# IE412 AI for Finance: Mini Project 3

## Factor analysis with clustering

TA: Youngbin Lee
young@unist.ac.kr

UNIST Financial Engineering Lab.

# Contents

# 1. Stock clustering

## 1 Overview: Time-series clustering

### Stock price data

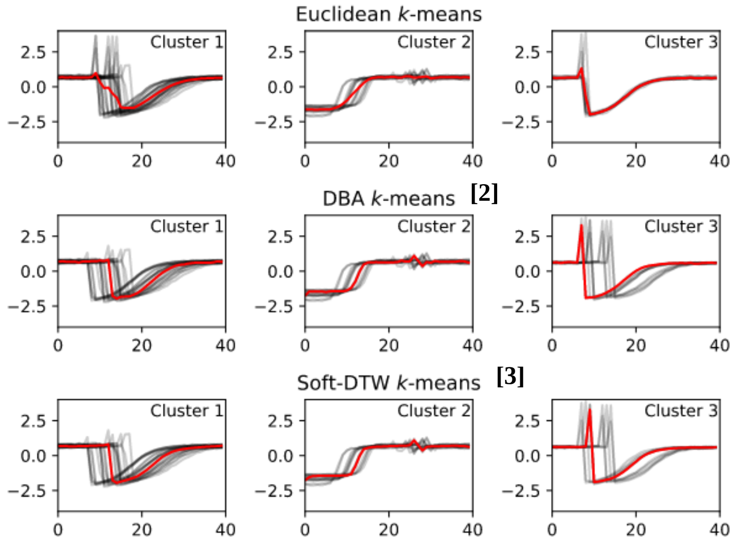
| Date | 2022-01-03 | 2022-01-04 | 2022-01-05 | 2022-01-06 | 2022-01-07 | 2022-01-10 | 2022-01-11 | 2022-01-12 | 2022-01-13 | 2022-01-14 |
|---|---|---|---|---|---|---|---|---|---|---|
| A | 155.204315 | 149.957458 | 147.388580 | 147.904327 | 143.966690 | 143.976624 | 145.444565 | 148.291153 | 143.986542 | 143.500519 |
| AAL | 18.750000 | 19.020000 | 18.680000 | 18.570000 | 19.280001 | 18.790001 | 19.020000 | 18.500000 | 19.340000 | 18.490000 |
| AAPL | 180.434280 | 178.144302 | 173.405685 | 170.510956 | 170.679489 | 170.699326 | 173.564301 | 174.010406 | 170.699326 | 171.571701 |
| ABBV | 127.937485 | 127.691849 | 128.362625 | 127.757980 | 127.427338 | 128.853882 | 129.401840 | 129.704147 | 127.451431 | 129.694611 |
| ABC | 130.200043 | 128.963043 | 130.082260 | 128.069672 | 130.690948 | 132.183182 | 133.960175 | 133.233673 | 132.428635 | 133.714722 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| XYL | 115.033981 | 116.254288 | 114.669853 | 114.512405 | 113.597168 | 112.445747 | 114.345093 | 114.335258 | 112.603203 | 109.768921 |
| YUM | 133.341415 | 134.054382 | 132.355026 | 133.722336 | 132.189011 | 130.343124 | 127.608521 | 126.133774 | 125.801712 | 124.092590 |
| ZBH | 124.095253 | 125.354172 | 124.940941 | 123.903046 | 123.153473 | 121.952202 | 123.374496 | 120.683662 | 123.076576 | 122.413490 |
| ZBRA | 583.900024 | 587.599976 | 558.179993 | 555.159973 | 530.859985 | 535.409973 | 538.570007 | 538.440002 | 525.799988 | 528.000000 |
| ZTS | 231.284103 | 222.478653 | 214.019089 | 214.869003 | 208.613281 | 210.105545 | 210.303223 | 210.313080 | 204.571243 | 203.760864 |

472 rows × 10 columns

Sample

Feature

### Time-series clustering[1]




The constantly changing stock price data is **a time series data** that has values at every time point.
Such time series data can be clustered with stocks as **samples** and prices at each time points as **features**.

The most common method is **k-means clustering**, where distance measures such as **Euclidean** distance and **DTW** are possible.

[1] https://tslearn.readthedocs.io/en/stable/auto_examples/clustering/plot_kmeans.html#sphx-glr-auto-examples-clustering-plot-kmeans-py
[2] F. Petitjean, A. Ketterlin & P. Gancarski. A global averaging method for dynamic time warping, with applications to clustering. Pattern Recognition, Elsevier, 2011, Vol. 44, Num. 3, pp. 678-693
[3] M. Cuturi, M. Blondel “Soft-DTW: a Differentiable Loss Function for Time-Series,” ICML 2017.

