Portfolio Re-allocation with Hidden Shocks

Background and Assumptions

Given N assets $S_1 \ldots S_N$ such that

$$dS_1 = S_1\mu_1 dt + S_1\sigma_{11}dW_1 + \cdots + S_1\sigma_{1M}dW_M$$

$$\ldots \tag{1}$$

$$dS_N = S_N\mu_N dt + S_N\sigma_{N1}dW_1 + \cdots + S_N\sigma_{NM}dW_M$$

Where $dW_1 \ldots dW_M$ are M Brownian motions that can be either dependent or independent.

Assuming we want to form a portfolio with those N assets + a risk-free asset, to tell the story in matrix notations, we have:

$$\mathbf{A}\times\mathbf{B}=\begin{pmatrix} \mu_1 - r & \mu_2 - r & \cdots & \mu_N - r \\ \sigma_{11} & \sigma_{21} & \ddots & \vdots \\ \ldots & \ldots & \ldots & \ldots \\ \sigma_{1M} & \sigma_{2M} & \cdots & \sigma_{NM} \end{pmatrix}\begin{pmatrix} u_1 \\ u_2 \\ \ldots \\ u_N \end{pmatrix} = \begin{pmatrix} \beta \\ 0 \\ \ldots \\ 0 \end{pmatrix} \tag{2}$$

such that $\beta > 0$, and r is the risk-free rate.

The weight vector **B** (N by 1) follows the relationship of $u_1 + \cdots + u_N = 1 - u_B$, where u_B is the risk-free asset allocation (e.g. risk-free bonds).

By no-arbitrage theory, matrix **A** (M+1 by N) needs to be singular to prevent having a solution that drives β infinitely high, therefore, each asset's risk-premium can be written as a linear combination of its diffusion terms to avoid invertibility; namely:

$$\mu_i - r = \sum_{j=1}^{M} \sigma_{ij}\lambda_j \,;\, \lambda_j > 0 \tag{2}$$

Following the abovementioned relationship should be the best way to define each asset's drift and diffusion terms under normal market conditions.

Trading strategy and its Ito's Lemma derivation

We define our claim to be:

$$F = \mathrm{c} \cdot \{\gamma_1 \log(S_1) + \cdots + \gamma_N \log(S_N) + (1 - \gamma_1 - \cdots - \gamma_N)\log(B)\}$$

$$where\ \gamma_B = 1 - \gamma_1 - \cdots - \gamma_N \tag{3}$$

and c is a positive number and γ_i which belongs to the set of $\{\gamma_i : i \in 1, \ldots, N\}$ is the weight on an individual asset. To simplify the model, I assume M=2 which means two Brownian motions are playing parts in the diffusion terms for all assets, and I assume them to be independent in a way such that $dW_1 \cdot dW_2 = \rho \cdot dt$; hence, in my following section, assumptions will be based on matrix **A** being 3 by N. Using Ito's lemma:

$$
\begin{aligned}
dF &= F(S_1 + dS_1, \dots, S_N + dS_N, B + dB, t + dt) - F(S_1, \dots, S_N, B, t) \\
&= \frac{\partial F}{\partial t} dt + \frac{\partial F}{\partial S_1} dS_1 + \cdots + \frac{\partial F}{\partial S_N} dS_N + \frac{\partial F}{\partial B} dB + \frac{1}{2}\frac{\partial^2 F}{\partial S_1^2} dS_1^2 + \cdots + \frac{1}{2}\frac{\partial^2 F}{\partial S_N^2} dS_N^2 \\
&= c\left\{\frac{\gamma_1}{S_1}(S_1\mu_1 dt + S_1\sigma_{11} dW_1 + S_1\sigma_{12} dW_2) + \cdots\right. \\
&+ \frac{\gamma_N}{S_N}(S_N\mu_N dt + S_N\sigma_{N1} dW_1 + S_1\sigma_{N2} dW_2) + \frac{\gamma_B}{B} rB dt \\
&+ \frac{1}{2}\left(-\frac{\gamma_1}{S_1^2}\right)(S_1^2\sigma_{11}^2 dt + S_1^2\sigma_{12}^2 dt + 2S_1^2\sigma_{11}\sigma_{12}\rho dt) + \cdots \\
&\left.+ \frac{1}{2}\left(-\frac{\gamma_N}{S_N^2}\right)(S_N^2\sigma_{N1}^2 dt + S_N^2\sigma_{N2}^2 dt + 2S_N^2\sigma_{N1}\sigma_{N2}\rho dt)\right\} \\
&= c\left\{\left[\gamma_1\mu_1 + \cdots + \gamma_N\mu_N + \gamma_B r\right.\right. \\
&- \frac{1}{2}(\gamma_1\sigma_{11}^2 + \gamma_1\sigma_{12}^2 + \cdots + \gamma_N\sigma_{N1}^2 + \gamma_N\sigma_{N2}^2 + 2\gamma_1\rho\sigma_{11}\sigma_{12} + \cdots \\
&\left.\left.+ 2\gamma_N\rho\sigma_{N1}\sigma_{N2})\right] dt + (\gamma_1\sigma_{11} + \cdots + \gamma_N\sigma_{N1}) dW_1 + (\gamma_1\sigma_{12} + \cdots + \gamma_N\sigma_{N2}) dW_2\right\}
\end{aligned}
\tag{4}
$$

