# Regression_3112_1037_1037_Hanyang_Securities

February 8, 2021

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
import os
import sys
import tensorflow as tf

from sklearn.preprocessing import MinMaxScaler
from tqdm import tqdm
```

[2]:
```
# 예측할 종목은 한양증권(001750) 입니다
# 기간은 00-01-04 ~ 21-02-05 입니다
# Yahoo Finanace에서 다운로드 받아 null값을 제거한 데이터를 사용합니다

stock = pd.read_csv('C:\Jupyter_Project\Hanyang_Securities_F.csv')
df = stock.dropna()
df.head()
```

[2]:

| | Date | Open | High | Low | Close | Adj Close | Volume |
|---|---|---|---|---|---|---|---|
| 0 | 2000-01-04 | 6300 | 7100 | 6300 | 7000 | 1619.266357 | 56800 |
| 1 | 2000-01-05 | 6700 | 7100 | 6610 | 6700 | 1549.868774 | 52100 |
| 2 | 2000-01-06 | 7000 | 7000 | 6250 | 6300 | 1457.339844 | 64900 |
| 3 | 2000-01-07 | 6350 | 6600 | 6300 | 6370 | 1473.532349 | 61800 |
| 4 | 2000-01-10 | 6610 | 6700 | 6300 | 6500 | 1503.603882 | 56100 |

[3]:
```
df.tail()
```

[3]:

| | Date | Open | High | Low | Close | Adj Close | Volume |
|---|---|---|---|---|---|---|---|
| 5181 | 2021-02-01 | 9200 | 9480 | 9100 | 9380 | 9380.0 | 81355 |
| 5182 | 2021-02-02 | 9460 | 9810 | 9460 | 9700 | 9700.0 | 105755 |
| 5183 | 2021-02-03 | 9850 | 10200 | 9800 | 9990 | 9990.0 | 170966 |
| 5184 | 2021-02-04 | 10100 | 10200 | 9940 | 10150 | 10150.0 | 133504 |
| 5185 | 2021-02-05 | 10200 | 10800 | 10150 | 10650 | 10650.0 | 247224 |

[4]:
```
# OHLC를 Adj OHLC로 바꾸기 위한 비율입니다
# Adj OHLC는 과거의 절대가격을 현재 가격의 시점으로 보기위한 수정된 가격입니다
# 과거 발생한 액면분할과 현금배당을 반영한 Adj Close를 기준으로 조정합니다

ratio = df['Adj Close']/df['Close']
ratio
```

[4]:
```
0       0.231324
1       0.231324
2       0.231324
3       0.231324
4       0.231324
          ...
5181    1.000000
5182    1.000000
5183    1.000000
5184    1.000000
5185    1.000000
Length: 5186, dtype: float64
```

[5]:
```
df['Adj Open'] = df['Open']*ratio
df['Adj High'] = df['High']*ratio
df['Adj Low'] = df['Low']*ratio
```

[6]:
```
df.drop(['Open','High','Low','Close'], axis=1, inplace=True)
```

[7]:
```
df.rename(columns={'Adj Open':'Open', 'Adj High':'High', 'Adj Low':'Low', 'Adj␣
↪Close':'Close'}, inplace=True)
```

[8]:
```
df = df[['Open', 'High', 'Low', 'Close', 'Volume']]

df
```

[8]:

| | Open | High | Low | Close | Volume |
|---|---|---|---|---|---|
| 0 | 1457.339721 | 1642.398734 | 1457.339721 | 1619.266357 | 56800 |
| 1 | 1549.868774 | 1642.398253 | 1529.049641 | 1549.868774 | 52100 |
| 2 | 1619.266493 | 1619.266493 | 1445.773655 | 1457.339844 | 64900 |
| 3 | 1468.905874 | 1526.736814 | 1457.339686 | 1473.532349 | 61800 |
| 4 | 1529.049486 | 1549.868617 | 1457.339147 | 1503.603882 | 56100 |
| ... | ... | ... | ... | ... | ... |
| 5181 | 9200.000000 | 9480.000000 | 9100.000000 | 9380.000000 | 81355 |
| 5182 | 9460.000000 | 9810.000000 | 9460.000000 | 9700.000000 | 105755 |
| 5183 | 9850.000000 | 10200.000000 | 9800.000000 | 9990.000000 | 170966 |
| 5184 | 10100.000000 | 10200.000000 | 9940.000000 | 10150.000000 | 133504 |
| 5185 | 10200.000000 | 10800.000000 | 10150.000000 | 10650.000000 | 247224 |

[5186 rows x 5 columns]

[9]:
```
# MinMaxScaler 클래스를 사용하여 데이터를 스케일링 합니다.
# MinMaxScaler는 데이터의 최대값이 1, 최소값이 0이 되도록 변환합니다

scaler = MinMaxScaler()
scale_cols = ['Open', 'High', 'Low', 'Close', 'Volume']
df_scaled = scaler.fit_transform(df[scale_cols])

# 정규화가 완료된 데이터들은 pandas dataframe으로 변환합니다
# pandas는 시계열 자료에 대한 다양한 기능을 제공하여 LSTM에서 사용하는 window를 만들
때 유용합니다

df_scaled = pd.DataFrame(df_scaled)
df_scaled.columns = scale_cols

print(df_scaled)
```

```
          Open      High       Low     Close    Volume
0     0.069093  0.078420  0.072692  0.082280  0.020301
1     0.076891  0.078420  0.078924  0.076473  0.018620
2     0.082740  0.076587  0.071686  0.068730  0.023197
3     0.070068  0.069256  0.072692  0.070085  0.022088
4     0.075136  0.071089  0.072692  0.072601  0.020050
...        ...       ...       ...       ...       ...
5181  0.721622  0.699387  0.736878  0.731697  0.029080
5182  0.743534  0.725532  0.768164  0.758474  0.037804
5183  0.776402  0.756432  0.797711  0.782742  0.061119
5184  0.797472  0.756432  0.809878  0.796130  0.047725
5185  0.805899  0.803969  0.828128  0.837970  0.088383

[5186 rows x 5 columns]
```

[10]:
```
# window는 LSTM을 훈련하기 위한 단위로 고정된 사이즈를 가집니다
# window가 12개라면 과거 시간 데이터 12개를 사용해서 다음 시간 단위의 값의 예측하게 됩
니다
# 테스트 기간은 21일, 따라서 5299-21 : train / 21 : test

window_size = 20
TEST_SIZE = 1037
```

[11]:
```
train = df_scaled[:-TEST_SIZE]
test = df_scaled[-TEST_SIZE:]
```

[12]:
```
test.describe()
```

[12]:

| | Open | High | Low | Close | Volume |
|---|---|---|---|---|---|
| count | 1037.000000 | 1037.000000 | 1037.000000 | 1037.000000 | 1037.000000 |
| mean | 0.555879 | 0.526694 | 0.567086 | 0.551764 | 0.012608 |
| std | 0.079533 | 0.076859 | 0.080378 | 0.079329 | 0.036624 |

```
min       0.319620     0.315917     0.326686     0.314971     0.000169
25%       0.502964     0.475662     0.513952     0.498957     0.002769
50%       0.539574     0.509858     0.550341     0.535049     0.005645
75%       0.568782     0.538644     0.580720     0.565572     0.012352
max       0.831182     0.803969     0.828128     0.837970     1.000000
```

[13]:
```
# 정해진 window_size에 기반하여 20일 기간의 데이터 셋을 묶어준다

def make_dataset(data, label, window_size=20):
    feature_list = []
    label_list = []
    for i in range(len(data) - window_size):
        feature_list.append(np.array(data.iloc[i:i+window_size]))
        label_list.append(np.array(label.iloc[i+window_size]))
    return np.array(feature_list), np.array(label_list)
```

[14]:
```
from sklearn.model_selection import train_test_split

feature_cols = ['Open', 'High', 'Low', 'Volume']
label_cols = ['Close']

train_feature = train[feature_cols]
train_label = train[label_cols]

# train dataset
train_feature, train_label = make_dataset(train_feature, train_label, 20)

# train set : 모델을 학습하는 유일한 dataset
# validation set : 학습이 이미 완료된 모델을 검증하기 위한 dataset(비율 0.2)
# validation_split : X_train과 y_train에서 일정 비율을 분리하여 검증데이터로 사용
# 훈련 자체에는 반영되지 않고 훈련 과정을 지켜보기 위한 용도로 사용된다
# train_test_split 함수는 전체 데이터 셋 배열을 받아서 랜덤하게 훈련/테스트 데이터 셋
으로 분리해주는 함수입니다

x_train, x_valid, y_train, y_valid = train_test_split(train_feature,␣
 ↪train_label, test_size=0.25)

x_train.shape, x_valid.shape
# ((4206, 20, 4), (1052, 20, 4))