# 1. Stock clustering

## 2 Load data

### S&P500 daily price

| Date | A | AAL | AAPL | ABBV | ABC | ABT | ACGL | ACN | ADBE | |
|---|---|---|---|---|---|---|---|---|---|---|
| 2022-01-03 | 155.204315 | 18.750000 | 180.434280 | 127.937485 | 130.200043 | 135.452087 | 44.549999 | 398.527893 | 564.369995 | 17. |
| 2022-01-04 | 149.957458 | 19.020000 | 178.144302 | 127.691849 | 128.963043 | 132.266479 | 45.130001 | 395.679932 | 554.000000 | 17 |
| 2022-01-05 | 147.388580 | 18.680000 | 173.405685 | 128.362625 | 130.082260 | 131.672226 | 44.599998 | 388.711761 | 514.429993 | 16 |
| 2022-01-06 | 147.904327 | 18.570000 | 170.510956 | 127.757980 | 128.069672 | 131.652710 | 44.860001 | 369.940704 | 514.119995 | 16 |
| 2022-01-07 | 143.966690 | 19.280001 | 170.679489 | 127.427338 | 130.690948 | 132.061874 | 45.070000 | 362.845306 | 510.700012 | 16 |

5 rows × 472 columns

### S&P500 cumulative return

| Date | A | AAL | AAPL | ABBV | ABC | ABT | ACGL | ACN | ADBE | ADI |
|---|---|---|---|---|---|---|---|---|---|---|
| 2022-01-04 | -0.033806 | 0.014400 | -0.012691 | -0.001920 | -0.009501 | -0.023518 | 0.013019 | -0.007146 | -0.018374 | -0.009032 |
| 2022-01-05 | -0.050937 | -0.003476 | -0.039291 | 0.003333 | -0.000822 | -0.028011 | 0.001275 | -0.024757 | -0.089800 | -0.024469 |
| 2022-01-06 | -0.047438 | -0.009365 | -0.055985 | -0.001377 | -0.016294 | -0.028159 | 0.007105 | -0.073047 | -0.090403 | -0.021287 |
| 2022-01-07 | -0.074060 | 0.028869 | -0.054996 | -0.003965 | 0.004174 | -0.025051 | 0.011786 | -0.092227 | -0.097055 | -0.047528 |
| 2022-01-10 | -0.073991 | 0.003454 | -0.054880 | 0.007230 | 0.015592 | -0.027264 | 0.032864 | -0.086158 | -0.067429 | -0.038289 |

5 rows × 472 columns

* Date: Jan 1, 2022 ~ Dec 31, 2022

### S&P500 market cap

```
{'MMM': 55663685632,
 'AOS': 10413592576,
 'ABT': 192483524608,
 'ABBV': 259615277056,
 'ACN': 175054372864,
 'ATVI': 60825120768,
 'ADM': 41054584832,
 'ADBE': 157802070016,
 'ADP': 88825552896,
 'AAP': 7476755968,
 'AES': 15153751040,
 'AFL': 40314007552,
 'A': 37867589632,
 'APD': 62405439488,
 'AKAM': 13335857152,
 'ALK': 5514254848,
 'ALB': 22973214720,
 'ARE': 21156126720,
 'ALGN': 23357552640,
 'ALLE': 9561566208,
 'LNT': 13866568704,
 'ALL': 31084877824,
 'GOOGL': 1422000259072,
 'GOOG': 1421997244416,
 'MO': 82004590592,
...
```

* Date: May 18, 2023

To perform clustering, we retrieve **daily closing price** data from Yahoo Finance[1] and convert it into **cumulative returns**.

Additionally, we obtain **market capitalization information** to examine the clustering results based on it.

