PSS-BEM_114_PS3

May 3, 2024

[ ]:
```
import pandas as pd
import statsmodels.api as sm
from sklearn.linear_model import LinearRegression
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.dates import AutoDateLocator, AutoDateFormatter
```

[ ]:
```
crsp_data = pd.read_csv("/content/crsp_1926_2020.csv")
crsp_data['PRC'] = crsp_data['PRC'].where(crsp_data['PRC'] >= 0)
crsp_data['ME'] = (crsp_data['PRC'] * crsp_data['SHROUT'])/1000

ff5 = pd.read_csv('/content/ff5_factors.csv')
mom_data = pd.read_csv('/content/mom_factor.csv')
industry_data = pd.read_csv('/content/12IndustryPortfolios.csv')

data_datemod = crsp_data.copy()
data_datemod['date'] = np.floor(data_datemod['date'].str.replace('-','').
 ↪astype(float)/100).astype(int)

crsp_data = pd.merge(data_datemod, ff5, how = 'inner', on='date')
crsp_data = pd.merge(crsp_data, mom_data, how = 'inner', on='date')
crsp_data = pd.merge(crsp_data, industry_data, how = 'inner', on='date')
crsp_data.columns = crsp_data.columns.str.strip()

bcw_data = pd.read_excel("/content/bcwlist.xls")
```

[ ]:
```
crsp_data['date'] = [str(n)[:4] + "-" + str(n)[4:] for n in crsp_data['date']]
```

[ ]:
```
crsp_data['date'] = pd.to_datetime(crsp_data['date'])
crsp_data['month'] =crsp_data['date'].dt.month
crsp_data['year'] = crsp_data['date'].dt.year
```

1 1A

The code below is used to create the signals for taking positions with the Edmans (2011) strategy as well as create weightings for each of the assets with a signal. The code will be further described cell-by-cell below.

1.1 Signal

The code below creates a new column for signalling assets in the crsp dataset based on if they are ranked in that year. It indexes each of the rating years and if a company is listed that year on the bcw list and also in the crsp data it will write that column value as a 1, indicating that it is listed at that time and should be included in the portfolio.

```
[ ]: # Create column in pricing dataframe 'ranked' that for each row indicates 1 if the company was ranked
# at that timestep in the bcw list, and 0 if not

signal_years = sorted(bcw_data['year'].unique())
test_years = sorted(crsp_data['year'].unique())

crsp_data['ranked'] = 0

for year in signal_years:
    ranked_ids = bcw_data[bcw_data['year'] == year]['permno']
    ranked_ids = ranked_ids.dropna()

    if year == 1984:
      crsp_data.loc[(crsp_data['year'] == year) & (crsp_data['month'] >= 3) & (crsp_data['PERMNO'].isin(ranked_ids)), 'ranked'] = 1
      crsp_data.loc[(crsp_data['year'] == year) & (crsp_data['month'] >= 3) & ~(crsp_data['PERMNO'].isin(ranked_ids)), 'ranked'] = 0

      crsp_data.loc[(crsp_data['year'] > year) & (crsp_data['PERMNO'].isin(ranked_ids)), 'ranked'] = 1
      crsp_data.loc[(crsp_data['year'] > year) & ~(crsp_data['PERMNO'].isin(ranked_ids)), 'ranked'] = 0
    elif year == 1993:
      crsp_data.loc[(crsp_data['year'] == year) & (crsp_data['month'] >= 2) & (crsp_data['PERMNO'].isin(ranked_ids)), 'ranked'] = 1
      crsp_data.loc[(crsp_data['year'] == year) & (crsp_data['month'] >= 2) & ~(crsp_data['PERMNO'].isin(ranked_ids)), 'ranked'] = 0

      crsp_data.loc[(crsp_data['year'] > year) & (crsp_data['PERMNO'].isin(ranked_ids)), 'ranked'] = 1
      crsp_data.loc[(crsp_data['year'] > year) & ~(crsp_data['PERMNO'].isin(ranked_ids)), 'ranked'] = 0
    else:
      crsp_data.loc[(crsp_data['year'] >= year) & (crsp_data['PERMNO'].isin(ranked_ids)), 'ranked'] = 1
      crsp_data.loc[(crsp_data['year'] >= year) & ~(crsp_data['PERMNO'].isin(ranked_ids)), 'ranked'] = 0
```

[ ]:
```
def rank_shift(group):
  group = group.sort_values(by='date')
  group['ranked'] = group['ranked'].shift(1)
  return group
```

1.2 Building portfolio weights

The code below builds the portfolio weights at each timestep for the value weighted and equal weighted portfolios. The reweighting is done at each time step for both the value and equal weighted portfolios. The reason for this is although ratings are only re-evaluated every year, the value weighted portfolios will automatically rebalance themselves each timestep by the nature the weights should be updated at each timestep to reflect their current relative weights. The equal weighted portfolios are also recalculated every timestep, ideally we would want to keep a tracker of portfolio values after equal weighting to adjust the weights at each timestep respective to the returns after equal weighting but for simplicity we will expect that these weights do not change to a statistically signficant scale to affect the returns calculations and factor models later on.

[ ]:
```
def eval_weights(group):
  group['val_weighting'] = group['ME'] / group['ME'].sum()
  group['eq_weighting'] = 1 / len(list(group['ME']))
  return group

def calc_weights(data):
  data['val_weighting'] = 0
  data = data.groupby('date').apply(eval_weights)
  data = data.reset_index(drop = True)
  return(data)

def calc_weights_2(data):
  data['val_weighting'] = 0
  data = data.groupby(['date', 'ranked']).apply(eval_weights)
  data = data.reset_index(drop = True)
  return(data)
```

[ ]:
```
def weight_shift(group):
  group = group.sort_values(by='date')
  group['val_weighting'] = group['val_weighting'].shift(1)
  group['eq_weighting'] = group['eq_weighting'].shift(1)
  return group
```

1.3 Data processing and filtering

The code below calculates the weights and shifts indices of ranking signal and weights to adjust for position taking. We also filters all crsp data that was not ranked in the signalling cell above so accessing and analyzing data visually becomes easier and quicker to run.

[ ]:
```
crsp_data = calc_weights_2(crsp_data)
crsp_data = crsp_data.groupby('PERMNO').apply(rank_shift)
```

```
crsp_data = crsp_data.reset_index(drop = True)
crsp_data = crsp_data.groupby('PERMNO').apply(weight_shift)
crsp_data = crsp_data.reset_index(drop = True)
crsp_data = crsp_data[crsp_data['ranked'] == 1]
```

2 Testing / Sanity Check code

The testing functions below test whether the rankings signals we developed are correct. The function test_rankings shows 74 companies are ranked in 1984 and 65 in 1993 so it seems to work. The test_weights function also sums all weighted assets to 1 over all timesteps so the value weightings seem to be properly set as well.

```
[ ]: def test_weights(data):
       years = sorted(set(data['year']))
       months = sorted(set(data['month']))
       prev_weighting = []
       for year in years:
         if year >= 1984:
           print('------------------------')
           print(year)
           for month in months:
             condition = (data['year'] == year) & (data['month'] == month) &␣
      ↪(data['val_weighting'] != 0)
             # print(data.loc[condition])

             curr_weight = list(data.loc[condition]['val_weighting'])
             # if curr_weight != prev_weighting:
             #   print('weight change')
             # print(len(list(data.loc[condition]['val_weighting'])))
             print(data.loc[condition]['val_weighting'].sum())
             prev_weighting = curr_weight

     def test_rankings(data):
       dates = sorted(set(data['date']))
       curr_permno = []
       for date in dates:
         df = data.loc[(data['date'] == date) & (data['ranked'] == 1)]
         permnos = list(df['PERMNO'])
         print(date)
         print(len(permnos))
         curr_permno = permnos
```

3 Returns and Model Estimation Code

Below is the code we use to calculate returns as well as estimate the loadings on the CAPM, FF3, FF5, Carhart, and 12 Industry Portfolio implied returns.

