Getting started with Time-series features

Python notebook using data from Store Item Demand Forecasting Challenge · 3,367 views

Submission
✔ Ran successfully
Submitted by Dan Ofer 9 months ago

Private Score	Public Score
150.77992	148.57800

A very basic kernel with some naive features for predicting seasonal or hierachical autoregressive problems (such as store sales)

- Kernel will be updated

In [1]:
```
import pandas as pd
import datetime
import xgboost as xgb
from sklearn.model_selection import train_test_split, TimeSeriesSp
lit

# Input data files are available in the "../input/" directory.
# For example, running this (by clicking run or pressing Shift+Ente
r) will list the files in the input directory

import os
print(os.listdir("../input"))
# Any results you write to the current directory are saved as outpu
t.

train = pd.read_csv('../input/train.csv')
test = pd.read_csv('../input/test.csv')
sample = pd.read_csv('../input/sample_submission.csv')
```

```
['train.csv', 'sample_submission.csv', 'test.csv']
```

merge train and test for easier feature engineering.

- Beware leaks and the offset needed in predicting the future!!
- Can also be useful for creating a baseline; see: https://machinelearningmastery.com/model-residual-errors-correct-time-series-forecasts-python/ (https://machinelearningmastery.com/model-residual-errors-correct-time-series-forecasts-python/)
- Note that here, train has no ID column

In [2]:
```
train.columns
```

Out[2]:
```
Index(['date', 'store', 'item', 'sales'], dtype='object')
```

In [3]:
```
test.columns
```

Out[3]:
```
Index(['id', 'date', 'store', 'item'], dtype='object')
```

In [4]:
```
print("train shape:", train.shape)
print("Test shape:", test.shape)
df = pd.concat([train,test])
print(df.shape)
df.head()
```

```
train shape: (913000, 4)
Test shape: (45000, 4)
(958000, 5)


/opt/conda/lib/python3.6/site-packages/ipykernel_launcher.py:3: Fu
tureWarning: Sorting because non-concatenation axis is not aligne
d. A future version
of pandas will change to not sort by default.

To accept the future behavior, pass 'sort=False'.

To retain the current behavior and silence the warning, pass 'sort
=True'.

  This is separate from the ipykernel package so we can avoid doin
g imports until
```

Out[4]:

	date	id	item	sales	store
0	2013-01-01	NaN	1	13.0	1
1	2013-01-02	NaN	1	11.0	1
2	2013-01-03	NaN	1	14.0	1
3	2013-01-04	NaN	1	13.0	1
4	2013-01-05	NaN	1	10.0	1

naive datetime features:

- Could also add holidays, weekends, work-hours if relevant and known

In [5]:
```
df['date'] = pd.to_datetime(df['date'],infer_datetime_format=True)


df['month'] = df['date'].dt.month
df['weekday'] = df['date'].dt.dayofweek
df['year'] = df['date'].dt.year
# df['date'].dt.
df['week_of_year']  = df.date.dt.weekofyear
```

In [6]:
```
df.set_index("date",inplace=True)
```

Add historical / seasonal features

- Additional features could includes slopes, trends, item-basket level features, but require more work to avoid leakage. Will add in future
- For now - naive features: we expect sales to be what they were at the same time of year, in the past, for each store+item combo
  - Could be done more elegantly with pandas's agg_func

In [7]:

In [7]:
```
df["median-store_item-month"] = df.groupby(['month',"item","store"
])["sales"].transform("median")
df["mean-store_item-week"] = df.groupby(['week_of_year',"item","st
ore"])["sales"].transform("mean")
df["item-month-sum"] = df.groupby(['month',"item"])["sales"].trans
form("sum") # total sales of that item  for all stores
df["store-month-sum"] = df.groupby(['month',"store"])["sales"].tra
nsform("sum") # total sales of that store  for all items
```

In [8]:
```
# get shifted features for grouped data. Note need to sort first!
df['store_item_shifted-90'] = df.groupby(["item","store"])['sales'
].transform(lambda x:x.shift(90)) # sales for that item 90 days = 3
months ago
df['store_item_shifted-180'] = df.groupby(["item","store"])['sale
s'].transform(lambda x:x.shift(180)) # sales for that item 180 days
= 3 months ago
df['store_item_shifted-365'] = df.groupby(["item","store"])['sale
s'].transform(lambda x:x.shift(365)) # sales for that 1 year  ago

df["item-week_shifted-90"] = df.groupby(['week_of_year',"item"])[
"sales"].transform(lambda x:x.shift(12).sum()) # shifted total sale
s for that item 12 weeks (3 months) ago
df["store-week_shifted-90"] = df.groupby(['week_of_year',"store"])
["sales"].transform(lambda x:x.shift(12).sum()) # shifted total sal
es for that store 12 weeks (3 months) ago
df["item-week_shifted-90"] = df.groupby(['week_of_year',"item"])[
"sales"].transform(lambda x:x.shift(12).mean()) # shifted mean sale
s for that item 12 weeks (3 months) ago
df["store-week_shifted-90"] = df.groupby(['week_of_year',"store"])
["sales"].transform(lambda x:x.shift(12).mean()) # shifted mean sal
es for that store 12 weeks (3 months) ago
```

In [9]:
```
df.tail()
```

Out[9]:

	id	item	sales	store	month	weekday	year	week_of_year	median-store_ite month
date									
2018-03-27	44995.0	50	NaN	10	3	1	2018	13	67.0
2018-03-28	44996.0	50	NaN	10	3	2	2018	13	67.0
2018-03-29	44997.0	50	NaN	10	3	3	2018	13	67.0
2018-03-30	44998.0	50	NaN	10	3	4	2018	13	67.0
2018-03-31	44999.0	50	NaN	10	3	5	2018	13	67.0

We should do one hot encoding at this point on the store and ite mIDs to avoid silly range based features.

