749174	12.633801	11.759837	11.525150
23	11.951958	13.749174	12.633801	11.759837
24	11.957446	11.951958	13.749174	12.633801
25	12.392776	11.957446	11.951958	13.749174

26	12.307079	12.392776	11.957446	11.951958
27	13.376080	12.307079	12.392776	11.957446
28	13.511968	13.376080	12.307079	12.392776
29	14.732271	13.511968	13.376080	12.307079
...
800	11.620778	11.409880	11.300414	11.109560
801	12.729659	11.620778	11.409880	11.300414
802	11.753871	12.729659	11.620778	11.409880
803	11.344805	11.753871	12.729659	11.620778
804	11.800777	11.344805	11.753871	12.729659
805	11.685169	11.800777	11.344805	11.753871
806	11.857957	11.685169	11.800777	11.344805
807	11.710582	11.857957	11.685169	11.800777
808	12.078164	11.710582	11.857957	11.685169
809	11.280011	12.078164	11.710582	11.857957
810	11.095584	11.280011	12.078164	11.710582
811	11.415105	11.095584	11.280011	12.078164
812	11.445403	11.415105	11.095584	11.280011
813	10.972318	11.445403	11.415105	11.095584
814	11.569300	10.972318	11.445403	11.415105
815	12.202967	11.569300	10.972318	11.445403
816	11.264175	12.202967	11.569300	10.972318
817	11.452649	11.264175	12.202967	11.569300
818	11.679099	11.452649	11.264175	12.202967
819	11.285737	11.679099	11.452649	11.264175
820	11.816914	11.285737	11.679099	11.452649
821	11.490470	11.816914	11.285737	11.679099
822	11.582159	11.490470	11.816914	11.285737
823	10.979566	11.582159	11.490470	11.816914
824	10.781898	10.979566	11.582159	11.490470
825	10.674624	10.781898	10.979566	11.582159
826	10.573835	10.674624	10.781898	10.979566
827	10.518126	10.573835	10.674624	10.781898
828	10.776242	10.518126	10.573835	10.674624
829	11.480411	10.776242	10.518126	10.573835

[830 rows x 10 columns]

In [5]: *#Ens quedem amb energies*

```
daily_dia=daily_dia[['energy_sum','t-1','t-2','t-3','t-4','t-5','t-6','t-7']]
daily_dia.head(5)
```

```
Out[5]:
```

	energy_sum	t-1	t-2	t-3	t-4	t-5	t-6	t-7
0	6.952692	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	8.536480	6.952692	NaN	NaN	NaN	NaN	NaN	NaN
2	9.499781	8.536480	6.952692	NaN	NaN	NaN	NaN	NaN
3	10.267707	9.499781	8.536480	6.952692	NaN	NaN	NaN	NaN
4	10.850805	10.267707	9.499781	8.536480	6.952692	NaN	NaN	NaN

```
In [6]: #Eliminem les 7 primeres files ja que contenen NaN (valors buits)
```

```
daily_dia=daily_dia.drop([0,1,2,3,4,5,6])
daily_dia.head(5)
```

```
Out[6]:
```

	energy_sum	t-1	t-2	t-3	t-4	t-5	\
7	8.813513	9.274873	9.103382	10.850805	10.267707	9.499781	
8	9.227707	8.813513	9.274873	9.103382	10.850805	10.267707	
9	10.145910	9.227707	8.813513	9.274873	9.103382	10.850805	
10	10.780273	10.145910	9.227707	8.813513	9.274873	9.103382	
11	12.163127	10.780273	10.145910	9.227707	8.813513	9.274873	

	t-6	t-7
7	8.536480	6.952692
8	9.499781	8.536480
9	10.267707	9.499781
10	10.850805	10.267707
11	9.103382	10.850805

```
In [39]: len(daily_dia)
```

```
Out[39]: 823
```

```
In [7]: # definim model
```

```
import tensorflow as tf
model =Sequential()
model.add(LSTM(50, activation='relu', input_shape=(7, 1)))
model.add(Dense(1))
model.compile(optimizer='adam', loss='mse', metrics=['accuracy'])
```