[1] https://github.com/ranaroussi/yfinance

# 1. Stock clustering

## 3 Perform clustering

### Model and hyperparameters

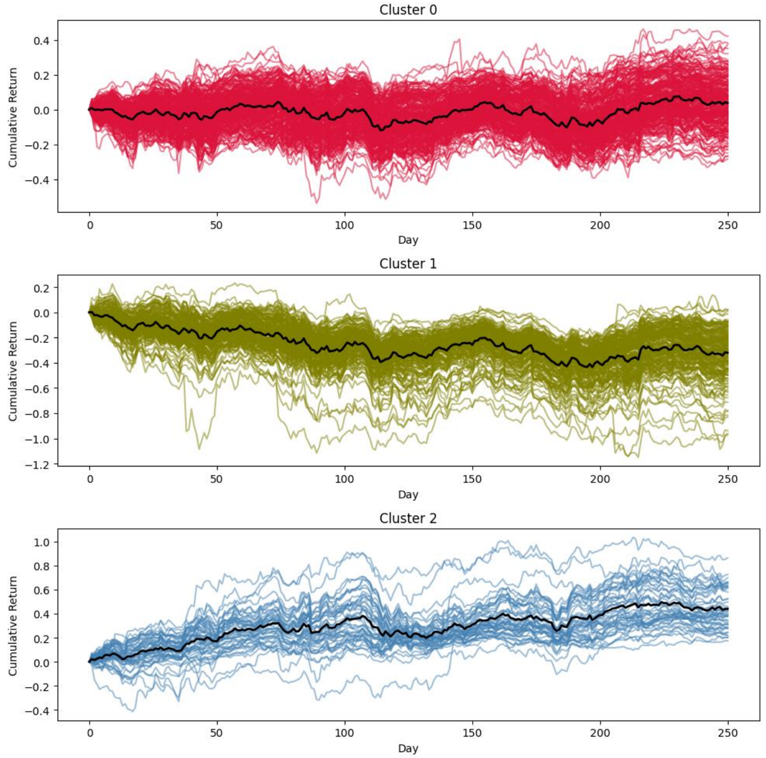
```python
"""
Perform k-means clustering
"""
ts_return = ts_return.T # transpose to (n_samples, n_features)

model = TimeSeriesKMeans(n_clusters=3, metric='euclidean', n_init=10)
model.fit(ts_return)
```
✓ 1.7s

**A clustering model** can be implemented using the machine learning package "tslearn[1]" for time series data.

The following **hyperparameters** are used:

- n_cluster=3
  - To compare it with the Fama-French 3 Factor model, stocks were grouped into 3 clusters.
- metric='Euclidean'
  - The distance between stocks was calculated using the Euclidean distance.
- n_init=10
  - Since clustering is sensitive to initialization, the final result will be the best output of n_init consecutive runs.

### Result




**A visualization of the stock time series** corresponding to each cluster and the cluster centers

[1] https://tslearn.readthedocs.io/en/stable/auto_examples/clustering/plot_kmeans.html#sphx-glr-auto-examples-clustering-plot-kmeans-py

# 1. Stock clustering

## 4 Performance evaluation

### Silhouette coefficient

- A measure of cohesion compared to separation
- Range: [-1, 1]
  - Higher scores indicate dense and well separated clusters

For data point $i$ in cluster $C_I$, the **Silhouette coefficient**

$$s_i = \frac{b_i - a_i}{\max(a_i, b_i)}$$

where

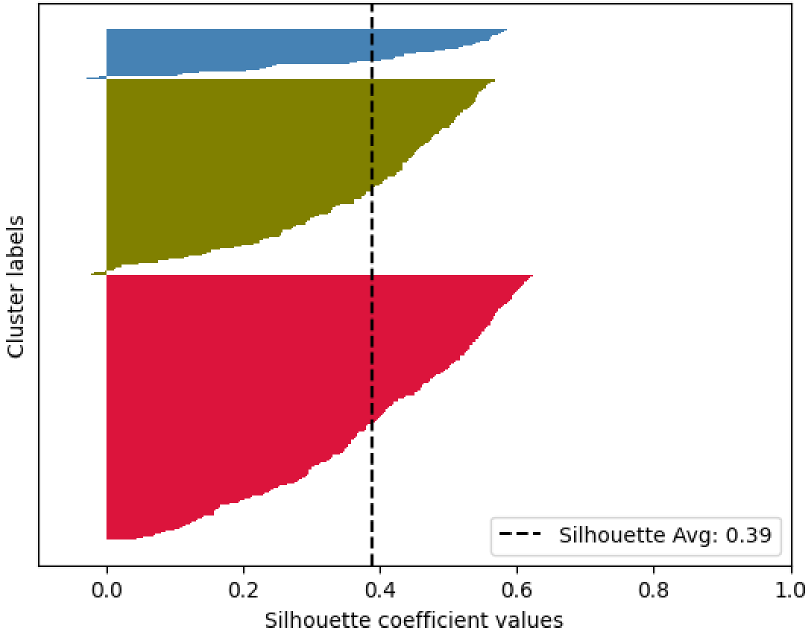
$a_i$ is the mean distance between a sample $i$ and all other points in the same cluster (cohesion)

$$a_i = \frac{1}{|C_I| - 1} \sum_{j \in C_I, i \neq j} d(i, j)$$

$b_i$ is the mean distance between a sample $i$ and all other points in the next nearest cluster (separation)