- We'll do that later, as it's also possible the numbers/order has meaning.

split our data for modelling

In [10]:
```
col = [i for i in df.columns if i not in ['date','id']]
y = 'sales'
```

In [11]:
```
train.columns
```

Out[11]:
```
Index(['date', 'store', 'item', 'sales'], dtype='object')
```

In [12]:
```
train = df.loc[~df.sales.isna()]
print("new train",train.shape)
test = df.loc[df.sales.isna()]
print("new test",test.shape)
```

```
new train (913000, 17)
new test (45000, 17)
```

Evaluation should use **temporal train test** split or temporal CV

- we can define it manually or use sklearn's functions. these aren't trivial to plug and play with xgboost, so i'll skip for this version of the kernel, but without it, our local score is meaningless!

In [13]:
```
train_x, train_cv, y, y_cv = train_test_split(train[col],train[y],
test_size=0.15, random_state=42)
# train_x, train_cv, y, y_cv = TimeSeriesSplit(train[col],train[y],
 test_size=0.1, random_state=42)
```

In [14]:
```
def XGB_regressor(train_X, train_y, test_X, test_y, feature_names=
None, seed_val=2017, num_rounds=500):
    param = {}
    param['objective'] = 'reg:linear'
    param['eta'] = 0.1
    param['max_depth'] = 6
    param['silent'] = 1
    param['eval_metric'] = 'mae'
    param['min_child_weight'] = 1
    param['subsample'] = 0.8
    param['colsample_bytree'] = 0.8
    param['seed'] = seed_val
```

```
    num_rounds = num_rounds

    plst = list(param.items())

    xgtrain = xgb.DMatrix(train_X, label=train_y)

    if test_y is not None:
        xgtest = xgb.DMatrix(test_X, label=test_y)
        watchlist = [ (xgtrain,'train'), (xgtest, 'test') ]
        model = xgb.train(plst, xgtrain, num_rounds, watchlist, ea
rly_stopping_rounds=20)
    else:
        xgtest = xgb.DMatrix(test_X)
        model = xgb.train(plst, xgtrain, num_rounds)

    return model
```