WARNING:tensorflow:From c:\users\laura\appdata\local\programs\python\python37\lib\site-packages: Instructions for updating:
Colocations handled automatically by placer.

```
In [9]: #Partim en X i y(valor a predir)
```

```
#normalitzem
scaler=preprocessing.MinMaxScaler(feature_range=(0, 1))
daily_dia_norm=scaler.fit_transform(daily_dia)
```

```
#Seleccionem dades per y i X
y_daily=daily_dia_norm[:,0]
X_daily=daily_dia_norm[:,1:8]
```

```
#Reshape de [samples,timesteps] a [samples,timesteps,features]
#X_daily_list=X_daily.values#.tolist()
X_daily=np.reshape(X_daily, (X_daily.shape[0],X_daily.shape[1],1))
```

```
In [44]: X_daily[0]
        daily_dia_norm
```

```
Out[44]: array([[0.20375985, 0.25530572, 0.2361457 , ..., 0.28043381, 0.17280805,
                0.          ],
               [0.250036   , 0.20375985, 0.25530572, ..., 0.36623108, 0.28043381,
                0.17622026],
               [0.35262316, 0.250036   , 0.20375985, ..., 0.43137821, 0.36623108,
                0.28340205],
               ...,
               [0.36928    , 0.38228562, 0.50172153, ..., 0.40043353, 0.41169419,
                0.42605687],
               [0.35899548, 0.36928    , 0.38228562, ..., 0.3942094 , 0.40043353,
                0.41412098],
               [0.37613476, 0.35899548, 0.36928    , ..., 0.42304761, 0.3942094 ,
                0.40290677]])
```

```
In [10]: # definim model
import tensorflow as tf
model = Sequential()
model.add(LSTM(50, activation='relu', input_shape=(7, 1)))
model.add(Dense(1))
model.compile(optimizer='adam', loss='mse', metrics=['accuracy'])

#Walk forward per test i train
minim=100
n_train=465
lenght=len(daily_dia)

llista_evaluate=list()
llista_prediccions=list()
llista_preditrain=list()
llista_scores=list()
llista_scoretrain=list()
sumScores=0

for i in range(n_train,lenght):
    minim=minim+1
    X_train,X_test= X_daily[minim:i],X_daily[i:i+1]
    y_train,y_test= y_daily[minim:i],y_daily[i:i+1]

    #fem fit al model
    model.fit(X_train, y_train, epochs=50, verbose=0)

    #mostrem score per cada model
    score=model.evaluate(X_test,y_test,verbose=0)
    llista_evaluate.append(score)
```

```

#Predim per cadascun
preditest=model.predict(X_test)
llista_prediccions.append(preditest)

preditrain=model.predict(X_train)
llista_preditrain.append(preditrain)

trainScore = math.sqrt(mean_squared_error(y_train, predictrain))
llista_scoretrain.append(trainScore )

testScore = math.sqrt(mean_squared_error(y_test, predictest))
llista_scores.append(testScore)

sumScores=sumScores+testScore

```

WARNING:tensorflow:From c:\users\laura\appdata\local\programs\python\python37\lib\site-packages: Instructions for updating:
Use tf.cast instead.

In [77]:

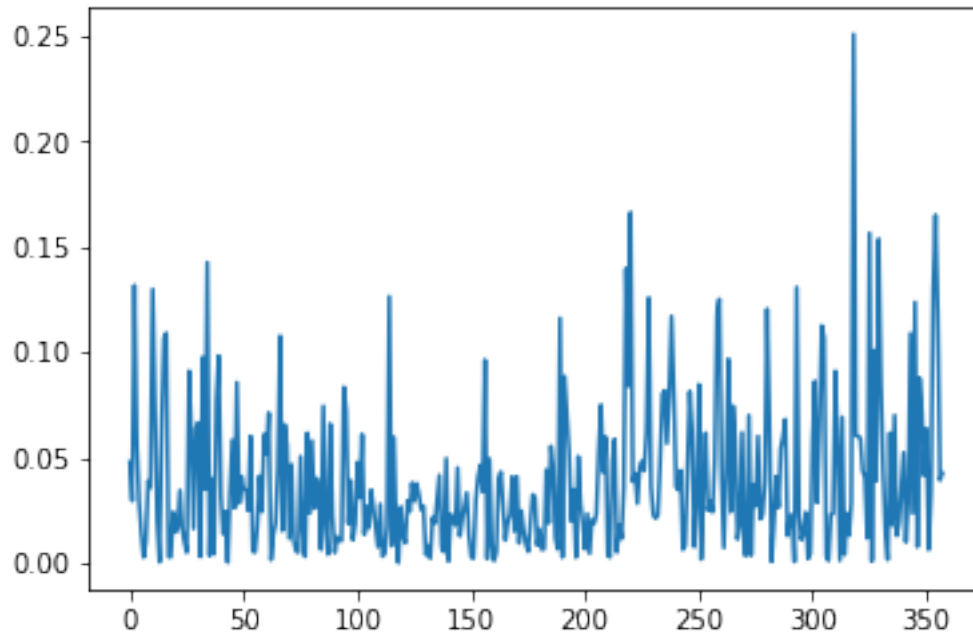
Out[77]: <keras.callbacks.History at 0x24fa43a5198>