[ ]:

```python
# Code to calculate return list based on each companies

def get_returns(data, ret_type):
  date_set = sorted(set(data['date']))

  returns = []

  for date in date_set:
    # print(np.sum(data.loc[data['date'] == date]['val_weighting'])) - used to check portfolio weights at each timestep sum to 1 (they do)
    ret = 1 + np.sum(data.loc[data['date'] == date][ret_type])
    returns.append(ret)

  return returns

def estim_CAPM(portfolio_ret, data_unique_dates):
  model1 = sm.OLS(portfolio_ret, sm.add_constant(data_unique_dates['Mkt-RF'])).fit()
  alpha, beta = model1.params
  print("CAPM ESTIMATES")
  print(model1.summary())
  return (beta, alpha)

def estim_FF3(portfolio_ret, data_unique_dates):
  model1=sm.OLS(portfolio_ret, sm.add_constant(data_unique_dates[['Mkt-RF', 'SMB', 'HML']])).fit()
  alpha, beta_1, beta_2, beta_3 = model1.params
  # print(model1.summary())
  return (beta_1, beta_2, beta_3, alpha)

def estim_FF5(portfolio_ret, data_unique_dates):
  model1=sm.OLS(portfolio_ret, sm.add_constant(data_unique_dates[['Mkt-RF', 'SMB', 'HML', 'RMW', 'CMA']])).fit()
  alpha, beta_1, beta_2, beta_3, beta_4, beta_5 = model1.params
  print("FF5 ESTIMATES")
  print(model1.summary())
  return (beta_1, beta_2, beta_3, beta_4, beta_5, alpha)

def estim_FF5mom(portfolio_ret, data_unique_dates):
  model1=sm.OLS(portfolio_ret, sm.add_constant(data_unique_dates[['Mkt-RF', 'SMB', 'HML', 'RMW', 'CMA', 'Mom']])).fit()
  alpha, beta_1, beta_2, beta_3, beta_4, beta_5, beta_6 = model1.params
  print("FF5 + MOMENTUM ESTIMATES")
  print(model1.summary())
  return (beta_1, beta_2, beta_3, beta_4, beta_5, beta_6, alpha)

def estim_industry(portfolio_ret, data_unique_dates):
```

```python
    model1 = sm.OLS(portfolio_ret, sm.add_constant(data_unique_dates[['NoDur', 'Durbl', 'Manuf', 'Enrgy', 'Chems', 'BusEq', 'Telcm', 'Utils', 'Shops', 'Hlth', 'Money', 'Other']])).fit()
    alpha, beta_1, beta_2, beta_3, beta_4, beta_5, beta_6, beta_7, beta_8, beta_9, beta_10, beta_11, beta_12 = model1.params
    print("INDUSTRY LOADINGS ESTIMATES")
    print(model1.summary())
    return (beta_1, beta_2, beta_3, beta_4, beta_5, beta_6, beta_7, beta_8, beta_9, beta_10, beta_11, beta_12, alpha)

def estim_models(portfolio_ret, input_ret, data_unique_months, industry):
    rf = list(data_unique_months['RF'])
    ret = [input_ret[i] - (rf[i] / 100) for i in range(len(input_ret))]
    mean_ret = np.mean(ret) - 1
    volatility = np.std(ret)
    sharpe_ratio = mean_ret / volatility
    print("Mean returns = {:.3f}".format(mean_ret))
    print("Portfolio volatility = {:3f}".format(volatility))
    print("Strategy Sharpe Ratio = {:3f}".format(sharpe_ratio))

    portfolio_ret = [(item - 1) * 100 for item in portfolio_ret]

    beta, alpha = estim_CAPM(portfolio_ret, data_unique_months)
    CAPM_implied = list(data_unique_months['RF'] +  beta * data_unique_months['Mkt-RF'])
    CAPM_implied_percent = [1 + CAPM_implied[i] / 100 for i in range(len(CAPM_implied))]
    CAPM_implied_cumulative = [1 * np.prod(CAPM_implied_percent[0:i+1]) for i in range(len(CAPM_implied_percent))]

    beta_1, beta_2, beta_3, alpha = estim_FF3(portfolio_ret, data_unique_months)
    FF3_implied = list(data_unique_months['RF'] +  beta_1 * data_unique_months['Mkt-RF'] + beta_2 * data_unique_months['SMB'] + beta_3 * data_unique_months['HML'])
    FF3_implied_percent = [1 + FF3_implied[i] / 100 for i in range(len(FF3_implied))]
    FF3_implied_cumulative = [1 * np.prod(FF3_implied_percent[0:i+1]) for i in range(len(FF3_implied_percent))]

    beta_1, beta_2, beta_3, beta_4, beta_5, alpha = estim_FF5(portfolio_ret, data_unique_months)
    FF5_implied = list(data_unique_months['RF'] +  beta_1 * data_unique_months['Mkt-RF'] + beta_2 * data_unique_months['SMB'] + beta_3 * data_unique_months['HML'] + beta_4 * data_unique_months['RMW'] + beta_5 * data_unique_months['CMA'])
```

```python
    FF5_implied_percent = [1 + FF5_implied[i] / 100 for i in range(len(FF5_implied))]
    FF5_implied_cumulative = [1 * np.prod(FF5_implied_percent[0:i+1]) for i in range(len(FF5_implied_percent))]

    beta_1, beta_2, beta_3, beta_4, beta_5, beta_6, alpha = estim_FF5mom(portfolio_ret, data_unique_months)
    FF5mom_implied = list(data_unique_months['RF'] +  beta_1 * data_unique_months['Mkt-RF'] + beta_2 * data_unique_months['SMB'] + beta_3 * data_unique_months['HML'] + beta_4 * data_unique_months['RMW'] + beta_5 * data_unique_months['CMA'] + beta_6 * data_unique_months['Mom'])
    FF5mom_implied_percent = [1 + FF5mom_implied[i] / 100 for i in range(len(FF5mom_implied))]
    FF5mom_implied_cumulative = [1 * np.prod(FF5mom_implied_percent[0:i+1]) for i in range(len(FF5mom_implied_percent))]

    if industry is True:

        beta_1, beta_2, beta_3, beta_4, beta_5, beta_6, beta_7, beta_8, beta_9, beta_10, beta_11, beta_12, alpha = estim_industry(portfolio_ret, data_unique_months)
        Industry_implied = list(data_unique_months['RF'] +  beta_1 * data_unique_months['NoDur'] + beta_2 * data_unique_months['Durbl'] + beta_3 * data_unique_months['Manuf'] + beta_4 * data_unique_months['Enrgy'] + beta_5 * data_unique_months['Chems'] + beta_6 * data_unique_months['BusEq'] + beta_7 * data_unique_months['Telcm'] + beta_8 * data_unique_months['Utils'] + beta_9 * data_unique_months['Shops'] + beta_10 * data_unique_months['Hlth'] + beta_11 * data_unique_months['Money'] + beta_12 * data_unique_months['Other'])
        Industry_implied_percent = [1 + Industry_implied[i] / 100 for i in range(len(Industry_implied))]
        Industry_implied_cumulative = [1 * np.prod(Industry_implied_percent[0:i+1]) for i in range(len(Industry_implied_percent))]

        return CAPM_implied_cumulative, FF3_implied_cumulative, FF5_implied_cumulative, FF5mom_implied_cumulative, Industry_implied_cumulative


    return CAPM_implied_cumulative, FF3_implied_cumulative, FF5_implied_cumulative, 0, 0
```

4 Portfolio Simulation and Plotting Code

Below is the code we use to simulate this portfolio given our analyzed and signal-included dataset as well as estimate the model-implied returns for the various models described in the text cell above.