In [15]:

```
model = XGB_regressor(train_X = train_x, train_y = y, test_X = tra
in_cv, test_y = y_cv)
y_test = model.predict(xgb.DMatrix(test[col]), ntree_limit = model
.best_ntree_limit)
```

```
[0]     train-mae:46.5709       test-mae:46.6042
Multiple eval metrics have been passed: 'test-mae' will be used fo
r early stopping.

Will train until test-mae hasn't improved in 20 rounds.
[1]     train-mae:41.9145       test-mae:41.9444
[2]     train-mae:37.7236       test-mae:37.7475
[3]     train-mae:33.9518       test-mae:33.973
[4]     train-mae:30.5571       test-mae:30.5759
[5]     train-mae:27.5019       test-mae:27.5188
[6]     train-mae:24.7527       test-mae:24.7659
[7]     train-mae:22.278        test-mae:22.2899
[8]     train-mae:20.0505       test-mae:20.0606
[9]     train-mae:18.0459       test-mae:18.0546
[10]    train-mae:16.2416       test-mae:16.2493
[11]    train-mae:14.6187       test-mae:14.6242
[12]    train-mae:13.1572       test-mae:13.162
[13]    train-mae:11.8419       test-mae:11.8458
[14]    train-mae:10.6581       test-mae:10.6613
[15]    train-mae:9.5928        test-mae:9.59569
[16]    train-mae:8.63406       test-mae:8.63654
[17]    train-mae:7.77105       test-mae:7.77293
[18]    train-mae:6.99453       test-mae:6.99617
[19]    train-mae:6.29563       test-mae:6.29698
[20]    train-mae:5.6668        test-mae:5.66798
[21]    train-mae:5.10097       test-mae:5.10203
[22]    train-mae:4.59362       test-mae:4.59399
[23]    train-mae:4.13564       test-mae:4.13559
[24]    train-mae:3.7238        test-mae:3.72356
[25]    train-mae:3.35328       test-mae:3.35287
[26]    train-mae:3.0205        test-mae:3.0199
[27]    train-mae:2.72117       test-mae:2.72051
[28]    train-mae:2.45228       test-mae:2.45155
[29]    train-mae:2.21091       test-mae:2.21018
[30]    train-mae:1.99974       test-mae:1.99897
[31]    train-mae:1.80571       test-mae:1.80511
[32]    train-mae:1.63234       test-mae:1.63171
[33]    train-mae:1.4774        test-mae:1.47653
[34]    train-mae:1.34759       test-mae:1.34661
[35]    train-mae:1.22455       test-mae:1.22372
```

```
[36]    train-mae:1.11544       test-mae:1.11462
[37]    train-mae:1.02866       test-mae:1.02789
[38]    train-mae:0.943466      test-mae:0.942917
[39]    train-mae:0.867303      test-mae:0.866845
[40]    train-mae:0.799735      test-mae:0.799523
[41]    train-mae:0.740692      test-mae:0.740688
[42]    train-mae:0.699018      test-mae:0.699104
[43]    train-mae:0.664232      test-mae:0.664462
[44]    train-mae:0.623668      test-mae:0.62419
[45]    train-mae:0.598709      test-mae:0.599438
[46]    train-mae:0.565124      test-mae:0.56607
[47]    train-mae:0.535944      test-mae:0.537041
[48]    train-mae:0.520377      test-mae:0.521546
[49]    train-mae:0.507561      test-mae:0.508841
[50]    train-mae:0.497039      test-mae:0.498451
[51]    train-mae:0.479871      test-mae:0.481495
[52]    train-mae:0.462514      test-mae:0.464291
[53]    train-mae:0.445863      test-mae:0.447645
[54]    train-mae:0.431323      test-mae:0.433141
[55]    train-mae:0.419604      test-mae:0.421475
[56]    train-mae:0.408989      test-mae:0.410967
[57]    train-mae:0.399146      test-mae:0.401153
[58]    train-mae:0.394201      test-mae:0.396241
[59]    train-mae:0.382849      test-mae:0.385019
[60]    train-mae:0.372249      test-mae:0.374578
[61]    train-mae:0.361764      test-mae:0.364175
[62]    train-mae:0.353053      test-mae:0.355537
[63]    train-mae:0.344338      test-mae:0.346792
[64]    train-mae:0.337156      test-mae:0.339662
[65]    train-mae:0.329217      test-mae:0.331739
[66]    train-mae:0.327101      test-mae:0.329633
[67]    train-mae:0.324889      test-mae:0.327425
[68]    train-mae:0.318867      test-mae:0.321391
[69]    train-mae:0.314144      test-mae:0.316696
[70]    train-mae:0.309754      test-mae:0.312285
[71]    train-mae:0.30795       test-mae:0.310491
[72]    train-mae:0.306408      test-mae:0.308935
[73]    train-mae:0.304842      test-mae:0.307383
[74]    train-mae:0.299131      test-mae:0.301679
[75]    train-mae:0.293245      test-mae:0.295827
[76]    train-mae:0.292187      test-mae:0.294747
[77]    train-mae:0.28727       test-mae:0.289732
[78]    train-mae:0.282926      test-mae:0.285394
[79]    train-mae:0.281248      test-mae:0.283728
[80]    train-mae:0.279979      test-mae:0.282469
[81]    train-mae:0.275542      test-mae:0.278021
[82]    train-mae:0.270448      test-mae:0.272973
[83]    train-mae:0.265268      test-mae:0.267861
