Bayesian Approach in Finance: Understanding the Black-Litterman Model

Gian Martin Keller

MSc in Finance
Università della Svizzera Italiana
Lugano, Switzerland
09/01/2024

Abstract

This report presents an accessible exploration of applying Bayesian statistics in finance, focusing on the Black-Litterman model. It illustrates how Bayesian principles, notably the concept that the posterior is proportional to the likelihood times the prior, enhance financial modeling and decision-making. The report compares the advantages of Bayesian methods to frequentist approaches, emphasizing the former's flexibility and ability to integrate subjective insights. Additionally, it discusses the application of Monte Carlo simulations within the Bayesian framework for risk assessment. Aimed at the average reader, this report highlights the significance and utility of Bayesian techniques in modern financial analysis and portfolio management.

Contents

1 Introduction

Bayesian statistics is a branch of statistics based on Bayes' Theorem, a fundamental principle that describes how to update the probabilities of hypotheses when given more evidence. The bayesian approach contrasts with traditional (frequentist) statistics by incorporating prior knowledge or beliefs, which are then updated with new data. Bayesian methods are invaluable in areas like machine learning, medical research, environmental science, and many other disciplines where making informed decisions under uncertainty is crucial.
This report aims to explain the Black-Litterman model, a sophisticated application of Bayesian statistics in finance. The focus is on making this financial model comprehensible to the average reader, highlighting how Bayesian principles are applied to improve investment decision-making and risk assessment in the financial world.

2 Bayesian Statistics Basics

Bayesian statistics is a framework for statistical inference in which probabilities are assigned to hypotheses or parameters. Its foundational principles come from Bayes' Theorem, which mathematically describes how to update the probability of a hypothesis as more evidence or information becomes available. The theorem can be expressed as:

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

Where:

- $P(A|B)$ is the posterior probability: the probability of hypothesis A given the evidence B.
- $P(B|A)$ is the likelihood: the probability of observing the evidence B given that A is true.
- $P(A)$ is the prior probability: the initial belief in A.
- $P(B)$ is the marginal probability of the evidence (across all hypotheses).

The essence of Bayesian statistics lies in the prior $P(A)$ and how it's updated with new information (through $P(B|A)$) to produce the posterior $P(A|B)$. This posterior distribution embodies everything known about the parameters after considering both the prior information and the new data. In contrast, frequentist statistics rely only on data from current experiments or studies, interpreting probability as the long-run frequency of events and often treating parameters as having fixed, true values that do not vary. However, these true values are unknown.

The following analogy helps illustrate how Bayesian statistics work:
Imagine predicting rain. Consider your prior belief based on experience, such as believing there's

a 30% chance of rain today. Now, you see dark clouds (new evidence), so you update your belief. Bayesian statistics is like adjusting your rain forecast based on both your prior experience and the current sight of dark clouds.

3 Introduction to the Black-Litterman Model

The Black-Litterman model represents a Bayesian approach to portfolio optimization that combines market equilibrium, represented by the Capital Asset Pricing Model (CAPM), with investors' unique views. In this part, the focus will be in defining the components of the Black Litterman model.

3.1 The Prior

The Black-Litterman model starts with what's called "equilibrium returns." These are the returns needed to balance the market. To find these returns, a method called reverse optimization is used. This method takes known information and uses a special formula (1) to figure out the equilibrium returns. This is our Prior in the context of Bayesian statistics.

$$\Pi = \lambda \Sigma w_{mkt} \tag{1}$$

for N being the number of assets, we have

- Π is the Implied Excess Equilibrium Return Vector ($N \times 1$ column vector),
- λ is the risk aversion coefficient,
- Σ is the covariance matrix of excess returns ($N \times N$ matrix), and
- w_{mkt} is the market capitalization weight ($N \times 1$ column vector) of the assets.

3.1.1 CAPM for the Implied Returns

The Implied Excess Equilibrium Return Vector (Π) corresponds to the CAPM returns vector, if the CAPM returns vector is computed based on implied betas. Implied betas are computed using the CAPM formula, with the returns of the market being the returns of a portfolio weighted by market capitalization.

$$E(R_A) = R_f + \beta_{implied,A} \times [E(R_{MktCap}) - R_f] \tag{2}$$

where:

- $E(R_A)$ is the expected return on the asset,

- R_f is the risk-free rate of return,
- $\beta_{implied,A}$ is the implied beta of the asset,
- $E(R_{MktCap})$ is the expected return of the market capitalization-weighted portfolio.