# test dataset : 학습과 검증이 완료된 모델의 성능을 평가하기 위한 dataset
x_test = test[feature_cols]
y_test = test[label_cols]

x_test.shape, y_test.shape
#     (21, 4),          (21, 1)
```

[14]:
```
((1037, 4), (1037, 1))
```

[15]:
```
x_test, y_test = make_dataset(x_test, y_test, 20)

x_test.shape, y_test.shape
# (21-20, 20, 4),     (21-20, 1)
```

[15]:
```
((1017, 20, 4), (1017, 1))
```

[16]:
```
x_train.shape, x_valid.shape, y_train.shape, y_valid.shape, x_test.shape, y_test.
 ↪shape
```

[16]:
```
((3096, 20, 4), (1033, 20, 4), (3096, 1), (1033, 1), (1017, 20, 4), (1017, 1))
```

[17]:
```
# print proportions
print('train: {}% | validation: {}% | test {}%'.format(round(len(y_train)/
 ↪len(df_scaled),2),
                                                       round(len(y_valid)/
 ↪len(df_scaled),2),
                                                       round(len(y_test)/
 ↪len(df_scaled),2)))
```

```
train: 0.6% | validation: 0.2% | test 0.2%
```

데이터 비율을 지정하는 방법에 대한 규칙은 없다. 다만 모델에 제공하는 데이터를 제한하면 학습할 수 있는 내용이 제한된다. 그러나 테스트 세트가 너무 작으면 모델 성능에 대한 정확한 추정치를 제공하지 않는다. 교차 검증을 통해 이 상황을 쉽게 처리할 수 있다

[18]:
```
from keras.models import Sequential
from keras.layers import Dense
from keras.callbacks import EarlyStopping, ModelCheckpoint
from keras.layers import LSTM

# LSTM은 RNN 알고리즘의 특별한 한 종류입니다.
# LSTM은 긴 의존기간을 필요로 하는 데이터를 학습하는데 효과적인 모델입니다
# 이 모델은 add함수를 사용하여 레이어들을 선형으로 쌓는 Sequential Model 입니다
# 16 메모리 셀을 가진 LSTM 레이어 하나와 Dense 레이어 하나(output)을 사용합니다
# input_shape는 input이 어떤 모양으로 들어올지에 대한 정보입니다. 데이터 개수는 중요하
지 않기에 window_size와 feature만 알려주면 된다
# train_feature.shape[1] = window_size / train_feature.shape[2] = ['Open',␣
 ↪'High', 'Low', 'Volume']
# 예측하고자 하는 target 개수가 하나이므로 Dense(1)dl 출력으로 사용됩니다

model = Sequential()
model.add(LSTM(20,
               input_shape=(x_test.shape[1], x_test.shape[2]),
               activation='relu',
```

```
                return_sequences=False)
          )
model.add(Dense(1))
```

[19]:
```
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm (LSTM)                  (None, 20)                2000
_________________________________________________________________
dense (Dense)                (None, 1)                 21
=================================================================
Total params: 2,021
Trainable params: 2,021
Non-trainable params: 0
_________________________________________________________________
```

[20]:
```
x_test.shape
```

[20]:
```
(1017, 20, 4)
```

[21]:
```
# val_loss가 10회 같을 시 early_stop, batch_size(=K)는 K문제 풀고 답보고 하는 식
# 위에서 모델을 구성한 후 compile 메서드를 호출하여 학습과정을 설정합니다
# optimizer : 훈련 과정을 설정한다
# loss : 최적화 과정에서 최소화될 손실 함수(loss function)을 설정합니다
# metrics : 훈련을 모니터링하기 위해 사용됩니다
# validation_data = 검증 데이터를 사용합니다. 각 에포크마다 정확도도 함께 출력됩니다
# 이 정확도는 훈련이 잘 되고 있는지를 보여줄 뿐이며 실제로 모델이 검증데이터를 학습하지는 않습니다
# 검증 데이터의 loss가 낮아지다가 높아지기 시작하면 overfitting의 신호입니다
# verbose / 0 : 출력 없음 / 1 : 훈련 진행도 보여주는 진행 막대 보여줌 / 2 : 미니 배치마다 손실 정보 출력

from numpy import array
from keras.models import Sequential
from keras.layers import Dense
from keras import backend as K

def RMSE(y_true, y_pred):
    return K.sqrt(K.mean(K.square(y_pred - y_true)))

def soft_acc(y_true, y_pred):
    return K.mean(K.equal(K.round(y_true), K.round(y_pred)))

def MPE(y_true, y_pred):
    return K.mean((y_true - y_pred) / y_true) * 100
```

```
def MSLE(y_true, y_pred):
    return K.mean(K.square(K.log(y_true+1) - K.log(y_pred+1)), axis=-1)

def RMSLE(y_true, y_pred):
    return K.sqrt(K.mean(K.square(K.log(y_true+1) - K.log(y_pred+1)), axis=-1))

def R2(y_true, y_pred):
    SS_res = K.sum(K.square(y_true - y_pred))
    SS_tot = K.sum(K.square(y_true - K.mean(y_true)))
    return ( 1 - SS_res/(SS_tot + K.epsilon()))

model.compile(loss = RMSE, optimizer='adam', metrics=[soft_acc, 'mse', 'mae',␣
 ↪RMSE, 'mape', MPE, MSLE, RMSLE, R2])

early_stop = EarlyStopping(monitor='val_loss', patience=10)
filename = os.path.join('tmp', 'ckeckpointer.ckpt')
checkpoint = ModelCheckpoint(filename, monitor='val_loss', verbose=1,␣
 ↪save_best_only=True, mode='auto')

history = model.fit(x_train, y_train,
                    epochs=200,
                    batch_size=128,
                    validation_data=(x_valid, y_valid),
                    callbacks=[early_stop, checkpoint])

# score_test를 만들면 테스트가 더이상 테스트가 아니고, 처음부터 모든 데이터에 대해 학습
한 것과 같기 때문에 일반화 할 수 없는 모델을 만드는 것과 같다.


score_train = model.evaluate(x_train, y_train, batch_size=128)
score_validation = model.evaluate(x_valid, y_valid, batch_size=128)
```

```
Epoch 1/200
25/25 [==============================] - 4s 81ms/step - loss: 0.3880 - soft_acc:
0.8765 - mse: 0.1513 - mae: 0.3176 - RMSE: 0.3878 - mape: 1869.4305 - MPE: -inf
- MSLE: 0.1038 - RMSLE: 0.2679 - R2: -2.7005 - val_loss: 0.3190 - val_soft_acc:
0.8687 - val_mse: 0.1020 - val_mae: 0.2512 - val_RMSE: 0.3133 - val_mape:
85.8997 - val_MPE: 38.0505 - val_MSLE: 0.0626 - val_RMSLE: 0.2014 - val_R2:
-1.2675

Epoch 00001: val_loss improved from inf to 0.31901, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 2/200
25/25 [==============================] - 0s 8ms/step - loss: 0.2765 - soft_acc:
0.8759 - mse: 0.0778 - mae: 0.2219 - RMSE: 0.2761 - mape: 26597.1178 - MPE: -inf
```

```
- MSLE: 0.0469 - RMSLE: 0.1762 - R2: -0.8739 - val_loss: 0.0954 - val_soft_acc:
0.9269 - val_mse: 0.0091 - val_mae: 0.0882 - val_RMSE: 0.0959 - val_mape:
143.6386 - val_MPE: -110.3536 - val_MSLE: 0.0055 - val_RMSLE: 0.0691 - val_R2:
0.7838

Epoch 00002: val_loss improved from 0.31901 to 0.09544, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 3/200
25/25 [==============================] - 0s 9ms/step - loss: 0.0830 - soft_acc:
0.9552 - mse: 0.0070 - mae: 0.0652 - RMSE: 0.0829 - mape: 16373.9791 - MPE: -inf
- MSLE: 0.0042 - RMSLE: 0.0518 - R2: 0.8344 - val_loss: 0.0638 - val_soft_acc:
0.9651 - val_mse: 0.0041 - val_mae: 0.0460 - val_RMSE: 0.0639 - val_mape:
117.1206 - val_MPE: -107.0139 - val_MSLE: 0.0025 - val_RMSLE: 0.0370 - val_R2:
0.9043

Epoch 00003: val_loss improved from 0.09544 to 0.06384, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 4/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0523 - soft_acc:
0.9657 - mse: 0.0028 - mae: 0.0407 - RMSE: 0.0522 - mape: 7257.6892 - MPE: -inf
- MSLE: 0.0017 - RMSLE: 0.0323 - R2: 0.9312 - val_loss: 0.0358 - val_soft_acc:
0.9668 - val_mse: 0.0013 - val_mae: 0.0208 - val_RMSE: 0.0363 - val_mape:
33.0372 - val_MPE: -26.7043 - val_MSLE: 5.3926e-04 - val_RMSLE: 0.0152 - val_R2:
0.9682

Epoch 00004: val_loss improved from 0.06384 to 0.03581, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 5/200
25/25 [==============================] - 0s 9ms/step - loss: 0.0275 - soft_acc:
0.9705 - mse: 7.8456e-04 - mae: 0.0165 - RMSE: 0.0275 - mape: 2546.9461 - MPE:
-inf - MSLE: 3.4091e-04 - RMSLE: 0.0120 - R2: 0.9813 - val_loss: 0.0278 -
val_soft_acc: 0.9555 - val_mse: 7.9019e-04 - val_mae: 0.0176 - val_RMSE: 0.0292
- val_mape: 10.9165 - val_MPE: 3.8695 - val_MSLE: 3.4885e-04 - val_RMSLE: 0.0127
- val_R2: 0.9786