[84]    train-mae:0.261257      test-mae:0.263835
[85]    train-mae:0.259933      test-mae:0.262519
[86]    train-mae:0.254996      test-mae:0.257564
[87]    train-mae:0.251288      test-mae:0.253897
[88]    train-mae:0.250218      test-mae:0.252836
[89]    train-mae:0.246028      test-mae:0.248682
[90]    train-mae:0.241431      test-mae:0.244073
[91]    train-mae:0.240222      test-mae:0.242861
[92]    train-mae:0.236095      test-mae:0.238753
[93]    train-mae:0.234881      test-mae:0.237522
[94]    train-mae:0.230486      test-mae:0.23312
[95]    train-mae:0.227025      test-mae:0.229635
[96]    train-mae:0.223826      test-mae:0.22645
[97]    train-mae:0.222766      test-mae:0.225382
[98]    train-mae:0.22193       test-mae:0.224545
[99]    train-mae:0.218901      test-mae:0.221567
```

```
[100]   train-mae:0.216934      test-mae:0.21962
[101]   train-mae:0.21519       test-mae:0.217879
[102]   train-mae:0.213 test-mae:0.215663
[103]   train-mae:0.21094       test-mae:0.213584
[104]   train-mae:0.208383      test-mae:0.211048
[105]   train-mae:0.207514      test-mae:0.210176
[106]   train-mae:0.204621      test-mae:0.207289
[107]   train-mae:0.202833      test-mae:0.205505
[108]   train-mae:0.201326      test-mae:0.203945
[109]   train-mae:0.199122      test-mae:0.201725
[110]   train-mae:0.198285      test-mae:0.200876
[111]   train-mae:0.196761      test-mae:0.199343
[112]   train-mae:0.196045      test-mae:0.198621
[113]   train-mae:0.193814      test-mae:0.196418
[114]   train-mae:0.19331       test-mae:0.195892
[115]   train-mae:0.192759      test-mae:0.195339
[116]   train-mae:0.190324      test-mae:0.192881
[117]   train-mae:0.188752      test-mae:0.191276
[118]   train-mae:0.187018      test-mae:0.189571
[119]   train-mae:0.18644       test-mae:0.188998
[120]   train-mae:0.185719      test-mae:0.188282
[121]   train-mae:0.183486      test-mae:0.18603
[122]   train-mae:0.181478      test-mae:0.184012
[123]   train-mae:0.17997       test-mae:0.182481
[124]   train-mae:0.179578      test-mae:0.182088
[125]   train-mae:0.177479      test-mae:0.179966
[126]   train-mae:0.176171      test-mae:0.178651
[127]   train-mae:0.175652      test-mae:0.178125
[128]   train-mae:0.173938      test-mae:0.176407
[129]   train-mae:0.172464      test-mae:0.174919
[130]   train-mae:0.170911      test-mae:0.17337
[131]   train-mae:0.16987       test-mae:0.172306
[132]   train-mae:0.168701      test-mae:0.171114
[133]   train-mae:0.167274      test-mae:0.169684
[134]   train-mae:0.165313      test-mae:0.167704
[135]   train-mae:0.163955      test-mae:0.166329
[136]   train-mae:0.162348      test-mae:0.164701
[137]   train-mae:0.161868      test-mae:0.164227
[138]   train-mae:0.160148      test-mae:0.162504
[139]   train-mae:0.159066      test-mae:0.161394
[140]   train-mae:0.158035      test-mae:0.160362
[141]   train-mae:0.156677      test-mae:0.159002
[142]   train-mae:0.155251      test-mae:0.157546
[143]   train-mae:0.15413       test-mae:0.156394
[144]   train-mae:0.152788      test-mae:0.155025
[145]   train-mae:0.151448      test-mae:0.153688
[146]   train-mae:0.149857      test-mae:0.152118
[147]   train-mae:0.148714      test-mae:0.150972
[148]   train-mae:0.147764      test-mae:0.15002
[149]   train-mae:0.146894      test-mae:0.149148
[150]   train-mae:0.146132      test-mae:0.148374
[151]   train-mae:0.145258      test-mae:0.147508
[152]   train-mae:0.144166      test-mae:0.146403
[153]   train-mae:0.143307      test-mae:0.145535
[154]   train-mae:0.142621      test-mae:0.144853
[155]   train-mae:0.14184       test-mae:0.144048
[156]   train-mae:0.141323      test-mae:0.143529
[157]   train-mae:0.140737      test-mae:0.142943
[158]   train-mae:0.140164      test-mae:0.142362
[159]   train-mae:0.139712      test-mae:0.141921
[160]   train-mae:0.138748      test-mae:0.140959
[161]   train-mae:0.137571      test-mae:0.139778
[162]   train-mae:0.136849      test-mae:0.139025
```

```
[163]   train-mae:0.13588       test-mae:0.138048
[164]   train-mae:0.135492      test-mae:0.137661
[165]   train-mae:0.134723      test-mae:0.136861
[166]   train-mae:0.133857      test-mae:0.135982
[167]   train-mae:0.13322       test-mae:0.135333
[168]   train-mae:0.132135      test-mae:0.134237
[169]   train-mae:0.131815      test-mae:0.133918
[170]   train-mae:0.131249      test-mae:0.133351
[171]   train-mae:0.13064       test-mae:0.132731
[172]   train-mae:0.130069      test-mae:0.132156
[173]   train-mae:0.129405      test-mae:0.131495