3.1.2 Implied Betas

The previous formula can be seen as a linear regression, which leads us to the following formula to compute the implied betas:

$$\beta_{implied} = \frac{\Sigma w_{mkt}}{w_{mkt}^T \Sigma w_{mkt}} = \frac{\Sigma w_{mkt}}{\sigma^2} \tag{3}$$

where

- $\beta_{implied}$ is the vector of implied betas $(N, 1)$;
- Σ is the covariance matrix of excess returns (N, N);
- w_{mkt} is the market capitalization weights $(N, 1)$; and,
- $\sigma^2 = w_{mkt}^T \Sigma w_{mkt}$ is the variance of the market capitalization-weighted portfolio excess returns.

3.1.3 Recommended Weights

From $\Pi = \lambda \Sigma w_{mkt}$, we can substitute Π (Implied Return Vector) for μ, which represents any vector of excess returns. We get the weights:

$$w = (\lambda \Sigma)^{-1} \mu \tag{4}$$

Which is the solution to the unconstrained maximization problem:

$$\max_{w} w' \mu - \frac{\lambda}{2} w' \Sigma w \tag{5}$$

if $\mu = \Pi$, then $w = w_{mkt}$.

The risk-aversion coefficient (λ) characterizes the expected risk-return tradeoff:

$$\lambda = \frac{E(R_{MktCap}) - R_f}{\sigma^2} \tag{6}$$

The weights formula (4) will be a useful tool for getting the posterior portfolio weights.

3.2 The Posterior

Now that we have our prior, we will start computing the posterior, by first introducing the Black-Litterman model formula.

The formula for the new Combined Return Vector $E[R]$ is:

$$E[R] = \left[(\tau\Sigma)^{-1} + P'\Omega^{-1}P\right]^{-1} \left[(\tau\Sigma)^{-1}\Pi + P'\Omega^{-1}Q\right] \tag{7}$$

where

- $E[R]$ is the new (posterior) Combined Return Vector ($N \times 1$);
- τ is a scalar;
- P is a matrix that identifies the assets involved in the views ($K \times N$;
- Ω is a diagonal covariance matrix of error terms from the views, representing the uncertainty in each view ($K \times K$);
- Q is the View Vector ($K \times 1$).

3.2.1 Investor Views

External information can be incorporated into the model in the form of views, which represents investors estimates (or opinions) on how an asset may perform in the future. These views, in the context of Bayesian statistics, are what we call the Likelihood. Views from an investor can be either absolute (e.g., Bitcoin will grow 30%), or relative (e.g., Ethereum will outperform Bitcoin by 10%), and it is not required to specify views on all assets. Views are modelled via 2 components, which are P and Q.

The first component that is going to be introduced is P, which is the matrix that identifies the assets involved in a view. There are different methods to incorporate the views using the matrix P. However, in this project we'll focus on the market capitalization weighting scheme. Using this method, the relative weight of each individual asset is proportional to the asset's market capitalization divided by the total market capitalization of either the outperforming or underperforming assets of that particular view. In the case where the view is absolute, the used weight in the matrix P is 1, because none of the other assets are involved. In the case of relative views, if only two assets are involved, then we assign 1 to the outperformer and -1 to the underperformer. If more than 2 assets are involved, the previously explained market capitalization weighting scheme is used. A detailed explanation on how to compute the components of P can be found in the Notes section.

3.2.2 Variance of the Views

The individual variance associated with each view provides insight about the level of confidence one might have in a particular view. There's a method that uses this information to compute the variances of the error terms (ω), which are the diagonal components of the matrix Ω. For this project a different method was used to compute Ω. However, even if it's not directly incorporated in the model, the variance of the views still provides an insight on how uncertain a view might be, given the covariance of the assets involved.

The variance of an individual view is $\sigma_{p_k}^2 = p_k' \Sigma p_k$, where p_k is a $1 \times N$ row vector from Matrix P that corresponds to the k^{th} view.

3.2.3 View Vector Q, the Likelihood

A view is composed of an estimate or opinion about the future performance of an asset. This estimate (or opinion) comes with an uncertainty or error. Thus, a view has the form $\mathbf{Q} + \boldsymbol{\varepsilon}$. The uncertainty of the views corresponds to an Error Term Vector $\boldsymbol{\varepsilon}$ with mean 0 and covariance Ω. The Error Term Vector ($\boldsymbol{\varepsilon}$) does not directly enter the Black-Litterman formula, it is the variance of each error term (ω) that enters into the formula.