Epoch 00005: val_loss improved from 0.03581 to 0.02781, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 6/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0243 - soft_acc:
0.9732 - mse: 6.1030e-04 - mae: 0.0152 - RMSE: 0.0243 - mape: 6330.5447 - MPE:
-inf - MSLE: 2.7210e-04 - RMSLE: 0.0110 - R2: 0.9858 - val_loss: 0.0234 -
val_soft_acc: 0.9694 - val_mse: 5.5810e-04 - val_mae: 0.0132 - val_RMSE: 0.0248
- val_mape: 10.3574 - val_MPE: -3.9734 - val_MSLE: 2.3525e-04 - val_RMSLE:
0.0094 - val_R2: 0.9845
```

```
Epoch 00006: val_loss improved from 0.02781 to 0.02339, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 7/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0210 - soft_acc:
0.9788 - mse: 4.5956e-04 - mae: 0.0123 - RMSE: 0.0210 - mape: 1740.6762 - MPE:
-inf - MSLE: 2.1007e-04 - RMSLE: 0.0089 - R2: 0.9891 - val_loss: 0.0231 -
val_soft_acc: 0.9712 - val_mse: 5.4654e-04 - val_mae: 0.0131 - val_RMSE: 0.0244
- val_mape: 10.3307 - val_MPE: -5.1015 - val_MSLE: 2.2895e-04 - val_RMSLE:
0.0093 - val_R2: 0.9851

Epoch 00007: val_loss improved from 0.02339 to 0.02314, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 8/200
25/25 [==============================] - 0s 9ms/step - loss: 0.0213 - soft_acc:
0.9790 - mse: 4.7408e-04 - mae: 0.0121 - RMSE: 0.0214 - mape: 333.6890 - MPE:
-inf - MSLE: 2.0479e-04 - RMSLE: 0.0087 - R2: 0.9889 - val_loss: 0.0221 -
val_soft_acc: 0.9720 - val_mse: 4.9916e-04 - val_mae: 0.0123 - val_RMSE: 0.0233
- val_mape: 10.0658 - val_MPE: -5.0937 - val_MSLE: 2.0983e-04 - val_RMSLE:
0.0087 - val_R2: 0.9862

Epoch 00008: val_loss improved from 0.02314 to 0.02208, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 9/200
25/25 [==============================] - 0s 9ms/step - loss: 0.0200 - soft_acc:
0.9738 - mse: 4.1408e-04 - mae: 0.0122 - RMSE: 0.0200 - mape: 12253.4055 - MPE:
-inf - MSLE: 1.8608e-04 - RMSLE: 0.0088 - R2: 0.9907 - val_loss: 0.0214 -
val_soft_acc: 0.9729 - val_mse: 4.6988e-04 - val_mae: 0.0116 - val_RMSE: 0.0226
- val_mape: 7.4660 - val_MPE: -2.4444 - val_MSLE: 1.9435e-04 - val_RMSLE: 0.0082
- val_R2: 0.9871

Epoch 00009: val_loss improved from 0.02208 to 0.02138, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 10/200
25/25 [==============================] - ETA: 0s - loss: 0.0199 - soft_acc:
0.9830 - mse: 4.1242e-04 - mae: 0.0112 - RMSE: 0.0199 - mape: 7.1733 - MPE:
-1.1896 - MSLE: 1.8140e-04 - RMSLE: 0.0081 - R2: 0.989 - 0s 8ms/step - loss:
0.0197 - soft_acc: 0.9823 - mse: 4.0163e-04 - mae: 0.0111 - RMSE: 0.0196 - mape:
293.6921 - MPE: -inf - MSLE: 1.7680e-04 - RMSLE: 0.0080 - R2: 0.9903 - val_loss:
0.0217 - val_soft_acc: 0.9703 - val_mse: 4.8178e-04 - val_mae: 0.0124 -
val_RMSE: 0.0229 - val_mape: 7.9044 - val_MPE: -1.5238 - val_MSLE: 2.0242e-04 -
val_RMSLE: 0.0088 - val_R2: 0.9866

Epoch 00010: val_loss did not improve from 0.02138
```

```
Epoch 11/200
25/25 [==============================] - 0s 10ms/step - loss: 0.0188 - soft_acc:
0.9789 - mse: 3.6545e-04 - mae: 0.0110 - RMSE: 0.0188 - mape: 1170.1042 - MPE:
-inf - MSLE: 1.6173e-04 - RMSLE: 0.0079 - R2: 0.9915 - val_loss: 0.0205 -
val_soft_acc: 0.9729 - val_mse: 4.3492e-04 - val_mae: 0.0110 - val_RMSE: 0.0217
- val_mape: 6.9849 - val_MPE: -1.8560 - val_MSLE: 1.7883e-04 - val_RMSLE: 0.0077
- val_R2: 0.9880

Epoch 00011: val_loss improved from 0.02138 to 0.02054, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 12/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0188 - soft_acc:
0.9783 - mse: 3.6398e-04 - mae: 0.0103 - RMSE: 0.0187 - mape: 780.7187 - MPE:
-inf - MSLE: 1.5757e-04 - RMSLE: 0.0073 - R2: 0.9914 - val_loss: 0.0203 -
val_soft_acc: 0.9720 - val_mse: 4.2666e-04 - val_mae: 0.0108 - val_RMSE: 0.0215
- val_mape: 6.6287 - val_MPE: -1.2335 - val_MSLE: 1.7467e-04 - val_RMSLE: 0.0076
- val_R2: 0.9883

Epoch 00012: val_loss improved from 0.02054 to 0.02032, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 13/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0186 - soft_acc:
0.9835 - mse: 3.5768e-04 - mae: 0.0101 - RMSE: 0.0187 - mape: 48.1493 - MPE:
-inf - MSLE: 1.5274e-04 - RMSLE: 0.0072 - R2: 0.9914 - val_loss: 0.0226 -
val_soft_acc: 0.9616 - val_mse: 5.2460e-04 - val_mae: 0.0138 - val_RMSE: 0.0234
- val_mape: 10.9509 - val_MPE: -8.8762 - val_MSLE: 2.1896e-04 - val_RMSLE:
0.0098 - val_R2: 0.9865

Epoch 00013: val_loss did not improve from 0.02032
Epoch 14/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0202 - soft_acc:
0.9802 - mse: 4.3468e-04 - mae: 0.0120 - RMSE: 0.0203 - mape: 974.8507 - MPE:
-inf - MSLE: 1.8545e-04 - RMSLE: 0.0086 - R2: 0.9901 - val_loss: 0.0200 -
val_soft_acc: 0.9712 - val_mse: 4.1229e-04 - val_mae: 0.0109 - val_RMSE: 0.0212
- val_mape: 6.3493 - val_MPE: 0.1461 - val_MSLE: 1.7014e-04 - val_RMSLE: 0.0077
- val_R2: 0.9885

Epoch 00014: val_loss improved from 0.02032 to 0.01998, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 15/200
25/25 [==============================] - 0s 9ms/step - loss: 0.0192 - soft_acc:
0.9808 - mse: 3.8097e-04 - mae: 0.0105 - RMSE: 0.0192 - mape: 375.2020 - MPE:
-inf - MSLE: 1.6094e-04 - RMSLE: 0.0075 - R2: 0.9913 - val_loss: 0.0198 -
val_soft_acc: 0.9720 - val_mse: 4.0708e-04 - val_mae: 0.0113 - val_RMSE: 0.0209
- val_mape: 6.3737 - val_MPE: 1.6391 - val_MSLE: 1.6979e-04 - val_RMSLE: 0.0080
```

```
- val_R2: 0.9889

Epoch 00015: val_loss improved from 0.01998 to 0.01985, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 16/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0175 - soft_acc:
0.9828 - mse: 3.1733e-04 - mae: 0.0099 - RMSE: 0.0175 - mape: 110.8316 - MPE:
-inf - MSLE: 1.4014e-04 - RMSLE: 0.0071 - R2: 0.9923 - val_loss: 0.0194 -
val_soft_acc: 0.9703 - val_mse: 3.9178e-04 - val_mae: 0.0106 - val_RMSE: 0.0204
- val_mape: 6.5014 - val_MPE: 2.6541 - val_MSLE: 1.6205e-04 - val_RMSLE: 0.0076
- val_R2: 0.9895

Epoch 00016: val_loss improved from 0.01985 to 0.01945, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 17/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0186 - soft_acc:
0.9826 - mse: 3.5554e-04 - mae: 0.0104 - RMSE: 0.0186 - mape: 690.9058 - MPE:
-inf - MSLE: 1.5286e-04 - RMSLE: 0.0075 - R2: 0.9918 - val_loss: 0.0190 -
val_soft_acc: 0.9703 - val_mse: 3.7433e-04 - val_mae: 0.0102 - val_RMSE: 0.0200
- val_mape: 5.9632 - val_MPE: 0.6051 - val_MSLE: 1.5380e-04 - val_RMSLE: 0.0072
- val_R2: 0.9900