[174]   train-mae:0.128651      test-mae:0.13075
[175]   train-mae:0.128161      test-mae:0.130265
[176]   train-mae:0.127298      test-mae:0.129384
[177]   train-mae:0.126901      test-mae:0.128969
[178]   train-mae:0.126498      test-mae:0.128555
[179]   train-mae:0.125888      test-mae:0.127941
[180]   train-mae:0.125292      test-mae:0.127336
[181]   train-mae:0.124565      test-mae:0.12661
[182]   train-mae:0.123778      test-mae:0.125825
[183]   train-mae:0.123444      test-mae:0.125477
[184]   train-mae:0.123101      test-mae:0.125121
[185]   train-mae:0.122534      test-mae:0.124547
[186]   train-mae:0.122054      test-mae:0.124057
[187]   train-mae:0.121692      test-mae:0.123701
[188]   train-mae:0.121343      test-mae:0.12336
[189]   train-mae:0.12055       test-mae:0.122554
[190]   train-mae:0.120021      test-mae:0.122023
[191]   train-mae:0.119725      test-mae:0.121734
[192]   train-mae:0.119451      test-mae:0.121453
[193]   train-mae:0.119171      test-mae:0.121179
[194]   train-mae:0.118844      test-mae:0.120856
[195]   train-mae:0.118206      test-mae:0.120211
[196]   train-mae:0.118071      test-mae:0.120075
[197]   train-mae:0.117643      test-mae:0.119645
[198]   train-mae:0.117313      test-mae:0.119326
[199]   train-mae:0.116821      test-mae:0.118811
[200]   train-mae:0.116514      test-mae:0.118504
[201]   train-mae:0.116059      test-mae:0.118046
[202]   train-mae:0.115478      test-mae:0.117464
[203]   train-mae:0.115218      test-mae:0.117197
[204]   train-mae:0.114951      test-mae:0.116926
[205]   train-mae:0.114738      test-mae:0.116718
[206]   train-mae:0.114406      test-mae:0.116377
[207]   train-mae:0.11415       test-mae:0.116112
[208]   train-mae:0.113827      test-mae:0.115788
[209]   train-mae:0.113565      test-mae:0.115525
[210]   train-mae:0.113183      test-mae:0.115128
[211]   train-mae:0.112694      test-mae:0.114629
[212]   train-mae:0.112501      test-mae:0.114436
[213]   train-mae:0.112163      test-mae:0.114091
[214]   train-mae:0.111657      test-mae:0.113586
[215]   train-mae:0.11134       test-mae:0.113278
[216]   train-mae:0.11085       test-mae:0.11279
[217]   train-mae:0.11051       test-mae:0.112453
[218]   train-mae:0.110241      test-mae:0.112179
[219]   train-mae:0.110013      test-mae:0.111939
[220]   train-mae:0.109677      test-mae:0.11159
[221]   train-mae:0.109312      test-mae:0.111223
[222]   train-mae:0.109017      test-mae:0.110928
[223]   train-mae:0.108577      test-mae:0.110477
[224]   train-mae:0.108256      test-mae:0.110165
[225]   train-mae:0.107979      test-mae:0.10988
[226]   train-mae:0.107671      test-mae:0.109573
```

```
[227]   train-mae:0.107374      test-mae:0.109256
[228]   train-mae:0.106937      test-mae:0.108818
[229]   train-mae:0.106649      test-mae:0.108526
[230]   train-mae:0.106196      test-mae:0.108065
[231]   train-mae:0.105825      test-mae:0.107686
[232]   train-mae:0.105587      test-mae:0.10744
[233]   train-mae:0.105132      test-mae:0.106987
[234]   train-mae:0.104739      test-mae:0.106596
[235]   train-mae:0.104558      test-mae:0.106416
[236]   train-mae:0.104353      test-mae:0.106214
[237]   train-mae:0.104211      test-mae:0.106073
[238]   train-mae:0.103868      test-mae:0.105746
[239]   train-mae:0.103689      test-mae:0.105558
[240]   train-mae:0.103512      test-mae:0.105378
[241]   train-mae:0.103311      test-mae:0.105184
[242]   train-mae:0.103039      test-mae:0.104897
[243]   train-mae:0.102867      test-mae:0.104729
[244]   train-mae:0.10253       test-mae:0.104397
[245]   train-mae:0.102243      test-mae:0.104113
[246]   train-mae:0.102006      test-mae:0.10386
[247]   train-mae:0.101803      test-mae:0.103656
[248]   train-mae:0.101676      test-mae:0.103536
[249]   train-mae:0.10123       test-mae:0.103091
[250]   train-mae:0.101092      test-mae:0.102948
[251]   train-mae:0.100716      test-mae:0.102556
[252]   train-mae:0.100445      test-mae:0.102288
[253]   train-mae:0.100208      test-mae:0.102047
[254]   train-mae:0.099901      test-mae:0.101728
[255]   train-mae:0.099664      test-mae:0.101482
[256]   train-mae:0.099504      test-mae:0.10131
[257]   train-mae:0.09923       test-mae:0.101039
[258]   train-mae:0.098856      test-mae:0.100663
[259]   train-mae:0.098565      test-mae:0.100368
[260]   train-mae:0.098385      test-mae:0.100188
[261]   train-mae:0.098275      test-mae:0.10007
[262]   train-mae:0.097983      test-mae:0.099768
[263]   train-mae:0.097729      test-mae:0.099505