$$b_i = \min_{J \neq I} \frac{1}{|C_J|} \sum_{j \in C_J} d(i, j)$$

### Result



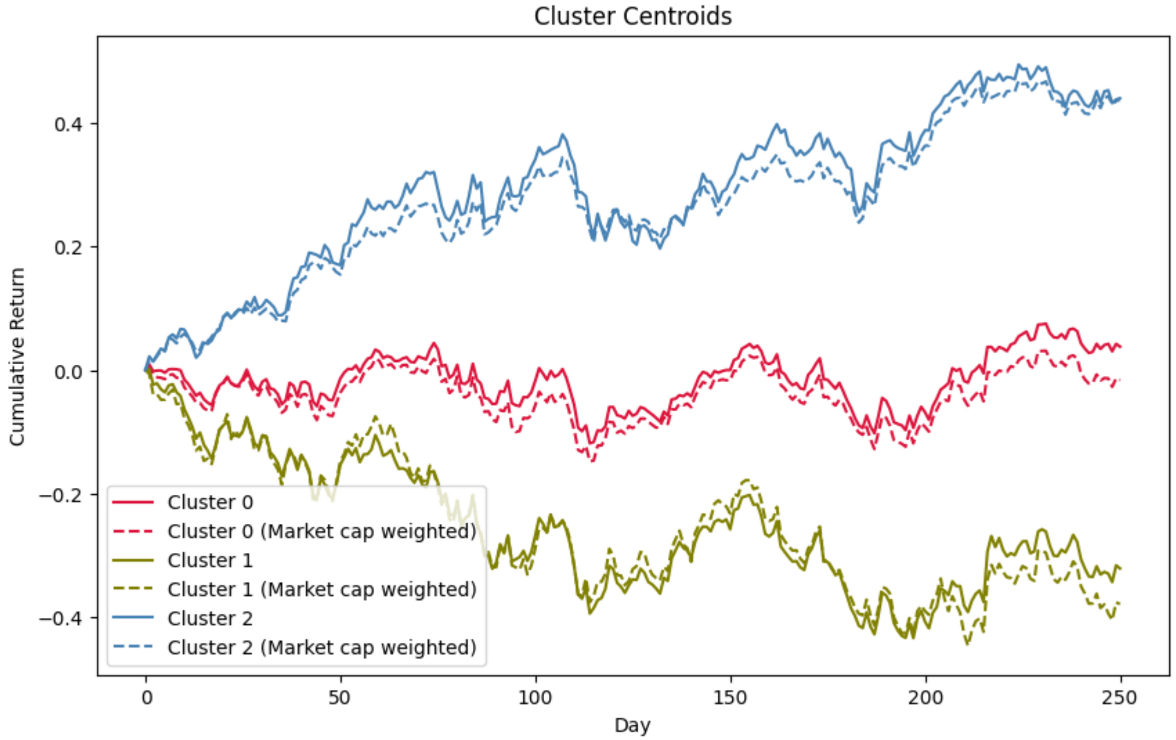

We calculated the **Silhouette scores using the Euclidean distance** measure for our clustering results and visualized them.

When calculating Silhouette scores, various pairwise distances such as cosine, L1, and L2 can be used as the distance measure[1].

[1] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise_distances.html

# 1. Stock clustering

## 5 Cluster centers as factors




The cluster centers were visualized based on the **three clusters** obtained from clustering.

In the figure, the solid line represents the **simple average** of the cumulative return values within the cluster, while the dotted line represents the **weighted average based on market capitalization**.

Clustering can be used to find factors, but it is also possible to use other machine learning techniques such as **PCA or AutoEncoder** to identify factors.

# 2. Fama-French 3 Factors

## 1 Overview: Fama-French 3 Factors

$$r_i - r_f = \alpha + \beta_i(r_M - r_f) + \beta_{si} \cdot \boldsymbol{f}_{\boldsymbol{SMB}} + \beta_{vi} \cdot \boldsymbol{f}_{\boldsymbol{HML}} + \varepsilon_i$$

1. **Market excess return**
2. **Outperformance of small versus big companies**
3. **Outperformance of high value versus low value companies**

# 2. Fama-French 3 Factors

## 2 Load data

### F-F daily return

| | Mkt-RF | SMB | HML | RF |
|---|---|---|---|---|
| 20220103 | 0.0073 | 0.0033 | 0.0078 | 0.00000 |
| 20220104 | -0.0029 | -0.0082 | 0.0362 | 0.00000 |
| 20220105 | -0.0228 | -0.0146 | 0.0260 | 0.00000 |
| 20220106 | 0.0000 | 0.0021 | 0.0175 | 0.00000 |
| 20220107 | -0.0048 | -0.0132 | 0.0202 | 0.00000 |
| ... | ... | ... | ... | ... |
| 20221223 | 0.0051 | -0.0060 | 0.0115 | 0.00016 |
| 20221227 | -0.0051 | -0.0073 | 0.0143 | 0.00016 |
| 20221228 | -0.0123 | -0.0024 | -0.0029 | 0.00016 |
| 20221229 | 0.0187 | 0.0126 | -0.0107 | 0.00016 |
| 20221230 | -0.0022 | 0.0010 | -0.0003 | 0.00016 |