Epoch 00017: val_loss improved from 0.01945 to 0.01901, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 18/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0174 - soft_acc:
0.9855 - mse: 3.2335e-04 - mae: 0.0097 - RMSE: 0.0174 - mape: 79.2738 - MPE: inf
- MSLE: 1.4420e-04 - RMSLE: 0.0070 - R2: 0.9926 - val_loss: 0.0188 -
val_soft_acc: 0.9729 - val_mse: 3.6781e-04 - val_mae: 0.0101 - val_RMSE: 0.0198
- val_mape: 6.3844 - val_MPE: -1.9029 - val_MSLE: 1.5069e-04 - val_RMSLE: 0.0071
- val_R2: 0.9901

Epoch 00018: val_loss improved from 0.01901 to 0.01884, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 19/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0173 - soft_acc:
0.9841 - mse: 3.2168e-04 - mae: 0.0099 - RMSE: 0.0173 - mape: 6298.2121 - MPE:
-inf - MSLE: 1.3933e-04 - RMSLE: 0.0072 - R2: 0.9928 - val_loss: 0.0184 -
val_soft_acc: 0.9720 - val_mse: 3.5183e-04 - val_mae: 0.0102 - val_RMSE: 0.0194
- val_mape: 8.1452 - val_MPE: -4.7442 - val_MSLE: 1.4613e-04 - val_RMSLE: 0.0073
- val_R2: 0.9904

Epoch 00019: val_loss improved from 0.01884 to 0.01842, saving model to
tmp\ckeckpointer.ckpt
```

```
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 20/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0172 - soft_acc:
0.9810 - mse: 3.0739e-04 - mae: 0.0096 - RMSE: 0.0172 - mape: 2382.2863 - MPE:
-inf - MSLE: 1.3669e-04 - RMSLE: 0.0069 - R2: 0.9925 - val_loss: 0.0189 -
val_soft_acc: 0.9712 - val_mse: 3.6875e-04 - val_mae: 0.0105 - val_RMSE: 0.0197
- val_mape: 7.5009 - val_MPE: -4.6798 - val_MSLE: 1.5221e-04 - val_RMSLE: 0.0074
- val_R2: 0.9903

Epoch 00020: val_loss did not improve from 0.01842
Epoch 21/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0171 - soft_acc:
0.9805 - mse: 2.9841e-04 - mae: 0.0100 - RMSE: 0.0170 - mape: 444.5656 - MPE:
-inf - MSLE: 1.3373e-04 - RMSLE: 0.0072 - R2: 0.9928 - val_loss: 0.0192 -
val_soft_acc: 0.9712 - val_mse: 3.8082e-04 - val_mae: 0.0107 - val_RMSE: 0.0200
- val_mape: 7.1463 - val_MPE: -4.2206 - val_MSLE: 1.5638e-04 - val_RMSLE: 0.0075
- val_R2: 0.9901

Epoch 00021: val_loss did not improve from 0.01842
Epoch 22/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0173 - soft_acc:
0.9823 - mse: 3.1395e-04 - mae: 0.0100 - RMSE: 0.0173 - mape: 7.5945 - MPE: -inf
- MSLE: 1.3748e-04 - RMSLE: 0.0072 - R2: 0.9925 - val_loss: 0.0180 -
val_soft_acc: 0.9720 - val_mse: 3.3642e-04 - val_mae: 0.0100 - val_RMSE: 0.0190
- val_mape: 5.9931 - val_MPE: 2.0080 - val_MSLE: 1.4036e-04 - val_RMSLE: 0.0071
- val_R2: 0.9908

Epoch 00022: val_loss improved from 0.01842 to 0.01803, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 23/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0167 - soft_acc:
0.9843 - mse: 2.8995e-04 - mae: 0.0095 - RMSE: 0.0167 - mape: 3838.1649 - MPE:
-inf - MSLE: 1.2549e-04 - RMSLE: 0.0068 - R2: 0.9931 - val_loss: 0.0179 -
val_soft_acc: 0.9720 - val_mse: 3.3162e-04 - val_mae: 0.0100 - val_RMSE: 0.0189
- val_mape: 5.8429 - val_MPE: 0.8401 - val_MSLE: 1.3806e-04 - val_RMSLE: 0.0071
- val_R2: 0.9909

Epoch 00023: val_loss improved from 0.01803 to 0.01790, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 24/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0166 - soft_acc:
0.9820 - mse: 2.8415e-04 - mae: 0.0094 - RMSE: 0.0165 - mape: 1147.4901 - MPE:
-inf - MSLE: 1.2470e-04 - RMSLE: 0.0067 - R2: 0.9933 - val_loss: 0.0179 -
val_soft_acc: 0.9703 - val_mse: 3.3238e-04 - val_mae: 0.0097 - val_RMSE: 0.0188
- val_mape: 5.8527 - val_MPE: 1.1157 - val_MSLE: 1.3738e-04 - val_RMSLE: 0.0069
- val_R2: 0.9911
```

```
Epoch 00024: val_loss did not improve from 0.01790
Epoch 25/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0182 - soft_acc:
0.9788 - mse: 3.3881e-04 - mae: 0.0110 - RMSE: 0.0182 - mape: 1339.5308 - MPE:
-inf - MSLE: 1.5040e-04 - RMSLE: 0.0080 - R2: 0.9918 - val_loss: 0.0178 -
val_soft_acc: 0.9703 - val_mse: 3.2603e-04 - val_mae: 0.0097 - val_RMSE: 0.0186
- val_mape: 5.8893 - val_MPE: 1.4070 - val_MSLE: 1.3537e-04 - val_RMSLE: 0.0069
- val_R2: 0.9913

Epoch 00025: val_loss improved from 0.01790 to 0.01777, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 26/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0169 - soft_acc:
0.9786 - mse: 3.0087e-04 - mae: 0.0098 - RMSE: 0.0169 - mape: 2374.9665 - MPE:
-inf - MSLE: 1.3281e-04 - RMSLE: 0.0071 - R2: 0.9932 - val_loss: 0.0177 -
val_soft_acc: 0.9703 - val_mse: 3.2450e-04 - val_mae: 0.0096 - val_RMSE: 0.0185
- val_mape: 6.2016 - val_MPE: -2.5596 - val_MSLE: 1.3371e-04 - val_RMSLE: 0.0067
- val_R2: 0.9914

Epoch 00026: val_loss improved from 0.01777 to 0.01770, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 27/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0159 - soft_acc:
0.9835 - mse: 2.6730e-04 - mae: 0.0086 - RMSE: 0.0159 - mape: 1857.4715 - MPE:
-inf - MSLE: 1.1299e-04 - RMSLE: 0.0062 - R2: 0.9933 - val_loss: 0.0172 -
val_soft_acc: 0.9738 - val_mse: 3.0634e-04 - val_mae: 0.0093 - val_RMSE: 0.0181
- val_mape: 5.9751 - val_MPE: -1.3736 - val_MSLE: 1.2709e-04 - val_RMSLE: 0.0066
- val_R2: 0.9917

Epoch 00027: val_loss improved from 0.01770 to 0.01721, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 28/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0165 - soft_acc:
0.9868 - mse: 2.8310e-04 - mae: 0.0089 - RMSE: 0.0165 - mape: 2446.6873 - MPE:
-inf - MSLE: 1.2080e-04 - RMSLE: 0.0064 - R2: 0.9933 - val_loss: 0.0175 -
val_soft_acc: 0.9712 - val_mse: 3.1682e-04 - val_mae: 0.0099 - val_RMSE: 0.0183
- val_mape: 8.6990 - val_MPE: -6.4365 - val_MSLE: 1.3339e-04 - val_RMSLE: 0.0071
- val_R2: 0.9916

Epoch 00028: val_loss did not improve from 0.01721
Epoch 29/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0168 - soft_acc:
0.9846 - mse: 2.9124e-04 - mae: 0.0091 - RMSE: 0.0169 - mape: 1831.8570 - MPE:
-inf - MSLE: 1.2642e-04 - RMSLE: 0.0066 - R2: 0.9932 - val_loss: 0.0171 -
```

```
val_soft_acc: 0.9729 - val_mse: 3.0252e-04 - val_mae: 0.0098 - val_RMSE: 0.0180
- val_mape: 5.7162 - val_MPE: 1.2424 - val_MSLE: 1.2721e-04 - val_RMSLE: 0.0069
- val_R2: 0.9918

Epoch 00029: val_loss improved from 0.01721 to 0.01712, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 30/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0160 - soft_acc:
0.9803 - mse: 2.6481e-04 - mae: 0.0090 - RMSE: 0.0159 - mape: 916.9306 - MPE:
-inf - MSLE: 1.1269e-04 - RMSLE: 0.0064 - R2: 0.9940 - val_loss: 0.0171 -
val_soft_acc: 0.9729 - val_mse: 3.0074e-04 - val_mae: 0.0098 - val_RMSE: 0.0180
- val_mape: 5.7832 - val_MPE: 1.7475 - val_MSLE: 1.2723e-04 - val_RMSLE: 0.0070
- val_R2: 0.9918

