ISSN 2229-5518

β (Beta) Neutral Long-Short Equity Strategy using Ranking.

Ayan Mukherjee
B.Tech CSE, KIIT University Bhubneswar, Odisha,India
ayan721301@gmail.com

Abstract— The paper devises a beta neutral strategy for building a portfolio that is not exposed to market risk by implementing long short equity strategy using ranking for a selected basket of stocks.The portfolio so created has a weighted beta value of 0 indicating no correlation with the capital markets thus ensuring its risk free nature.

Index Terms— Beta Neutral, Long-Short, Equity, Portfolio,Morden Portfolio Theory, Securities,Strategy

1.Introduction

According to Modern Portfolio Theory (MPT), the capital markets and securities exhibit correlation and securities with positive correlation exhibit beta value of 1 and securities that are negatively correlation exhibit a beta value of -1 and a beta value of 0 is assigned to a security that has no correlation with the market.

Beta neutral portfolios consist of securities which are have a weighted average beta value 0, thus the portfolio has no market exposure. Strategies of these kind are implemented by hedge funds so as to generate profit without being exposed to market risk.

The paper deals with a β Neutral Long-Short Equity Strategy using Ranking wherein trades are entered both in long and short positions simultaneously in the market. The strategy has a weighted average beta of 0. In this strategy a basket of securities is selected and then the securities are ranked to identify which securities are relatively cheap(under valued) and expensive(over valued). It then goes long (buys) for top *n* equities/securities based on the ranking, and short (sells) the bottom *n* for equal amounts of money.

2. Strategy and Mathematical Analysis

Trades are entered both in long and short positions simultaneously in the market. The strategy has a weighted average beta of 0.

A basket of securities is initially selected where in the securities will then be ranked to identify which securities are relatively cheap (under valued) and expensive (overvalued). It will then go long (buys) the top n equities/securities based on the ranking, and short (sells) the bottom n for equal amounts of money.

Total value of long position = Total value of short position

The strategy is also statistically robust since by ranking securities and entering multiple positions, one is making many bets on the ranking model rather than just a few risky bets.

A ranking scheme is any model that can assign each security a number based on how they are expected to perform. Examples could be value factors, technical indicators, pricing models, or a combination of all of the above. For example, a momentum indicator can be used to give a ranking to a basket of trend following securities: securities with highest momentum are expected to continue to do well and get the highest ranks; securities with lowest momentum will perform the worst and get lowest ranks.

Upon determining the ranking scheme, profits are likely to be generated from the same. This is done by investing an equal amount of money into buying securities at the top of the ranking, and selling securities at the bottom. This ensures that the strategy will make money proportionally to the quality of the ranking only, and will be market neutral.

Let's assume m equities/securities have been ranked, and n dollars are to be invested, and total holding position is of 2p positions (where m > 2p). If the security at rank 1 is expected to perform the worst and security at rank m is expected to perform the best,then:

One takes the securities in position 1,…p in the ranking, sell n/2p dollars worth of each security

For each security in position m−,…m in the ranking, buy n/2p dollars worth of each security

Friction Because of Prices will exist as security prices will not always divide n/2p evenly, and securities must be bought in integer amounts, there will be some imprecision and the algorithm should get as close as it can to this number.

For a strategy running with n=100000 and p=500, we see that n/2p=100000/1000 =100

This will cause big problems for securities with prices > 100 since fractional securities cannot be purchased or sold.This is alleviated by trading fewer equities i.e by flooring(floor function) on the capital invested in trading or increasing the capital I.e using leverage.

Strategy testing on hypothetical data set

A random factor is used to rank the random generated stock/securities names. Let's assume our future returns are dependent on these factor values.The strategy was tested on hypothetical security data generated using python:

```
import numpy as np
import statsmodels.api as sm
import scipy.stats as stats import scipy
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
## PROBLEM SETUP ##
# Generate securities and a random factor val-
ue for them
security_names = ['security ' + str(x) for x
in range(10000)] current_factor_values =
np.random.normal(0, 1, 10000)
# Generate future returns for these are de-
pendent on our factor values
future_returns = current_factor_values +
np.random.normal(0, 1, 10000)
# Put both the factor values and returns into
one dataframe
data = pd.DataFrame(index = security_names,
columns=['Factor Value','Returns'])
data['Factor Value'] = current_factor_values
data['Returns'] = future_returns
# Take a look data.head(10)
```

	Factor Value	Returns
stock 0	0.931224	0.178167
stock 1	1.519926	0.554476
stock 2	-0.832380	-0.770446
stock 3	-0.383235	0.693043
stock 4	-0.457871	-0.232366
stock 5	0.438764	2.064121
stock 6	-0.603939	-1.950858
stock 7	0.739928	1.199309
stock 8	-1.602303	-2.121320
stock 9	-0.003074	-0.579422