251 rows × 4 columns

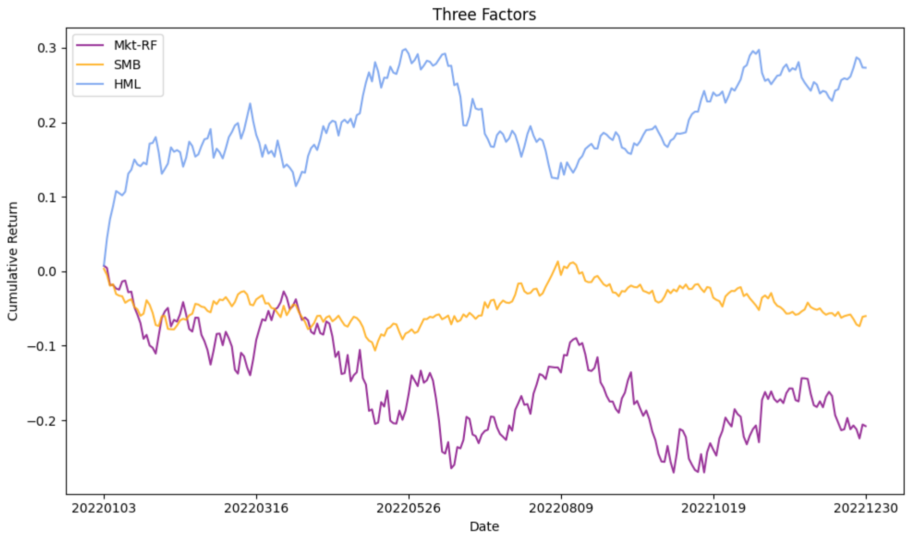
### F-F cumulative return

| | Mkt-RF | SMB | HML | RF |
|---|---|---|---|---|
| 20220103 | 0.0073 | 0.0033 | 0.0078 | 0.00000 |
| 20220104 | 0.0044 | -0.0049 | 0.0440 | 0.00000 |
| 20220105 | -0.0184 | -0.0195 | 0.0700 | 0.00000 |
| 20220106 | -0.0184 | -0.0174 | 0.0875 | 0.00000 |
| 20220107 | -0.0232 | -0.0306 | 0.1077 | 0.00000 |
| ... | ... | ... | ... | ... |
| 20221223 | -0.2071 | -0.0641 | 0.2729 | 0.01354 |
| 20221227 | -0.2122 | -0.0714 | 0.2872 | 0.01370 |
| 20221228 | -0.2245 | -0.0738 | 0.2843 | 0.01386 |
| 20221229 | -0.2058 | -0.0612 | 0.2736 | 0.01402 |
| 20221230 | -0.2080 | -0.0602 | 0.2733 | 0.01418 |

251 rows × 4 columns

* Date: Jan 1, 2022 ~ Dec 31, 2022

### Visualization



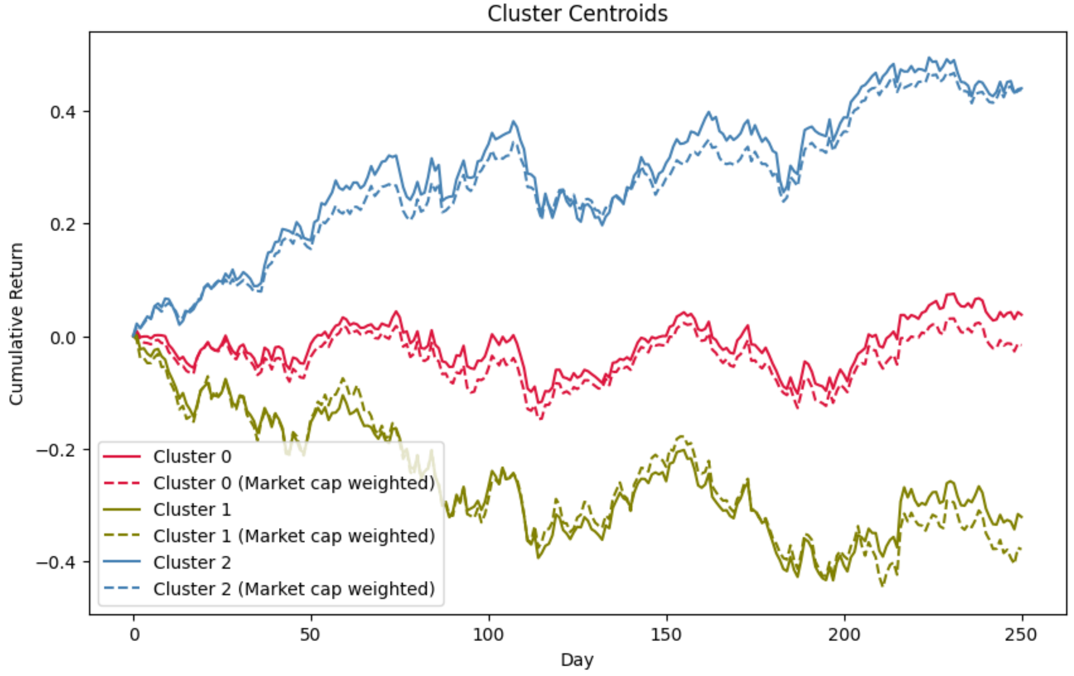

To compare with the factors obtained from clustering, we obtain daily data for 3 factors[1] and convert them into cumulative returns.