Epoch 00030: val_loss improved from 0.01712 to 0.01708, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 31/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0169 - soft_acc:
0.9802 - mse: 2.9786e-04 - mae: 0.0098 - RMSE: 0.0169 - mape: 357.1914 - MPE:
-inf - MSLE: 1.2675e-04 - RMSLE: 0.0070 - R2: 0.9931 - val_loss: 0.0179 -
val_soft_acc: 0.9712 - val_mse: 3.2770e-04 - val_mae: 0.0114 - val_RMSE: 0.0188
- val_mape: 7.3403 - val_MPE: 5.4104 - val_MSLE: 1.4370e-04 - val_RMSLE: 0.0082
- val_R2: 0.9911

Epoch 00031: val_loss did not improve from 0.01708
Epoch 32/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0169 - soft_acc:
0.9835 - mse: 2.9948e-04 - mae: 0.0098 - RMSE: 0.0169 - mape: 2251.9771 - MPE:
-inf - MSLE: 1.3007e-04 - RMSLE: 0.0071 - R2: 0.9931 - val_loss: 0.0165 -
val_soft_acc: 0.9746 - val_mse: 2.8149e-04 - val_mae: 0.0091 - val_RMSE: 0.0173
- val_mape: 7.3964 - val_MPE: -4.6022 - val_MSLE: 1.1812e-04 - val_RMSLE: 0.0065
- val_R2: 0.9925

Epoch 00032: val_loss improved from 0.01708 to 0.01651, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 33/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0153 - soft_acc:
0.9821 - mse: 2.4631e-04 - mae: 0.0090 - RMSE: 0.0153 - mape: 4213.2886 - MPE:
-inf - MSLE: 1.0888e-04 - RMSLE: 0.0065 - R2: 0.9943 - val_loss: 0.0164 -
val_soft_acc: 0.9729 - val_mse: 2.7815e-04 - val_mae: 0.0089 - val_RMSE: 0.0172
- val_mape: 5.5174 - val_MPE: -1.1442 - val_MSLE: 1.1596e-04 - val_RMSLE: 0.0063
- val_R2: 0.9926

Epoch 00033: val_loss improved from 0.01651 to 0.01643, saving model to
tmp\ckeckpointer.ckpt
```

```
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 34/200
25/25 [==============================] - 0s 9ms/step - loss: 0.0162 - soft_acc:
0.9847 - mse: 2.7626e-04 - mae: 0.0088 - RMSE: 0.0161 - mape: 372.5754 - MPE:
-inf - MSLE: 1.1942e-04 - RMSLE: 0.0062 - R2: 0.9937 - val_loss: 0.0163 -
val_soft_acc: 0.9746 - val_mse: 2.7321e-04 - val_mae: 0.0091 - val_RMSE: 0.0171
- val_mape: 6.1135 - val_MPE: -1.9166 - val_MSLE: 1.1473e-04 - val_RMSLE: 0.0064
- val_R2: 0.9926

Epoch 00034: val_loss improved from 0.01643 to 0.01628, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 35/200
25/25 [==============================] - 0s 9ms/step - loss: 0.0152 - soft_acc:
0.9839 - mse: 2.3452e-04 - mae: 0.0086 - RMSE: 0.0152 - mape: 6208.0151 - MPE:
-inf - MSLE: 1.0278e-04 - RMSLE: 0.0062 - R2: 0.9945 - val_loss: 0.0161 -
val_soft_acc: 0.9746 - val_mse: 2.6850e-04 - val_mae: 0.0088 - val_RMSE: 0.0170
- val_mape: 6.0768 - val_MPE: -2.6731 - val_MSLE: 1.1247e-04 - val_RMSLE: 0.0062
- val_R2: 0.9928

Epoch 00035: val_loss improved from 0.01628 to 0.01615, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 36/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0152 - soft_acc:
0.9842 - mse: 2.3840e-04 - mae: 0.0089 - RMSE: 0.0152 - mape: 437.0283 - MPE:
-inf - MSLE: 1.0652e-04 - RMSLE: 0.0064 - R2: 0.9941 - val_loss: 0.0162 -
val_soft_acc: 0.9712 - val_mse: 2.6981e-04 - val_mae: 0.0088 - val_RMSE: 0.0169
- val_mape: 5.4708 - val_MPE: -1.2429 - val_MSLE: 1.1285e-04 - val_RMSLE: 0.0062
- val_R2: 0.9928

Epoch 00036: val_loss did not improve from 0.01615
Epoch 37/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0157 - soft_acc:
0.9784 - mse: 2.5590e-04 - mae: 0.0089 - RMSE: 0.0157 - mape: 1690.8989 - MPE:
-inf - MSLE: 1.1299e-04 - RMSLE: 0.0064 - R2: 0.9939 - val_loss: 0.0163 -
val_soft_acc: 0.9738 - val_mse: 2.7187e-04 - val_mae: 0.0094 - val_RMSE: 0.0171
- val_mape: 5.4822 - val_MPE: 1.2683 - val_MSLE: 1.1566e-04 - val_RMSLE: 0.0067
- val_R2: 0.9926

Epoch 00037: val_loss did not improve from 0.01615
Epoch 38/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0157 - soft_acc:
0.9812 - mse: 2.5378e-04 - mae: 0.0089 - RMSE: 0.0157 - mape: 1160.1034 - MPE:
-inf - MSLE: 1.0897e-04 - RMSLE: 0.0064 - R2: 0.9939 - val_loss: 0.0160 -
val_soft_acc: 0.9764 - val_mse: 2.6231e-04 - val_mae: 0.0086 - val_RMSE: 0.0167
- val_mape: 5.2619 - val_MPE: 0.4669 - val_MSLE: 1.0971e-04 - val_RMSLE: 0.0061
- val_R2: 0.9930
```

```
Epoch 00038: val_loss improved from 0.01615 to 0.01596, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 39/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0153 - soft_acc:
0.9843 - mse: 2.4178e-04 - mae: 0.0084 - RMSE: 0.0152 - mape: 267.0457 - MPE:
-inf - MSLE: 1.0312e-04 - RMSLE: 0.0060 - R2: 0.9945 - val_loss: 0.0164 -
val_soft_acc: 0.9738 - val_mse: 2.7692e-04 - val_mae: 0.0090 - val_RMSE: 0.0171
- val_mape: 6.8158 - val_MPE: -4.2522 - val_MSLE: 1.1516e-04 - val_RMSLE: 0.0064
- val_R2: 0.9928

Epoch 00039: val_loss did not improve from 0.01596
Epoch 40/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0158 - soft_acc:
0.9825 - mse: 2.5872e-04 - mae: 0.0089 - RMSE: 0.0158 - mape: 211.0300 - MPE:
-inf - MSLE: 1.1244e-04 - RMSLE: 0.0064 - R2: 0.9939 - val_loss: 0.0171 -
val_soft_acc: 0.9703 - val_mse: 2.9813e-04 - val_mae: 0.0102 - val_RMSE: 0.0176
- val_mape: 8.0166 - val_MPE: -6.2196 - val_MSLE: 1.2719e-04 - val_RMSLE: 0.0073
- val_R2: 0.9924

Epoch 00040: val_loss did not improve from 0.01596
Epoch 41/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0156 - soft_acc:
0.9863 - mse: 2.5294e-04 - mae: 0.0088 - RMSE: 0.0156 - mape: 456.3876 - MPE:
-inf - MSLE: 1.0975e-04 - RMSLE: 0.0063 - R2: 0.9940 - val_loss: 0.0155 -
val_soft_acc: 0.9755 - val_mse: 2.4728e-04 - val_mae: 0.0085 - val_RMSE: 0.0163
- val_mape: 5.1752 - val_MPE: 0.4380 - val_MSLE: 1.0442e-04 - val_RMSLE: 0.0061
- val_R2: 0.9933

Epoch 00041: val_loss improved from 0.01596 to 0.01550, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 42/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0144 - soft_acc:
0.9886 - mse: 2.2062e-04 - mae: 0.0080 - RMSE: 0.0144 - mape: 3459.1965 - MPE:
-inf - MSLE: 9.8270e-05 - RMSLE: 0.0058 - R2: 0.9948 - val_loss: 0.0162 -
val_soft_acc: 0.9720 - val_mse: 2.6871e-04 - val_mae: 0.0090 - val_RMSE: 0.0168
- val_mape: 5.9219 - val_MPE: -3.0058 - val_MSLE: 1.1255e-04 - val_RMSLE: 0.0064
- val_R2: 0.9930

Epoch 00042: val_loss did not improve from 0.01550
Epoch 43/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0159 - soft_acc:
0.9824 - mse: 2.7160e-04 - mae: 0.0089 - RMSE: 0.0159 - mape: 603.6849 - MPE:
-inf - MSLE: 1.1694e-04 - RMSLE: 0.0064 - R2: 0.9937 - val_loss: 0.0164 -
val_soft_acc: 0.9720 - val_mse: 2.7514e-04 - val_mae: 0.0088 - val_RMSE: 0.0169
- val_mape: 5.1839 - val_MPE: -0.6919 - val_MSLE: 1.1364e-04 - val_RMSLE: 0.0062
```

