

## **A report on stock analysis**

Group 6

Peihao Cheng, Bhavya Gala, Haoran Li, Zheng Li, Kamil Yusubov

04/21/2020

### **Contents**

1. Introduction.....	
2. Data.....	
3. Results.....	
4. Conclusion.....	

## Introduction

Momentum investing is a strategy which capitalizes on the continuance of an existing market trend. It involves going long on securities that show upward-trending prices and short the respective assets with downward-trending prices. In this report, we observe the results of the momentum strategy used on historical stock prices. We used CAPM, Factor model, Shapiro-Wilk Test and the visualisation of historical drawdown to find conclusive results that the strategy return is always larger than the risk free return.

## Data

In order to prepare the data for our analysis, we imported pandas, numpy, matplotlib and statsmodel. Then, we import the CRSP file which holds stock data and the FF file to get the three factor model data and the risk free rate. Further, we set the data in an exploitable format and calculate the past 12 month rolling returns for each month. Next, we form portfolios for every month, a portfolio consisting of different stocks and divide those into 10 groups. We shift two months backward to be certain that we use previous returns to predict future returns. We then merge the previous data and the portfolios, this is the final step in our data set creation.

We then discover some statistics for the portfolios we just created. We calculate characteristics such as mean and standard deviation for the top and bottom portfolios. Using the function 'describe' we calculate the characteristics of all the portfolios for all months. For further analysis, we plot and calculate the mean and standard deviation of the 10 portfolios we created.

We use the difference between top portfolio return and bottom portfolio return as strategy return. Then we take risk free rate away from it, we can get strategy excess. In order to make calculation convenient, we convert all returns into log returns, which shows as portfolio2 in codes(Graph 1). Here we also plot strategy return and market return. Besides, the standard deviation of the strategy return and market return are 8.40 and 5.34 individually. While the 12-month rolling excess return is 0.47.

## Graph 1: portfolios2

```
In [16]: portfolios2
```

```
Out[16]:
```

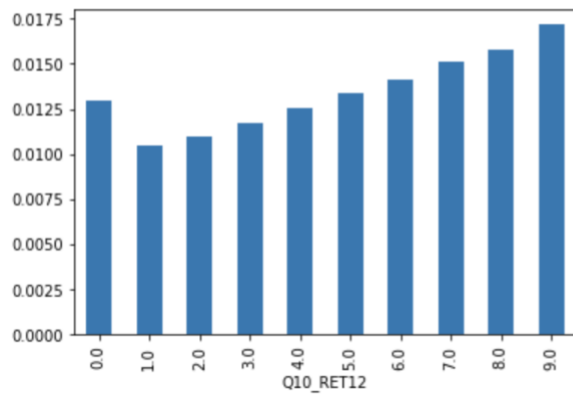
	Mkt_RF	SMB	HML	RF	Mkt	strategy_ret	strategy_excess	log_mktrf	log_strategyrf	log_mkt	log_strategy
date											
1928-02-29	-1.70	-2.03	-0.69	0.33	-1.37	-7.213916	-7.433916	-0.017146	-0.077247	-0.013795	-0.074874
1928-03-31	8.81	-0.26	-1.20	0.29	9.10	5.433933	5.183933	0.084433	0.050540	0.087095	0.052914
1928-04-30	4.23	3.82	3.67	0.22	4.45	-6.128900	-6.358900	0.041430	-0.065701	0.043538	-0.063248
1928-05-31	1.52	2.98	-3.46	0.32	1.84	-10.353505	-10.673505	0.015086	-0.112872	0.018233	-0.109296
1928-06-30	-4.85	-3.50	-0.06	0.31	-4.54	1.910393	1.600393	-0.049716	0.015877	-0.046463	0.018924
...	...	...	...	...	...	...	...	...	...	...	...
2019-08-31	-2.58	-2.41	-4.99	0.16	-2.42	4.227480	4.117480	-0.026139	0.040350	-0.024498	0.041406
2019-09-30	1.44	-0.90	6.71	0.18	1.62	-8.108368	-8.218368	0.014297	-0.085758	0.016070	-0.084560
2019-10-31	2.06	0.25	-2.07	0.15	2.21	6.493778	6.373778	0.020391	0.061789	0.021859	0.062916
2019-11-30	3.87	0.87	-1.86	0.12	3.99	2.770538	2.630538	0.037970	0.025965	0.039125	0.027329
2019-12-31	2.77	0.69	1.82	0.14	2.91	-9.106043	-9.246043	0.027323	-0.097018	0.028685	-0.095477