Therefore, we have a drift term of $c\sum_{i=1}^{N}\gamma_i\mu_i + \gamma_B r - \frac{1}{2}(\gamma_i\sigma_{i1}^2 + \gamma_i\sigma_{i2}^2 + 2\gamma_i\rho\sigma_{i1}\sigma_{i2})$ and two diffusion terms of $c\sum_{i=1}^{N}\gamma_i\sigma_{i1}$ and $c\sum_{i=1}^{N}\gamma_i\sigma_{i2}$

Under no market shocks, the portfolio is equally weighted with no bond allocation; namely, $\gamma_i = \frac{1}{N}; i = 1, \dots, N$ and $\gamma_B = 0$, however, when shocks are introduced to the market, an idiosyncratic jump diffusion is added to each asset. To ensure our portfolio performance doesn't vary by large margins, we want to recalculate the weights on each asset, and potentially, to add bond allocation to match our actual return and volatility under normal market conditions. Nevertheless, we don't know those shocks' distributions, but we know they follow some hidden functions with respect to their theoretical drifts and volatilities. My goal is to develop a framework that chooses the best weights when jump diffusions are attached on our pool of assets.

To define our claim $\tilde{F}$ under shock conditions, we write:

$$\tilde{F} = \mathrm{c} \cdot \{\gamma_1 \log(\tilde{S}_1) + \cdots + \gamma_{\mathrm{N}} \log(\tilde{S}_N) + (1 - \gamma_1 - \cdots - \gamma_N)\log(B)\}$$

$$where\ d\tilde{S}_i = \tilde{S}_i\mu_i dt + \tilde{S}_i\sigma_{i1} dW_1 + \cdots + \tilde{S}_i\sigma_{iM} dW_M + J_i dP$$

$$st. J_i = f\left(\mu_i, \sigma_{ij}\right); j = 1 \dots M$$

$$and\ \gamma_B = 1 - \gamma_1 - \cdots - \gamma_N; M = 2 \quad (4)$$

Using Ito's lemma, we then have:

$$\begin{aligned} d\tilde{F} = c \Bigg\{ \Big[&\gamma_1\mu_1 + \cdots + \gamma_N\mu_N + \gamma_B r \\ &- \frac{1}{2}(\gamma_1\sigma_{11}^2 + \gamma_1\sigma_{12}^2 + \cdots + \gamma_N\sigma_{N1}^2 + \gamma_N\sigma_{N2}^2 + 2\gamma_1\rho\sigma_{11}\sigma_{12} + \cdots \\ &+ 2\gamma_N\rho\sigma_{N1}\sigma_{N2}) \Big] dt + (\gamma_1\sigma_{11} + \cdots + \gamma_N\sigma_{N1})dW_1 + (\gamma_1\sigma_{12} + \cdots + \gamma_N\sigma_{N2})dW_2 \\ &+ (\gamma_1 J_1 + \cdots + \gamma_N J_N)dP \Bigg\} \end{aligned} \quad (5)$$

Since we can derive the explicit solutions for the PDEs under both normal and shocked markets ((4)&(5)), we can simulate the actual portfolio returns and volatilities easily.

Model Implementation

Assuming we are given H number of assets and to choose N out of it to form a portfolio, we have $\frac{H!}{(H-N)!}$ choices of combinations. Each of those portfolios comprise assets with different theoretical returns μ_i and theoretical volatilities σ_{i1}^2 and σ_{i2}^2, and hidden functions of jump structures with respect to their μ_i, σ_{i1}^2 and σ_{i2}^2. However, at the time of re-allocating the weights to assets and bond, we lack the information of the hidden function of jumps $J_i = f(\mu_i, \sigma_{i1}, \sigma_{i2})$ which prevents us from obtaining closed-form solutions. As one example of model's X variables, it could be written in the following form when multiplying to its corresponding weight vector, say $\Gamma\ st. \Gamma \subseteq S$, and S is the sample space for all possible weight structures:

$$M = X \cdot \Gamma = \begin{pmatrix} \mu_1 & \mu_2 & \cdots & \mu_N \\ \sigma_{11} & \sigma_{21} & \dots & \sigma_{N1} \\ \sigma_{12} & \sigma_{22} & \cdots & \sigma_{N2} \end{pmatrix} \begin{pmatrix} \gamma_1 \\ \gamma_2 \\ \dots \\ \gamma_N \end{pmatrix} = \begin{pmatrix} \gamma_1\mu_1 & \gamma_2\mu_2 & \cdots & \gamma_N\mu_N \\ \gamma_1\sigma_{11} & \gamma_2\sigma_{21} & \dots & \gamma_N\sigma_{N1} \\ \gamma_1\sigma_{12} & \gamma_2\sigma_{22} & \cdots & \gamma_N\sigma_{N2} \end{pmatrix} \quad (6)$$

Our first step is to randomly generate the sample space S by utilizing PyTorch's Convolutional Neural Network (CNN) function *torch.nn.conv2d(in channels, out channels, kernel size)*; assuming we want K number of weight vectors in S, we have *in channels* = 1, *out channels* = K, and *kernel size* = (1, N). As a result, each *out channel* produces a set of randomly generated weights and bias, and after the multiplication of X and $\Gamma_{\mathrm{i}}, i = 1, \dots, K$, we obtain K number of weighted sums of μ_i, σ_{i1}^2 and σ_{i2}^2; namely, $\sum_{i=1}^{N} \gamma_{ik}\mu_i$, $\sum_{i=1}^{N} \gamma_{ik}\sigma_{i1}$ and $\sum_{i=1}^{N} \gamma_{ik}\sigma_{i2}$ $where\ k = 1, \ldots, K.$

The second step is to define β_0 and $(\beta_1, \beta_2, \beta_3)$ in a way they work as logistic regression intercept and coefficients in the following ways for K number of weighted sums of μ_i, σ_{i1}^2 and σ_{i2}^2:

$$\Pr(Y = 1) = \frac{\exp\,(A)}{1 + \exp\,(A)}$$

$$\text{where A} = \left(\beta_0 + \beta_1 \sum_{i=1}^{N} \gamma_{ik}\mu_i + \beta_2 \sum_{i=1}^{N} \gamma_{ik}\sigma_{i1} + \beta_3 \sum_{i=1}^{N} \gamma_{ik}\sigma_{i2}\right) \tag{7}$$

The formation of A could be done by using *m=torch.nn.Linear(in features, out features, bias=True)* by setting *in features* = 3 and *out features* = 1. By turning bias to be true, we will get one intercept β_0 (*m.bias*) and three coefficients $\beta_1, \beta_2, \beta_3$ (*m.weight*) that correspond to one weighted drift and two weighted diffusions; calling this linear function J times yields J sets of $(\beta_0, \beta_1, \beta_2, \beta_3)$ and each of those parameter sets calculates A, which works as a score of the weighted drift and diffusions. Since we have K number of out channels, for each X variable, we have K by J times of scores A. Plug A into the probability function $\Pr(Y = 1)$ as showing above is as simple as calling *torch.nn.functional.sigmoid(x)* from the PyTorch library, which results in K by J number of probabilities $\Pr(Y = 1)$.

To Define Our Y Variables

Following from the last section, my goal is to maximize the probability of choosing weights that best reconcile discrepancies from market shocks. There are two questions we need to answer:

1. What does Pr(Y=1) mean, or what outcome we wish to get here?
2. How is maximizing Pr(Y=1) in our case different from Maximum Likelihood Estimation for Logistic Regression?

To answer question 1, we first need to review our hidden shock functions with respect to their corresponding theoretical returns and volatilities. Under normal market conditions, the equally weighted portfolio generates actual portfolio return Ret and actual portfolio volatilities Vol that can be calculated by simulating equation (4). Under the hidden market shocks, we need to reallocate the assets in a way its new actual portfolio return $\widetilde{Ret}$, and new actual portfolio volatilities $\widetilde{Vol}$ are as close to Ret and Vol as possible. By applying weighted loss function we can tell the model how important to match $\widetilde{Ret}$ with Ret and $\widetilde{Vol}$

with Vol. For example, by setting the weights to be (5, 1) is equivalent saying matching return is 5 times more important than matching volatility. On top of that, we don't want to jeopardize our Sharpe ratio $\widetilde{Ret}/\widetilde{Vol}$; therefore, Y=1, namely having a desirable re-allocated portfolio under market shocks, means it has an equal or better Sharpe ratio than its equally weighted portfolio under no market shocks.