We create lists of returns and cumulative portfolio returns in the code below as well as a plotting function to plot the raw returns against model implied returns over the time frame.

```
[ ]: # Calculated value weighted and equal weighted returns for each asset at each␣
     ↪time step then create list of monthly returns
     # Use list of monthly returns (from get_returns function) to create cumulative␣
     ↪portfolio values
     # Feed returns into estimation models to develop factor implied returns and␣
     ↪portfolio values

     def sim_portfolio(input_data, equal, date_split = None, type_split = None,␣
     ↪industry = False):
       if date_split is not None:
         if type_split == 'Pre':
           data = input_data[(input_data['date'] < pd.to_datetime(date_split)) &␣
     ↪(input_data['date'] > pd.to_datetime('1984-01-01'))]
         else:
           data = input_data[(input_data['date'] >= pd.to_datetime(date_split))]
       else:
         data = input_data[(input_data['date'] > pd.to_datetime('1984-01-01'))]
       data['RET'] = pd.to_numeric(data['RET'], errors='coerce')
       data['weighted_val_ret'] = data['val_weighting'] * data['RET']
       data['weighted_eq_ret'] = data['eq_weighting'] * data['RET']

       if equal:
         returns = get_returns(data, 'weighted_eq_ret')
       else:
         returns = get_returns(data, 'weighted_val_ret')

       portfolio_ret_mom = returns
       portfolio_val_mom = [1 * np.prod(portfolio_ret_mom[0:i+1]) for i in␣
     ↪range(len(portfolio_ret_mom))]

       date_df = data.reset_index(drop = True).drop_duplicates(subset='date',␣
     ↪keep='first')

       date_df = date_df.sort_values(by = 'date')

       # ff5mom, industry

       capm, ff3, ff5, ff5mom, industry = estim_models(portfolio_ret_mom,␣
     ↪portfolio_ret_mom, date_df, industry)

       return data.reset_index(drop = True), portfolio_ret_mom, portfolio_val_mom,␣
     ↪capm, ff3, ff5, ff5mom, industry
```

```
[ ]: # Plot portfolio cumulative returns and model implied returns for each model

     def disp_portfolio(data, rets, vals, capm, ff3, ff5, ff5mom =None, industry =␣
      ↪None):
       mean_ret = np.mean([i - 1 for i in rets])
       var_ret = np.std(rets)
       x_axis = sorted(set(data['date']))
       print(mean_ret)
       print(var_ret)
       print('Sharpe Ratio: {}'.format(mean_ret / var_ret))
       plt.figure(figsize = (20, 8))
       plt.plot(x_axis, vals, label = 'Returns')
       plt.plot(x_axis, ff3, label = 'FF3 Implied Returns')
       plt.plot(x_axis, capm, label = 'CAPM Implied Returns')
       if ff5 is not None:
         plt.plot(x_axis, ff5, label = 'FF5 Implied Returns')
       if ff5mom is not None:
         plt.plot(x_axis, ff5mom, label = 'FF5+Momentum Implied Returns')
       if industry is not None:
         plt.plot(x_axis, industry, label = 'Industry Implied Returns')
       plt.gca().xaxis.set_major_locator(AutoDateLocator())
       plt.legend(loc = 'best')
       plt.show()
```