## Results

**Graph 2: Mean Return of 10 Portfolios**

```
# this chart shows the mean return of 10 portfolios in all months  
port_ret['mean'].plot(kind = 'bar')
```

<matplotlib.axes.\_subplots.AxesSubplot at 0x1a4311b790>

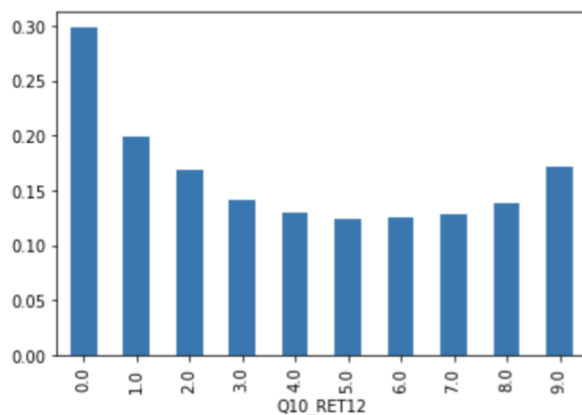


We have divided the stocks in 10 portfolios according to the momentum strategy. This graph shows the mean return of these portfolios. Except for the bottom portfolio, we observe an upward trend, where the top portfolio has the highest mean return.

**Graph 3: Standard Deviation of 10 Portfolios**

```
# this chart shows the standard deviation of 10 portfolios in all months  
port_ret['std'].plot(kind = 'bar')
```

<matplotlib.axes.\_subplots.AxesSubplot at 0x1a20f19b90>

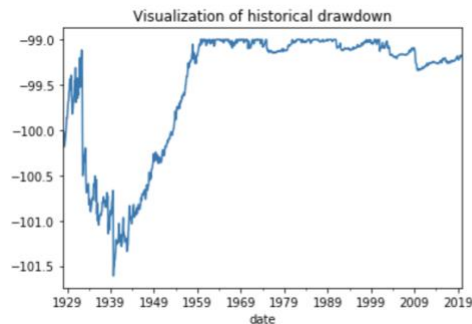


This graph shows the standard deviation of all the portfolios, we notice that the top and bottom portfolios have high standard deviations whereas portfolios with stocks that haven't moved have a lower deviation.

#### Graph 4: Historical drawdown

```
In [70]: # Visualization of historical drawdown
portfolios2['cum_mom'] = portfolios2['strategy_ret'].cumsum()
cum_mom = portfolios2['cum_mom']
running_max = np.maximum.accumulate(cum_mom)
running_max[running_max < 100] = 100
drawdown = (cum_mom)/running_max - 100
plt.title('Visualization of historical drawdown')
drawdown.plot()
```