```
- val_R2: 0.9929

Epoch 00043: val_loss did not improve from 0.01550
Epoch 44/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0160 - soft_acc:
0.9809 - mse: 2.6917e-04 - mae: 0.0091 - RMSE: 0.0160 - mape: 1159.5668 - MPE:
-inf - MSLE: 1.1488e-04 - RMSLE: 0.0065 - R2: 0.9937 - val_loss: 0.0155 -
val_soft_acc: 0.9764 - val_mse: 2.4720e-04 - val_mae: 0.0083 - val_RMSE: 0.0163
- val_mape: 5.2653 - val_MPE: -1.2176 - val_MSLE: 1.0375e-04 - val_RMSLE: 0.0059
- val_R2: 0.9934

Epoch 00044: val_loss did not improve from 0.01550
Epoch 45/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0151 - soft_acc:
0.9845 - mse: 2.3497e-04 - mae: 0.0083 - RMSE: 0.0151 - mape: 294.7668 - MPE:
-inf - MSLE: 9.9587e-05 - RMSLE: 0.0059 - R2: 0.9946 - val_loss: 0.0155 -
val_soft_acc: 0.9755 - val_mse: 2.4735e-04 - val_mae: 0.0089 - val_RMSE: 0.0163
- val_mape: 8.3404 - val_MPE: -6.1452 - val_MSLE: 1.0654e-04 - val_RMSLE: 0.0064
- val_R2: 0.9934

Epoch 00045: val_loss did not improve from 0.01550
Epoch 46/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0144 - soft_acc:
0.9860 - mse: 2.1565e-04 - mae: 0.0082 - RMSE: 0.0144 - mape: 2779.3721 - MPE:
-inf - MSLE: 9.5685e-05 - RMSLE: 0.0060 - R2: 0.9948 - val_loss: 0.0154 -
val_soft_acc: 0.9764 - val_mse: 2.4420e-04 - val_mae: 0.0084 - val_RMSE: 0.0161
- val_mape: 5.2409 - val_MPE: -0.8130 - val_MSLE: 1.0221e-04 - val_RMSLE: 0.0059
- val_R2: 0.9935

Epoch 00046: val_loss improved from 0.01550 to 0.01542, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 47/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0152 - soft_acc:
0.9878 - mse: 2.4544e-04 - mae: 0.0083 - RMSE: 0.0152 - mape: 651.2839 - MPE:
-inf - MSLE: 1.0323e-04 - RMSLE: 0.0059 - R2: 0.9943 - val_loss: 0.0152 -
val_soft_acc: 0.9764 - val_mse: 2.3618e-04 - val_mae: 0.0084 - val_RMSE: 0.0159
- val_mape: 5.6613 - val_MPE: 2.6650 - val_MSLE: 1.0111e-04 - val_RMSLE: 0.0061
- val_R2: 0.9937

Epoch 00047: val_loss improved from 0.01542 to 0.01518, saving model to
tmp\ckeckpointer.ckpt
INFO:tensorflow:Assets written to: tmp\ckeckpointer.ckpt\assets
Epoch 48/200
25/25 [==============================] - 0s 9ms/step - loss: 0.0152 - soft_acc:
0.9857 - mse: 2.4011e-04 - mae: 0.0083 - RMSE: 0.0152 - mape: 1562.8400 - MPE:
-inf - MSLE: 1.0407e-04 - RMSLE: 0.0060 - R2: 0.9943 - val_loss: 0.0152 -
val_soft_acc: 0.9729 - val_mse: 2.3743e-04 - val_mae: 0.0083 - val_RMSE: 0.0159
```

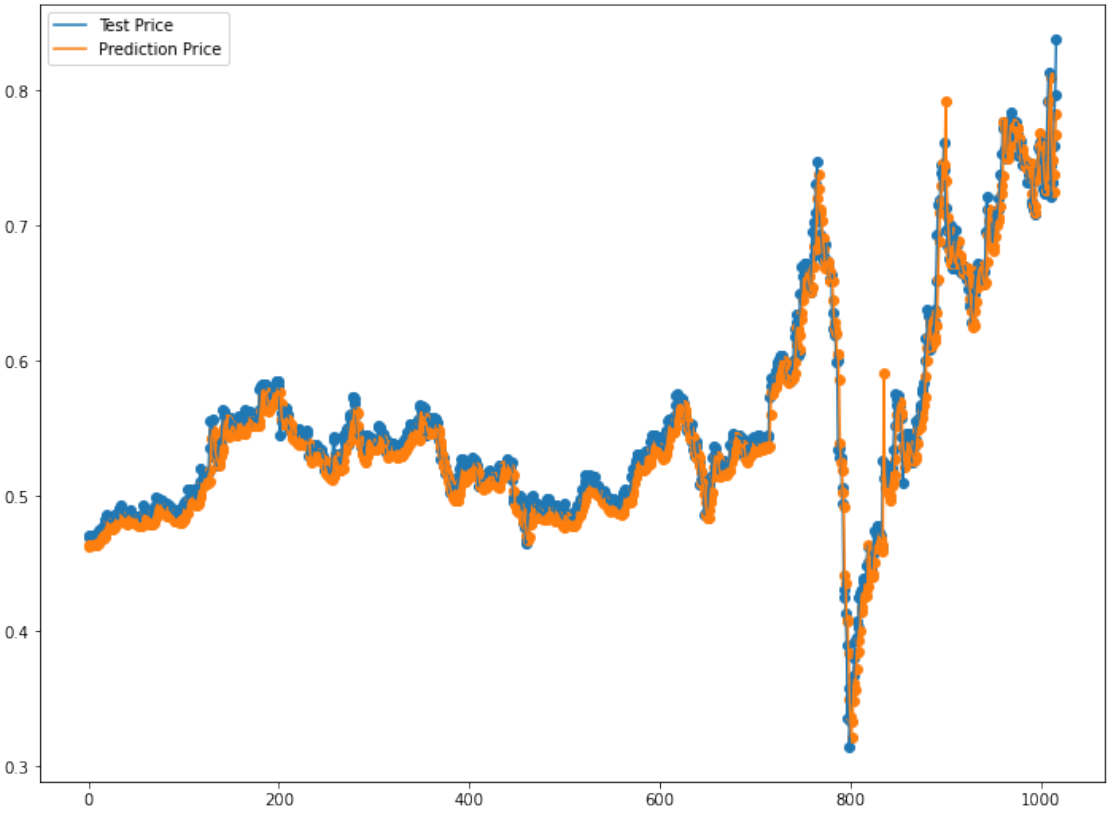
```
- val_mape: 5.2055 - val_MPE: 1.0778 - val_MSLE: 1.0079e-04 - val_RMSLE: 0.0059
- val_R2: 0.9937

Epoch 00048: val_loss did not improve from 0.01518
Epoch 49/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0162 - soft_acc:
0.9847 - mse: 2.6783e-04 - mae: 0.0090 - RMSE: 0.0162 - mape: 349.3843 - MPE:
-inf - MSLE: 1.1463e-04 - RMSLE: 0.0065 - R2: 0.9937 - val_loss: 0.0160 -
val_soft_acc: 0.9738 - val_mse: 2.6133e-04 - val_mae: 0.0101 - val_RMSE: 0.0168
- val_mape: 6.5535 - val_MPE: 4.6092 - val_MSLE: 1.1626e-04 - val_RMSLE: 0.0073
- val_R2: 0.9929

Epoch 00049: val_loss did not improve from 0.01518
Epoch 50/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0147 - soft_acc:
0.9814 - mse: 2.2277e-04 - mae: 0.0083 - RMSE: 0.0147 - mape: 2574.2157 - MPE:
-inf - MSLE: 9.6752e-05 - RMSLE: 0.0060 - R2: 0.9948 - val_loss: 0.0155 -
val_soft_acc: 0.9720 - val_mse: 2.4425e-04 - val_mae: 0.0086 - val_RMSE: 0.0161
- val_mape: 6.2979 - val_MPE: -3.6401 - val_MSLE: 1.0329e-04 - val_RMSLE: 0.0062
- val_R2: 0.9936

Epoch 00050: val_loss did not improve from 0.01518
Epoch 51/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0153 - soft_acc:
0.9823 - mse: 2.3845e-04 - mae: 0.0091 - RMSE: 0.0154 - mape: 127.7951 - MPE:
-inf - MSLE: 1.0409e-04 - RMSLE: 0.0065 - R2: 0.9945 - val_loss: 0.0152 -
val_soft_acc: 0.9764 - val_mse: 2.3685e-04 - val_mae: 0.0084 - val_RMSE: 0.0159
- val_mape: 5.4101 - val_MPE: 2.1620 - val_MSLE: 1.0086e-04 - val_RMSLE: 0.0060
- val_R2: 0.9937

Epoch 00051: val_loss did not improve from 0.01518
Epoch 52/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0150 - soft_acc:
0.9864 - mse: 2.3084e-04 - mae: 0.0079 - RMSE: 0.0150 - mape: 1809.9790 - MPE:
-inf - MSLE: 9.6272e-05 - RMSLE: 0.0057 - R2: 0.9946 - val_loss: 0.0168 -
val_soft_acc: 0.9712 - val_mse: 2.8834e-04 - val_mae: 0.0114 - val_RMSE: 0.0176
- val_mape: 7.4638 - val_MPE: 5.9537 - val_MSLE: 1.3111e-04 - val_RMSLE: 0.0083
- val_R2: 0.9923

