

Better Incentives for Participants in Crowd-Sourced Hedge Funds

Nuo Wen Lei and David Kane

Abstract

Numerai is a crowd-sourced hedge fund with \$300 million in assets under management that collects participant trading ideas based on data that Numerai provides. Participants submit scores periodically for stocks in a pre-determined universe. Numerai combines these submissions into an overall score and then uses that score to create its actual portfolio. We introduce the Rnumerai package for participating in the Numerai tournament. Participants are paid on the correlation between their scores and future returns. We propose two changes. First, participants should submit portfolio weights instead of scores. Second, participants should be paid on the basis of their approximate marginal contribution to Numerai's overall Sharpe Ratio.

Introduction

Lee (2023) provides an overview of how the Numerai process works. The company collates trading ideas submitted from participants who could be freelance data wonks, engineers and scientists with some free time. These submissions after being pooled, weighted by participant stakes, and neutralized to market risk factors, become the “Meta Model” that Numerai invests in. Based on the results of the Meta Model and the contribution of each participant’s submission to the Meta Model, these participants are rewarded with its native cryptocurrency, Numeraire or NMR for short.

Numerai’s submission system lets participants submit scores for each of the stocks. Numerai ranks these scores such that the magnitude of the scores are not important. The score of each stock for each participant’s submission is weighted only by its relative rank to other scores provided in the submission. Additionally, Numerai obfuscates the data that they provide by anonymizing every stock and bucketizing and normalizing every feature. This obfuscation forces participants to use data science methods to create submission scores.

Numerai (2023) describes their overall process of evaluating participant submissions with a method called Meta Model Contribution, or MMC. This method first neutralizes all individual submissions to the “Meta Model”, and then calculates the covariance between the live market target and each participant’s neutralized submission. The idea is that a higher covariance represents a stronger unique positive contribution to the Meta Model, and therefore should be rewarded more.

However, there are 3 flaws with the MMC payout system. First, participants are not incentivized to submit scores that improve Numerai’s Sharpe Ratio. Introduced by Sharpe (~ Sharpe 1966), Sharpe Ratio measures investment success adjusted for the investment risk. Although Numerai uses Sharpe Ratio as their key performance indicator, the inclusion of participant

scores with low correlation to the Meta Model do not increase Numerai’s overall Sharpe Ratio. Second, it is difficult for a payout system like MMC to incorporate real-world constraints like liquidity and short locates into its calculations. Third, since the calculation is based on the Meta Model, there is a lack of transparency for participants to calculate and understand how to act upon their MMC result.

We propose two fixes. First, we propose that participants should be able to submit weights instead of scores. We define portfolio weights as a vector of N values where N is the number of stocks in the universe. Since Numerai invests in both long and short positions, the submitted weights should also allow negative values to represent short positions. Therefore, the constraints for a submission of weights is that the sum of all weights must be 0, the sum of all positive weights must be 1, and the sum of all negative weights must be -1.

The second change we propose is to evaluate participants’ portfolio weights based on their marginal contribution to Numerai’s overall Sharpe Ratio. Numerai should provide greater reward for portfolio weights that perform well when most portfolios perform poorly as those weights contribute the most to Numerai’s Sharpe Ratio. We also discuss ways to introduce other constraints like liquidity and short locates, thereby minimizing the influence of participant submissions that short stocks that cannot be borrowed or buy stocks that are too expensive to trade.

Rnumerai

Rnumerai (~ Omni Analytics Group 2022) is a CRAN package that provides an R interface for participants to use the Numerai data science tournament API. Following Rnumerai, we provide some basic functions to participate in Numerai’s data science tournament.

Load the Rnumerai package.

```
library(Rnumerai)
```

Download the latest data provided by Numerai, which is now V4.2 (Rain).

```
download_dataset("v4.2/train.parquet", "train.parquet")
download_dataset("v4.2/validation.parquet", "validation.parquet")
download_dataset("v4.2/live.parquet", "live.parquet")
download_dataset("v4.2/live_example_preds.parquet",
                 "live_example_preds.parquet")
download_dataset("v4.2/validation_example_preds.parquet",
                 "validation_example_preds.parquet")
download_dataset("v4.2/features.json", "features.json")
```

Load the downloaded data into R.

```
data_train <- data.table::setDT(arrow::read_parquet("train.parquet"))
data_validation <- data.table::setDT(arrow::read_parquet("validation.parquet"))
data_live <- data.table::setDT(arrow::read_parquet("live.parquet"))
```

Given this data, users can use any data science methods to create scores. We make a tibble named `df` that has columns named `id` and `prediction`, representing our predicted score for every obfuscated stock. Then, we upload our predictions to the corresponding model registered on Numerai under the variable name `model_id`.

```
upload_predictions(model_id = model_id, df=df)
```

Rnumerai is working with the constraints of the current Numerai contest, in which participants can only submit scores. It would be better if participants can submit weights.

