#임포트 및 세팅

import sys

import os

import IPython

import multiprocessing

import copy

import pickle

import warnings

from datetime import datetime

from time import time

from matplotlib import font\_manager as fm, rc, rcParams

import matplotlib.pyplot as plt

import seaborn as sns

import numpy as np

from numpy import array, nan, random as rnd, where as which

import pandas as pd

from pandas import DataFrame as dataframe, Series as series, isna, isnull, read\_csv

from pandas.tseries.offsets import DateOffset

from scipy.special import boxcox1p

from scipy.stats import skew

import pandas as pd

import numpy as np

import os

import FinanceDataReader as fdr

from sklearn.linear\_model import LinearRegression

import xgboost as xgb

from catboost import CatBoostClassifier

from catboost import CatBoostRegressor

from xgboost import XGBRegressor

from lightgbm import LGBMRegressor

from tqdm import tqdm

# display setting

warnings.filterwarnings(action='ignore')

rcParams['axes.unicode\_minus'] = False

pd.set\_option('display.max\_columns', 20)

pd.set\_option('display.max\_rows', 400)

pd.set\_option('display.width', 1500)

# plot setting

# 폰트 경로 본인 PC에 맞춰 설정

font\_path = 'c:/windows/fonts/KoPub Dotum Light.ttf'

font\_obj = fm.FontProperties(fname=font\_path, size=12).get\_name()

rc('font', family=font\_obj)

myColors = sns.hls\_palette(20, s=0.4)

#기본 세팅 및 파일 불러오기

# Get Stock List

path = 'C:/Users/nashv/Desktop/asiaE\_FinalProject/DACON/open\_week4'

list\_name = 'Stock\_List.csv'

sample\_name = 'sample\_submission\_week4.csv'

# path와 list\_name을 조인하여 csv 파일 불러옴

stock\_list = pd.read\_csv(os.path.join(path ,list\_name))

# 종목코드를 6자리로 맞추고 부족하면 앞에 0으로 채움

stock\_list['종목코드'] = stock\_list['종목코드'].apply(lambda x : str(x).zfill(6))

stock\_list

# Get Data & Modeling

start\_date = '20210104'

end\_date = '20210917'

# 시작 날짜를 weekday() 메서드를 사용하여 Python에서 요일 이름 가져 옴

start\_weekday = pd.to\_datetime(start\_date).weekday()

# 마지막 날짜를 53주 내에서 몇주차 인지 전환

max\_weeknum = pd.to\_datetime(end\_date).strftime('%V')

Business\_days = pd.DataFrame(pd.date\_range(start\_date ,end\_date ,freq='B'), columns = ['Date'])

print(f'WEEKDAY of "start\_date" : {start\_weekday}')

print(f'NUM of WEEKS to "end\_date" : {max\_weeknum}')

print(f'HOW MANY "Business\_days" : {Business\_days.shape}', )

display(Business\_days.head())

# 전체 모델링

sample\_name = 'sample\_submission\_week4.csv'

sample\_submission = pd.read\_csv(os.path.join(path ,sample\_name))

# CatBoost 3주차

model = CatBoostRegressor()

for code in tqdm(stock\_list['종목코드'].values):

data = fdr.DataReader(code, start = start\_date, end = end\_date)[['Close']].reset\_index()

data = pd.merge(Business\_days, data, how = 'outer')

data['weekday'] = data.Date.apply(lambda x : x.weekday())

data['weeknum'] = data.Date.apply(lambda x : x.strftime('%V'))

data.Close = data.Close.ffill()

data = pd.pivot\_table(data = data, values = 'Close', columns = 'weekday', index = 'weeknum')

x = data.iloc[0:-2].to\_numpy() # 2021년 1월 04일 ~ 2021년 8월 13일까지의 데이터로

y = data.iloc[1:-1].to\_numpy() # 2021년 1월 11일 ~ 2021년 8월 20일까지의 데이터를 학습한다.

y\_0 = y[: ,0]

y\_1 = y[: ,1]

y\_2 = y[: ,2]

y\_3 = y[: ,3]

y\_4 = y[: ,4]

y\_values = [y\_0, y\_1, y\_2, y\_3, y\_4]

x\_public = data.iloc[-2].to\_numpy() # 2021년 8월 23일부터 8월 27일까지의 데이터를 예측할 것이다.