5 2A & 2B

The output from the cell below displays the mean returns, volatility, and Sharpe ratio as well as the factor models OLS estimates for value and equal weighted portfolios respectively. From the factor model estimates calculated in the cells below, we can clearly see that the strategy has a significant positive alpha.

```
[ ]: print("VALUE WEIGHTED PORTFOLIO RESULTS:")
     data_val_w, rets_w, vals_w, capm_w, ff3_w, ff5_w, ff5mom_w, industry_w =␣
      ↪sim_portfolio(crsp_data, False)
     print("#######################################################################################
     print("EQUAL WEIGHTED PORTFOLIO RESULTS:")
     data_val_e, rets_e, vals_e, capm_e, ff3_e, ff5_e, ff5mom_e, industry_e =␣
      ↪sim_portfolio(crsp_data, True)
```

```
VALUE WEIGHTED PORTFOLIO RESULTS:
Mean returns = 0.009
Portfolio volatility = 0.051827
Strategy Sharpe Ratio = 0.167496
CAPM ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.825
```

```
Model:                            OLS   Adj. R-squared:                  0.825
Method:                 Least Squares   F-statistic:                     2073.
Date:                Sat, 27 Apr 2024   Prob (F-statistic):          2.17e-168
Time:                        04:51:52   Log-Likelihood:                -966.47
No. Observations:                 441   AIC:                             1937.
Df Residuals:                     439   BIC:                             1945.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.3590      0.105      3.427      0.001       0.153       0.565
Mkt-RF         1.0494      0.023     45.531      0.000       1.004       1.095
==============================================================================
Omnibus:                       22.348   Durbin-Watson:                   2.105
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               64.023
Skew:                           0.061   Prob(JB):                     1.25e-14
Kurtosis:                       4.863   Cond. No.                         4.61
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
FF5 ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.861
Model:                            OLS   Adj. R-squared:                  0.859
Method:                 Least Squares   F-statistic:                     538.0
Date:                Sat, 27 Apr 2024   Prob (F-statistic):          1.08e-183
Time:                        04:51:52   Log-Likelihood:                -916.31
No. Observations:                 441   AIC:                             1845.
Df Residuals:                     435   BIC:                             1869.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.4999      0.098      5.095      0.000       0.307       0.693
Mkt-RF         1.0092      0.024     42.795      0.000       0.963       1.056
SMB           -0.2167      0.035     -6.134      0.000      -0.286      -0.147
HML           -0.1008      0.043     -2.348      0.019      -0.185      -0.016
RMW           -0.0907      0.046     -1.967      0.050      -0.181   -5.02e-05
CMA           -0.2927      0.066     -4.434      0.000      -0.422      -0.163
==============================================================================
Omnibus:                       11.971   Durbin-Watson:                   2.130
Prob(Omnibus):                  0.003   Jarque-Bera (JB):               21.231
Skew:                           0.120   Prob(JB):                     2.45e-05
```

```
Kurtosis:                       4.048   Cond. No.                         5.31
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
FF5 + MOMENTUM ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.862
Model:                            OLS   Adj. R-squared:                  0.860
Method:                 Least Squares   F-statistic:                     451.0
Date:                Sat, 27 Apr 2024   Prob (F-statistic):          5.72e-183
Time:                        04:51:52   Log-Likelihood:                -914.76
No. Observations:                 441   AIC:                             1844.
Df Residuals:                     434   BIC:                             1872.
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.5219      0.099      5.289      0.000       0.328       0.716
Mkt-RF         1.0015      0.024     41.842      0.000       0.954       1.049
SMB           -0.2152      0.035     -6.105      0.000      -0.284      -0.146
HML           -0.1231      0.045     -2.756      0.006      -0.211      -0.035
RMW           -0.0823      0.046     -1.778      0.076      -0.173       0.009
CMA           -0.2788      0.066     -4.202      0.000      -0.409      -0.148
Mom           -0.0386      0.022     -1.753      0.080      -0.082       0.005
==============================================================================
Omnibus:                       12.777   Durbin-Watson:                   2.116
Prob(Omnibus):                  0.002   Jarque-Bera (JB):               22.430
Skew:                           0.149   Prob(JB):                     1.35e-05
Kurtosis:                       4.064   Cond. No.                         5.62
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
################################################################################
############
EQUAL WEIGHTED PORTFOLIO RESULTS:
Mean returns = 0.009
Portfolio volatility = 0.053696
Strategy Sharpe Ratio = 0.174329
CAPM ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.899
```

```
Model:                            OLS   Adj. R-squared:                  0.899
Method:                 Least Squares   F-statistic:                     3915.
Date:                Sat, 27 Apr 2024   Prob (F-statistic):          7.79e-221
Time:                        04:51:52   Log-Likelihood:                -860.29
No. Observations:                 441   AIC:                             1725.
Df Residuals:                     439   BIC:                             1733.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.3642      0.082      4.422      0.000       0.202       0.526
Mkt-RF         1.1335      0.018     62.569      0.000       1.098       1.169
==============================================================================
Omnibus:                       36.269   Durbin-Watson:                   1.803
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              148.286
Skew:                           0.169   Prob(JB):                     6.31e-33
Kurtosis:                       5.821   Cond. No.                         4.61
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
FF5 ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.915
Model:                            OLS   Adj. R-squared:                  0.914
Method:                 Least Squares   F-statistic:                     935.9
Date:                Sat, 27 Apr 2024   Prob (F-statistic):          3.44e-230
Time:                        04:51:52   Log-Likelihood:                -822.76
No. Observations:                 441   AIC:                             1658.
Df Residuals:                     435   BIC:                             1682.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.3520      0.079      4.436      0.000       0.196       0.508
Mkt-RF         1.1093      0.019     58.154      0.000       1.072       1.147
SMB            0.2187      0.029      7.657      0.000       0.163       0.275
HML            0.1404      0.035      4.043      0.000       0.072       0.209
RMW            0.0845      0.037      2.266      0.024       0.011       0.158
CMA           -0.1296      0.053     -2.427      0.016      -0.235      -0.025
==============================================================================
Omnibus:                       23.275   Durbin-Watson:                   1.939
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               52.987
Skew:                           0.244   Prob(JB):                     3.12e-12
```

```
Kurtosis:                       4.626   Cond. No.                         5.31
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
FF5 + MOMENTUM ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.927
Model:                            OLS   Adj. R-squared:                  0.926
Method:                 Least Squares   F-statistic:                     915.2
Date:                Sat, 27 Apr 2024   Prob (F-statistic):          9.43e-243
Time:                        04:51:52   Log-Likelihood:                -789.82
No. Observations:                 441   AIC:                             1594.
Df Residuals:                     434   BIC:                             1622.
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.4311      0.074      5.800      0.000       0.285       0.577
Mkt-RF         1.0815      0.018     59.985      0.000       1.046       1.117
SMB            0.2241      0.027      8.439      0.000       0.172       0.276
HML            0.0601      0.034      1.786      0.075      -0.006       0.126
RMW            0.1148      0.035      3.295      0.001       0.046       0.183
CMA           -0.0795      0.050     -1.591      0.112      -0.178       0.019
Mom           -0.1386      0.017     -8.363      0.000      -0.171      -0.106
==============================================================================
Omnibus:                       10.071   Durbin-Watson:                   1.880
Prob(Omnibus):                  0.007   Jarque-Bera (JB):               17.535
Skew:                           0.058   Prob(JB):                     0.000156
Kurtosis:                       3.970   Cond. No.                         5.62
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

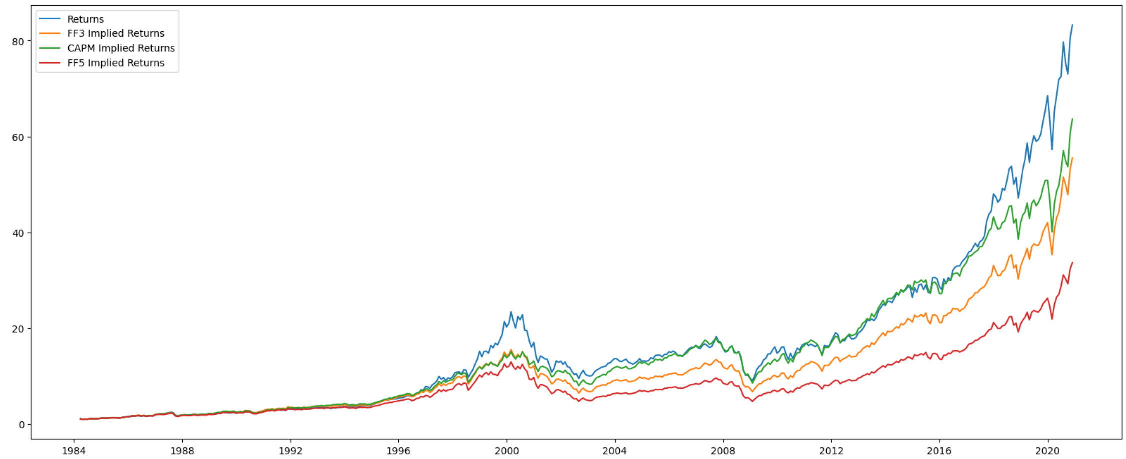
6 2C

The output in the 2 cells below displays: * The plot of cumulative returns for the value weighted portfolio as well as the model implied cumulative returns for the various factor models * The plot of cumulative returns for the equal weighted portfolio as well as the model implied cumulative returns for the various factor models

We can clearly see for both portfolios relative to the CAPM implied returns and the general market returns (around 50-60x calculated in a test not displayed) that the portfolio does tend to outperform the benchmark of the market over the time period of the test (1984-2020)

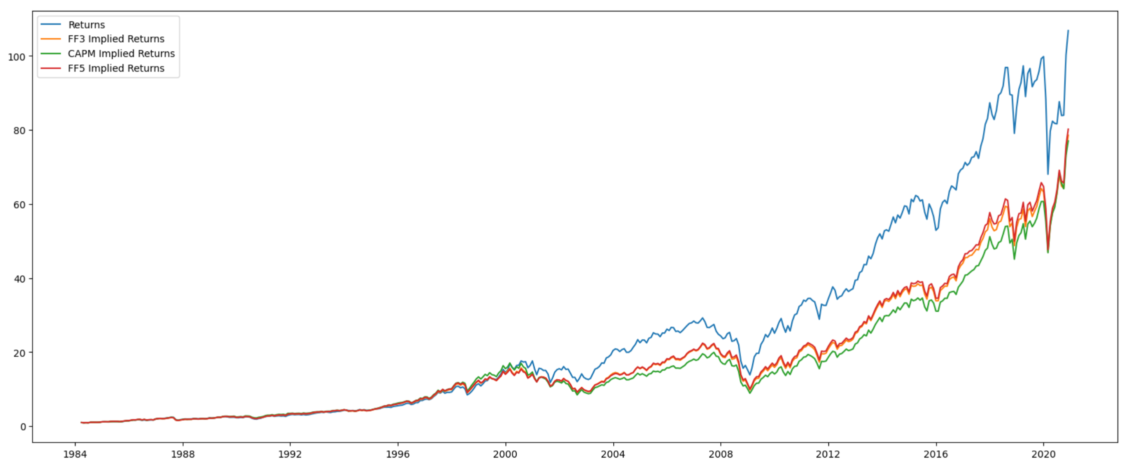
```
[ ]: disp_portfolio(data_val_w, rets_w, vals_w, capm_w, ff3_w, ff5_w)
```

```
0.011430050589911093
0.05179853869304044
Sharpe Ratio: 0.22066357233832185
```