Methodology

Here are the details on portfolio weight instead of score submissions.

N is the number of stocks in the universe at a point in time and i is an individual stock.

s is a vector of user-submitted scores. s_i is the score for stock i .

r is the vector of raw stock returns. r_i is the return for stock i .

b is the vector of bucketed residual returns used to calculate payout. b_i is the bucketed residual return for stock i . An example of the vector b is `target_nomi20`.

w is the vector of portfolio weights that the user would submit to the portfolio weights API. w_i is the weight for stock i . The portfolio weights w would be required to mimic the long/short structure of a standard equity hedge fund portfolio.

$$\sum_{i=1}^N w_i = 0$$

$$\sum_{i=1}^N [w_i > 0] \cdot w_i = 1$$

From which it follows that:

$$\sum_{i=1}^N [w_i < 0] \cdot w_i = -1$$

And:

$$\sum_{i=1}^N |w_i| = 2$$

Further constraints might be added, but it is not clear if such additions are either necessary or even useful. Nothing prevents users from neutralizing their portfolios to beta, sectors or any other factor. Some will choose to do so. Others will not.

$PR = w \cdot r$ is the portfolio return over a round.

$PB = w \cdot b$ is a unit-less measure of pseudo portfolio return in bucketed residual return space.

We have seen the term of **Meta Model** used in two ways. The first way is as a vector of length N where each value is between 0 and 1, which is calculated as the stake-weighted average of user-submitted scores. The second way is as an N-length vector of weights that has been created via a complex optimization process, which uses the stake-weighted scores as one of its inputs. Therefore, we clarify between **Meta Model scores** and **Meta Model weights**. Since Meta Model weights are also in the form of portfolio weights, that means that in the same way we have portfolio return PR and portfolio bucket PB , we can define Meta Model return.

MR is defined as the Meta Model weights times r .

MB is defined as the Meta Model weights times b .

We refer to the starting Long Market Value invested in a portfolio as LMV . Then, we can define portfolio dollar return as $PR \cdot LMV$ and Meta Model dollar return as $MR \cdot LMV$.

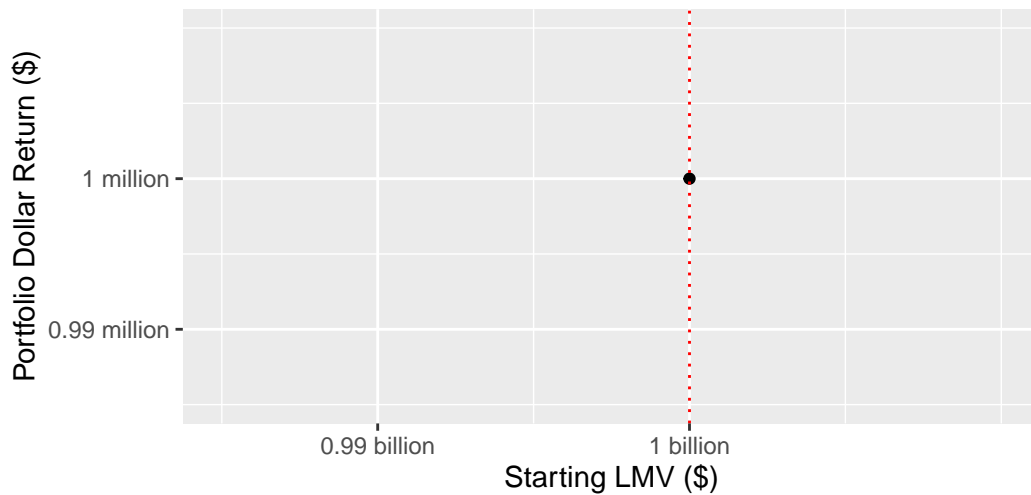
Sharpe Contribution

This section describes a diagram that shows the Sharpe contribution of differing portfolios over a fixed time period, e.g., 20 days. In this section, we ignore liquidity cap and transaction costs.