Epoch 00052: val_loss did not improve from 0.01518
Epoch 53/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0153 - soft_acc:
0.9819 - mse: 2.3990e-04 - mae: 0.0092 - RMSE: 0.0153 - mape: 2224.9102 - MPE:
-inf - MSLE: 1.0726e-04 - RMSLE: 0.0067 - R2: 0.9941 - val_loss: 0.0153 -
val_soft_acc: 0.9746 - val_mse: 2.4133e-04 - val_mae: 0.0090 - val_RMSE: 0.0161
- val_mape: 5.2285 - val_MPE: 1.6214 - val_MSLE: 1.0401e-04 - val_RMSLE: 0.0064
- val_R2: 0.9934
```

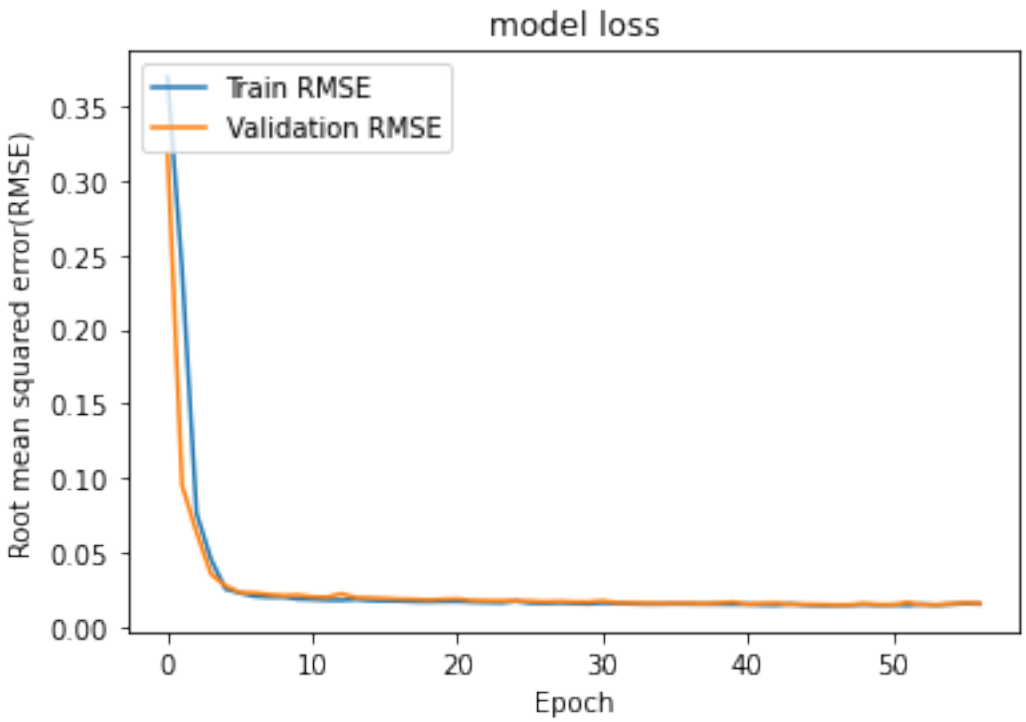
```
Epoch 00053: val_loss did not improve from 0.01518
Epoch 54/200
25/25 [==============================] - 0s 9ms/step - loss: 0.0147 - soft_acc:
0.9833 - mse: 2.2280e-04 - mae: 0.0085 - RMSE: 0.0147 - mape: 42.4492 - MPE: inf
- MSLE: 9.7055e-05 - RMSLE: 0.0061 - R2: 0.9947 - val_loss: 0.0152 -
val_soft_acc: 0.9746 - val_mse: 2.3687e-04 - val_mae: 0.0088 - val_RMSE: 0.0159
- val_mape: 8.8350 - val_MPE: -6.6440 - val_MSLE: 1.0291e-04 - val_RMSLE: 0.0064
- val_R2: 0.9936

Epoch 00054: val_loss did not improve from 0.01518
Epoch 55/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0157 - soft_acc:
0.9818 - mse: 2.4979e-04 - mae: 0.0093 - RMSE: 0.0157 - mape: 2354.6518 - MPE:
-inf - MSLE: 1.1090e-04 - RMSLE: 0.0068 - R2: 0.9943 - val_loss: 0.0159 -
val_soft_acc: 0.9720 - val_mse: 2.5746e-04 - val_mae: 0.0091 - val_RMSE: 0.0164
- val_mape: 7.3763 - val_MPE: -5.4096 - val_MSLE: 1.0940e-04 - val_RMSLE: 0.0066
- val_R2: 0.9934

Epoch 00055: val_loss did not improve from 0.01518
Epoch 56/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0155 - soft_acc:
0.9796 - mse: 2.4692e-04 - mae: 0.0096 - RMSE: 0.0155 - mape: 404.0689 - MPE:
-inf - MSLE: 1.1181e-04 - RMSLE: 0.0070 - R2: 0.9942 - val_loss: 0.0165 -
val_soft_acc: 0.9703 - val_mse: 2.7844e-04 - val_mae: 0.0101 - val_RMSE: 0.0170
- val_mape: 8.1351 - val_MPE: -6.4739 - val_MSLE: 1.1985e-04 - val_RMSLE: 0.0073
- val_R2: 0.9930

Epoch 00056: val_loss did not improve from 0.01518
Epoch 57/200
25/25 [==============================] - 0s 8ms/step - loss: 0.0150 - soft_acc:
0.9778 - mse: 2.3164e-04 - mae: 0.0097 - RMSE: 0.0150 - mape: 190.1999 - MPE:
-inf - MSLE: 1.0646e-04 - RMSLE: 0.0071 - R2: 0.9946 - val_loss: 0.0161 -
val_soft_acc: 0.9746 - val_mse: 2.6542e-04 - val_mae: 0.0105 - val_RMSE: 0.0169
- val_mape: 9.0060 - val_MPE: 7.3535 - val_MSLE: 1.2074e-04 - val_RMSLE: 0.0077
- val_R2: 0.9929

Epoch 00057: val_loss did not improve from 0.01518
25/25 [==============================] - 0s 2ms/step - loss: 0.0154 - soft_acc:
0.9859 - mse: 2.4505e-04 - mae: 0.0098 - RMSE: 0.0154 - mape: 802.3535 - MPE:
inf - MSLE: 1.1239e-04 - RMSLE: 0.0073 - R2: 0.9943
9/9 [==============================] - 0s 3ms/step - loss: 0.0161 - soft_acc:
0.9746 - mse: 2.6542e-04 - mae: 0.0105 - RMSE: 0.0169 - mape: 9.0060 - MPE:
7.3535 - MSLE: 1.2074e-04 - RMSLE: 0.0077 - R2: 0.9929
```

[22]:
```
pred = model.predict(x_test)
pred.shape
```

[22]: (1017, 1)

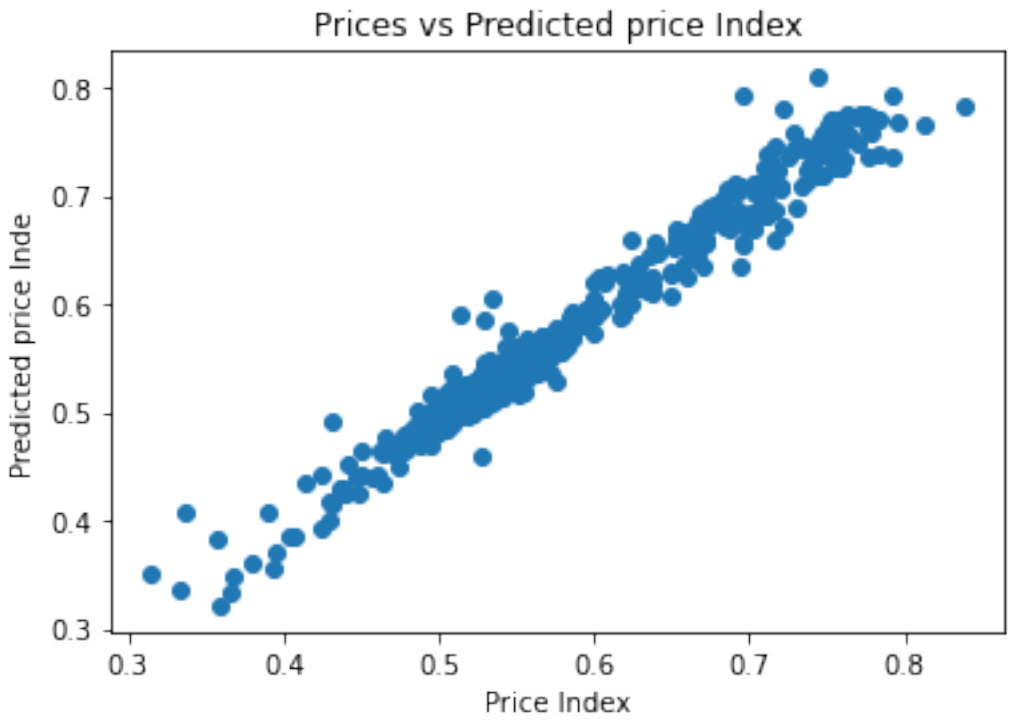
[23]:
```
pred = model.predict(x_test)
pred.shape