The return paths of the three factors are visualized.

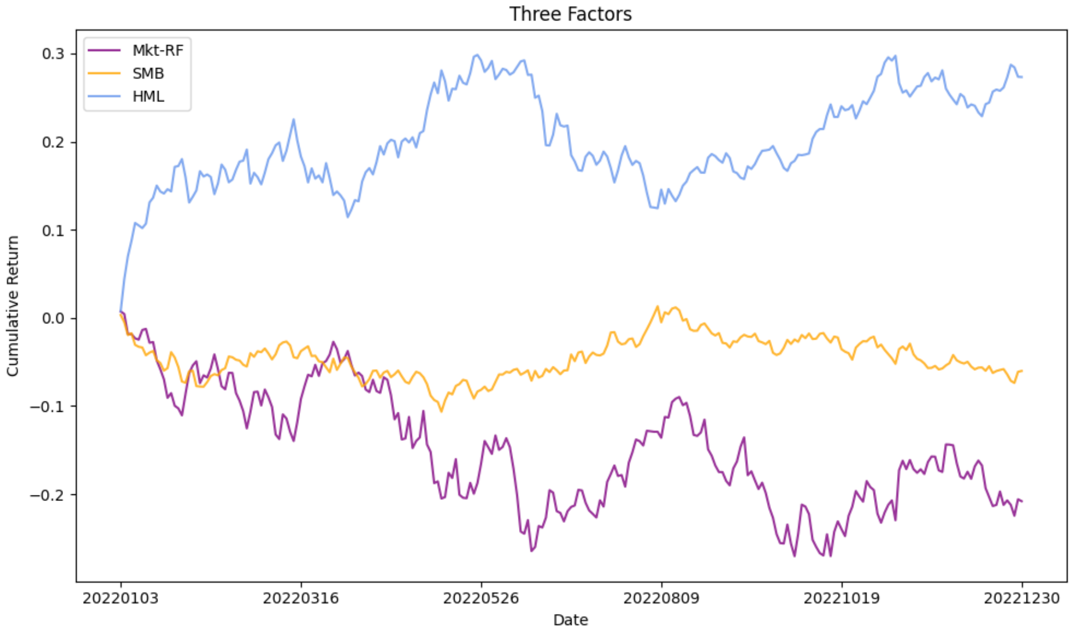
[1] https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html

# 2. Fama-French 3 Factors

## 3 Comparison with cluster factors




**VS**



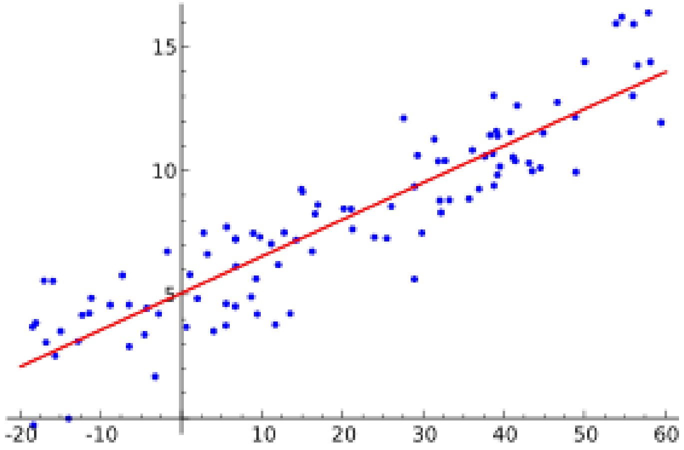

Comparing with the three cluster centers obtained earlier:

**Cluster 0** resembles the **SMB factor**, which oscillates around zero during the given period.
**Cluster 1** exhibits a movement similar to the **Mkt-RF** factor.
**Cluster 2** resembles the **HML factor**, which shows a consistent upward trend during the given period.

# 3. Regression analysis

## 1 Overview: Linear regression

$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^\mathsf{T} \boldsymbol\beta + \varepsilon_i, \qquad i = 1, \ldots, n,$$




Example of simple linear regression, which has one independent variable[1].