Recall from equation (7), if to apply logistic regression to K numbers of weighted returns $\sum_{i=1}^{N}\gamma_{ik}\mu_i$ and volatility sets $(\sum_{i=1}^{N}\gamma_{ik}\sigma_{i1}, \sum_{i=1}^{N}\gamma_{ik}\sigma_{i2})$; the likelihood function tells us to maximize:

$$L(\beta_0,\beta_1,\beta_2,\beta_3) = \Pr\,(Y_1 |\, \textstyle\sum_{i=1}^{N}\gamma_{i1}\mu_i\,, \sum_{i=1}^{N}\gamma_{i1}\sigma_{i1}, \sum_{i=1}^{N}\gamma_{i1}\sigma_{i2}) \times \ldots \times \\ \Pr\,(Y_K |\, \textstyle\sum_{i=1}^{N}\gamma_{iK}\mu_i\,, \sum_{i=1}^{N}\gamma_{iK}\sigma_{i1}, \sum_{i=1}^{N}\gamma_{iK}\sigma_{i2}) \tag{8}$$

However, in our actual process, instead of directly applying logistic regression, or namely using equation (8) to obtain the intercept and coefficients, $(\beta_0,\beta_1,\beta_2,\beta_3)$ are being updated by the designated optimization algorithm but only to maximize "the best" re-allocation. Our ultimate goal is not to have a regression model that's in its best performance of all X variables but the focus is only towards "the best" X variable and to maximize $\Pr\,(Y_k = 1 |\, \sum_{i=1}^{N}\gamma_{ik}\mu_i\,, \sum_{i=1}^{N}\gamma_{ik}\sigma_{i1}, \sum_{i=1}^{N}\gamma_{ik}\sigma_{i2})$.

Our Optimization Algorithm

Assuming we are given N number of assets and to choose N from it, so there is one choice of combination of X variable. Last section has briefly touched on how intercept and coefficients are updated which is part of our optimization process; in addition, our K sample of portfolio weights $\Gamma_k = (\gamma_{1k}, \ldots, \gamma_{Nk})$ are being updated to maximize the best allocation. Our objective function can be written as:

$$\max \Pr\,(Y_k = 1 |\, \textstyle\sum_{i=1}^{N}\gamma_{ik}\mu_i\,, \sum_{i=1}^{N}\gamma_{ik}\sigma_{i1}, \sum_{i=1}^{N}\gamma_{ik}\sigma_{i2}) \\ where\ k = \{x: x \in \{1, \ldots, K\}\}\ and\ \Gamma_k = \{x: x \in S\} \tag{9}$$

and S is the collection of all portfolio weights, or namely a sample space.

To write equation (9) in terms of minimizing the cost function:

$$\min MSE(\textstyle\sum_{i=1}^{N}\gamma_{ik}\mu_i\,, \sum_{i=1}^{N}\gamma_{ik}\sigma_{i1}, \sum_{i=1}^{N}\gamma_{ik}\sigma_{i2})$$

$$= \left(\delta_{Ret} \cdot (\widetilde{Ret} - Ret)\right)^2 + \left(\delta_{Vol} \cdot (\widetilde{Vol} - Vol)\right)^2$$

(10)

δ_{Ret} and δ_{Vol} are the tuning weights on return and volatility. In the previous section, I set $\delta_{Ret} = 5$ and $\delta_{Vol} = 1$ by saying matching return is 5 times more important than matching volatility. And our constraint function is as simple as:

$$\widetilde{Ret} / \widetilde{Vol} \geq Ret / Vol \quad (11)$$

Results

Seed = 10

$H = 10,\ N = 10,\ K = 500$; number of optimizations = 1000:

X=

([0.0613, 0.0582, 0.0584, 0.0606, 0.0603, 0.0582, 0.0584, 0.0600, 0.0589, 0.0578],

[0.3428, 0.1552, 0.3084, 0.3372, 0.2746, 0.2062, 0.1995, 0.3401, 0.1923, 0.1721],

[0.3213, 0.3883, 0.1510, 0.2780, 0.3532, 0.3031, 0.3304, 0.2230, 0.3794, 0.3286])

The initial loss before optimization: 7.1951

The final loss after optimization: 0.0566

Final Asset Allocation:

[[0.010419570025624696, 0.017949788455818 01, 0.014145524086563265, 0.019812500356003262, 0.015395207182580 62, 0.003912777435760183, 0.009383646521554703, 0.0, 0.018859917837433633, 0.012362024957391757]]

Equally Weighted Accumulated Daily Return and Daily Vol:

[-0.0124, 0.0046]

Final Accumulated Daily Return and Daily Vol:

[0.007158850204848077, 0.0005660226472913477]