In [11]: *#Dividim la suma de scores de test entre el nombre de prediccions per obtenir la mitja*
sumScores/(lenght-n_train)

Out[11]: 0.03935483565166944

In [18]: plt.plot(llista_scores)

Out[18]: [<matplotlib.lines.Line2D at 0x18ee8ca0b70>]



```
In [12]: predis=list()
```

```
for i in range(len(llista_prediccions)):
    predi=llista_prediccions[i].tolist()
    predis.append(predi)
```

```
predis=np.reshape(predis, (358) )
```

```
predis
```

```
Out[12]: array([ 0.48804915,  0.48116875,  0.52007627,  0.62581819,  0.62542796,
                  0.58745068,  0.53324944,  0.50420082,  0.54182315,  0.58864218,
                  0.66909075,  0.56484556,  0.50241619,  0.50483775,  0.50104076,
                  0.61280262,  0.69562191,  0.68760961,  0.66496444,  0.63937914,
                  0.58003765,  0.56885421,  0.57093424,  0.53623843,  0.56202257,
                  0.56274593,  0.56888103,  0.48957837,  0.59859759,  0.59371835,
                  0.52535123,  0.46279198,  0.50020993,  0.40196937,  0.52260923,
                  0.39545405,  0.41802436,  0.39990181,  0.41854599,  0.3584792 ,
                  0.28190097,  0.25838739,  0.27698126,  0.27453393,  0.31160754,
                  0.27768716,  0.29667521,  0.27277124,  0.1977337 ,  0.18542451,
                  0.16483492,  0.24877556,  0.31255493,  0.29170656,  0.21272179,
                  0.19648422,  0.19289121,  0.19796965,  0.18565354,  0.26348597,
                  0.23668006,  0.21830527,  0.16771424,  0.17799339,  0.19853178,
                  0.2112467 ,  0.20719621,  0.18854749,  0.19142768,  0.19754438,
                  0.19499309,  0.20310643,  0.25224411,  0.24509333,  0.23166414,
                  0.24878189,  0.2207659 ,  0.21674323,  0.22531863,  0.23883852,
```