It is easy to input the views, for example if the view says that bitcoin will increase 30%, then the first row of $\mathbf{Q}$ is 0.3, with unknown error term ε_1.

3.2.4 Uncertainty Matrix Ω

The Black-Litterman model represents a mix of the Prior (Π) and the Likelihood (Q), where the balance between them depends on the scalar (τ) and the uncertainty associated with the views (Ω). The higher the confidence towards the views(Q), the closer the posterior $E(R)$ will be to the views. Conversely, the lower the level of confidence towards the views, the closer the posterior $E(R)$ will be to the Prior (Π).

As previously mentioned, the Ω matrix is composed of the variance of the error term ω, which are based on the variances of the views $p_k \Sigma p_k^T$ multiplied by the scalar τ. However, there may be other sources of information, in addition to the previous source, that affect an investor's confidence in a view. Additional factors like credibility of the analyst, should be combined with the variance of the view to get the best possible estimates of the certainty on the views.

There are different methods to set the value of the scalar τ and Ω. For τ, assumptions about the value can be made. For Ω, the tilt method will be used in this project. A detailed explanation of the steps involved to compute Ω can be found in the Notes section.

Once all the inputs are calculated, the Posterior can be computed using formula (7), and then the

New Recommended weights can be retreived using formula (4). These weights correspond to the allocation of the assets in the portfolio that we want to build.

Now that the concepts that form the Black-Litterman model have been introduced, we will dive into an application of it.

4 Application of the Black-Litterman Model

To visualize in a clearer way how the Black-Litterman model can be used, we'll create a portfolio of cryptocurrencies. The chosen cryptocurrencies are the top 14 by market capitalization, the frequency of the used data is weekly and more precisely 170 average weekly returns and average weekly market capitalizations. All the results presented will be weekly, unless stated otherwise.

The first step after gathering the returns of the different assets, is to compute the covariance matrix, which represents how much the returns of two assets change together. Each element in the matrix quantifies the covariance between a pair of assets. If the returns on two assets tend to increase or decrease together, they have a positive covariance; if one tends to increase when the other decreases, they have a negative covariance. The diagonal elements of the covariance matrix are simply the variances of each asset, showing the individual variability of each asset's return. This matrix is crucial in portfolio optimization, as it helps in understanding the degree to which assets diversify the portfolio.

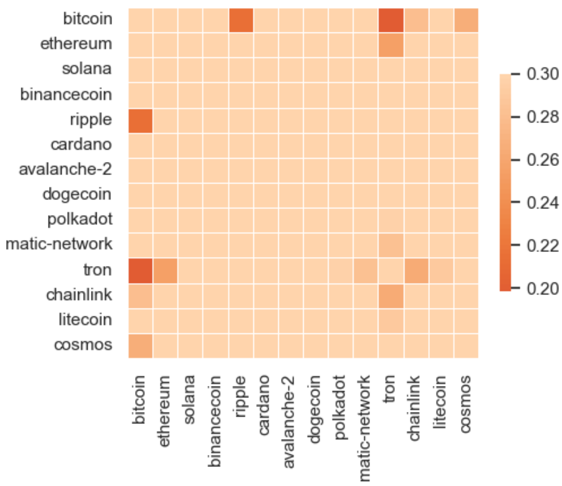

Figure 1: Covariance Matrix Σ (Annualized)

4.0.1 Computing the Prior

As mentioned in the part 3.1, to compute the Prior we first need to perform a linear regression where the variable we want to explain is the average excess returns of the asset A, and the explanatory variable is the average excess returns of the market capitalization-weighted portfolio. This procedure gives implied betas, and then we can get the implied return of each asset by using the CAPM formula (2) with the implied betas.

The following table shows the Implied Returns of each asset, in comparison with an Historical approach and another CAPM approach where the market is the MVDA10 crypto index.