The x-axis of the diagram is the starting LMV of the hedge fund portfolio at the start of the 20 days and the y-axis is the portfolio dollar return after the 20 days.

Starting LMV and Dollar Return Space

We can place any portfolio in a space defined by its starting LMV and Dollar Return over a time period.

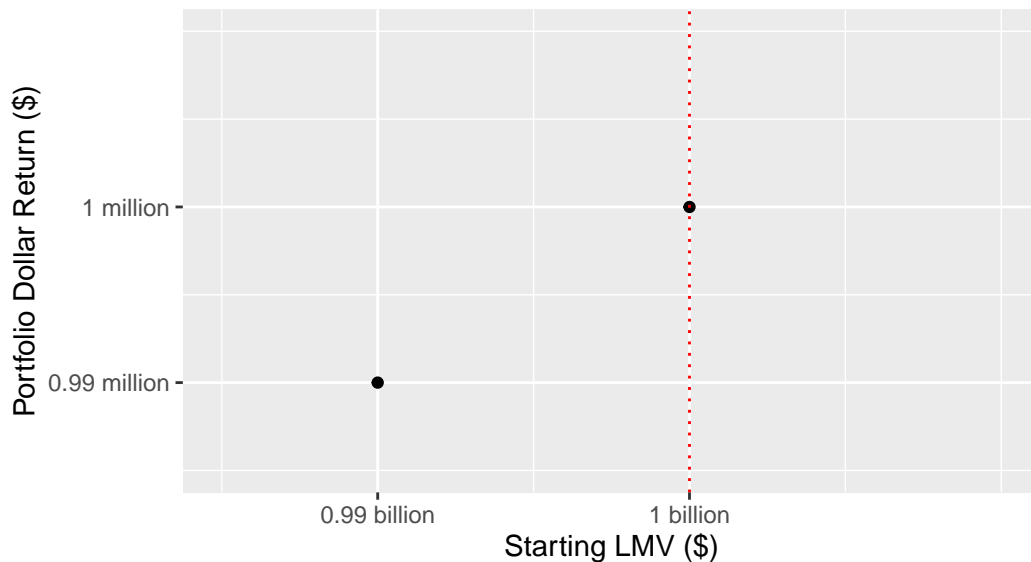


Any investment that has a starting LMV of \$1 billion would be a point on the dotted red line.

Suppose a hedge fund invested \$1 billion at the start of 20 days and the return over this 20 days is \$1 million of profit. This investment and return would be represented as the black point on the diagram above.

Starting LMV and Dollar Return Space

We can place multiple portfolios or the same portfolio with different :

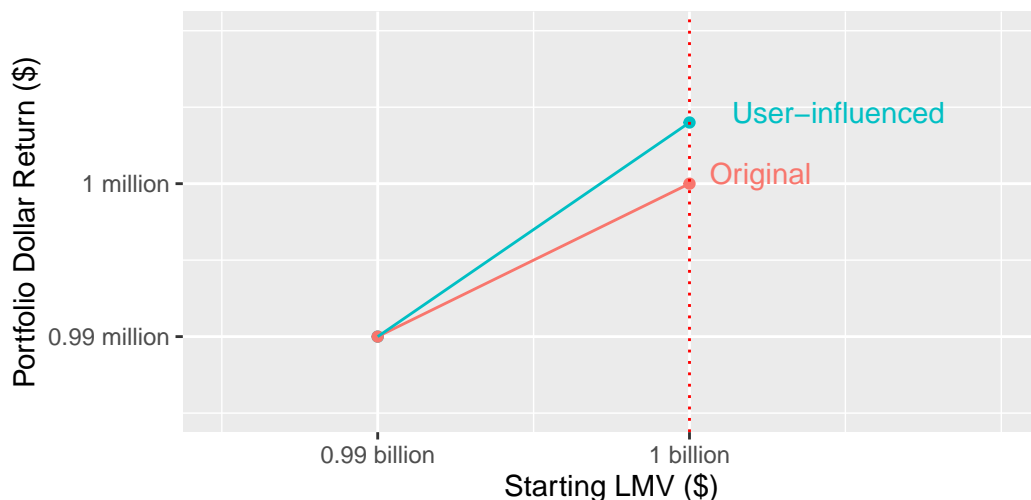


Suppose the hedge fund decides to invest in 99% of their \$1 billion instead of the entire billion. Then the black dot that is not on the red line represents this investment and return.