Out[70]: <matplotlib.axes.\_subplots.AxesSubplot at 0x1a1babbfd0>

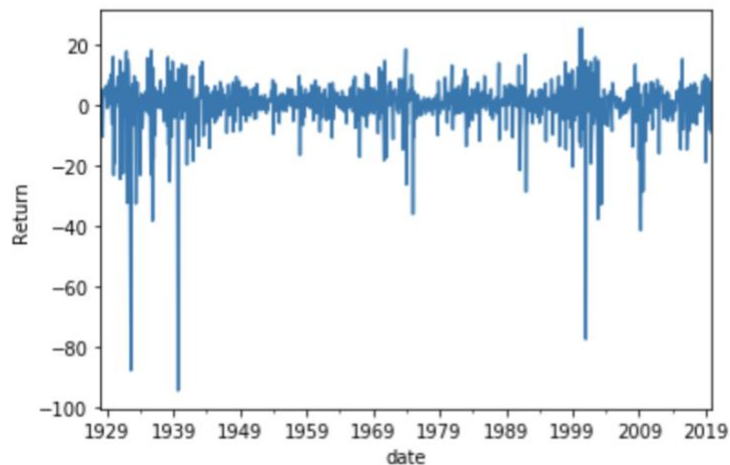


The Graph 4 shows historical drawdown of the strategy return, we use the cumsum function to get this chart. We visualize history drawdown of our momentum strategy and we can see from the graph that our momentum strategy has very low drawdown risk after 1959.

#### Graph 5: Strategy Returns

```
portfolios2['strategy_ret'].plot()
plt.ylabel('Return')
```

Text(0, 0.5, 'Return')

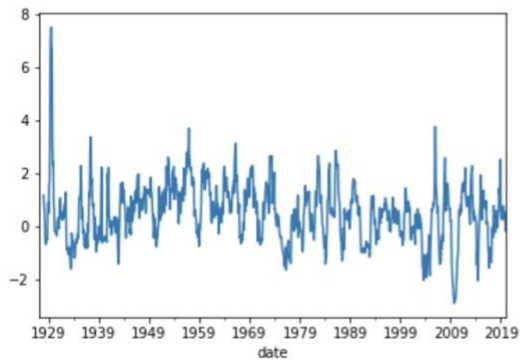


As the Graph 5 shows, the strategy return fluctuated during the 90-years period. There are several significant increases that occurred at around 1934, 1941 and 2001.

**Graph 6: Sharpe ratio of strategy return**

```
def SR(x):  
    return np.sqrt(12) * np.mean(x) / np.std(x)  
portfolios2['strategy_excess'].rolling(window = 12).apply(SR, raw = True).plot()
```

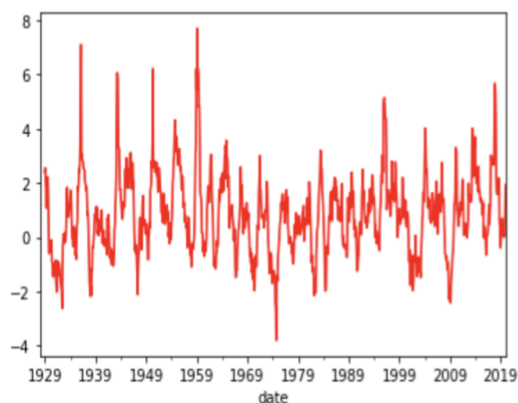
<matplotlib.axes.\_subplots.AxesSubplot at 0x19003624a08>



**Graph 7: Sharpe Ratio of market return**

```
(portfolios2['Mkt_RF']).rolling(window = 12).apply(SR, raw = True).plot(color = 'red')
```

<matplotlib.axes.\_subplots.AxesSubplot at 0x1a1b4ec150>



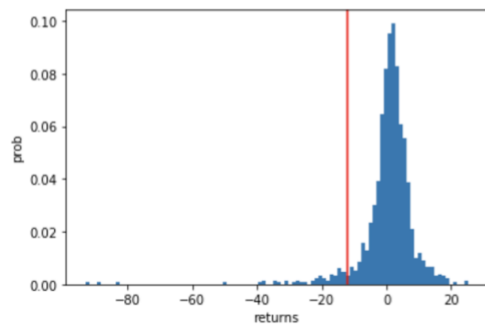
Sharp ratio could be calculated via the following formula:

$$\text{Sharpe Ratio} = \sigma_p / (R_p - R_f)$$

The Sharpe ratio of the strategy return is plotted in Graph 6. The Sharpe ratio for the momentum strategy is 0.47 while that of the market return is 0.82. The Sharpe ratio is the risk adjusted return of a portfolio, in this scenario to our surprise we see that the Sharpe ratio of the momentum strategy is less than that of the market.