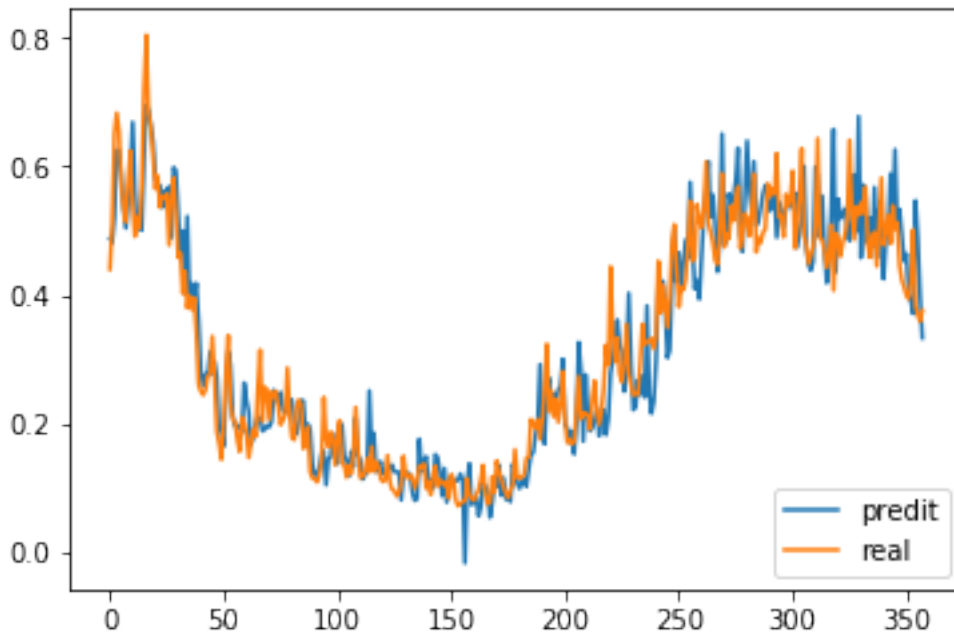
0.23625296,	0.2007964 ,	0.18392789,	0.20237242,	0.2365638 ,
0.23558763,	0.1833766 ,	0.19929563,	0.19146629,	0.13784878,
0.1190792 ,	0.12147431,	0.12997679,	0.15352368,	0.15765841,
0.1048416 ,	0.14545441,	0.1483043 ,	0.18453662,	0.15431581,
0.13801405,	0.17432788,	0.19892347,	0.14944576,	0.14327595,
0.11861844,	0.15660045,	0.1328363 ,	0.20903997,	0.15221937,
0.1437898 ,	0.11377032,	0.11993967,	0.12799749,	0.25132942,
0.15654121,	0.18543494,	0.12493565,	0.13463774,	0.14111921,
0.12198199,	0.12199495,	0.12126657,	0.13377821,	0.13577797,
0.12511316,	0.12494876,	0.12458546,	0.08128938,	0.12280819,
0.12417248,	0.12369771,	0.11880851,	0.09140267,	0.08041166,
0.0837869 ,	0.17608207,	0.12539643,	0.14288574,	0.14894529,
0.11081685,	0.11249059,	0.09953192,	0.15212491,	0.14765982,
0.12555774,	0.08761493,	0.13057765,	0.07774635,	0.09723365,
0.11837457,	0.11082317,	0.11058697,	0.11257194,	0.12199457,
0.11254969,	-0.01662399,	0.11294782,	0.13797133,	0.07361326,
0.08156757,	0.08571304,	0.05652468,	0.07061562,	0.10925595,
0.10231915,	0.09419477,	0.05420019,	0.07694857,	0.10771452,
0.10124557,	0.13539232,	0.08836038,	0.11578146,	0.08255433,
0.09008051,	0.07859118,	0.13674645,	0.12883963,	0.10734221,
0.09916636,	0.10991646,	0.11415239,	0.10232177,	0.12905605,
0.15138032,	0.15491924,	0.18140709,	0.20011094,	0.2922028 ,
0.20291391,	0.16870356,	0.25381395,	0.27113372,	0.23188229,
0.24594237,	0.23506519,	0.25278604,	0.25785437,	0.3005515 ,
0.19782551,	0.19401455,	0.17081347,	0.18838857,	0.15283591,
0.20085771,	0.32668629,	0.28229761,	0.17258675,	0.27644339,
0.21486169,	0.18866031,	0.19338377,	0.20799866,	0.2238818 ,
0.1811043 ,	0.20649591,	0.2218371 ,	0.18258061,	0.20748699,
0.27819556,	0.30116928,	0.33872759,	0.36149508,	0.32248515,
0.31299415,	0.25026757,	0.29251185,	0.40311214,	0.29536259,
0.22160459,	0.22604392,	0.26757628,	0.24517259,	0.27490318,
0.24131934,	0.38347632,	0.24762349,	0.21601497,	0.23520209,
0.29765007,	0.418311 ,	0.41617358,	0.42200744,	0.38148805,
0.30242732,	0.31435031,	0.41975233,	0.50232726,	0.42040861,
0.46622443,	0.40943843,	0.44708014,	0.48598832,	0.45625114,
0.57584536,	0.47704846,	0.40968594,	0.4229005 ,	0.39384645,
0.46443427,	0.52700955,	0.56361657,	0.60805726,	0.52168119,
0.5560168 ,	0.51171768,	0.43648893,	0.4894318 ,	0.650962 ,
0.47865608,	0.4908523 ,	0.55780756,	0.5427115 ,	0.55288601,
0.57824928,	0.6288383 ,	0.49387944,	0.46755463,	0.56516516,
0.64013648,	0.56037968,	0.52972662,	0.60803288,	0.50803095,
0.51568556,	0.53606361,	0.55744183,	0.5707956 ,	0.56078702,
0.53125477,	0.54213667,	0.55826283,	0.48970997,	0.5444482 ,
0.53298569,	0.50463414,	0.53278565,	0.54093349,	0.55470681,
0.53549099,	0.55922812,	0.47414738,	0.49594721,	0.51555032,
0.60062599,	0.47459388,	0.44766426,	0.43854517,	0.45229053,
0.59926945,	0.58877599,	0.48991469,	0.5544529 ,	0.47258726,
0.4189657 ,	0.45059219,	0.53766727,	0.6581102 ,	0.43585265,

```

0.54939848, 0.51960748, 0.52525079, 0.53432143, 0.50571156,
0.48487306, 0.53153843, 0.58740443, 0.49850488, 0.67817748,
0.45811749, 0.57178897, 0.55446357, 0.48086077, 0.5203892 ,
0.47658288, 0.56780624, 0.4578566 , 0.53070694, 0.54770714,
0.42523801, 0.48872846, 0.49502307, 0.58885109, 0.51572597,
0.62649471, 0.50552005, 0.53349936, 0.49558198, 0.45304227,
0.46407291, 0.40042958, 0.38997298, 0.37117833, 0.5472343 ,
0.48827213, 0.39819586, 0.3335371 ])
```

```

In [13]: ##Mostrem
plt.plot(predis, label="predit")
plt.plot(y_daily[n_train:lenght], label="real")
plt.legend(loc="lower right")
plt.show()
```



```

In [14]: prova=daily_dia.iloc[n_train:lenght]
prova
#len(predis)
#lenght-n_train
prova['predi']=predis
prova['y']=y_daily[n_train:lenght]
prova=prova.drop(['energy_sum','t-1'], axis=1)
prova
```

c:\users\laura\appdata\local\programs\python\python37\lib\site-packages\ipykernel_launcher.py:
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>

c:\users\laura\appdata\local\programs\python\python37\lib\site-packages\ipykernel_launcher.py:

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>

import sys

```
Out[14]:
```

	t-2	t-3	t-4	t-5	t-6	t-7	\
472	10.675248	10.860481	11.481859	12.735907	12.308851	12.048499	
473	10.889469	10.675248	10.860481	11.481859	12.735907	12.308851	
474	10.930170	10.889469	10.675248	10.860481	11.481859	12.735907	
475	11.559878	10.930170	10.889469	10.675248	10.860481	11.481859	
476	12.823073	11.559878	10.930170	10.889469	10.675248	10.860481	
477	13.106773	12.823073	11.559878	10.930170	10.889469	10.675248	
478	12.852295	13.106773	12.823073	11.559878	10.930170	10.889469	
479	12.119938	12.852295	13.106773	12.823073	11.559878	10.930170	
480	11.786082	12.119938	12.852295	13.106773	12.823073	11.559878	
481	11.590859	11.786082	12.119938	12.852295	13.106773	12.823073	
482	12.186487	11.590859	11.786082	12.119938	12.852295	13.106773	
483	12.577783	12.186487	11.590859	11.786082	12.119938	12.852295	
484	11.816573	12.577783	12.186487	11.590859	11.786082	12.119938	
485	11.387627	11.816573	12.577783	12.186487	11.590859	11.786082	
486	11.663214	11.387627	11.816573	12.577783	12.186487	11.590859	
487	11.504756	11.663214	11.387627	11.816573	12.577783	12.186487	
488	12.071173	11.504756	11.663214	11.387627	11.816573	12.577783	
489	13.429271	12.071173	11.504756	11.663214	11.387627	11.816573	
490	14.191591	13.429271	12.071173	11.504756	11.663214	11.387627	
491	13.118295	14.191591	13.429271	12.071173	11.504756	11.663214	
492	12.916559	13.118295	14.191591	13.429271	12.071173	11.504756	
493	12.496044	12.916559	13.118295	14.191591	13.429271	12.071173	
494	12.050954	12.496044	12.916559	13.118295	14.191591	13.429271	
495	12.231576	12.050954	12.496044	12.916559	13.118295	14.191591	
496	11.791904	12.231576	12.050954	12.496044	12.916559	13.118295	
497	11.932721	11.791904	12.231576	12.050954	12.496044	12.916559	
498	11.932721	11.932721	11.791904	12.231576	12.050954	12.496044	
499	11.982423	11.932721	11.932721	11.791904	12.231576	12.050954	
500	11.266252	11.982423	11.932721	11.932721	11.791904	12.231576	
501	11.923226	11.266252	11.982423	11.932721	11.932721	11.791904	
..	
800	11.753871	12.729659	11.620778	11.409880	11.300414	11.109560	
801	11.344805	11.753871	12.729659	11.620778	11.409880	11.300414	
802	11.800777	11.344805	11.753871	12.729659	11.620778	11.409880	
803	11.685169	11.800777	11.344805	11.753871	12.729659	11.620778	

804	11.857957	11.685169	11.800777	11.344805	11.753871	12.729659
805	11.710582	11.857957	11.685169	11.800777	11.344805	11.753871
806	12.078164	11.710582	11.857957	11.685169	11.800777	11.344805
807	11.280011	12.078164	11.710582	11.857957	11.685169	11.800777
808	11.095584	11.280011	12.078164	11.710582	11.857957	11.685169
809	11.415105	11.095584	11.280011	12.078164	11.710582	11.857957
810	11.445403	11.415105	11.095584	11.280011	12.078164	11.710582
811	10.972318	11.445403	11.415105	11.095584	11.280011	12.078164
812	11.569300	10.972318	11.445403	11.415105	11.095584	11.280011
813	12.202967	11.569300	10.972318	11.445403	11.415105	11.095584
814	11.264175	12.202967	11.569300	10.972318	11.445403	11.415105
815	11.452649	11.264175	12.202967	11.569300	10.972318	11.445403
816	11.679099	11.452649	11.264175	12.202967	11.569300	10.972318
817	11.285737	11.679099	11.452649	11.264175	12.202967	11.569300
818	11.816914	11.285737	11.679099	11.452649	11.264175	12.202967
819	11.490470	11.816914	11.285737	11.679099	11.452649	11.264175
820	11.582159	11.490470	11.816914	11.285737	11.679099	11.452649
821	10.979566	11.582159	11.490470	11.816914	11.285737	11.679099
822	10.781898	10.979566	11.582159	11.490470	11.816914	11.285737
823	10.674624	10.781898	10.979566	11.582159	11.490470	11.816914
824	10.573835	10.674624	10.781898	10.979566	11.582159	11.490470
825	10.518126	10.573835	10.674624	10.781898	10.979566	11.582159
826	10.776242	10.518126	10.573835	10.674624	10.781898	10.979566
827	11.480411	10.776242	10.518126	10.573835	10.674624	10.781898
828	10.411403	11.480411	10.776242	10.518126	10.573835	10.674624
829	10.294997	10.411403	11.480411	10.776242	10.518126	10.573835

