Black-Litterman Model:
Machine Learning Methods to form investor views

MF796 Project

Group members:
Xuan Wang (U47863561)

Yuzhe Zhang (U14627415)

Sihai Feng (U44893255)

Yuyang Zhao (U42414906)

Lingyi Zeng (U77149480)

Yifan Nie (U16669110)

Questrom School of Business

Boston University

May 3, 2019

Contents

1 Background

1.1 Introduction

1.1.1 Black-Litterman Portfolios:

The Black-Litterman model is a one-step further from mean-variance optimization, it uses a Bayesian approach to combine the subjective views of an investor regarding the expected returns of one or more assets with the market equilibrium vector of expected returns.

It allows unique views and overcomes extreme value issue by choosing a neutral reference point.

1.1.2 Machine Learning Views:

Instead of specified views, we apply multiple Machine Learning methods to find sets of optimized views and use them as inputs.

Using these methods, we are able to access a group of stable "views".

Machine Learning Methods we used:

- Bayesian
- Adaptive Boosting
- Decision Tree
- KNN
- Logistic Regression
- Random Forest
- Support Vector Machine

1.2 Research Goals

- Use machine learning methods to form investor view in Black-Litterman strategies and see whether it outperforms the result from mean variance optimization.
- Study the economic context of when & under what scenarios will Black-Litterman Portfolio strategy outperform or underperform.
- Find the best investor view generating model, comparing different machine learning methods.

2 Data

2.1 Data Source

Source: 6 ETF from Yahoo finance market weight from Bloomberg

- EMB - ISHARES TR/JPMORGAN USD emerging market
- ETFGLD - SPDR Gold Shares
- ETFTLT - iShares 20+ Year Treasury Bond
- ETFIGIB - iShares Intermediate-Term Corporate Bond
- ETFIYR - iShares U.S. Real Estate
- ETFIJH - iShares Core S&P Mid Cap ETF

Timeline: 2008– 2019 weekly data

2.2 Data Analysis

The ETFs we selected can be divided in to two groups, bonds and stocks. The combination of ETFs guarantees we have a insight on overall market behavior on different underlying assets.

We used data in a weekly step. To avoid abnormal inputs, we cut down values with a thresh hold and cleaned up empty periods. Yahoo was a good enough source for the pricing curve, and we further used Bloomberg as the source for market weight – which helps the shares to be consistent. Fortunately, we did not have large gaps and abnormal that needs to do interpolation.

Table 1 Data Correlation Table

	EMB	GLD	TLT	IYR	IGIB	IJH
EMB	1	0.21	0.106	0.466	0.491	0.442
GLD	0.21	1	0.108	0.216	0.226	0.188
TLT	0.106	0.108	1	-0.149	0.53	-0.386
IYR	0.466	0.216	-0.149	1	0.087	0.764
IGIB	0.491	0.226	0.53	0.087	1	-0.028
IJH	0.442	0.188	-0.386	0.764	-0.028	1

3 Methodology

The Black-Litterman model consists of two parts: the equilibrium returns and the subjective views. In our research, we implemented different methods to model these two parts. The equilibrium returns are derived from the mean-variance framework, and the subjective views are obtained from machine learning methods. Finally, we use the Bayesian methods to combine those two parts and get the posteriors' mean and variance.