plt.figure(figsize=(12,9))
plt.plot(np.asarray(y_test), label='Test Price')
plt.plot(pred, label='Prediction Price')
x_values = list(range(1017))
plt.scatter(x_values, np.asarray(y_test))
plt.scatter(x_values, pred)
plt.legend()
plt.show()

plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('model loss')
plt.ylabel('Root mean squared error(RMSE)')
plt.xlabel('Epoch')
plt.legend(['Train RMSE', 'Validation RMSE'], loc='upper left')
plt.show()
```



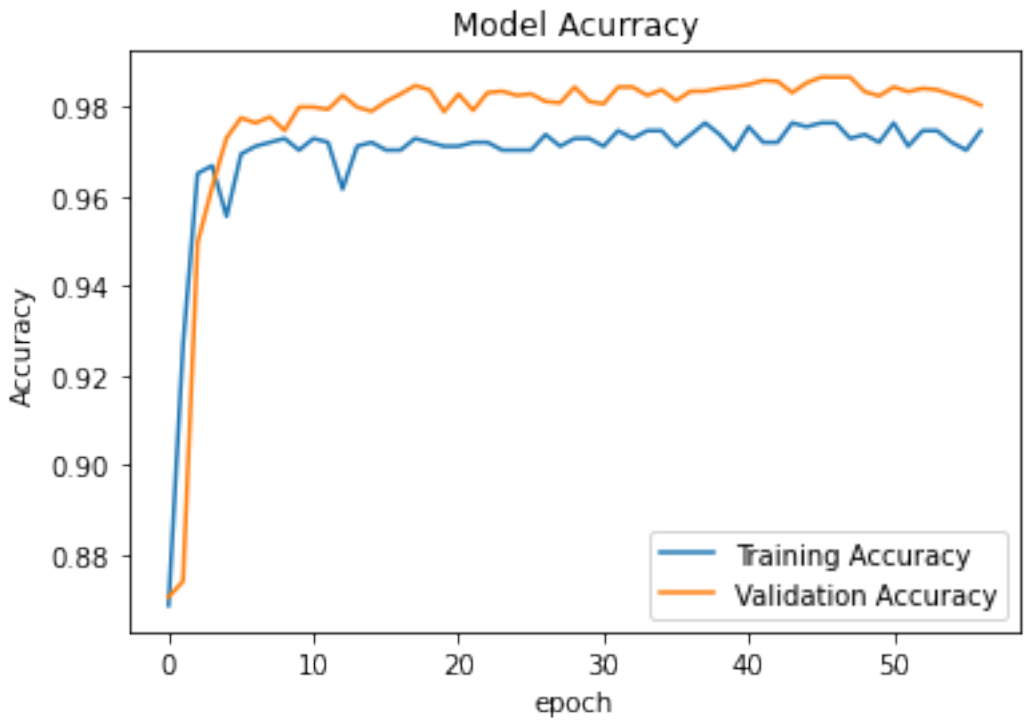



[24]:
```
# 원래값과 예측 값이 일치하면 직선에 가깝게 분포가 된다

%matplotlib inline
import matplotlib.pyplot as plt

plt.scatter(np.asarray(y_test), pred)
plt.xlabel("Price Index")
plt.ylabel("Predicted price Inde")
plt.title("Prices vs Predicted price Index")
```

[24]:
```
Text(0.5, 1.0, 'Prices vs Predicted price Index')
```




[25]:
```
plt.plot(history.history['val_soft_acc'])
plt.plot(history.history['soft_acc'])
plt.title('Model Acurracy')
plt.ylabel('Accuracy')
plt.xlabel('epoch')
plt.legend(['Training Accuracy', 'Validation Accuracy'], loc='lower right')
plt.show()
```




[26]:
```
accuracy_train = 100*score_train[1]
accuracy_validation = 100*score_validation[1]

print('train accuracy: %.4f%%' % accuracy_train)
print('validation accuracy: %.4f%%' % accuracy_validation)
```

```
train accuracy: 98.5937%
validation accuracy: 97.4634%
```

R2_Score가 train data에서 계산되면 모델이 샘플 내 분산을 얼마나 설명하는지에 대해 알려주고 test data에서 계산되면 모델의 예측품질에 대해 알려준다. 기계학습 세계에서는 검증과 테스트 정확도를 모두 제시하는 것이 매우 일반적이지만 가장 중요한 것은 테스트 정확도입니다. 그러나 한 쪽에서 낮은 R2 점수를 받고 그렇지 않다면 뭔가 꺼진 것입니다. R2 test « R2training이면 모델이 잘 일반화되지 않았음을 나타낸다. 예를들어 테스트 세트에 future 데이타 만 포함되어 있으면 모델이 잘 외삽되지 않는 것처럼 보입니다. 결론적으로 당신은 그것을 비교해야 합니다. 그러나 대부분의 경우 가장 관심이 있는 테스트 세트 결과입니다

[27]:
```
import numpy as np
from sklearn.metrics import r2_score
from sklearn.metrics import mean_squared_log_error

Y = np.asarray(y_test)
Y_hat = pred
```

```
def MSE(y_true, y_pred):
    return np.mean(np.square((y_true - y_pred)))

def MAE(y_true, y_pred):
    return np.mean(np.abs((y_true - y_pred)))

def RMSE(y_true, y_pred):
    return np.sqrt(np.mean((y_pred-y_true)**2))

def MAPE(y_true, y_pred):
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100

def MPE(y_true, y_pred):
    return np.mean((y_true - y_pred) / y_true) * 100

def root_mean_squared_log_error(y_true, y_pred):
    return np.sqrt(mean_squared_log_error(y_true, y_pred))

print('R2_Score')
print('-' * 40)
print('train error: {} |\nvalid error: {} |\ntest error : {}\n'.
 ↪format(score_train[9], score_validation[9], r2_score(Y, Y_hat)))

print('Mean Squared Error')
print('-' * 40)
print('train error: {} |\nvalid error: {} |\ntest error : {}\n'.
 ↪format(score_train[2], score_validation[2], MSE(Y, Y_hat)))

print('Mean Absolute Error')
print('-' * 40)
print('train error: {} |\nvalid error: {} |\ntest error : {}\n'.
 ↪format(score_train[3], score_validation[3], MAE(Y, Y_hat)))

print('Root Mean Squared Error')
print('-' * 40)
print('train error: {} |\nvalid error: {} |\ntest error : {}\n'.
 ↪format(score_train[4], score_validation[3], RMSE(Y, Y_hat)))

print('Mean Squared Logarithmic Error')
print('-' * 40)
print('train error: {} |\nvalid error: {} |\ntest error : {}\n'.
 ↪format(score_train[7], score_validation[7], mean_squared_log_error(Y, Y_hat)))

print('Root Mean Squared Logarithmic Error')
print('-' * 40)
```

```
print('train error: {} |\nvalid error: {} |\ntest error : {}\n'.
 ↪format(score_train[8], score_validation[8], root_mean_squared_log_error(Y,␣
 ↪Y_hat)))

print('Mean Absolute Percentage Error')
print('-' * 40)
print('train error: {} |\nvalid error: {} |\ntest error : {}\n'.
 ↪format(score_train[5], score_validation[3], MAPE(Y, Y_hat)))

print('Mean Percentage Error')
print('-' * 40)
print('train error: {} |\nvalid error: {} |\ntest error : {}\n'.
 ↪format(score_train[6], score_validation[3], MPE(Y, Y_hat)))
```

```
R2_Score
----------------------------------------
train error: 0.9942859411239624 |
valid error: 0.992901086807251 |
test error : 0.969569823591676

Mean Squared Error
----------------------------------------
train error: 0.00024504572502337396 |
valid error: 0.0002654152922332287 |
test error : 0.00019094254030111796

Mean Absolute Error
----------------------------------------
train error: 0.009836779907345772 |
valid error: 0.01048602070659399 |
test error : 0.009594846344983357

Root Mean Squared Error
----------------------------------------
train error: 0.015370653010904789 |
valid error: 0.01048602070659399 |
test error : 0.013818195985768835

Mean Squared Logarithmic Error
----------------------------------------
train error: 0.00011239355808356777 |
valid error: 0.0001207444875035435 |
test error : 7.643460276778266e-05

Root Mean Squared Logarithmic Error
----------------------------------------
train error: 0.007310476154088974 |
```

```
valid error: 0.007747378665953875 |
test error : 0.008742688532012488

Mean Absolute Percentage Error
---------------------------------------
train error: 802.353515625 |
valid error: 0.01048602070659399 |
test error : 1.7292413517444112

Mean Percentage Error
---------------------------------------
train error: inf |
valid error: 0.01048602070659399 |
test error : 1.037522508518689
```

[28]:
```
# count : 개수, std : 표준편차

test[label_cols].describe()
```

[28]:
```
              Close
count  1037.000000
mean      0.551764
std       0.079329
min       0.314971
25%       0.498957
50%       0.535049
75%       0.565572
max       0.837970
```

[ ]:
```

```