[1] https://en.wikipedia.org/wiki/Linear_regression

# 3. Regression analysis

## 2 Load data

### AAPL daily price

| | Close |
|---|---|
| 2022-01-03 | 180.434296 |
| 2022-01-04 | 178.144302 |
| 2022-01-05 | 173.405670 |
| 2022-01-06 | 170.510971 |
| 2022-01-07 | 170.679504 |
| ... | ... |
| 2022-12-23 | 131.477127 |
| 2022-12-27 | 129.652435 |
| 2022-12-28 | 125.674019 |
| 2022-12-29 | 129.233658 |
| 2022-12-30 | 129.552719 |

251 rows × 1 columns

### AAPL log return

| | Close |
|---|---|
| 2022-01-03 | 0.000000 |
| 2022-01-04 | -0.012773 |
| 2022-01-05 | -0.026960 |
| 2022-01-06 | -0.016834 |
| 2022-01-07 | 0.000988 |
| ... | ... |
| 2022-12-23 | -0.002802 |
| 2022-12-27 | -0.013976 |
| 2022-12-28 | -0.031166 |
| 2022-12-29 | 0.027931 |
| 2022-12-30 | 0.002466 |

251 rows × 1 columns

### Cluster factors

| | Cluster 0 | Cluster 1 | Cluster 2 |
|---|---|---|---|
| Date | | | |
| 2022-01-03 | 0.000000 | 0.000000 | 0.000000 |
| 2022-01-04 | 0.009006 | 0.001512 | 0.022123 |
| 2022-01-05 | -0.010998 | -0.024916 | -0.008799 |
| 2022-01-06 | 0.001227 | 0.001656 | 0.011276 |
| 2022-01-07 | 0.000047 | -0.009480 | 0.010117 |

### F-F 3 factors

| | Mkt-RF | SMB | HML | RF |
|---|---|---|---|---|
| 20220103 | 0.0073 | 0.0033 | 0.0078 | 0.0 |
| 20220104 | -0.0029 | -0.0082 | 0.0362 | 0.0 |
| 20220105 | -0.0228 | -0.0146 | 0.0260 | 0.0 |
| 20220106 | 0.0000 | 0.0021 | 0.0175 | 0.0 |
| 20220107 | -0.0048 | -0.0132 | 0.0202 | 0.0 |

To fit the daily price movements of individual stocks to the factors, we retrieve **Apple stock data** and convert it into log returns.

# 3. Regression analysis

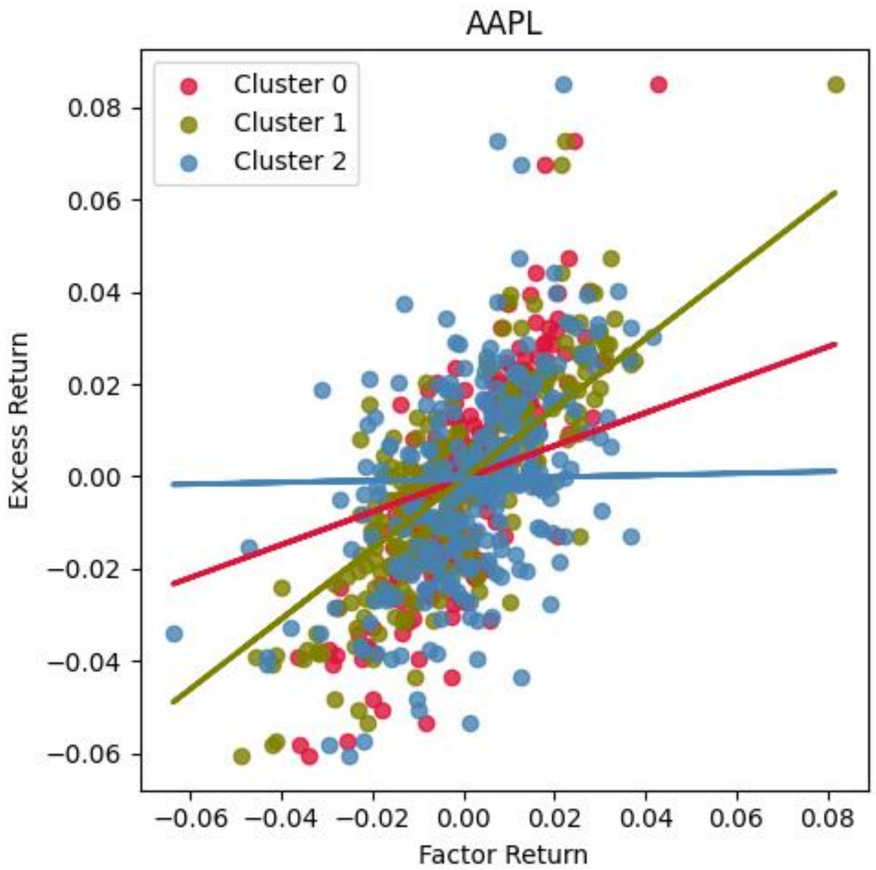
## 3 Perform linear regression

### Fitting the cluster factors

```python
AAPL_regr = linear_model.LinearRegression().fit(
    X = df.iloc[:,:3].values,                              # shape (n_samples, 3)
    y = ts_return.values - df_FF3['RF'].values.reshape(-1,1)    # shape (n_samples, 1)
)

# print stock ticker
print("Stock: {}".format("AAPL"))

# print coefficients
print("Alpha: {:.4f}".format(AAPL_regr.intercept_[0]))
for i, coef in enumerate(AAPL_regr.coef_[0]):
    print("Beta for factor {} ({}): {:.4f}".format(i+1, df.columns[i], coef))

# print R^2
print("R^2: {:.4f}".format(AAPL_regr.score(df.iloc[:,:3].values, ts_return.values - df_FF3['RF'].values
```