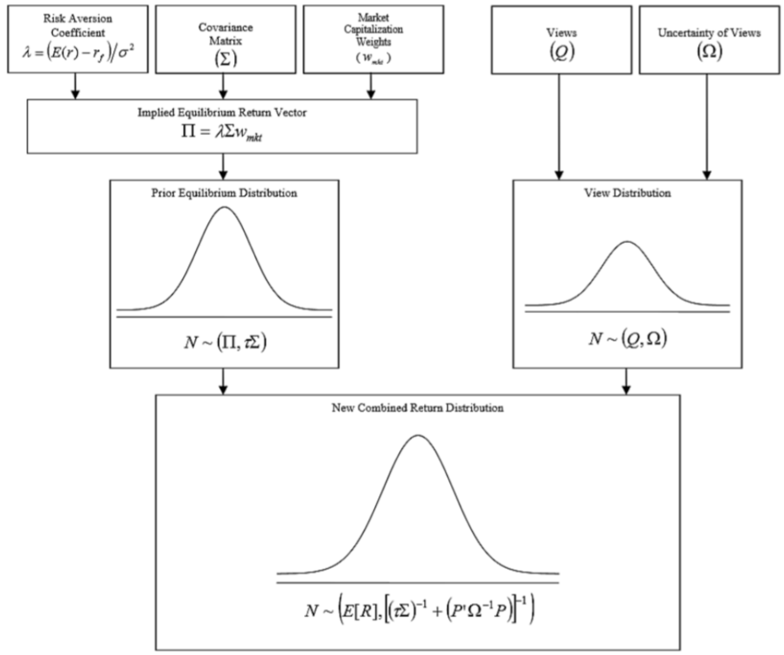


Figure 1 Black-Litterman Framework

3.1 Equilibrium Return

When CAPM holds, the equation of the equilibrium return as the following:

$$\pi_t = \delta \Sigma_t w_{t-1}$$

π_t is the N×1 mean vector, δ is the risk aversion coefficient, Σ_t is the N×N covariance matrix estimates of returns given information available at time t-1, and w_{t-1} is the market capitalization weights at time t-1.

3.1.1 Risk Aversion Coefficient $\boldsymbol{\delta}$

The risk aversion coefficient is based on the mean-variance formula. We used the market returns to get that, which is shown as the following equation:

$$\delta = \frac{E(r) - r_f}{\sigma^2}$$

σ refers to the standard deviation of market returns. Since the risk-free rate is relatively small compared with the market returns, we just ignore it in our calculation.

The result of δ is 0.53.

3.1.2 Covariance Matrix $\boldsymbol{\Sigma_t}$

We used four years rolling history returns to calculate the covariance matrix. The approach to forecast the covariance matrix is the EWMA model, shown as the following recursive form:

$$\Sigma_t = \lambda\Sigma_{t-1} + (1-\lambda)(r_{t-1} - \overline{r_{t-1}})(r_{t-1} - \overline{r_{t-1}})'$$

where λ refers to the decay rate measuring the weights of past information. We set $\lambda = 0.94$ based on the RiskMetrics (J.P.Morgan, 1994).

3.1.3 The Market Capitalization Weights w_{t-1}

The market capitalization weights is the weighted average of each ETFs' market capitalizations with data obtained from Bloomberg Terminal.

3.2 Feature Selection

As we all known that the features which is used as input data for our model is very important, the more representative the features are, the more accurate the prediction is. However, there is no need to include all the features in our model, since too many features would not improve the model, but contradictorily, it would introduce a lot of noise as well as overfitting. Therefore, the key to improve the accuracy of our prediction is to select the most proper and important features.

3.2.1 Information Set

Firstly, we classify 7 information sets. These sets include 3 types of time series data and 4 technical indicators. All the information sets are listed below:

Table 2 Information Sets

Information set	Explicit indicators
Time series data for return	Previous n week returns and their j lags for 6 ETFs along with S&P 500 (SPY), VIX and 10 Year treasury note yield index (IRX)
Time series data for excess return	Previous n week excess returns and their j lags for 6 ETFs along with S&P 500 (SPY), VIX and 10 Year treasury note yield index (IRX)
Time series data for volume change	Rate of change of volume in past n weeks and their j lags for 6 ETFs.
Momentum related technical indicators	CCI, RSI, ROC, Slow/Fast Stochastic Oscillator, William Indicator and Aroon indicator for 6 ETFs
Trend related technical indicators	SMA, EMA, MACD, ADX and T3 for 6 ETFs
Volume based technical indicators	OBV, Money Flow Index, Chaikin Oscillator up for 6 ETFs

Volatility based indicators	Bollinger Bands and Average True Range(ATR) for 6 ETFs

$n \in \{1,2,3,4\}, j \in \{0,1,2,3,4,5,6\}$

After collecting all the features, we obtain a large amount of features which is more than 100. However, there are too many features for our model, thus, in the next step, we need to find a way to select more representative features for our models.