	predi	y
472	0.488049	0.440245
473	0.481169	0.510600
474	0.520076	0.651732
475	0.625818	0.683428
476	0.625428	0.654997
477	0.587451	0.573173
478	0.533249	0.535873
479	0.504201	0.514061
480	0.541823	0.580609
481	0.588642	0.624326
482	0.669091	0.539280
483	0.564846	0.491355
484	0.502416	0.522145
485	0.504838	0.504442
486	0.501041	0.567725
487	0.612803	0.719460
488	0.695622	0.804631
489	0.687610	0.684716
490	0.664964	0.662177
491	0.639379	0.615194

492	0.580038	0.565466
493	0.568854	0.585646
494	0.570934	0.536523
495	0.536238	0.552256
496	0.562023	0.552256
497	0.562746	0.557809
498	0.568881	0.477794
499	0.489578	0.551195
500	0.598598	0.582339
501	0.593718	0.529772
..
800	0.498505	0.537515
801	0.678177	0.524598
802	0.458117	0.543903
803	0.571789	0.527438
804	0.554464	0.568506
805	0.480861	0.479332
806	0.520389	0.458726
807	0.476583	0.494425
808	0.567806	0.497810
809	0.457857	0.444954
810	0.530707	0.511653
811	0.547707	0.582450
812	0.425238	0.477562
813	0.488728	0.498620
814	0.495023	0.523920
815	0.588851	0.479971
816	0.515726	0.539318
817	0.626495	0.502845
818	0.505520	0.513089
819	0.533499	0.445764
820	0.495582	0.423680
821	0.453042	0.411694
822	0.464073	0.400434
823	0.400430	0.394209
824	0.389973	0.423048
825	0.371178	0.501722
826	0.547234	0.382286
827	0.488272	0.369280
828	0.398196	0.358995
829	0.333537	0.376135

[358 rows x 8 columns]

In [15]: *# Convert predictions back to normal values*

```
predi = scaler.inverse_transform(prova)
print(predi)
```

```

print(predi[0][6])
print(predi[0][7])

[[102.53827552 104.19619152 109.75782223 ... 114.82951586 11.35803653
 10.90941785]
 [104.45565387 102.53827552 104.19619152 ... 117.15978906 11.29645368
 11.54173402]
 [104.81994666 104.45565387 102.53827552 ... 120.98214256 11.64469441
 12.81016101]
 ...
 [109.74485905 103.44221738 101.13196088 ... 103.4928401 11.36003231
 10.27161324]
 [100.17673598 109.74485905 103.44221738 ... 102.5326838 10.55380722
 10.17918068]
 [ 99.13484299 100.17673598 109.74485905 ... 101.63058181 9.9750808
 10.33322068]]
11.35803652710475
10.909417847727964

```

In [16]: *#Fem una llista amb les prediccions i una llista amb y(valor real)*

```

listpredi=list()
for i in range(len(predi)):
    listpredi.append(predi[i][6])
listpredi

listy=list()
for i in range(len(predi)):
    listy.append(predi[i][7])
listy

```

Out[16]: [10.909417847727964,
11.541734015843437,
12.810161008052187,
13.095036372869387,
12.839504586908841,
12.104113874587336,
11.768874594549265,
11.5728434534286,
12.170938362994406,
12.563854805851543,
11.799491807423067,
11.368769171374655,
11.645498007385225,
11.486383766157797,
12.055146498912347,
13.418870046058563,

14.18434826454072,
13.106605940356959,
12.904034964148906,
12.481777834780083,
12.034844377299484,
12.216213953779874,
11.774720732675483,
11.916121209122355,
11.916121209122355,
11.96602913488797,
11.246891425520584,
11.906586850226484,
12.186487843286868,
11.714046017023353,
11.077116388183462,
11.138487728712652,
10.57127320547445,
10.879512180409455,
10.368387785899902,
10.537463985883184,
10.349229865851477,
10.50921411168254,
10.019834486664964,
9.292251119455212,
9.202081418270591,
9.151266909548587,
9.222557748636955,
9.422124723547805,
9.458964521083429,
9.972057279719857,
9.384480482677608,
8.6351294416642,
8.475305748219682,
8.250345631403201,
8.748759948165972,
9.499277963802884,
9.98525961941407,
9.034552548724397,
8.814949844433727,
8.76390022824991,
8.526805688003604,
8.360955926860143,
8.839414316662916,
8.772971632255084,
8.615273829224325,
8.275378167726693,
8.472927941606807,
8.672224300861377,