If a hedge fund removes 1% of their portfolio and invests it directly into a different user-submitted portfolio, then we can plot some possible investment and returns in the diagram below. In practice, changing any part of a portfolio requires transaction costs. However, we ignore these costs for simplicity.

Original vs. User-influenced Returns

We can compare the returns of different portfolios with the same starting LMV.



A line drawn from the origin to any portfolio point shows the trend of the portfolio's dollar return over increments of starting LMV.

The red points in this diagram represent the original portfolio investments at 99% and 100% of \$1 billion. The blue point represents the portfolio investment and return if 1% of the starting *LMV* had been invested in a particular user.

User portfolios can either make more or less than the original portfolio. This diagram shows the addition of a user portfolio that would have made more than the original portfolio, hence

the blue point is higher than the red point on the red dotted line.

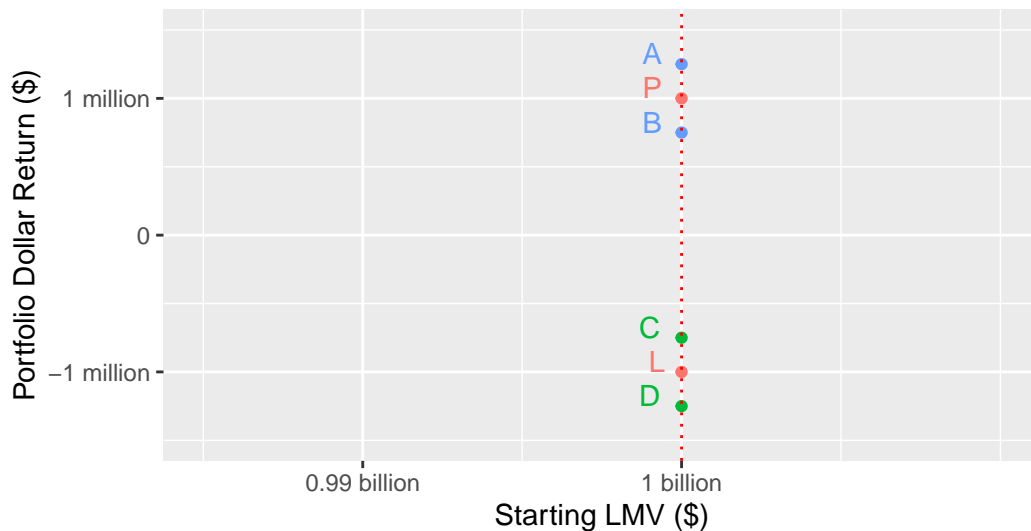
We now consider the 4 scenarios based on whether the original portfolio resulted in a profit or loss and whether the user portfolio would have performed better or worse than the original portfolio.

The 4 scenarios are listed here for clarity:

- Original made more than User
- Original made less than User
- Original lost more than User
- Original lost less than User

Four Scenarios of User Contribution

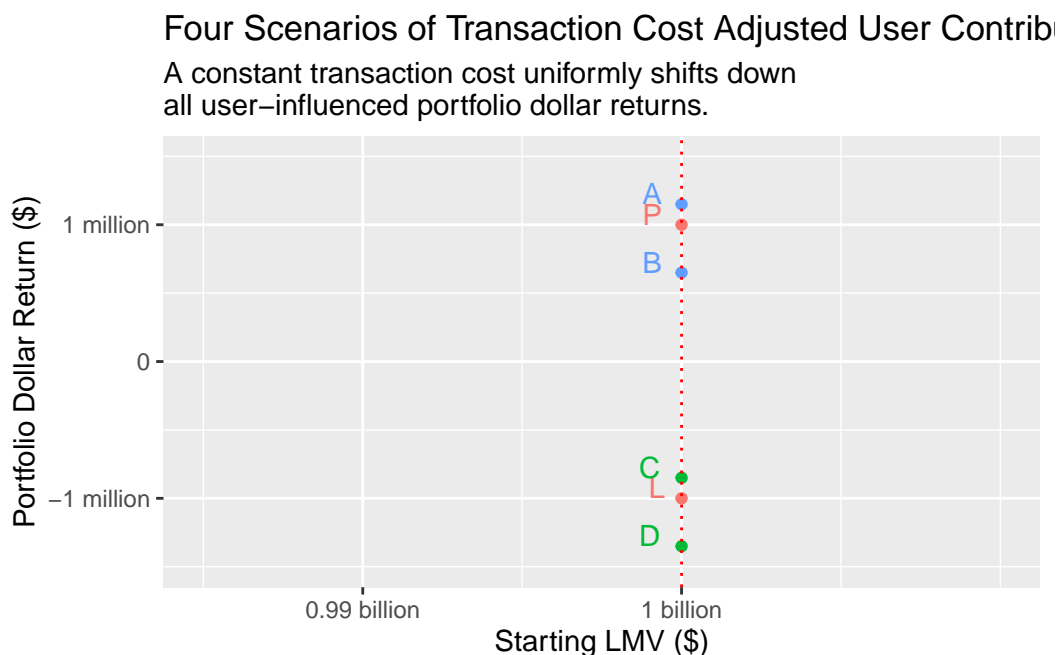
The user portfolio can either be a positive or negative addition.
And the original portfolio can either have profited or lost.