3.2.2 Gini Index Searching Method

After searching a lot of methods, like Weiss/Indurkhya 'independent features' significance testing method, we decide to select our features based on Gini index. Gini index is an impurity (or purity) measure used in building decision tree in CART. The lower the Gini index is, the more accuracy the feature contributes. To use Gini index, we apply the Random Forest with CART decision tree using all the features and get the feature importance which is computed by Gini index. Top 10 features in our feature important list are chosen as our final features. The list below shows the chosen features for each ETF. Data(i,j) means the previous i week data (like return, excess return and rate of change volume) with lag j, 'Slowk' means the Slow Stochastic Oscillator, 'BL' means the Bollinger Lowerband, and 'WI' means William Indicator:

Table 3 The Chosen Features for Each ETF

features	EMB	GLD	TLT	IGIB	IYR	IJH
1	VIX	IRX	Return(1,5)	$\text{Return}_{SPY}(1,0)$	Return(1,3)	IRX
2	Return(1,0)	VIX	$\text{Return}_{SPY}(1,5)$	Return(1,5)	Return(2,3)	TNX
3	Return(1,2)	Return(1,1)	SPY (2,1)	Volume(2,0)	Volume(2,3)	VIX
4	Volume(2,4)	Volume(1,4)	Volume(2,2)	SPY (2,0)	SPY (2,3)	Volume(1,3)
5	Volume(2,5)	Return(2,1)	Volume(3,2)	Volume(2,5)	Volume(3,0)	Volume(3,0)
6	$\text{Return}_{SPY}(3,0)$	EMA	Volume (3,4)	Volume(3,0)	SPY (3,0)	$\text{Return}_{SPY}(3,3)$
7	Volume(3,5)	T3	Return(3,5)	Volume(3,4)	$\text{Return}_{SPY}(4,0)$	Volume(4,1)
8	$\text{Return}_{SPY}(3,0)$	OBV	Return(4,2)	Return(4,3)	Volume(4,2)	Volume(4,3)
9	Volume(4,3)	BL	Volume(4,2)	OBV	Volume(4,3)	WI
10	$\text{Return}_{SPY}(4,2)$	ATR	T3	ATR	Slowk	T3

From the table, we can see that for stock ETF (EMB, GLD, IYR and IJH), time series data is highly important, since financial data always has autocorrelation. VIX, IRX and SPY is also important, since stocks would be significantly affected by market index. What's more, the volatility-based indicators (ATR s.t) and trend indicators (EMA s.t) affect the return of the next week, but they are less important than time series data. For bond ETF (TLT and IGIB), the features are similar including time series data of return, volume and market index, volatility-based indicators and trend indicators.

3.3 Building machine learning model

Table 4 Candidate Machine Learning Methods

Classifier	Pros	Cons
Bayesian	1.Feasible in small sample 2.Suitable for pretrain	Need assumption for sample independence
Adaptive boosting	1.Fast, simple, Flexible and Generally stable 2.Does not need prior knowledge	1.Need good weak classifier 2.Slow training speed
KNN	1.Easy to include new samples, no repeat training 2. No assumption, parameters	1.Need to handle missing values 2.Sensitive to class-outliers
Logistic Regression	1.Computationally cheap 2.No need for tuning, easy to regularize	1.Linear decision surface 2.High reliance on a proper presentation of data
Random Forest	1.Corrects decision tree's habit of overfitting 2.Can be used for both classification and regression tasks	1.Large number of trees will be computationally costly 2.It is a predictive tool, not good for description
Support Vector Machine	1.Suitable for small sample 2.Fits no-linear surface	1.Sensitive to missing data and class-outlier 2.Results depend on the choice of kernel

Divide return into two classification problem,

$$Y_1 = \begin{cases} -1, \text{ if sign of excess return of the next week's is negative} \\ +1, \text{ if sign of excess return of the next week's is positive} \end{cases}$$

$$Y_2 = \begin{cases} 1, \text{ if } z_t = \dfrac{r_t - \overline{r}_{t,3}}{\sigma_{t,3}} \le 1 \\ 2, \text{ if } z_t = \dfrac{r_t - \overline{r}_{t,3}}{\sigma_{t,3}} > 1 \end{cases}$$

We use the above two responses to identify the view of next week's excess return. i.e.,

$$Y = \begin{cases} -2, \text{ very Bearish} \\ -1, \quad \text{Bearish} \\ 1, \quad \text{Bullish} \\ 2, \text{ Very Bearish} \end{cases} = Y_1 Y_2$$

We use machine learning model with features selected from 4.2, and labels Y1 and Y2, we can get rolling subjective view about next week's return.