```
[ ]: disp_portfolio(data_val_e, rets_e, vals_e, capm_e, ff3_e, ff5_e)
```

```
0.012109906831284522
0.05360090604120217
Sharpe Ratio: 0.2259272785795044
```



7 2D

The 2 cells below estimate the factor models and plot cumulative returns for the dataset pre- and post- 2010 which is around when the Edmans paper was published. We notice that in the pre- timeframe the strategy has significant positive alpha according to the factor models, however after 2010 the alpha falls significantly to ~.1 as implied by the FF5 model. This suggests along with the cumulative return plots of strategy returns vs. CAPM benchmark returns that the strategy does not work in the post- period. From the data in part A-C we also notice the equal weighted portfolio outperforming the value weighted (higher alpha and sharpe ratio seen above) which follows along with table 4 in Edmans (2011) in which the equal weighted portfolios perform better in testing. Furthermore we see that the averages of the 2 alphas from pre- and post- is around the alpha from Edmans' results

[ ]:
```python
data_val, rets, vals, capm, ff3, ff5, ff5mom, industry =␣
    ↪sim_portfolio(crsp_data, False, '2010-01-01', 'Pre')
disp_portfolio(data_val, rets, vals, capm, ff3, ff5)
```

```
<ipython-input-182-f86afb83a410>:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['RET'] = pd.to_numeric(data['RET'], errors='coerce')
<ipython-input-182-f86afb83a410>:14: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['weighted_val_ret'] = data['val_weighting'] * data['RET']
<ipython-input-182-f86afb83a410>:15: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['weighted_eq_ret'] = data['eq_weighting'] * data['RET']
```

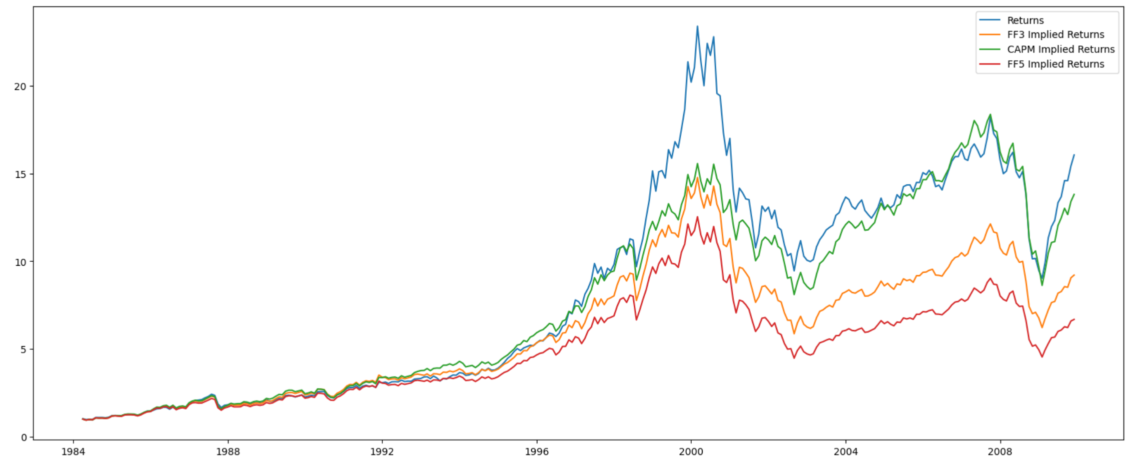
```
Mean returns = 0.007
Portfolio volatility = 0.053708
Strategy Sharpe Ratio = 0.125593
CAPM ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.830
Model:                            OLS   Adj. R-squared:                  0.829
Method:                 Least Squares   F-statistic:                     1498.
Date:                Sat, 27 Apr 2024   Prob (F-statistic):          4.09e-120
```

```
Time:                        04:52:09   Log-Likelihood:                -684.50
No. Observations:                 309   AIC:                             1373.
Df Residuals:                     307   BIC:                             1380.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.4461      0.128      3.499      0.001       0.195       0.697
Mkt-RF         1.0720      0.028     38.701      0.000       1.017       1.126
==============================================================================
Omnibus:                       19.985   Durbin-Watson:                   2.091
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               62.900
Skew:                           0.064   Prob(JB):                     2.19e-14
Kurtosis:                       5.207   Cond. No.                         4.64
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
FF5 ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.868
Model:                            OLS   Adj. R-squared:                  0.866
Method:                 Least Squares   F-statistic:                     398.7
Date:                Sat, 27 Apr 2024   Prob (F-statistic):          6.35e-131
Time:                        04:52:09   Log-Likelihood:                -645.26
No. Observations:                 309   AIC:                             1303.
Df Residuals:                     303   BIC:                             1325.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.6773      0.120      5.661      0.000       0.442       0.913
Mkt-RF         0.9905      0.029     33.957      0.000       0.933       1.048
SMB           -0.2094      0.041     -5.160      0.000      -0.289      -0.130
HML           -0.1814      0.053     -3.421      0.001      -0.286      -0.077
RMW           -0.0864      0.052     -1.656      0.099      -0.189       0.016
CMA           -0.2410      0.078     -3.105      0.002      -0.394      -0.088
==============================================================================
Omnibus:                       12.511   Durbin-Watson:                   2.085
Prob(Omnibus):                  0.002   Jarque-Bera (JB):               17.298
Skew:                           0.315   Prob(JB):                     0.000175
Kurtosis:                       3.974   Cond. No.                         5.64
==============================================================================
```

```
Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
FF5 + MOMENTUM ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.870
Model:                            OLS   Adj. R-squared:                  0.867
Method:                 Least Squares   F-statistic:                     336.3
Date:                Sat, 27 Apr 2024   Prob (F-statistic):         1.76e-130
Time:                        04:52:09   Log-Likelihood:                -643.19
No. Observations:                 309   AIC:                             1300.
Df Residuals:                     302   BIC:                             1327.
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.7107      0.120      5.914      0.000       0.474       0.947
Mkt-RF         0.9807      0.029     33.331      0.000       0.923       1.039
SMB           -0.2059      0.040     -5.096      0.000      -0.285      -0.126
HML           -0.2114      0.055     -3.857      0.000      -0.319      -0.104
RMW           -0.0737      0.052     -1.408      0.160      -0.177       0.029
CMA           -0.2208      0.078     -2.836      0.005      -0.374      -0.068
Mom           -0.0489      0.024     -2.021      0.044      -0.097      -0.001
==============================================================================
Omnibus:                       13.258   Durbin-Watson:                   2.070
Prob(Omnibus):                  0.001   Jarque-Bera (JB):               17.434
Skew:                           0.352   Prob(JB):                     0.000164
Kurtosis:                       3.927   Cond. No.                         5.94
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
0.010487723790098252
0.05376061361225146
Sharpe Ratio: 0.1950819212321679
```




```
[ ]: data_val, rets, vals, capm, ff3, ff5, ff5mom, industry =␣
     ↪sim_portfolio(crsp_data, False, '2010-01-01', 'Post')
     disp_portfolio(data_val, rets, vals, capm, ff3, ff5)
```

```
<ipython-input-182-f86afb83a410>:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['RET'] = pd.to_numeric(data['RET'], errors='coerce')
<ipython-input-182-f86afb83a410>:14: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['weighted_val_ret'] = data['val_weighting'] * data['RET']
<ipython-input-182-f86afb83a410>:15: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['weighted_eq_ret'] = data['eq_weighting'] * data['RET']
```