	Historical mu	MVDA10 CAPM mu	Implied Return pi
bitcoin	1.14%	0.71%	1.29%
ethereum	1.57%	0.91%	1.54%
solana	3.48%	0.99%	1.82%
binancecoin	2.07%	0.84%	1.56%
ripple	1.36%	0.62%	1.25%
cardano	1.68%	0.83%	1.64%
avalanche-2	2.92%	1.15%	2.18%
dogecoin	3.96%	1.12%	2.14%
polkadot	0.98%	0.82%	1.53%
matic-network	3.52%	0.92%	1.72%
tron	1.20%	0.54%	1.01%
chainlink	0.79%	0.81%	1.39%
litecoin	0.70%	0.80%	1.42%
cosmos	1.08%	0.79%	1.37%

Figure 2: Expected Excess Return Vectors

Using Formula (4), we can compute the weights of each strategy:

	Historical w	MVDA10 w	Implied Returns w	Mkt Cap w
bitcoin	82.32%	25.18%	59.46%	59.46%
ethereum	49.87%	27.12%	22.80%	22.80%
solana	43.07%	1.18%	1.60%	1.60%
binancecoin	27.94%	5.02%	4.48%	4.48%
ripple	-3.37%	-0.87%	2.76%	2.76%
cardano	-46.42%	-4.05%	2.36%	2.36%
avalanche-2	14.23%	0.41%	0.71%	0.71%
dogecoin	18.56%	-0.77%	1.44%	1.44%
polkadot	-38.50%	-5.82%	1.43%	1.43%
matic-network	32.33%	-0.32%	0.65%	0.65%
tron	32.70%	-0.69%	0.57%	0.57%
chainlink	-26.62%	3.41%	0.63%	0.63%
litecoin	-79.74%	0.91%	0.72%	0.72%
cosmos	-25.37%	4.58%	0.40%	0.40%

Figure 3: Weights Comparison

The weights of the Implied Returns are equal to the Market Capitalization-Weighted Portfolio (theoretically as well), and as you can observe, the Historical approach provides more extreme weights than the other two methods. The MVDA10 CAPM strategy doesn't look bad, however the Implied Returns are superior because of the fact that we can incorporate additional information/biases, since it's the prior.

	Historical	MVDA10 CAPM	Implied Returns
Weighted Average	4.23%	0.45%	1.40%
Std. Deviation	1.07%	0.16%	0.32%

Figure 4: Descriptive Statistics

As you can see, the Historical approach which is based on computing the mean of each asset over the 170 weeks(frequentist), has a higher average return and standard deviation than the other two methods. The Implied Returns provide overall a good balance of risk-return.

Now that we have our Implied Returns, we can gather information in order to construct our views.

The following is a list of sample views, annualized:

1. **Bitcoin (BTC)**:
    - *Absolute View*: Expected growth of 100%.
2. **Ethereum (ETH) vs. Bitcoin (BTC)**:
    - *Relative View*: Ethereum might outperform Bitcoin by 40%.
3. **Solana (SOL) and Binance Coin (BNB) vs. Cardano (ADA) and Polkadot (DOT)**:
    - *Relative View*: Solana and Binance Coin could outperform Cardano and Polkadot, by 20%.
4. **Polygon (Matic Network) vs. Chainlink (LINK)**:
    - *Relative View*: Polygon likely to outperform Chainlink by 50%.

Using the market capitalization data, we can compute P:

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.26 & 0.74 & 0 & -0.62 & 0 & -0.38 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \end{bmatrix}$$

Then we can compute the variance of the views:

View	Formula	Variance
1	$p_1 \Sigma p_1'$	0.6018%
2	$p_2 \Sigma p_2'$	0.3596%
3	$p_3 \Sigma p_3'$	0.6301%
4	$p_4 \Sigma p_4'$	2.8365%

Since we know the views, we can construct the view vector Q. The views were given in an annualized basis, so in order to have the same frequency, we have to convert from annual returns to weekly. For example for view 1, which is 100%(=1), we have: $(1+1)^{\frac{1}{52}} - 1$.

$$Q + \varepsilon = \begin{bmatrix} \text{View1} \\ \text{View2} \\ \text{View3} \\ \text{View4} \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \end{bmatrix}$$

4.0.2 Computing the Posterior

With these inputs, all we have to do is set the value of the scalar τ, choose some confidence levels for each view, and then compute Ω. These are the chosen confidence levels, based on my opinion: 100%, 70%, 10%, 80% respectively. The scalar τ will be set to 1, which means that the model will put a decent(not low, not high) weight towards the views.