Then

$$Q_t(k) = (P_t\pi_t)(k) + \eta_k\sqrt{(P_t\Sigma P_t')(k,k)}$$

Here since our views are specific for each single ETF, $P = I_t$

By using Bayesian method, we can get posterior mean:

$$E[R|q] = \pi + \sum P^T * (P\sum P^T + \Omega)^{-1}(q - P\pi)$$

$$Var[R|q] = \sum - \sum P^T * (P\sum P^T + \Omega)^{-1}(P\sum)$$

And the final optimal weight is:

$$\omega = \xi\tilde{C}^{-1}\tilde{R}$$

4 Result comparison

4.1 Construct benchmark portfolio

We build two benchmark portfolios, one is when view equals to zero, which means that the optimal weight is just the capitalization weighted portfolio, we called it market portfolio. Another one is S&P500, which represents the general performance of the who equity market.

4.2 Compare with benchmark portfolio

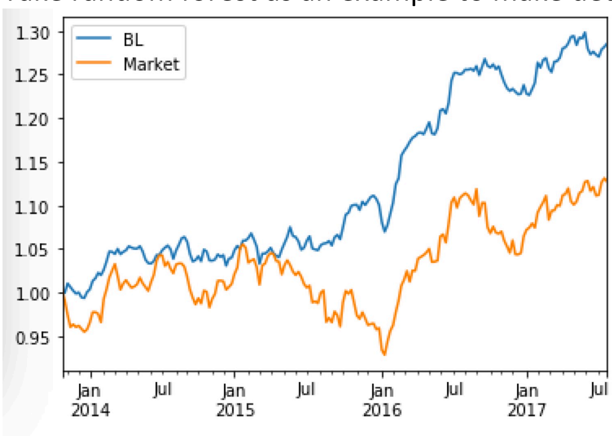
The overall performance compare with benchmark is:

Table 5 Overall performance

	Annualized Return	Annualized Volatility	Sharpe Ratio	Max Drawdown	VaR
Random Forest	0.0769	0.0523	1.4699	-0.053	-0.0089
Bayesian	0.0408	0.048	0.8491	-0.0756	-0.038
SVM	0.0615	0.0709	0.8679	-0.084	-0.0548
Logistic	0.0621	0.0457	1.3576	-0.04	-0.0129
KNN	0.0384	0.0484	0.7935	-0.0456	-0.041
Adaboosting	0.032	0.0464	0.6891	-0.0469	-0.0442
market	0.0344	0.0758	0.4536	-0.1201	-0.09
SPYClose	0.0985	0.1128	0.8738	-0.1297	-0.0865

As we can see from the table, all six machine learning models have higher Sharpe ratio than market weighted portfolio when subjective views equal to zero. Two of them have higher Sharpe ratio than SPY, 2 two of them have similar Sharpe ratio with SPY. All of them all relative low max drawdown and Value at Risk than market weighted portfolio and SPY.

Take random forest as an example to make detailed analysis:



	BL	Market	SPY
Annualized Return	0.07	0.0344	0.0985
Annualized Volatility	0.0496	0.0758	0.1128
Sharpe Ratio	1.4111	0.4536	0.8738
Beta	0.1224	0.3326	1.0051
Max Drawdown	-0.0469	-0.1201	-0.1297
VaR	-0.0114	-0.09	-0.0865

Figure 2 Compare With Marked Weighted Portfolio

As we can see from the picture, the market vibrated, Black-Litterman model just a little better than the market weighted portfolio. However, when the market went down, the Black-Litterman portfolio actually performed pretty well and outperform the benchmark. We thought this might show machine learning prediction actually have higher accuracy to predict downside risk.

4.3 Compare with direct investing by machine learning model

```
Back Test is initialized.
Running Back Test:
                          BL  Direct_view  Market     SPY
Annualized Return     0.0621       0.0552  0.0344  0.0985
Annualized Volatility 0.0457       0.0467  0.0758  0.1128
Sharpe Ratio          1.3576       1.1834  0.4536  0.8738
Beta                  0.1876       0.1646  0.3326  1.0051
Max Drawdown         -0.0400      -0.0481 -0.1201 -0.1297
VaR                  -0.0129      -0.0213 -0.0900 -0.0865
                   EMB       GLD       TLT       IYR       IG]
IJH
sign_predict   0.527919  0.578680  0.487310  0.548223  0.53299
0.512690
range_predict  0.568528  0.512690  0.568528  0.558376  0.53807
0.553299
total_predict  0.304569  0.279188  0.274112  0.279188  0.29441
0.284264
```

