#!/usr/bin/env python3

# -\*- coding: utf-8 -\*-

from cleandata import \*

import pandas as pd

class MLEngineer(cleandata):

def \_\_init\_\_(self, raw\_data, num\_stocks, trading\_window, algorithm, method='mean', selected\_columns=None, small\_sample=False):

"""

Method to initalize data prepartion

"""

cleandata.\_\_init\_\_(self, raw\_data, method,

selected\_columns, small\_sample)

self.num\_stocks = num\_stocks

self.trading\_window = trading\_window

self.algorithm = algorithm

def return\_prediction(self):

"""

Method to predict returns

"""

def opti\_stocks(self):

"""

Method to output optimally selected stocks

"""

def profit\_anaysis(self):

"""

Method to graphaically analyze portfolio returns

"""

if \_\_name\_\_ == '\_\_main\_\_':

col\_selected = pd.read\_excel("selected\_column.xlsx")

col\_set = set(col\_selected['col\_name'])

raw\_data = pd.read\_sas('rpsdata\_rfs.sas7bdat')

df = MLEngineer(raw\_data, 40, 20, 1, small\_sample= True)

clean\_data = df.\_\_main\_\_()

# To show vif distribution

vif\_df = df.calc\_vif()

vif\_df.plot.density()

Data Cleaning Algorithm:

1. Import Merged Dataset, and group by company:

For i in each company:

1. Filter out variables with missing values, drop columns that has all NAs.
2. Change Boolean, Logical vectors into numeric expression. Since stock price, the response variable is not missing, we use unsupervised learning.
3. Impute NAs with function na.roughfix(“Impute median or mode values”),

While iters(default = 5) > 0:

I. Use Random Forest model to compute proximities.

II. Update current missing values use K-nearest weighted average mean of current imputation values. Weights are calculated from Tree proximities in I.

III. Calculate the relative distance between imputed values in previous step and the original values. If converge, break the loop.

1. Concatenate the imputed dataset with variables with no missing values in sorted order.

2. Output: Concatenated datasets from all companies.

Pseudo Code (Outline)

#Creating a Pipeline

1. Obtaining Data
   1. Getting data from Yahoo & WRDS
   2. Cleaning the data
   3. Combining the data
2. Using the dataset to predict returns using indicators
   * 1. Split the dataset into training and testing
     2. Clustering(monthly)
3. Prediction

Select 15 stocks with highest predicted return

Track this portfolio of 15 stocks for one month

Backtrack 2 yrs

1. Compute performance metrics
2. Back testing/visualization