8.569827046179858,
9.248636480397565,
9.782949177205317,
8.78407110621045,
9.262864895726782,
9.228580614148845,
8.809022341042676,
9.198402951260073,
9.122928046760308,
9.205946753383158,
8.99039441509594,
8.73467580784989,
8.79505693254052,
8.873634078699684,
9.531136126523368,
8.887569084251147,
8.554449833093313,
8.526074666165814,
8.96670893011568,
9.084060970803332,
9.022101375737103,
8.401360411391364,
8.445002435020523,
8.707618292789272,
8.080598019861757,
7.986167965150143,
7.979425970849153,
7.933852188349128,
8.0269080683008,
8.237874154292523,
9.118918399933353,
8.545690125461526,
8.427830527450647,
8.636774690454713,
8.513206447510978,
8.172242914964519,
8.621977421169715,
8.796413561227569,
8.192563316001538,
8.174152977266415,
7.999329897635357,
8.167594639481628,
8.04961549345679,
8.382031844433065,
8.979698303776429,
8.388709126215051,
7.994412582623409,
8.003539590147223,

8.070914450318872,
8.259333507560532,
8.075620873767424,
8.431467753946913,
8.083098755408729,
8.21617176720151,
8.162964869199355,
7.981492296883346,
7.93726694560191,
7.9646096395929815,
8.308449837480422,
7.9354975468705256,
7.831604187634627,
7.813722029159717,
7.737085218778821,
7.782283454440081,
7.909753541488857,
8.298456178350863,
8.033943796839147,
7.99063574915203,
8.040299727996455,
7.967233799986828,
7.843376844393493,
7.980730399392371,
8.161686556968506,
8.146690092822242,
8.189987225466021,
7.846325420347436,
7.940944607720689,
7.7579440683531296,
8.04097395526863,
8.15976392422729,
7.8742814523868505,
7.966926850801755,
7.926720196741935,
7.879975173963482,
7.950190975918254,
7.707190505773298,
8.040597097188217,
7.967997837715548,
7.679937016413131,
7.5986441036577235,
7.632838197065269,
7.6595867937809805,
7.669189605188508,
7.9849040920466425,
7.748644667896798,
7.702247942117759,

7.676472082204107,
7.766959193216043,
7.822161074917124,
7.975039369345884,
8.170857996275723,
7.775680235449478,
7.626386244312332,
7.646361220764487,
8.011135498454033,
7.786595788453895,
8.231216283582087,
8.085296514820385,
7.969185684187949,
7.856526507750872,
7.77672973556362,
7.7138540329656085,
7.821559940058248,
7.891096132521677,
8.388331534415842,
7.992515578225606,
7.988412765500768,
7.997406126973563,
8.03843478020095,
8.269721703660037,
8.282245070492321,
8.810805953956507,
8.759353139654928,
8.6884374827898,
8.809740324593482,
8.535206219153517,
8.799950008720907,
9.264444648313265,
9.864896502412734,
8.923657253296597,
9.21468585759026,
8.849737770791489,
9.088705127647629,
8.7707417079098,
9.017281929794072,
9.486085058158912,
8.789006469986974,
8.48704948477753,
8.527618364775309,
8.46309069070408,
8.495554886986536,
8.967588503800933,
9.413931318395582,
8.816287557852858,