The equities are ranked according the their corresponding factor values and then trading positions are taken as per the strategy.The hypothetical security data so generated was then ranked using the following python code:

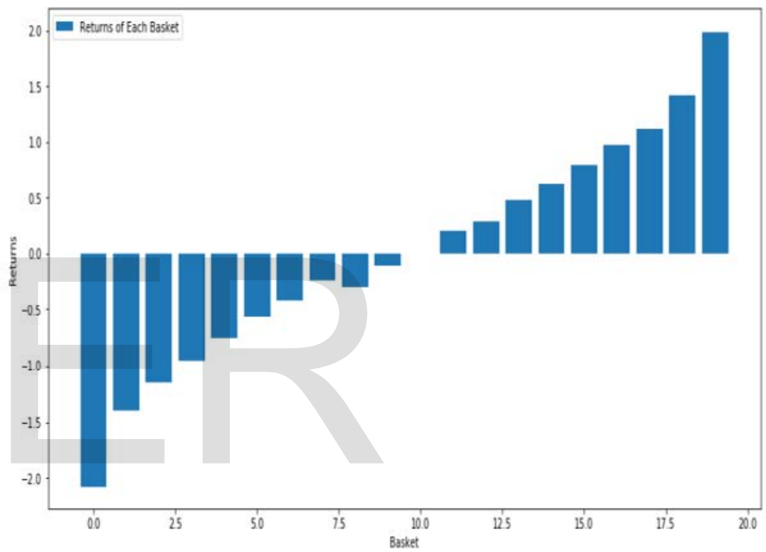
```
# Rank securities
ranked_data = data.sort_values('Factor Value')
# Compute the returns of each basket with a
basket size 500, so total (10000/500) baskets
number_of_baskets = int(10000/500) bas-
ket_returns = np.zeros(number_of_baskets)
for i in range(number_of_baskets):
start = i * 500 end = i * 500 + 500 bas-
ket_returns[i] =
ranked_data[start:end]['Returns'].mean()
# Plot the returns of each basket
plt.figure(figsize=(15,7))
plt.bar(range(number_of_baskets), bas-
ket_returns)
plt.ylabel('Returns') plt.xlabel('Basket')
plt.legend(['Returns of Each Basket'])
plt.show()
```

3. Results

The results so obtained from investment as per the strategy devised was computed using the following python code:

```
basket_returns[number_of_baskets-1] - bas-
ket_returns[0]
```

The returns of the this strategy upon testing with hypothetical data was found out to be **4.172**



.

4.References

[1] A. Agarwal, E. Hazan, S. Kale, and R. E. Schapire. Algorithms for portfolio management based on the newton method. In ICML, pages 9–16, 2006.
[2] A. Blum and A. Kalai. Universal portfolios with and without transaction costs. Machine Learning, 35(3): 193–205, 1999.
[3] A. Borodin, R. El-Yaniv, and V. Gogan. On the competitive theory and practice of portfolio selection (extended abstract). In LATIN, pages 173–196, 2000.
[4] A. Borodin, R. El-Yaniv, and V. Gogan. Can we learn to beat the best stock. Journal of Artificial Intelligence Research, 21:579–594, 2004.
[5] S. Boyd and L. Vandenberghe. Convex Optimization. Cambridge University Press, 2004.
[6] L. Breiman. Optimal gambling systems for favorable games. Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability, 1:65–78, 1961.
[7] T. M. Cover. Universal portfolios. Mathematical Finance, 1(1):1–29, 1991.

Leung, T. and Li, X. (2016). Optimal Mean Reversion Trading: Mathematical Analysis and Practical Applications. World Scientific Publishing Co.
[8] Leung, T. and Li, X. (2015). Optimal Mean Reversion Trading with Transaction Costs and Stop-loss Exit. International Journal of Theoretical & Applied Finance, vol. 18, no. 03, p.1550020.
[9] Leung, T., X. Li and Z. Wang (2015). Optimal Multiple Trading Times Under the Exponential OU Model with Transaction Costs. Stochastic Models, vol. 31, issue 4, p.554-587.
[10] Avellaneda, M., & Lee, J. H. (2010). Statistical Arbitrage in the US Equities Market. Quantitative Finance, 10(7), p.761-782.
[11] Gatev, E., Goetzmann, W. N., & Rouwenhorst, K. G. (2006). Pairs Trading: Performance of a Relativevalue Arbitrage Rule. Review of Financial Studies,19(3), p.797-827.