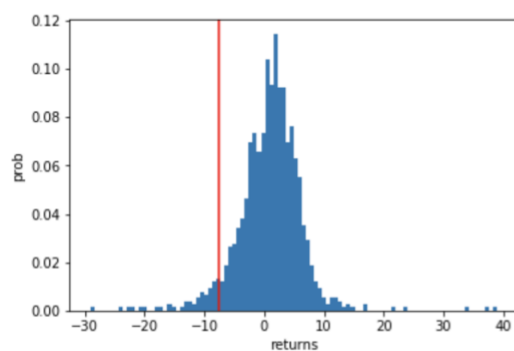
### Graph 8: 95% VAR - Momentum Strategy

```
var_95percent_st = np.percentile(port['strategy_ret'], 5)
var_95percent_st_sorted = sorted(port['strategy_ret'])
plt.hist(var_95percent_st_sorted, density = True, bins = 100)
plt.axvline(x=var_95percent_st, color='r', linestyle='-', label="var_95percent_st".format(var_95percent_st))
plt.xlabel('returns')
plt.ylabel('prob')
plt.show()
```



### Graph 9: 95% VAR - Market return

```
var_95percent_m = np.percentile(portfolios2['Mkt'], 5)
var_95percent_m_sorted = sorted(portfolios2['Mkt'])
plt.hist(var_95percent_m_sorted, density = True, bins = 100)
plt.axvline(x=var_95percent_m, color='r', linestyle='-', label="var_95percent_m".format(var_95percent_m))
plt.xlabel('returns')
plt.ylabel('prob')
plt.show()
```



The graph shows 95% VAR of our momentum strategy is -12.18%, which means we have 95% confidence that this strategy will not lose more than 12.18%, whereas the graph for the market return shows that there is 95% confidence that it won't lose more than 7.50%

**Graph 10: Regression Results**

OLS Regression Results						
Dep. Variable:	strategy_ret		R-squared:	0.237		
Model:	OLS		Adj. R-squared:	0.234		
Method:	Least Squares		F-statistic:	113.5		
Date:	Tue, 21 Apr 2020		Prob (F-statistic):	4.90e-64		
Time:	18:01:29		Log-Likelihood:	-3762.9		
No. Observations:	1103		AIC:	7534.		
Df Residuals:	1099		BIC:	7554.		
Df Model:	3					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
Intercept	0.9864	0.224	4.412	0.000	0.548	1.425
Mkt_RF	-0.2080	0.045	-4.655	0.000	-0.296	-0.120
SMB	-0.5692	0.073	-7.766	0.000	-0.713	-0.425
HML	-0.8121	0.065	-12.435	0.000	-0.940	-0.684
Omnibus:	684.142	Durbin-Watson:	2.073			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	17521.581			
Skew:	-2.394	Prob(JB):	0.00			
Kurtosis:	21.929	Cond. No.	5.70			

Fama and French Three-Factor Model is an extension model of Capital Asset Pricing Model. Here we try to create a regression model based on its formula

$$R_{it}-R_{ft}=\alpha_{it}+\beta_1(R_{Mt}-R_{ft})+\beta_2SMB_t+\beta_3HML_t+\epsilon$$

We use sm.ols function to create regression. From the regression results (Graph 10) we can find each independent variable has a p-value smaller than 0.001, so these independent variables show statistically significant relationship with the dependent variable. R square is equal to 0.237, which means around 23.7% stocks can be explained by this model. Then we do a Shapiro-Wilk test for normality test. Due to a p-value that is smaller than 0.05, null hypothesis of normality is rejected. Therefore, we can conclude that the strategy returns are not normally distributed.

## Conclusion

From the above analysis we observe that the standard deviation of the momentum strategy is 8.40% and on the market portfolio it's 5.34%, hence, the momentum strategy holds higher risks. Besides, the mean sharp ratio of the market is calculated as 0.47, which is smaller than the sharp ratio of the momentum strategy which is 0.82. We can also find that the selected strategy has 95% confidence that it will not lose more than 12.18%, while this figure of market return is 7.50%, which means this strategy is riskier. Thus, to some extent, we conclude that the momentum strategy is not a rational strategy.