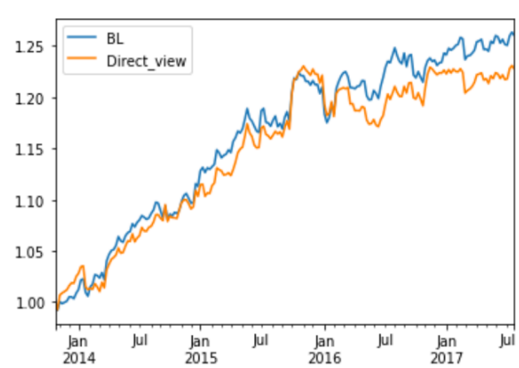


Figure 3 Compare With Direct Inversing By Machine Learning

As shown in the picture, compared with direct investing by machine learning, Black-Litterman portfolio actually have higher Sharpe ratio and lower max drawdown. Therefore, we think Black-Litterman framework which considered passive view and combine subjective view by Bayesian really improve the overall performance.

4.4 Out-of-sample Performance

Table 6 Out-of-Sample Performance

	Annualized Return	Annualized Volatility	Sharpe Ratio	Max Drawdown	VaR
Random Forest	0.0163	0.0433	0.3763	-0.0541	-0.0545
Bayesian	0.0468	0.0430	1.0879	-0.0439	-0.0236
SVM	0.0401	0.0651	0.6161	-0.0868	-0.0664
Logistic	0.0392	0.0545	0.7189	-0.0552	-0.0500
KNN	0.0220	0.0382	0.5759	-0.0557	-0.0404
Adaboosting	0.0199	0.0382	0.5191	-0.0314	-0.0427
market	0.0388	0.0779	0.4974	-0.1022	-0.0887
SPYClose	0.1014	0.1348	0.7527	-0.1561	-0.1190

As we can see from table, Bayesian, Support Vector Machine, Logistic model have more consistent performance on the out-of-sample test. This shows that under weekly frequency when rolling data is just nearly 200, simple machine learning model may be more robust than complicated ones.

5 Conclusion

5.1 Implications

- Well-built machine leaning model forming subjective views can improve the performance of Black-Litterman strategy. Its main contribution is when market goes down since machine learning model tend to have higher prediction accuracy for downside risk
- For stock ETF, features like time series data, market indicators like VIX, SPY, interest rate tend to have higher importance. For bond ETF, except for lag returns and market index, volatility-based indicators and trend indicators are also very important.
- Top performance machine learning model is Bayesian, Support Vector Machine, Logistic. Their average accuracy to predict signs of return is 55%. Their average accuracy to predict range of return is 58%.

5.2 Limitations

Due to time limit, we didn't try model ensemble to combine several outperform machine learning model together to get more accurate views. It still remains some space to make our subjective views more accurate.

Right now, we just simplify the prediction return into two classification problem since we thought direction predict is more robust than regression predict. But it is worth trying to use regression technique in machine learning to predict returns as subjective view and incorporate it into Black-Litterman model.

6 Model Validation

Table 7 Unit Test for Rolling Window

N	150	175	200	300
Annualized Return	0.0164	0.0414	0.0601	0.0708
Annualized Volatility	0.0685	0.0644	0.0599	0.0715
Sharpe Ratio	0.2397	0.6432	1.0032	0.9895
Beta	0.2916	0.2474	0.2903	0.4266
Max Drawdown	-0.099	-0.0731	-0.0639	-0.0656
VaR	-0.096	-0.0643	-0.0382	-0.0463

We do the unit test for the rolling window of training sets. The result are given above, from which we can see that when rolling window is short, which means training sets are small, the sharpe ratio is small. When the rolling window is reasonably long, more than 200 weeks as in the table, then the sharpe ratio is stable.

7 Reference

[1] Donthireddy, Pavan " Black-Litterman Portfolios with Machine Learning derived Views " (2018).*Web.* https://www.researchgate.net/publication/326489143_BlackLitterman_Portfolios_with_Machine_Learning_derived_Views

[2] Black, F., and R. Litterman. 1992. Global Portfolio Optimization. Financial Analysts Journal 48 (5): 28-43.

[3] Fischer Black and Robert Litterman. Global portfolio optimization. Financial Analysts Journal, pages 28–43, September-October 1992.

[4] Fisher Black. Universal hedging: Optimizing currency risk and reward in international equity portfolios. Financial Analysts Journal, pages 16–22, July-August 1989. missing.