<수요만 가지고 예측>

LASSO와 Ridge의 alpha 값은 Cross Validation을 통해 가장 적합한 alpha값 계산

LASSO :

alpha = 0.0012372103931348275

LASSO log변환 :

alpha = 0.00013763519994974846

Ridge :

alpha = 0.00013763519994974846

Ridge log 변환 :

alpha = 0.00013763519994974846

ARIMA:

p = 0

d = 1

q = 1

LSTM, GRU:

learning\_rate = 0.001

n\_step(stride) = 4

n\_hidden(hidden layer) = 32

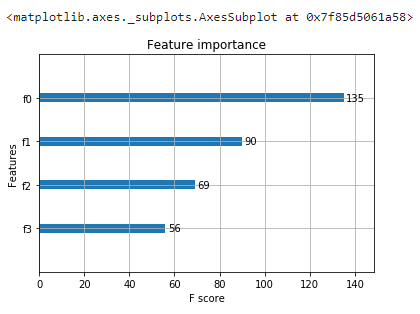
n\_multicell = 1~2

epoch = 5

batch\_size = 256

XGboost:

{'colsample\_bytree': 1.0,  
 'eta': 0.1,  
 'gamma': 0.0,  
 'max\_depth': 3,  
 'subsample': 0.5,  
 'objective': 'reg:linear',  
 'eval\_metric': 'rmse',  
 'tree\_method': 'gpu\_hist'}



최종

**xgboost**

|  |  |  |
| --- | --- | --- |
|  | 튜닝 범위 | 결과 값 |
| max\_depth | 3~8 | 3 |
| n\_estimators | 50~200 | 78 |
| booster | gbtree, gblinear, dart | dart |
| rmse |  | 2.13006 |

<수요+대여소정보만 가지고 예측>

LASSO와 Ridge의 alpha 값은 Cross Validation을 통해 가장 적합한 alpha값 계산

LASSO :

alpha =

LASSO log변환 :

alpha =

Ridge :

alpha =

Ridge log 변환 :

alpha =

ARIMA:

p = 0

d = 1

q = 1

XGboost :

{'colsample\_bytree': 1.0,  
 'eta': 0.1,  
 'gamma': 0.0,  
 'max\_depth': 3.0,  
 'subsample': 1.0,

'objective': 'reg:linear',  
 'eval\_metric': 'rmse',  
 'tree\_method': 'gpu\_hist'}