```
Mean returns = 0.013
Portfolio volatility = 0.046821
Strategy Sharpe Ratio = 0.282176
CAPM ESTIMATES
                            OLS Regression Results
==============================================================================
```

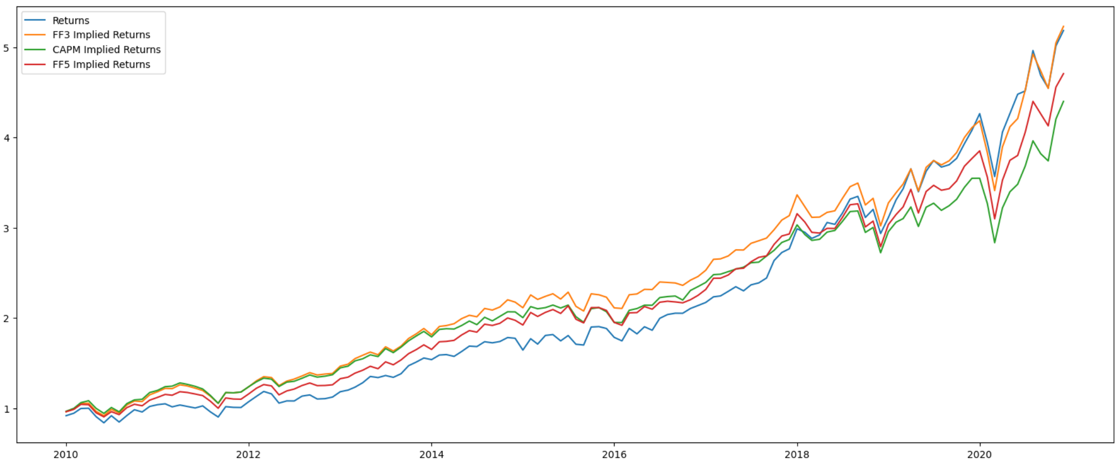
```
Dep. Variable:                      y   R-squared:                       0.817
Model:                            OLS   Adj. R-squared:                  0.816
Method:                 Least Squares   F-statistic:                     581.7
Date:                Sat, 27 Apr 2024   Prob (F-statistic):           7.84e-50
Time:                        04:52:10   Log-Likelihood:                -278.84
No. Observations:                 132   AIC:                             561.7
Df Residuals:                     130   BIC:                             567.4
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.1881      0.182      1.033      0.303      -0.172       0.548
Mkt-RF         0.9961      0.041     24.118      0.000       0.914       1.078
==============================================================================
Omnibus:                        0.498   Durbin-Watson:                   2.140
Prob(Omnibus):                  0.780   Jarque-Bera (JB):                0.183
Skew:                          -0.025   Prob(JB):                        0.912
Kurtosis:                       3.175   Cond. No.                         4.59
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
FF5 ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.864
Model:                            OLS   Adj. R-squared:                  0.858
Method:                 Least Squares   F-statistic:                     159.7
Date:                Sat, 27 Apr 2024   Prob (F-statistic):           9.37e-53
Time:                        04:52:10   Log-Likelihood:                -259.53
No. Observations:                 132   AIC:                             531.1
Df Residuals:                     126   BIC:                             548.3
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.1312      0.166      0.792      0.430      -0.197       0.459
Mkt-RF         1.0711      0.041     25.988      0.000       0.990       1.153
SMB           -0.3725      0.076     -4.916      0.000      -0.522      -0.223
HML            0.0945      0.075      1.266      0.208      -0.053       0.242
RMW           -0.0793      0.108     -0.733      0.465      -0.293       0.135
CMA           -0.4784      0.126     -3.798      0.000      -0.728      -0.229
==============================================================================
Omnibus:                        9.999   Durbin-Watson:                   2.216
Prob(Omnibus):                  0.007   Jarque-Bera (JB):               11.441
```

```
Skew:                          -0.502   Prob(JB):                      0.00328
Kurtosis:                       4.036   Cond. No.                         5.13
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
FF5 + MOMENTUM ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.864
Model:                            OLS   Adj. R-squared:                  0.857
Method:                 Least Squares   F-statistic:                     132.2
Date:                Sat, 27 Apr 2024   Prob (F-statistic):           1.15e-51
Time:                        04:52:10   Log-Likelihood:                -259.44
No. Observations:                 132   AIC:                             532.9
Df Residuals:                     125   BIC:                             553.1
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.1355      0.167      0.813      0.418      -0.194       0.465
Mkt-RF         1.0683      0.042     25.454      0.000       0.985       1.151
SMB           -0.3756      0.076     -4.914      0.000      -0.527      -0.224
HML            0.0835      0.080      1.046      0.297      -0.074       0.241
RMW           -0.0805      0.109     -0.741      0.460      -0.295       0.134
CMA           -0.4749      0.127     -3.749      0.000      -0.726      -0.224
Mom           -0.0209      0.052     -0.397      0.692      -0.125       0.083
==============================================================================
Omnibus:                        9.828   Durbin-Watson:                   2.211
Prob(Omnibus):                  0.007   Jarque-Bera (JB):               11.004
Skew:                          -0.506   Prob(JB):                      0.00408
Kurtosis:                       3.989   Cond. No.                         5.44
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
0.013635951962200245
0.046811394178296065
Sharpe Ratio: 0.2912955745403222
```



8 2E

The two cells below estimate the factor models and industry loadings as well as plot the strategy and models' cumulative returns for pre- and post- 1999 respectively. We can clearly see from the differences in the 12 industry loadings that the list of companies to work for has changed over time as the industry weightage of the portfolio varies from pre- and post-. In general, we see that BusEq was a strong weight that increased even further while industries such as Telecom's diminished over time, other industries also experience some level of weight shifting suggesting that the composition of top companies to work for changes over time.

[ ]:
```
data_val, rets, vals, capm, ff3, ff5, ff5mom, industry =␣
↪sim_portfolio(crsp_data, False, '1999-01-01', 'Pre', True)
disp_portfolio(data_val, rets, vals, capm, ff3, ff5, ff5mom, industry)
```

```
<ipython-input-182-f86afb83a410>:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['RET'] = pd.to_numeric(data['RET'], errors='coerce')
<ipython-input-182-f86afb83a410>:14: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['weighted_val_ret'] = data['val_weighting'] * data['RET']
<ipython-input-182-f86afb83a410>:15: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['weighted_eq_ret'] = data['eq_weighting'] * data['RET']