✓ 0.0s

```
Stock: AAPL
Alpha: -0.0005
Beta for factor 1 (Cluster 0): 0.3567
Beta for factor 2 (Cluster 1): 0.7583
Beta for factor 3 (Cluster 2): 0.0195
R^2: 0.6844
```




**The Apple stock**, with the largest market capitalization, has a strong correlation with the **Cluster 1** which is similar to the market factor.

# 3. Regression analysis

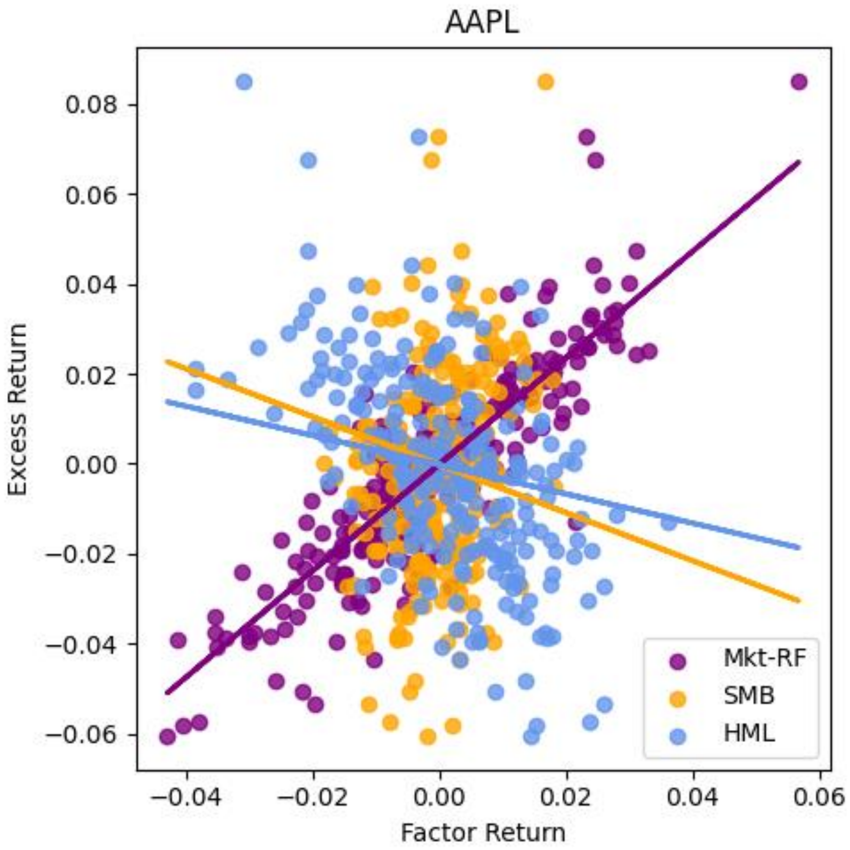
## 4 Perform linear regression

### Fitting the F-F 3 factors

```
AAPL_regr = linear_model.LinearRegression().fit(
    X = df_FF3.iloc[:,:3].values,                              # shape (n_samples, 3)
    y = ts_return.values - df_FF3['RF'].values.reshape(-1,1)   # shape (n_samples, 1)
)

# print stock ticker
print("Stock: {}".format("AAPL"))

# print coefficients
print("Alpha: {:.4f}".format(AAPL_regr.intercept_[0]))
for i, coef in enumerate(AAPL_regr.coef_[0]):
    print("Beta for factor {} ({}): {:.4f}".format(i+1, df_FF3.columns[i], coef))

# print R^2
print("R^2: {:.4f}".format(AAPL_regr.score(df_FF3.iloc[:,:3].values, ts_return.values - df_FF3['RF'].val
```

✓ 0.0s Pytho

```
Stock: AAPL
Alpha: -0.0002
Beta for factor 1 (Mkt-RF): 1.1820
Beta for factor 2 (SMB): -0.5327
Beta for factor 3 (HML): -0.3247
R^2: 0.7978
```




**The Apple stock**, with the largest market capitalization, has a significant negative correlation with the **size factor**.

# Questions & Answers

- E-mail: young@unist.ac.kr

# Thank You.