The blue and green points represent the results of the 4 scenarios described previously. The blue points represent the variation in returns when the original portfolio made money and the

green dots represent the variation when the original portfolio lost money.

This diagram, however, does not adjust for transaction cost. We can create a simple illustration of how that would affect the diagram by setting a fixed transaction cost, e.g., \$100,000, for all user-influenced portfolios.



All blue and green points represent user-influenced portfolios, therefore the dollar return for each portfolio is shifted down by a constant transaction cost of \$100,000.

From the diagram, we can derive 4 distances, each between a user-influenced portfolio dollar return and the original portfolio dollar return.

- Distance \overline{PA} between point A and point P
- Distance \overline{PB} between point B and point P

- Distance \overrightarrow{LC} between point C and point L
- Distance \overrightarrow{LD} between point D and point L

Each of these distances describes how much the investment would have made if the original portfolio replaced 1% of its starting LMV with a corresponding portfolio. For example, distance \overrightarrow{PA} shows that the investment would have made \$150,000 more than the original return P if 1% was invested in portfolio A. In other words, the marginal contribution of portfolio A is the distance \overrightarrow{PA} . Any user-submitted portfolio's marginal contribution to the original portfolio can be categorized as one of A, B, C, D.

Now suppose we compare points A and C on the diagram. The marginal contribution of both portfolios is \$150,000 from their respective original portfolio. However, while the marginal contribution may be the same, the marginal Sharpe contribution is different when comparing in the long term.

Two portfolios can have the same average contribution over many time periods of 20 days. However if one portfolio earns contributions mostly in the C area and another portfolio earns contributions mostly in the A area, then the portfolio that earns more C area contributions would also increase the Sharpe ratio more by lowering the variance of the original portfolio.

Therefore, we should incentivize users to submit portfolios that would be plotted in the C area, i.e., portfolios that perform well when the Meta Model performs poorly.

Payout with Portfolio Weights

The payout system should be how Numerai incentivizes users to submit portfolios that performs well especially when the Meta Model is performing poorly.

Payout with Portfolio Weights

The payout for a user submission would be an equation based on the following factors:

- Submitted Portfolio Weights w
- Raw Stock Returns r (or Bucketed Residual Return b)
- Meta Model return MR (or Meta Model bucket MB)

Payout Formula with Meta Model Performance

Given a submitted vector of portfolio weights w and the raw stock return r , we can calculate the portfolio return PR without taking transaction cost into account. PR is represented as the return percentage of the starting long market value LMV .

$$PR = w \cdot r$$

We adjust for some constant transaction cost t based on the final diagram in the [Sharpe Contribution section](#), defining APR as the transaction cost adjusted portfolio return.

$$APR = PR - t$$

In order to account for Meta Model return MR , we define a constant c that amplifies the portfolio return when the Meta Portfolio loses money. We refer to the portfolio payout conditionally scaled by c as PP .

$$PP = \begin{cases} c \cdot APR & \text{if } MR < 0 \\ APR & \text{otherwise} \end{cases}$$

Therefore, the PP of a portfolio is the payout percentage of NMR return or loss based on the portfolio's stake.