Mean returns = 0.011
Portfolio volatility = 0.047034
Strategy Sharpe Ratio = 0.237779
CAPM ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.856
Model:                            OLS   Adj. R-squared:                  0.855
Method:                 Least Squares   F-statistic:                     1042.
Date:                Sat, 27 Apr 2024   Prob (F-statistic):           1.34e-75
Time:                        04:52:15   Log-Likelihood:                -353.42
No. Observations:                 177   AIC:                             710.8
Df Residuals:                     175   BIC:                             717.2
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.6273      0.138      4.546      0.000       0.355       0.900
Mkt-RF         0.9909      0.031     32.274      0.000       0.930       1.051
==============================================================================
Omnibus:                        3.267   Durbin-Watson:                   2.053
Prob(Omnibus):                  0.195   Jarque-Bera (JB):                3.563
Skew:                           0.078   Prob(JB):                        0.168
Kurtosis:                       3.677   Cond. No.                         4.61
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
FF5 ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.890
Model:                            OLS   Adj. R-squared:                  0.887
Method:                 Least Squares   F-statistic:                     277.9
Date:                Sat, 27 Apr 2024   Prob (F-statistic):           4.09e-80
Time:                        04:52:15   Log-Likelihood:                -329.35
No. Observations:                 177   AIC:                             670.7
Df Residuals:                     171   BIC:                             689.8
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
```

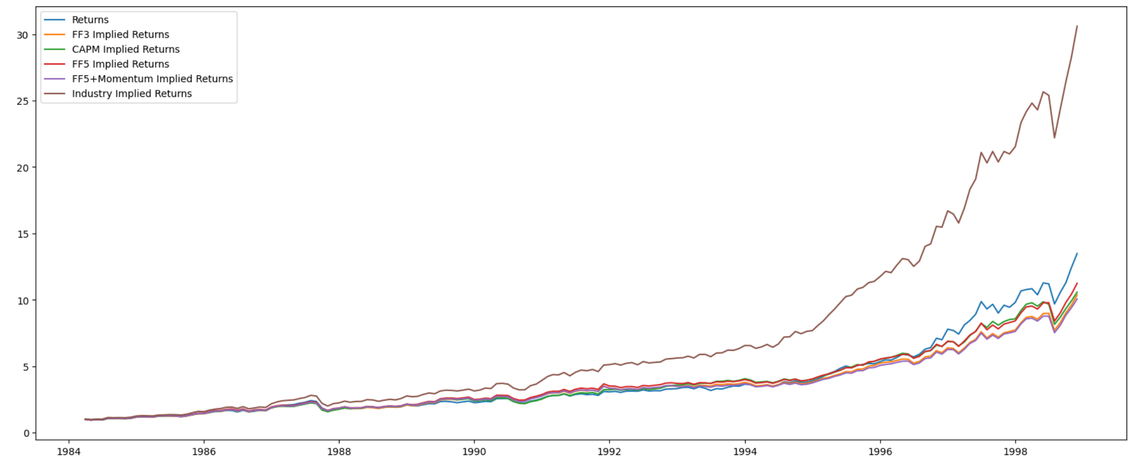
```
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.5894      0.138      4.270      0.000       0.317       0.862
Mkt-RF         0.9469      0.033     28.710      0.000       0.882       1.012
SMB           -0.2916      0.054     -5.430      0.000      -0.398      -0.186
HML           -0.2164      0.077     -2.820      0.005      -0.368      -0.065
RMW            0.1089      0.099      1.102      0.272      -0.086       0.304
CMA           -0.0701      0.108     -0.646      0.519      -0.284       0.144
==============================================================================
Omnibus:                        1.116   Durbin-Watson:                   2.046
Prob(Omnibus):                  0.572   Jarque-Bera (JB):                0.798
Skew:                           0.142   Prob(JB):                        0.671
Kurtosis:                       3.164   Cond. No.                         6.07
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
FF5 + MOMENTUM ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.894
Model:                            OLS   Adj. R-squared:                  0.890
Method:                 Least Squares   F-statistic:                     238.1
Date:                Sat, 27 Apr 2024   Prob (F-statistic):           5.61e-80
Time:                        04:52:15   Log-Likelihood:                -326.69
No. Observations:                 177   AIC:                             667.4
Df Residuals:                     170   BIC:                             689.6
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.6528      0.139      4.690      0.000       0.378       0.928
Mkt-RF         0.9563      0.033     29.115      0.000       0.891       1.021
SMB           -0.3150      0.054     -5.829      0.000      -0.422      -0.208
HML           -0.2349      0.076     -3.081      0.002      -0.385      -0.084
RMW            0.1234      0.098      1.261      0.209      -0.070       0.317
CMA           -0.0420      0.108     -0.389      0.697      -0.255       0.171
Mom           -0.0985      0.043     -2.278      0.024      -0.184      -0.013
==============================================================================
Omnibus:                        2.074   Durbin-Watson:                   2.058
Prob(Omnibus):                  0.355   Jarque-Bera (JB):                1.656
Skew:                           0.195   Prob(JB):                        0.437
Kurtosis:                       3.268   Cond. No.                         6.15
==============================================================================

Notes:
```

```
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
INDUSTRY LOADINGS ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.929
Model:                            OLS   Adj. R-squared:                  0.924
Method:                 Least Squares   F-statistic:                     179.5
Date:                Sat, 27 Apr 2024   Prob (F-statistic):           1.22e-87
Time:                        04:52:15   Log-Likelihood:                -290.61
No. Observations:                 177   AIC:                             607.2
Df Residuals:                     164   BIC:                             648.5
Df Model:                          12
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.0114      0.108      0.105      0.916      -0.202       0.225
NoDur          0.1751      0.054      3.219      0.002       0.068       0.283
Durbl          0.0126      0.036      0.351      0.726      -0.058       0.084
Manuf          0.1157      0.083      1.391      0.166      -0.049       0.280
Enrgy          0.1320      0.029      4.586      0.000       0.075       0.189
Chems          0.1513      0.057      2.643      0.009       0.038       0.264
BusEq          0.4451      0.029     15.227      0.000       0.387       0.503
Telcm          0.0770      0.035      2.199      0.029       0.008       0.146
Utils         -0.0325      0.042     -0.782      0.436      -0.115       0.050
Shops          0.0051      0.046      0.110      0.912      -0.086       0.097
Hlth           0.1496      0.038      3.953      0.000       0.075       0.224
Money          0.0628      0.042      1.513      0.132      -0.019       0.145
Other         -0.3327      0.067     -4.942      0.000      -0.466      -0.200
==============================================================================
Omnibus:                        7.559   Durbin-Watson:                   2.172
Prob(Omnibus):                  0.023   Jarque-Bera (JB):                8.690
Skew:                           0.332   Prob(JB):                       0.0130
Kurtosis:                       3.859   Cond. No.                         17.8
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
0.015920993548317903
0.04698664570712381
Sharpe Ratio: 0.338840819741769
```




```
[ ]: data_val, rets, vals, capm, ff3, ff5, ff5mom, industry =␣
      ↪sim_portfolio(crsp_data, False, '1999-01-01', 'Post', True)
     disp_portfolio(data_val, rets, vals, capm, ff3, ff5, ff5mom, industry)
```

```
<ipython-input-182-f86afb83a410>:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['RET'] = pd.to_numeric(data['RET'], errors='coerce')
<ipython-input-182-f86afb83a410>:14: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['weighted_val_ret'] = data['val_weighting'] * data['RET']
<ipython-input-182-f86afb83a410>:15: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  data['weighted_eq_ret'] = data['eq_weighting'] * data['RET']
```

```
Mean returns = 0.007
Portfolio volatility = 0.054743
Strategy Sharpe Ratio = 0.127922
CAPM ESTIMATES
                            OLS Regression Results
==============================================================================
```