Given all of this, we get the following Combined Return Vector(Posterior) using the Black-Litterman formula (7):

Asset	Combined Return E(R) (%)	100% Confidence on Views Returns (%)	Π (%)
bitcoin	1.30	1.34	1.29
ethereum	1.83	1.99	1.54
solana	2.14	2.50	1.82
binancecoin	1.70	2.04	1.56
ripple	1.42	1.60	1.25
cardano	1.86	1.86	1.64
avalanche-2	2.43	2.53	2.18
dogecoin	2.60	3.04	2.14
polkadot	1.67	1.73	1.53
matic-network	2.23	2.43	1.72
tron	1.10	1.28	1.01
chainlink	1.53	1.65	1.39
litecoin	1.53	1.66	1.42
cosmos	1.51	1.65	1.37

Plugin these returns to formula (4), gives use the weights, which can be visualized in the following figure:

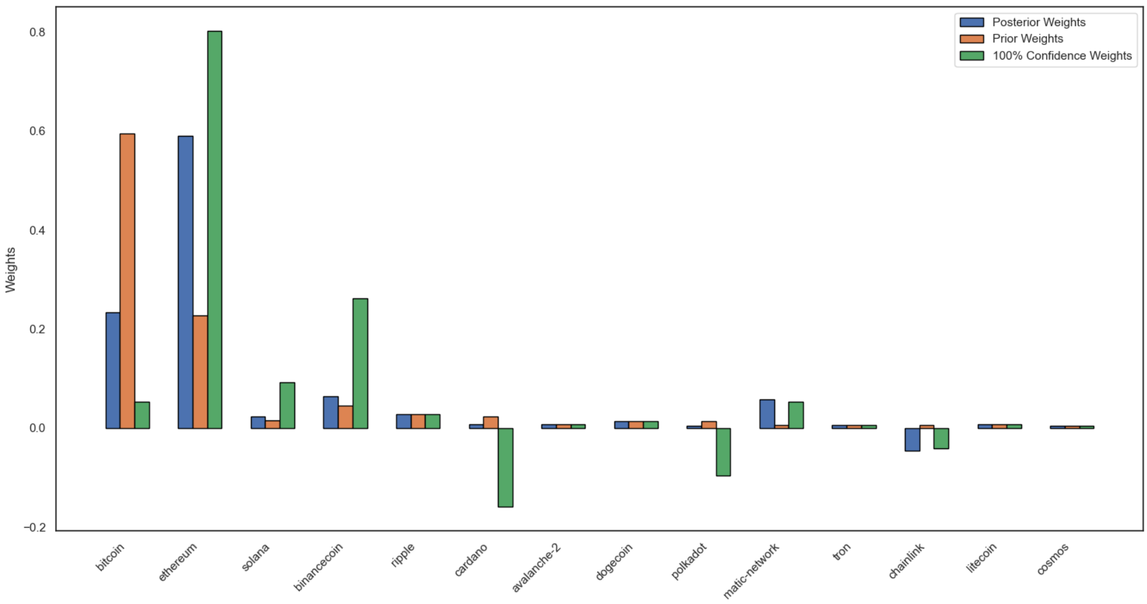

Figure 5: Posterior, Prior and 'Likelihood' Weights

The 100% Confidence on the Views portfolio represents the portfolio where the views have 0% uncertainty, and it can be seen as the likelihood in the Bayesian context. It is fully explained with the Tilt method in the Notes section.

It is clear that the posterior consists of a balance between the prior and the likelihood. View 1 states that Bitcoin will grow 100%, however View 2 states that Ethereum will outperform bitcoin by 40%, which means that it's better to invest in Ethereum. This can be seen in the 'Likelihood' portfolio, that has a very high allocation on Ethereum and a very low allocation on Bitcoin. The prior shows us that historically, Bitcoin has been a better investment, hence the bigger allocation. The posterior shows the balance between both, and the user can choose to be closer to the prior or to the likelihood by choosing the scalar τ and the confidence level in a view.

In the following figure, we can visualize the allocations for the different strategies that were previously mentioned:

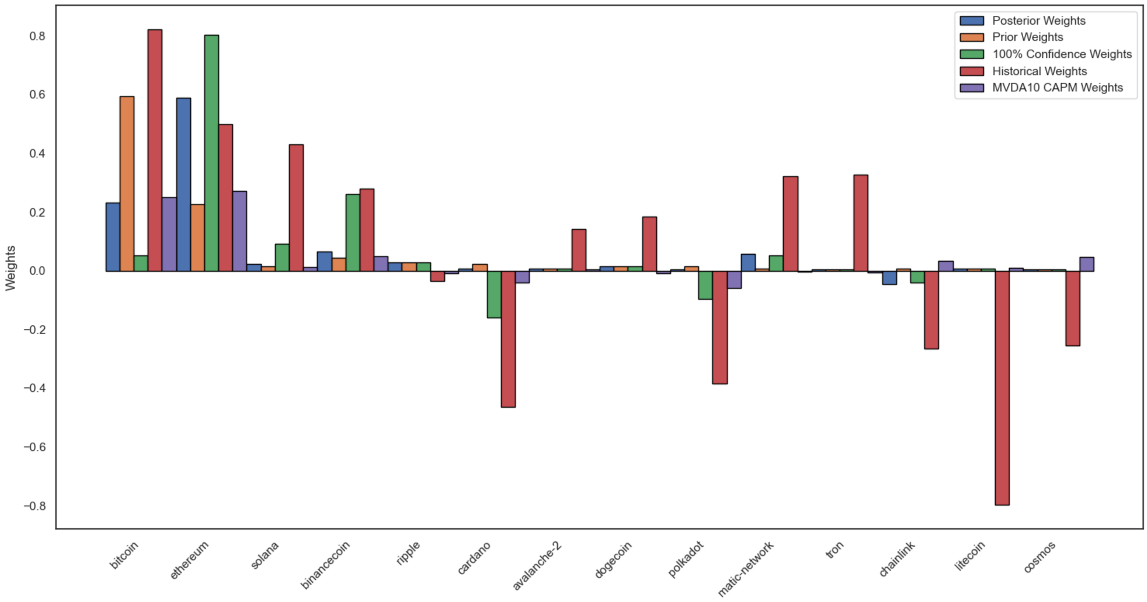

Figure 6: Weights Comparison with other methods

Assuming that the Historical strategy is a frequentist approach, we can see and say that a Bayesian approach for portfolio modelling is more flexible given the fact that the user can add some kind of bias towards a position. This is of course a double-edged sword because the added views can be totally wrong.

Let's evaluate the performance statistics of the portfolios:

	Market Cap-Weighted	**Black-Litterman**	**Historical**
Excess Returns	1.40%	1.74%	4.23%
Variance	0.006485	0.008043	0.019617
Std. Deviation	8.053%	8.968%	14.006%
Beta	1.0	1.08131	1.059696
Residual Return	–	0.0022%	0.0275%
Residual Risk	–	0.0214%	0.1111%
Active Return	–	0.0034%	0.0283%
Active Risk	–	0.0224%	0.1112%
Sharpe Ratio	0.173781	0.193521	0.302237
Information Ratio	–	0.141796	0.24728

In comparison to the Market capitalization-weighted portfolio, the Black-Litterman portfolio shows a solid improvement, with a higher Sharpe ratio (risk-adjusted return). However, the historical portfolio clearly outperforms the Black-Litterman portfolio. There's an obvious reason to that, which is the time period used to build the portfolio (the market has been increasing for the past one and a half years. This is one of the negative aspects of this kind of approach, since it relies heavily of the chosen time-period.

4.0.3 Posterior Distribution

After building our portfolio, which comes from the posterior, we can build a posterior distribution using the posterior mean, which is our expected return vector, and the posterior covariance. This assumes that the posterior distribution is a multivariate normal distribution:

$$N \sim (\mathbf{E}[\mathbf{R}], (\tau\Sigma)^{-1} + (P'\Omega^{-1}P)^{-1})$$

We can simulate the Posterior distribution with the posterior mean and posterior covariance matrix, using a Monte Carlo approach, which involves drawing random samples from the previously defined multivariate normal distribution.