Calculating the payout with portfolio weights does not require providing the raw stock returns. Instead, we can do all the necessary calculations in the same space as the bucketed residual returns by replacing all usages of stock return r with bucketed residual return b . Then, we may use portfolio bucket PB instead of portfolio return PR , which means the result will not be a percentage but some unit-less measure of residual score.

$$PB = w \cdot b$$

The constant used to amplify portfolio return conditional on Meta Portfolio performance MR has a different value due to being in bucketed residual return space, so we denote the adjusted constant as c_b . Similarly, the constant transaction cost is in a different unit, so we denote the adjusted transaction cost constant as t_b . Additionally, in order to scale the final result into a range resembling percentage return, we introduce another constant g . Therefore, PP calculated with bucketed residual return b uses different constant values.

$$PP = \begin{cases} \frac{c_b \cdot (PB - t_b)}{g} & \text{if } MR < 0 \\ \frac{PB - t_b}{g} & \text{otherwise} \end{cases}$$

We note that while c , c_b , t , t_c , g are referred to as constants for simplicity, they may be defined as functions of other variables.

Conclusion

Numerai crowd-sources participant submissions to create and manage a hedge fund with \$300 million in assets under management. A crucial component of crowd-sourced hedge funds like Numerai is their method of incentivizing participants to adjust their trading ideas to fit the hedge fund's needs. Numerai incentivizes participants by allowing participants to invest in their own model, through which they would win or lose money based on their model performance measured with MMC.

MMC, or Meta Model Contribution, measures a participant's model performance on live market targets after neutralization against the Meta Model. However, being scored on MMC, participants are not incentivized to submit scores that increase Numerai's Sharpe Ratio, which is Numerai's main goal. Additionally, MMC does not include real-world portfolio constraints such as liquidity or short locates, which should be an important factor for what submissions to weigh more heavily.

With portfolio weights, we introduce the payout formula with Meta Model performance, therefore focusing more on how participant submissions affect the result of the hedge fund rather than the uniqueness with regards to the Meta Model. This method allows Numerai to value

positively contributing strategies more when the hedge fund is performing poorly, which would increase the Sharpe Ratio of the fund's portfolio.

References

- Lee, Justina. 2023. "Quant Hedge Fund Jumps 20." *Bloomberg.com*. Bloomberg.
<https://www.bloomberg.com/news/articles/2023-02-28/tudor-jones-backed-quant-jumps-20-as-crowdsourced-bets-pay-off>.
- Numerai. 2023. "Meta Model Contribution (MMC & BMC) - Numerai Tournament." *Meta Model Contribution (MMC & BMC) - Numerai Tournament*. <https://docs.numer.ai/numerai-tournament/scoring/meta-model-contribution-mmc-and-bmc>.
- Omni Analytics Group. 2022. *Rnumerai: Interface to the Numerai Machine Learning Tournament API*. <https://CRAN.R-project.org/package=Rnumerai>.
- Sharpe, William F. 1966. "Mutual Fund Performance." *The Journal of Business* 39 (1): 119–38. <http://www.jstor.org/stable/2351741>.