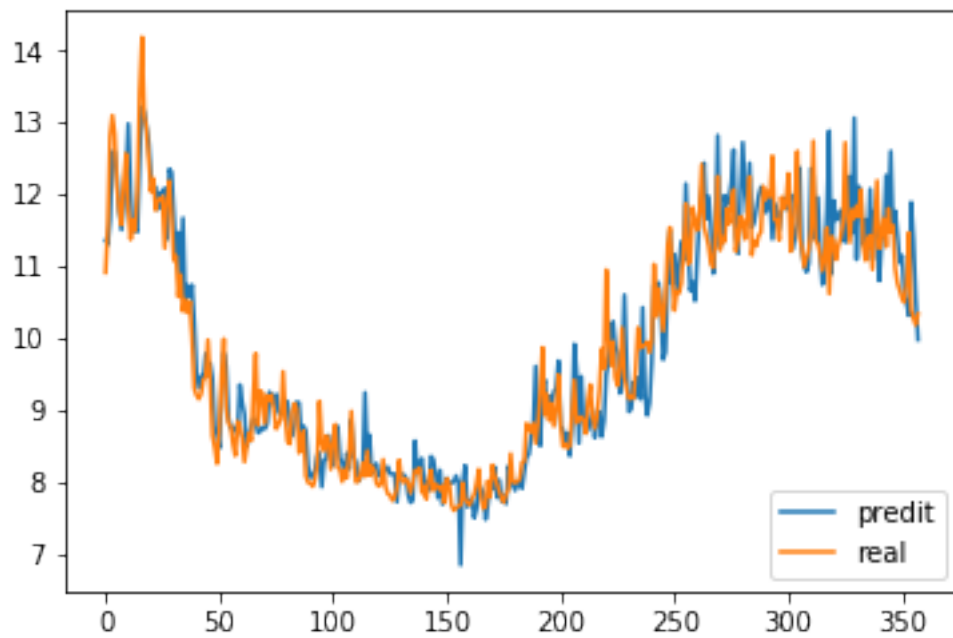
8.88986118611644,
8.900549693941644,
8.851638618481429,
8.672904981500528,
9.12419782228142,
9.349625345441478,
8.919862155636723,
8.739260455154465,
8.911849598083132,
9.26753256808381,
9.848602555278077,
9.570749355302713,
10.946909572416958,
10.006380374614903,
9.616263033037004,
9.948401093596452,
9.444237364853416,
9.332959777030139,
9.594861956368552,
10.141816549659852,
9.445828141920481,
9.28420500723586,
9.149516047613004,
9.173936990742419,
9.153846386000565,
9.589623967279916,
10.143475245833464,
9.856343978029473,
9.889450871459683,
9.904920695344611,
9.94557551769574,
9.79643787430105,
9.953012201546553,
11.023905520500147,
10.298646075869945,
10.688706294806515,
10.45588825412626,
10.09523578059129,
10.508768565408724,
11.287728740942534,
11.535731929015077,
10.98747864485505,
10.381783680340867,
10.647804005246456,
10.622401949674646,
10.766664346267493,
11.276561990023957,
11.864807179639488,

11.023992146704789,
11.074953465089989,
11.816852453441324,
11.616450877335833,
11.491630282956551,
11.753686258674128,
12.412632378166395,
11.549429348778885,
11.424641138629035,
11.280703586851473,
11.033242910892792,
10.976362559972356,
11.511879811375307,
12.247402989155825,
11.219569137698116,
11.395257976203919,
11.337116285866875,
11.798095044186056,
11.587218776514034,
11.907627195148779,
12.064553914855717,
11.20414149367243,
11.359534369142706,
11.690179257625214,
11.623473619682327,
11.373285729708359,
11.719242920481847,
12.242292563384435,
11.150302349242837,
11.351197600739617,
11.280702475240322,
11.425676567314516,
11.470881036124945,
12.108190463516383,
11.92545468202104,
12.030196060540819,
11.96527464349971,
12.528774496173462,
11.63811203398031,
11.64425731597274,
11.360132707583698,
11.957162404806581,
11.796517373529165,
11.897239672855578,
12.287518743179643,
11.203809184647207,
11.470608628551458,
11.978754980013164,

12.597485043606161,
11.38974536295075,
11.248881601971792,
10.98571332222799,
11.09960377204762,
11.227551079577204,
11.52074733805962,
12.739132484123028,
11.345600613184216,
11.314520733454554,
11.165151974531303,
10.929638574085304,
11.11884759178758,
11.528371134745226,
10.613439200667486,
11.412204513997853,
11.351673286441853,
11.089550763544295,
11.281195094564278,
11.391114872281667,
11.602886428432544,
12.71636005383942,
11.736530455723566,
11.325769792924136,
11.783630294495596,
11.667543798117368,
11.8410475626921,
11.693062123140267,
12.062167122521247,
11.260708192302587,
11.075517224671295,
11.396360825661187,
11.426785117284973,
10.95174045703851,
11.551195439803223,
12.187487148953624,
11.244806286933207,
11.434061112788763,
11.661449167564713,
11.266457151188172,
11.79983496585512,
11.472038070776696,
11.564107272307272,
10.959018211708038,
10.760531437390572,
10.652812713176369,
10.551607029217053,
10.495667376806587,

```
10.754852138077737,  
11.461937553711827,  
10.388501900283224,  
10.271613241190687,  
10.179180677830654,  
10.33322067736639]
```

```
In [17]: ##Mostrem  
plt.plot(listpredi, label="predict")  
plt.plot(listy, label="real")  
plt.legend(loc="lower right")  
plt.show()
```



```
In [ ]:
```

```
In [ ]:
```