[264]   train-mae:0.097492      test-mae:0.099255
[265]   train-mae:0.097272      test-mae:0.099033
[266]   train-mae:0.097023      test-mae:0.098786
[267]   train-mae:0.096813      test-mae:0.098579
[268]   train-mae:0.096743      test-mae:0.098512
[269]   train-mae:0.096607      test-mae:0.098383
[270]   train-mae:0.096327      test-mae:0.098104
[271]   train-mae:0.096274      test-mae:0.098051
[272]   train-mae:0.096081      test-mae:0.097851
[273]   train-mae:0.095853      test-mae:0.097621
[274]   train-mae:0.095587      test-mae:0.097357
[275]   train-mae:0.095341      test-mae:0.097109
[276]   train-mae:0.095204      test-mae:0.096969
[277]   train-mae:0.095146      test-mae:0.096907
[278]   train-mae:0.094905      test-mae:0.096668
[279]   train-mae:0.094832      test-mae:0.096598
[280]   train-mae:0.094665      test-mae:0.09643
[281]   train-mae:0.094515      test-mae:0.096278
[282]   train-mae:0.094301      test-mae:0.096058
[283]   train-mae:0.094193      test-mae:0.095947
[284]   train-mae:0.094053      test-mae:0.095815
[285]   train-mae:0.093962      test-mae:0.095725
[286]   train-mae:0.093807      test-mae:0.095562
[287]   train-mae:0.093682      test-mae:0.095444
[288]   train-mae:0.093518      test-mae:0.095276
[289]   train-mae:0.093374      test-mae:0.095126
[290]   train-mae:0.093313      test-mae:0.095067
```

```
[291]   train-mae:0.09321       test-mae:0.094967
[292]   train-mae:0.093075      test-mae:0.094825
[293]   train-mae:0.092978      test-mae:0.094726
[294]   train-mae:0.092861      test-mae:0.094615
[295]   train-mae:0.092726      test-mae:0.094484
[296]   train-mae:0.092449      test-mae:0.094203
[297]   train-mae:0.092282      test-mae:0.094037
[298]   train-mae:0.092226      test-mae:0.093981
[299]   train-mae:0.092123      test-mae:0.093878
[300]   train-mae:0.092067      test-mae:0.093824
[301]   train-mae:0.091997      test-mae:0.093746
[302]   train-mae:0.09185       test-mae:0.093598
[303]   train-mae:0.091709      test-mae:0.093456
[304]   train-mae:0.09148       test-mae:0.093219
[305]   train-mae:0.091329      test-mae:0.093065
[306]   train-mae:0.091141      test-mae:0.092883
[307]   train-mae:0.09105       test-mae:0.092797
[308]   train-mae:0.090836      test-mae:0.092587
[309]   train-mae:0.090622      test-mae:0.092367
[310]   train-mae:0.090361      test-mae:0.092109
[311]   train-mae:0.090301      test-mae:0.092051
[312]   train-mae:0.090171      test-mae:0.091918
[313]   train-mae:0.089979      test-mae:0.091728
[314]   train-mae:0.089803      test-mae:0.09155
[315]   train-mae:0.08961       test-mae:0.091357
[316]   train-mae:0.089443      test-mae:0.091194
[317]   train-mae:0.08939       test-mae:0.091144
[318]   train-mae:0.089345      test-mae:0.091095
[319]   train-mae:0.089121      test-mae:0.090865
[320]   train-mae:0.088955      test-mae:0.090693
[321]   train-mae:0.088777      test-mae:0.090514
[322]   train-mae:0.088704      test-mae:0.090436
[323]   train-mae:0.088564      test-mae:0.09029
[324]   train-mae:0.088475      test-mae:0.090197
[325]   train-mae:0.088428      test-mae:0.090153
[326]   train-mae:0.08838       test-mae:0.090105
[327]   train-mae:0.088097      test-mae:0.089818
[328]   train-mae:0.088079      test-mae:0.089803
[329]   train-mae:0.087941      test-mae:0.089668
[330]   train-mae:0.087859      test-mae:0.089589
[331]   train-mae:0.087761      test-mae:0.089501
[332]   train-mae:0.087706      test-mae:0.089445
[333]   train-mae:0.087608      test-mae:0.089343
[334]   train-mae:0.087551      test-mae:0.089284
[335]   train-mae:0.087511      test-mae:0.089242
[336]   train-mae:0.087394      test-mae:0.089127
[337]   train-mae:0.087245      test-mae:0.088973
[338]   train-mae:0.087193      test-mae:0.088924
[339]   train-mae:0.087057      test-mae:0.08879
[340]   train-mae:0.086888      test-mae:0.088625
[341]   train-mae:0.086519      test-mae:0.088256
[342]   train-mae:0.086424      test-mae:0.088165
[343]   train-mae:0.086353      test-mae:0.088092
[344]   train-mae:0.086295      test-mae:0.088036
[345]   train-mae:0.086249      test-mae:0.087992
[346]   train-mae:0.086202      test-mae:0.087943
[347]   train-mae:0.085953      test-mae:0.087701
[348]   train-mae:0.085787      test-mae:0.087531
[349]   train-mae:0.085752      test-mae:0.087496
[350]   train-mae:0.085612      test-mae:0.087355
[351]   train-mae:0.085305      test-mae:0.087045
[352]   train-mae:0.085253      test-mae:0.086992
[353]   train-mae:0.085223      test-mae:0.086964
```