predictions = []

for y\_value in y\_values :

model.fit(x ,y\_value)

prediction = model.predict(np.expand\_dims(x\_public ,0))

predictions.append(prediction[0])

sample\_submission.loc[: ,code] = predictions \* 2

sample\_submission.isna().sum().sum()

sample\_submission.columns

# 결과:

# Index(['Day', '000060', '000080', '000100', '000120', '000150', '000240',

# '000250', '000270', '000660',

# ...

# '330860', '336260', '336370', '347860', '348150', '348210', '352820',

# '357780', '363280', '950130'],

# dtype='object', length=377)

columns = list(sample\_submission.columns[1:])

columns = ['Day'] + [str(x).zfill(6) for x in columns]

sample\_submission.columns = columns

sample\_submission.to\_csv('BASELINE\_CatBoostRegressor\_Week4.csv' ,index=False)

sample\_submission

# xgboost 3추자

model = XGBRegressor()

for code in tqdm(stock\_list['종목코드'].values):

data = fdr.DataReader(code, start = start\_date, end = end\_date)[['Close']].reset\_index()

data = pd.merge(Business\_days, data, how = 'outer')

data['weekday'] = data.Date.apply(lambda x : x.weekday())

data['weeknum'] = data.Date.apply(lambda x : x.strftime('%V'))

data.Close = data.Close.ffill()

data = pd.pivot\_table(data = data, values = 'Close', columns = 'weekday', index = 'weeknum')

x = data.iloc[0:-2].to\_numpy() # 2021년 1월 04일 ~ 2021년 8월 13일까지의 데이터로

y = data.iloc[1:-1].to\_numpy() # 2021년 1월 11일 ~ 2021년 8월 20일까지의 데이터를 학습한다.

y\_0 = y[: ,0]

y\_1 = y[: ,1]

y\_2 = y[: ,2]

y\_3 = y[: ,3]

y\_4 = y[: ,4]

y\_values = [y\_0, y\_1, y\_2, y\_3, y\_4]

x\_public = data.iloc[-2].to\_numpy() # 2021년 8월 23일부터 8월 27일까지의 데이터를 예측할 것이다.

predictions = []

for y\_value in y\_values :

model.fit(x ,y\_value)

prediction = model.predict(np.expand\_dims(x\_public ,0))

predictions.append(prediction[0])

sample\_submission.loc[: ,code] = predictions \* 2

sample\_submission.isna().sum().sum()

sample\_submission.columns

# 결과:

# Index(['Day', '000060', '000080', '000100', '000120', '000150', '000240',

# '000250', '000270', '000660',

# ...

# '330860', '336260', '336370', '347860', '348150', '348210', '352820',

# '357780', '363280', '950130'],

# dtype='object', length=377)

columns = list(sample\_submission.columns[1:])

columns = ['Day'] + [str(x).zfill(6) for x in columns]

sample\_submission.columns = columns

sample\_submission.to\_csv('BASELINE\_XGBRegressor\_Week4.csv' ,index=False)

sample\_submission

# LGBM 3주차

model = LGBMRegressor()

for code in tqdm(stock\_list['종목코드'].values):

data = fdr.DataReader(code, start = start\_date, end = end\_date)[['Close']].reset\_index()

data = pd.merge(Business\_days, data, how = 'outer')

data['weekday'] = data.Date.apply(lambda x : x.weekday())

data['weeknum'] = data.Date.apply(lambda x : x.strftime('%V'))

data.Close = data.Close.ffill()

data = pd.pivot\_table(data = data, values = 'Close', columns = 'weekday', index = 'weeknum')

x = data.iloc[0:-2].to\_numpy() # 2021년 1월 04일 ~ 2021년 8월 13일까지의 데이터로

y = data.iloc[1:-1].to\_numpy() # 2021년 1월 11일 ~ 2021년 8월 20일까지의 데이터를 학습한다.

y\_0 = y[: ,0]

y\_1 = y[: ,1]

y\_2 = y[: ,2]

y\_3 = y[: ,3]

y\_4 = y[: ,4]

y\_values = [y\_0, y\_1, y\_2, y\_3, y\_4]

x\_public = data.iloc[-2].to\_numpy() # 2021년 8월 23일부터 8월 27일까지의 데이터를 예측할 것이다.

predictions = []

for y\_value in y\_values :

model.fit(x ,y\_value)

prediction = model.predict(np.expand\_dims(x\_public ,0))

predictions.append(prediction[0])

sample\_submission.loc[: ,code] = predictions \* 2

sample\_submission.isna().sum().sum()

sample\_submission.columns

# 결과:

# Index(['Day', '000060', '000080', '000100', '000120', '000150', '000240',

# '000250', '000270', '000660',

# ...

# '330860', '336260', '336370', '347860', '348150', '348210', '352820',

# '357780', '363280', '950130'],

# dtype='object', length=377)

columns = list(sample\_submission.columns[1:])

columns = ['Day'] + [str(x).zfill(6) for x in columns]

sample\_submission.columns = columns

sample\_submission.to\_csv('BASELINE\_LGBMRegressor\_Week4.csv' ,index=False)

sample\_submission