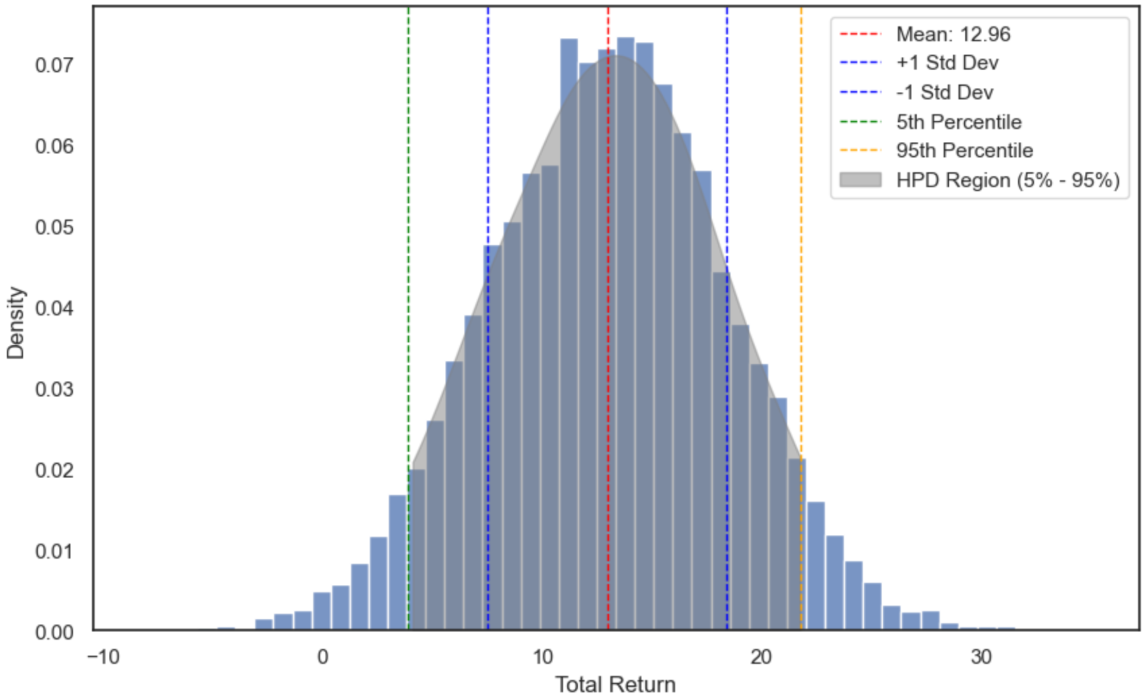

Figure 7: Distribution of Posterior returns

The previous graph shows an histogram of the posterior distribution, coming from 10,000 Monte Carlo simulations. The range of most probable outcomes is shown as well, which is the highest posterior region (HPD). A solid portfolio came from this model, showing a very little quantity in the negative area of returns. However, it is important to point out that the study is highly biased with the big assumptions of the future performance of the used cryptocurrencies.

We can include some Bayesian hypothesis testing. In Bayesian statistics, hypothesis testing is approached differently compared to the traditional frequentist hypothesis testing. Instead of testing a null hypothesis against an alternative hypothesis and using p-values, Bayesian hypothesis testing evaluates the probability of a hypothesis given the observed data. We can test the probability of the average return of the posterior distribution being greater than the average return of the prior:

```
Probability of H1 (mean return > pi): 0.9908
Probability of H2 (mean return <= pi): 0.009199999999999986
```

Figure 8: Hypothesis test

As you can see the odds are very high, which is not surprising because we gave a decent amount of weight towards the views, and the average confidence level was not low.

5 Conclusion

In conclusion, the Bayesian approach to portfolio modeling, exemplified by the Black-Litterman model, offers significant benefits over traditional frequentist methods. For simplicity, the shown frequentist method was simply using historical averages, however traditional mean-variance optimizations are very similar and provide less balanced portfolios, with a very high input sensitivity. Both approaches can be affected by the time-period bias, however the historical approach is way more sensitive to it because it's the only information it uses. This is why using a Bayesian approach with the Black-Litterman model may be superior. It essentially offers the possibility to incorporate a bias towards our portfolio, which could bring a negative outcome as well, so the view feature has to be carefully used. One important thing to mention is that a portfolio based on the Black-Litterman model is sensible to the assets chosen to build the prior, so it is very important to do a proper asset screening before creating the prior. That being said, combining a good prior and a good likelihood (estimates coming from reliable analysts) can lead to a very solid portfolio. So overall, this model can bring excellent results in he right hands.

6 References

1) A STEP-BY-STEP GUIDE TO THE BLACK-LITTERMAN MODEL, by: Thomas M. Idzorek
All the Black-Litterman model theory, especially the tilt method come from this paper.

2) Introduction to Bayesian Statistics and Computational Methods, by: Antonietta Mira

3) Introduction to Bayesian Computing, by: Antonietta Mira and Stefano Peluso

4) Data: coingecko.com ; marketvector.com

7 Notes

1) Matrix P example:

The matrix P is defined as:

$$P = \begin{bmatrix} p_{1,1} & \cdots & p_{1,n} \\ \vdots & \ddots & \vdots \\ p_{k,1} & \cdots & p_{k,n} \end{bmatrix} \tag{8}$$

In this example, we have a (3×7) matrix, which means that there are 3 views, and the portfolio is made of 7 assets.

$$P = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.9 & -0.9 & 0.1 & -0.1 & 0 \end{bmatrix}$$

The 1^{st} row has a 1 in the 6^{th} column, which corresponds to an absolute view on the 6^{th} asset of the portfolio.