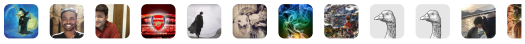
```
[354]   train-mae:0.085139      test-mae:0.086887
[355]   train-mae:0.08507       test-mae:0.08682
[356]   train-mae:0.085028      test-mae:0.086783
[357]   train-mae:0.084972      test-mae:0.086724
[358]   train-mae:0.084889      test-mae:0.086643
[359]   train-mae:0.084862      test-mae:0.086621
[360]   train-mae:0.084733      test-mae:0.086495
[361]   train-mae:0.084648      test-mae:0.086416
[362]   train-mae:0.084597      test-mae:0.086362
[363]   train-mae:0.084358      test-mae:0.086122
[364]   train-mae:0.08425       test-mae:0.086016
[365]   train-mae:0.084079      test-mae:0.085846
[366]   train-mae:0.083773      test-mae:0.085534
[367]   train-mae:0.083542      test-mae:0.085302
[368]   train-mae:0.08345       test-mae:0.085213
[369]   train-mae:0.083428      test-mae:0.085193
[370]   train-mae:0.083291      test-mae:0.085053
[371]   train-mae:0.083213      test-mae:0.084975
[372]   train-mae:0.083144      test-mae:0.084907
[373]   train-mae:0.083096      test-mae:0.084862
[374]   train-mae:0.082975      test-mae:0.084745
[375]   train-mae:0.082939      test-mae:0.08471
[376]   train-mae:0.082867      test-mae:0.084638
[377]   train-mae:0.082836      test-mae:0.084608
[378]   train-mae:0.08273       test-mae:0.084501
[379]   train-mae:0.082592      test-mae:0.084366
[380]   train-mae:0.082484      test-mae:0.084262
[381]   train-mae:0.082318      test-mae:0.084099
[382]   train-mae:0.082149      test-mae:0.083931
[383]   train-mae:0.08199       test-mae:0.083766
[384]   train-mae:0.081889      test-mae:0.08367
[385]   train-mae:0.081709      test-mae:0.083501
[386]   train-mae:0.081611      test-mae:0.0834
[387]   train-mae:0.081448      test-mae:0.083238
[388]   train-mae:0.081362      test-mae:0.083158
[389]   train-mae:0.081287      test-mae:0.083083
[390]   train-mae:0.081225      test-mae:0.083024
[391]   train-mae:0.081157      test-mae:0.082954
[392]   train-mae:0.081121      test-mae:0.082919
[393]   train-mae:0.081088      test-mae:0.082885
[394]   train-mae:0.080977      test-mae:0.082773
[395]   train-mae:0.080812      test-mae:0.08261
[396]   train-mae:0.080748      test-mae:0.082548
[397]   train-mae:0.080709      test-mae:0.082509
[398]   train-mae:0.080657      test-mae:0.082458
[399]   train-mae:0.080569      test-mae:0.08237
[400]   train-mae:0.080468      test-mae:0.082268
[401]   train-mae:0.080435      test-mae:0.082239
[402]   train-mae:0.08034       test-mae:0.082146
[403]   train-mae:0.080177      test-mae:0.08199
[404]   train-mae:0.080116      test-mae:0.081933
[405]   train-mae:0.079991      test-mae:0.081809
[406]   train-mae:0.079896      test-mae:0.081722
[407]   train-mae:0.079779      test-mae:0.081611
[408]   train-mae:0.079627      test-mae:0.081455
[409]   train-mae:0.0796        test-mae:0.081429
[410]   train-mae:0.079551      test-mae:0.081382
[411]   train-mae:0.07951       test-mae:0.081345
[412]   train-mae:0.079403      test-mae:0.081243
[413]   train-mae:0.079378      test-mae:0.081223
[414]   train-mae:0.079345      test-mae:0.081193
[415]   train-mae:0.079328      test-mae:0.081176
[416]   train-mae:0.079244      test-mae:0.081096
[417]   train-mae:0.079158      test-mae:0.081005
```

```
[418]   train-mae:0.07908       test-mae:0.080931
[419]   train-mae:0.078969      test-mae:0.080818
[420]   train-mae:0.078865      test-mae:0.080717
[421]   train-mae:0.078767      test-mae:0.080618
[422]   train-mae:0.078734      test-mae:0.080585
[423]   train-mae:0.078652      test-mae:0.080507
[424]   train-mae:0.078596      test-mae:0.080454
[425]   train-mae:0.078508      test-mae:0.080361
[426]   train-mae:0.078454      test-mae:0.080306
[427]   train-mae:0.078315      test-mae:0.080164
[428]   train-mae:0.078194      test-mae:0.080044
[429]   train-mae:0.078144      test-mae:0.079994
[430]   train-mae:0.078095      test-mae:0.079948
[431]   train-mae:0.078008      test-mae:0.079859
[432]   train-mae:0.077989      test-mae:0.07984
[433]   train-mae:0.077902      test-mae:0.079748
[434]   train-mae:0.077858      test-mae:0.079704
[435]   train-mae:0.077838      test-mae:0.079686
[436]   train-mae:0.077741      test-mae:0.079582
[437]   train-mae:0.077701      test-mae:0.079545
[438]   train-mae:0.077606      test-mae:0.079448
[439]   train-mae:0.077578      test-mae:0.079425
[440]   train-mae:0.077384      test-mae:0.079234
[441]   train-mae:0.077275      test-mae:0.07913
[442]   train-mae:0.077166      test-mae:0.079019
[443]   train-mae:0.077046      test-mae:0.078902
[444]   train-mae:0.076992      test-mae:0.078851
[445]   train-mae:0.076845      test-mae:0.078704
[446]   train-mae:0.076794      test-mae:0.078654
[447]   train-mae:0.076728      test-mae:0.078591
[448]   train-mae:0.076707      test-mae:0.078572
[449]   train-mae:0.076661      test-mae:0.078529
[450]   train-mae:0.076626      test-mae:0.078493
[451]   train-mae:0.076556      test-mae:0.078427
[452]   train-mae:0.076537      test-mae:0.078408
[453]   train-mae:0.076502      test-mae:0.078371
[454]   train-mae:0.076423      test-mae:0.078291
[455]   train-mae:0.076353      test-mae:0.078225
[456]   train-mae:0.076285      test-mae:0.078155
[457]   train-mae:0.076252      test-mae:0.078123
[458]   train-mae:0.076166      test-mae:0.078037
[459]   train-mae:0.076028      test-mae:0.077904
[460]   train-mae:0.07595       test-mae:0.077825
[461]   train-mae:0.075929      test-mae:0.077804
[462]   train-mae:0.075895      test-mae:0.077773
[463]   train-mae:0.07586       test-mae:0.077735
[464]   train-mae:0.075829      test-mae:0.077705
[465]   train-mae:0.075716      test-mae:0.077599
[466]   train-mae:0.075671      test-mae:0.077548
[467]   train-mae:0.075491      test-mae:0.077372
[468]   train-mae:0.075444      test-mae:0.077331
[469]   train-mae:0.075439      test-mae:0.077324
[470]   train-mae:0.075356      test-mae:0.077242
[471]   train-mae:0.075307      test-mae:0.077194
[472]   train-mae:0.075234      test-mae:0.077123
[473]   train-mae:0.075177      test-mae:0.077067
[474]   train-mae:0.075165      test-mae:0.077055
[475]   train-mae:0.075117      test-mae:0.077009
[476]   train-mae:0.075098      test-mae:0.076992
[477]   train-mae:0.074892      test-mae:0.076782
[478]   train-mae:0.074848      test-mae:0.076735
[479]   train-mae:0.074704      test-mae:0.076588
[480]   train-mae:0.07466       test-mae:0.076545
[481]   train-mae:0.074632      test-mae:0.07652
```

```
[482]   train-mae:0.074551      test-mae:0.07644
[483]   train-mae:0.07454       test-mae:0.076432
[484]   train-mae:0.074477      test-mae:0.076368
[485]   train-mae:0.074436      test-mae:0.076332
[486]   train-mae:0.074376      test-mae:0.076272
[487]   train-mae:0.074335      test-mae:0.076231
[488]   train-mae:0.074309      test-mae:0.076211
[489]   train-mae:0.0743        test-mae:0.076203
[490]   train-mae:0.074236      test-mae:0.076136
[491]   train-mae:0.074208      test-mae:0.076109
[492]   train-mae:0.074002      test-mae:0.075904
[493]   train-mae:0.073922      test-mae:0.075824
[494]   train-mae:0.073889      test-mae:0.07579
[495]   train-mae:0.073851      test-mae:0.075755
[496]   train-mae:0.073826      test-mae:0.075731
[497]   train-mae:0.07382       test-mae:0.075725
[498]   train-mae:0.073752      test-mae:0.075664
[499]   train-mae:0.073667      test-mae:0.075574
```