```
Dep. Variable:                      y   R-squared:                       0.813
Model:                            OLS   Adj. R-squared:                  0.812
Method:                 Least Squares   F-statistic:                     1135.
Date:                Sat, 27 Apr 2024   Prob (F-statistic):           3.13e-97
Time:                        04:52:17   Log-Likelihood:                -601.66
No. Observations:                 264   AIC:                             1207.
Df Residuals:                     262   BIC:                             1214.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.1971      0.147      1.338      0.182      -0.093       0.487
Mkt-RF         1.0834      0.032     33.697      0.000       1.020       1.147
==============================================================================
Omnibus:                       14.735   Durbin-Watson:                   2.116
Prob(Omnibus):                  0.001   Jarque-Bera (JB):               36.114
Skew:                           0.136   Prob(JB):                     1.44e-08
Kurtosis:                       4.791   Cond. No.                         4.62
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
FF5 ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.858
Model:                            OLS   Adj. R-squared:                  0.855
Method:                 Least Squares   F-statistic:                     312.1
Date:                Sat, 27 Apr 2024   Prob (F-statistic):          3.57e-107
Time:                        04:52:17   Log-Likelihood:                -564.89
No. Observations:                 264   AIC:                             1142.
Df Residuals:                     258   BIC:                             1163.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.3418      0.135      2.529      0.012       0.076       0.608
Mkt-RF         1.0539      0.034     31.439      0.000       0.988       1.120
SMB           -0.1480      0.047     -3.131      0.002      -0.241      -0.055
HML           -0.1161      0.055     -2.109      0.036      -0.225      -0.008
RMW           -0.0431      0.060     -0.716      0.475      -0.162       0.076
CMA           -0.3703      0.082     -4.520      0.000      -0.532      -0.209
==============================================================================
Omnibus:                        5.945   Durbin-Watson:                   2.146
Prob(Omnibus):                  0.051   Jarque-Bera (JB):                7.880
```

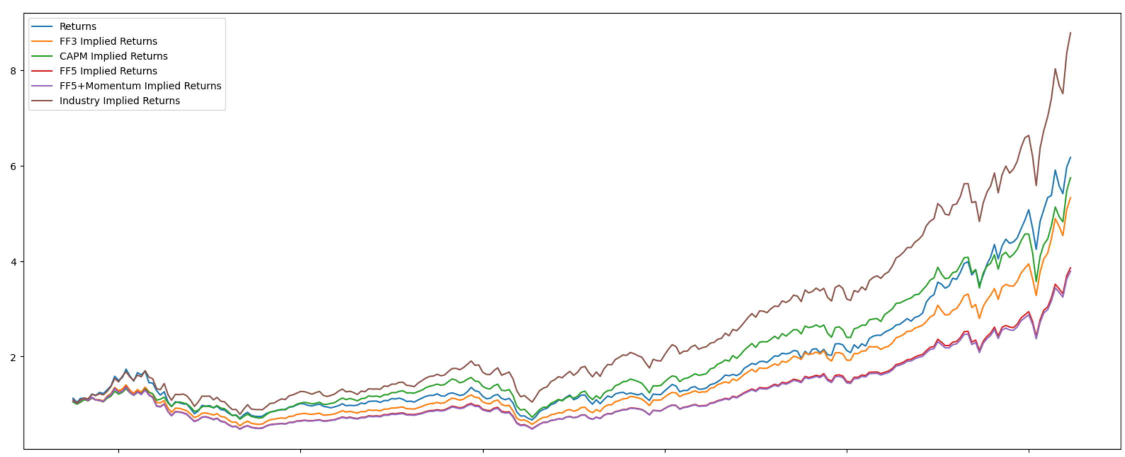
```
Skew:                           0.140   Prob(JB):                       0.0195
Kurtosis:                       3.798   Cond. No.                         5.51
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
FF5 + MOMENTUM ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.859
Model:                            OLS   Adj. R-squared:                  0.855
Method:                 Least Squares   F-statistic:                     260.2
Date:                Sat, 27 Apr 2024   Prob (F-statistic):         4.11e-106
Time:                        04:52:17   Log-Likelihood:                -564.39
No. Observations:                 264   AIC:                             1143.
Df Residuals:                     257   BIC:                             1168.
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.3489      0.135      2.577      0.011       0.082       0.615
Mkt-RF         1.0441      0.035     29.845      0.000       0.975       1.113
SMB           -0.1412      0.048     -2.955      0.003      -0.235      -0.047
HML           -0.1332      0.058     -2.307      0.022      -0.247      -0.019
RMW           -0.0362      0.061     -0.597      0.551      -0.156       0.083
CMA           -0.3626      0.082     -4.405      0.000      -0.525      -0.201
Mom           -0.0267      0.027     -0.982      0.327      -0.080       0.027
==============================================================================
Omnibus:                        6.026   Durbin-Watson:                   2.134
Prob(Omnibus):                  0.049   Jarque-Bera (JB):                7.754
Skew:                           0.161   Prob(JB):                       0.0207
Kurtosis:                       3.776   Cond. No.                         6.45
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
INDUSTRY LOADINGS ESTIMATES
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.896
Model:                            OLS   Adj. R-squared:                  0.891
Method:                 Least Squares   F-statistic:                     180.0
Date:                Sat, 27 Apr 2024   Prob (F-statistic):         8.25e-116
Time:                        04:52:17   Log-Likelihood:                -524.00
No. Observations:                 264   AIC:                             1074.
```

```
Df Residuals:                     251   BIC:                             1120.
Df Model:                          12
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.0231      0.117      0.198      0.843      -0.207       0.253
NoDur          0.0721      0.058      1.243      0.215      -0.042       0.186
Durbl          0.0175      0.024      0.725      0.469      -0.030       0.065
Manuf          0.0274      0.059      0.464      0.643      -0.089       0.144
Enrgy          0.0193      0.024      0.817      0.415      -0.027       0.066
Chems          0.0980      0.055      1.768      0.078      -0.011       0.207
BusEq          0.5820      0.029     20.376      0.000       0.526       0.638
Telcm          0.0129      0.036      0.361      0.719      -0.057       0.083
Utils         -0.0870      0.035     -2.475      0.014      -0.156      -0.018
Shops          0.0355      0.047      0.760      0.448      -0.057       0.128
Hlth           0.0678      0.038      1.789      0.075      -0.007       0.143
Money          0.1318      0.039      3.407      0.001       0.056       0.208
Other         -0.0788      0.070     -1.122      0.263      -0.217       0.060
==============================================================================
Omnibus:                        5.013   Durbin-Watson:                   2.184
Prob(Omnibus):                  0.082   Jarque-Bera (JB):                6.955
Skew:                           0.011   Prob(JB):                       0.0309
Kurtosis:                       3.795   Cond. No.                         17.0
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
0.008419077470070164
0.05458202003848453
Sharpe Ratio: 0.15424635189635827
```



9 Question 3

9.1 Part A

The reason the beta of this strategy is close to 1 is because this list of companies is related to the employee satisfaction and not correlated with the beta values of each company. Therefore, it is logical the average beta of all the companies is 1 and the portfolio is a long-only portfolio (market returns are not negated by any short positions). Since the companies are "randomly" selected from all companies in the market, which has a beta of 1.

Institutional investors care more about stable, consistant, uncorrelated returns. This is different from retail investors, which often care more about absolute returns. This long-short strategy will create a pure-alpha strategy with returns uncorrelated with the market, which is attractive for institutions. However, the absolute returns from the long-short may be smaller, which is why it is less attractive for retail investors.

9.2 Part B

Our results indicate the market does not fully price in employee satisfaction becuase there is alpha in going long the companies with the highest satisfaction. This means that there is an opportunity to make better risk-adjusted returns by investing in companies with higher employee satisfaction than investing in the market, which means the market has not fully priced in employee satisfaction.

9.3 Part C

As the knowledge that there was alpha in investing in companies with higher employee satisfaction spreads, investors will deploy capital in the strategy. This drives up the price of these stocks and means that the price of companies with higher employee satisfaction are fairly priced. This means that over time the market has priced in the benefit to companies from having higher employee satisfaction, meaning that the alpha has decreased.

9.4 Part D

There are several different angles that we can take for seeing how cookies data can be relevant for company stock prices.

One method for using cookie data is to see how much time employees are spending on non-productive sites during the work day. For instance, if someone works in finance and they are spending significant time on social media, watching sports, or reading about cooking during the work day, they are probably unsatisfied with their job. Furthermore, if employees are doing things unrelated to their job during the work day, they are not as productive in their role. We can use this reasoning to believe that companies whose employees spend less time on unrelated websites are more satisfied, and companies whose employees spend more time on unrelated websites are less satisfied.

Another method for using cookie data could be more related to evaluating employees at software companies. One thing about software engineering is that there is constant innovation, and people are constantly developing new frameworks for different tasks, and improved ways of doing things. In this context, software companies with employees that spend more time on sites that talk about innovative computer science ideas (such as Medium) may outperform those companies with less inquisitive engineers. Companies with more curious engineers may promote a better engineering culture and may be more likely to innovate due to their engineers spending more time thinking about advancements in technology.