The 2^{nd} row has a -1 and a 1, in the 1^{st} and 2^{nd} column, respectively. The underperforming asset (Asset 1) has a negative value and the outperforming asset (Asset 2) has a positive value. Since there are only two assets involved, the market capitalization weighting scheme is not being used.

For the 3^{rd} row, the market capitalization weighting scheme is being used. Let's say that Asset 3 and 4 have a market cap of \$90 million and Asset 5 and 6 have a market cap of \$10 million. Let's assume, from an investor's view, that Asset 3 & 5 will outperform Asset 4 & 6. Note that the 'quantity' of the outperformance is not needed for P.

To compute the weights for P, we grab the outperforming assets and compute their relative weight(by market cap). We do the same for the set of underperforming assets.

We have:

$$\text{Asset 3 relative weight} = \frac{90 \text{ million}}{90 \text{ million} + 10 \text{ million}} = 0.9$$

$$\text{Asset 5 relative weight} = \frac{10 \text{ million}}{90 \text{ million} + 10 \text{ million}} = 0.1$$

Since those two assets are outperformers, the sign stays positive. Assets 4 and 6 have the same computation (in this example), however since they are underperformers, the sign becomes negative.

2) Tilt Method for computing Ω

To explain the tilt method, we'll start by defining the 100% confidence on the views portfolio. Setting Ω to zero is the equivalent to say that we are 100% confident on the views.

The Black-Litterman formula becomes:

$$E[R_{100\%}] = \Pi + \tau\Sigma(P^T\Sigma P)^{-1}(Q - \Pi)$$

Substituting $E[R_{100\%}]$ for μ in Formula 2 gives us the weights based on 100% confidence in the views: $W_{100\%}$.

The tilt method to compute Ω involves intuitive confidence levels, which are specified by the user.

$$\text{Tilt}_k \approx (W_{100\%} - w_{\text{mkt}}) * C_k$$

where Tilt_k $(N \times 1)$ is the approximate tilt caused by the k^{th} view, and C_k is the confidence in the k^{th} view.

In the absence of other views, the approximate recommended weight vector resulting from the view is:

$$W_{k,\%} \approx W_{\text{mkt}} + \text{Tilt}_k$$

where

$W_{k,\%}$ is the target weight vector based on the tilt caused by the k^{th} view $(N \times 1)$.

This method involves several steps, which are the following:

Step 1: For each view k, calculate the New Combined Return Vector $E[R_{k,100\%}]$ using the Black-Litterman formula under 100% certainty, treating each view as if it was the only view.

$$E[R_{k,100\%}] = \Pi + \tau\Sigma(P_k^T\Sigma P_k)^{-1}(Q_k - P_k\Pi)$$

where

- $E[R_{k,100\%}]$ is the Expected Return Vector based on 100% confidence in the k^{th} view $(N \times 1)$;
- P_k identifies the assets involved in the k^{th} view $(1 \times N)$;
- Q_k is the k^{th} View (1×1).

Step 2: Calculate the weight vectors $W_{k,100\%}$ using formula 2.

Step 3: Compute $D_{k,100\%} = W_{k,100\%} - w_{\text{mkt}}$

where

$D_{k,100\%}$ is the departure from market capitalization weight based on 100% confidence in k^{th} view $(N \times 1)$.

Step 4: Compute

$$Tilt_k = D_{k,100\%} \times C_k$$

where

- $Tilt_k$ is the desired tilt (active weights) caused by the k^{th} view $(N \times 1)$; and,
- C_k is $(N \times 1)$. It contains the Confidence Level of the view.

The assets that are involved in the view receive the confidence level of the k^{th} view and the assets that are not part of the view are set to 0.

Step 5: Compute the target weight vector $W_{k,\%}$ based on the tilt.

$$W_{k,\%} = w_{\text{mkt}} + Tilt_k$$

Step 6: Find ω_k, which corresponds to the diagonal element of Ω in row k, by minimizing the sum of the squared differences between $W_{k,\%}$ and W_k.

$$\text{minimize } \sum (W_{k,\%} - W_k)^2$$

subject to $\omega_k > 0$.

where

$$W_k = [\lambda\Sigma]^{-1} \left((\tau\Sigma)^{-1} + P_k' \omega_k^{-1} P_k\right)^{-1} \left((\tau\Sigma)^{-1}\Pi + P_k' \omega_k^{-1} Q_k\right)$$

3) Code:

All the code that was used to make the computations can be found in my Github:

https://github.com/MartinKeke/Portfolio-Modelling-Black-Litterman