In [16]:

```
sample['sales'] = y_test
sample.to_csv('simple_starter.csv', index=False)
```

Data

Data Sources

- Store Item Demand F...
  - sample_submission....
  - test.csv
  - train.csv

Store Item Demand Forecasting Challenge

Predict 3 months of item sales at different stores

Last Updated: 9 months ago

About this Competition

The objective of this competition is to predict 3 months of item-level sales data at different store locations.

File descriptions

- **train.csv** - Training data
- **test.csv** - Test data (Note: the Public/Private split is time based)
- **sample_submission.csv** - a sample submission file in the correct format

Data fields

- **date** - Date of the sale data. There are no holiday effects or store closures.
- **store** - Store ID
- **item** - Item ID
- **sales** - Number of items sold at a particular store on a particular date.

Output Files

New Dataset | New Kernel | Download All

Output Files

simple_starter.csv

About this file

This file was created from a Kernel, it does not have a description.

simple_starter.csv

	id	sales
1	id	sales
2	0	1.903107
3	1	1.7500988
4	2	1.8341172
5	3	2.2187161
6	4	2.3308232
7	5	2.5820692
8	6	2.706662
9	7	1.9091817
10	8	1.7837312
11	9	1.847212
12	10	2.2395277
13	11	2.2989206
14	12	2.2414289
15	13	2.6800244
16	14	2.0092275
17	15	1.764232
18	16	1.7935684
19	17	1.9731407
20	18	2.3174295
21	19	2.5704925
22	20	2.4812877
23	21	1.9060259
24	22	1.8556648
25	23	1.7789823
26	24	2.2487745
27	25	2.1294398
28	26	2.3074546
29	27	2.6923084
30	28	1.8795525
31	29	1.836388
32	30	1.7481265

Run Info

Succeeded	True	Run Time	839.3 seconds
Exit Code	0	Queue Time	0 seconds
Docker Image Name	kaggle/python(Dockerfile)	Output Size	0
Timeout Exceeded	False	Used All Space	False
Failure Message			

Log

Download Log

```
Time  Line #  Log Message
5.4s       1  [NbConvertApp] Converting notebook script.ipynb to html
5.5s       2  [NbConvertApp] Executing notebook with kernel: python3
838.4s     3  [NbConvertApp] Writing 310789 bytes to __results__.html
838.4s     4
838.4s     6  Complete. Exited with code 0.
```

Comments (5)

Sort by

All Comments | Hotness

Click here to enter a comment...

Adithya Ram... • Posted on Latest Version • 8 months ago • Options • Reply 1

Hey ,Can you help me understand the significance of the columns ['median-store*item-month', 'mean-store*item-week',
'item-month-sum', 'store-month-sum'] ?
And also why have you shifted the features?

Aditya Soni • Posted on Latest Version • 8 months ago • Options • Reply 0

Have a look at this discusion too..
It's basically done so as to capture the previous months, weeks, years sales as they are not supossed to change abruptly...
https://www.kaggle.com/abhilashawasthi/feature-engineering-lgb-model/comments#362974

Leyla Bogdan... • Posted on Latest Version • 9 months ago • Options • Reply 0

Useful tutorial, thanks

Rashmi Jain • Posted on Latest Version • 5 months ago • Options • Reply 0

Hi,

Thanks for the tutorial, I have one question.

When you say:

col = [i for i in df.columns if i not in ['date','id']]
y = 'sales'

and

train*x, train*cv, y, y*cv = train*test*split(train[col],train[y], test*size=0.15, random_state=42)

Doesn't your train*x, train*cv contain your y variable as well i.e., sales?

9 months ago

This Comment was deleted.