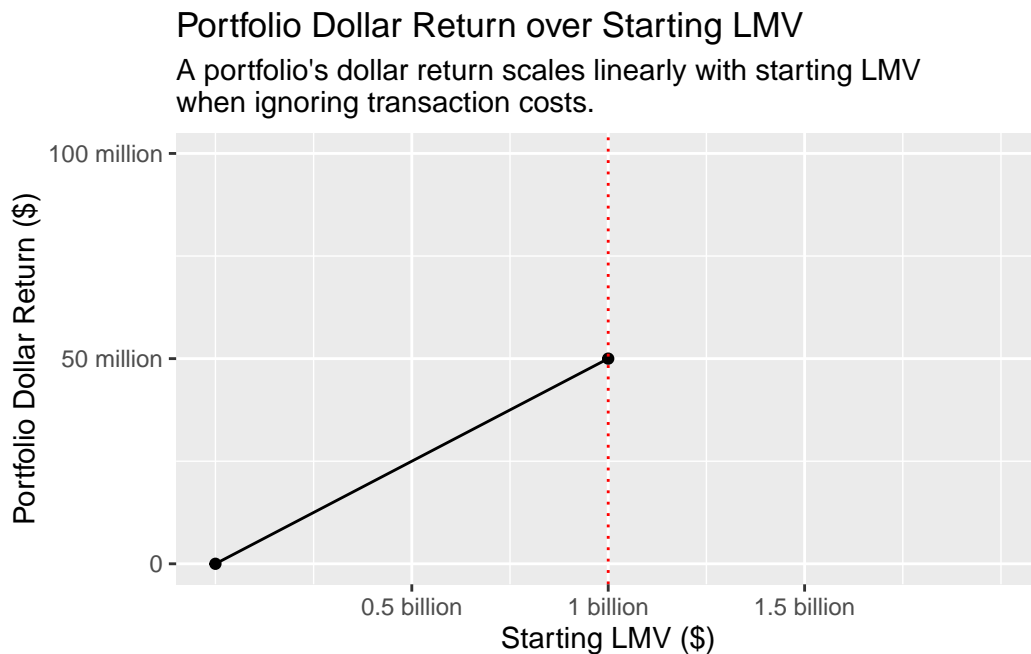
Appendix

The purpose of this appendix is to provide intuition as to why transaction cost adjustment is so important, especially as assets under management at Numerai grow.

Importance of Transaction Cost

We describe a diagram that shows the possible ways an additional portfolio can contribute to a pre-existing original portfolio over a fixed time period, e.g., 20 days. We define an additional portfolio as a portfolio with a new set of weights that is independent (not necessarily different) from the original portfolio.

The x-axis of the diagram is the starting *LMV* of the hedge fund portfolio at the start of the 20 days and the y-axis is the portfolio dollar return of the hedge fund after the 20 days.



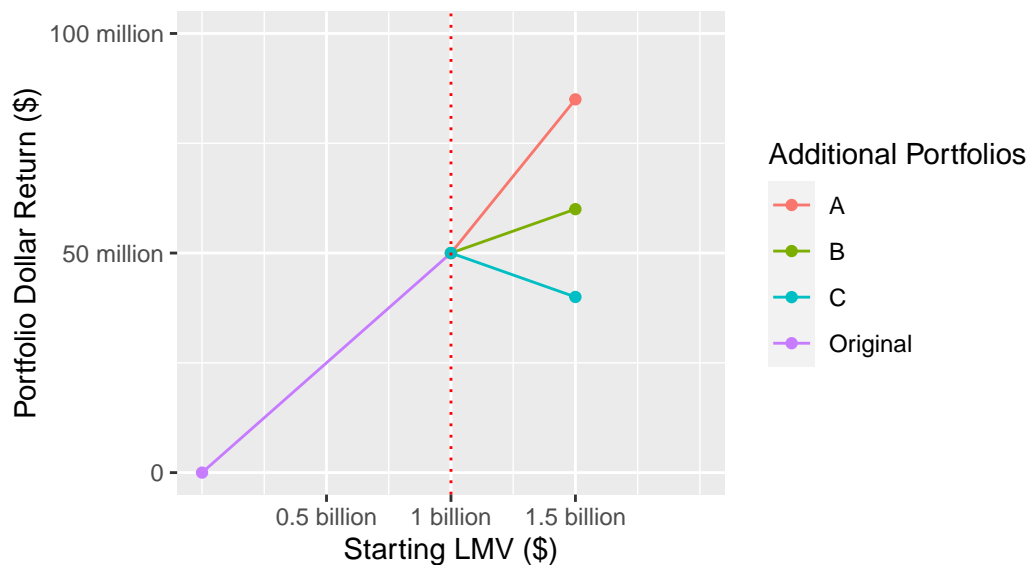
Ignoring liquidity and transaction costs, a hedge fund's investment in their original portfolio would linearly scale with their return. Suppose a hedge fund's original portfolio has a return rate of 5%, then any point on the black line shows what the dollar return (y) of the portfolio would be for any amount of dollars invested (x). For example, the dotted red line represents an investment of \$1 billion and the black dot on the red line represents the dollar return (\$50 million) of a portfolio with a return rate of 5% given the starting *LMV* of \$1 billion. For demonstration purposes, we call the black line the original portfolio at different amount of money invested. Additionally, we call the 5% return rate of the original portfolio the original return rate.

Building from the original portfolio, we introduce three additional portfolios with extra investments of \$500 million and consider how the diagram changes with each.

- Portfolio A (7%): new return rate $>$ original return rate
- Portfolio B (2%): $0 <$ new return rate $<$ original return rate
- Portfolio C (-2%): new return rate < 0

Portfolio investment and returns

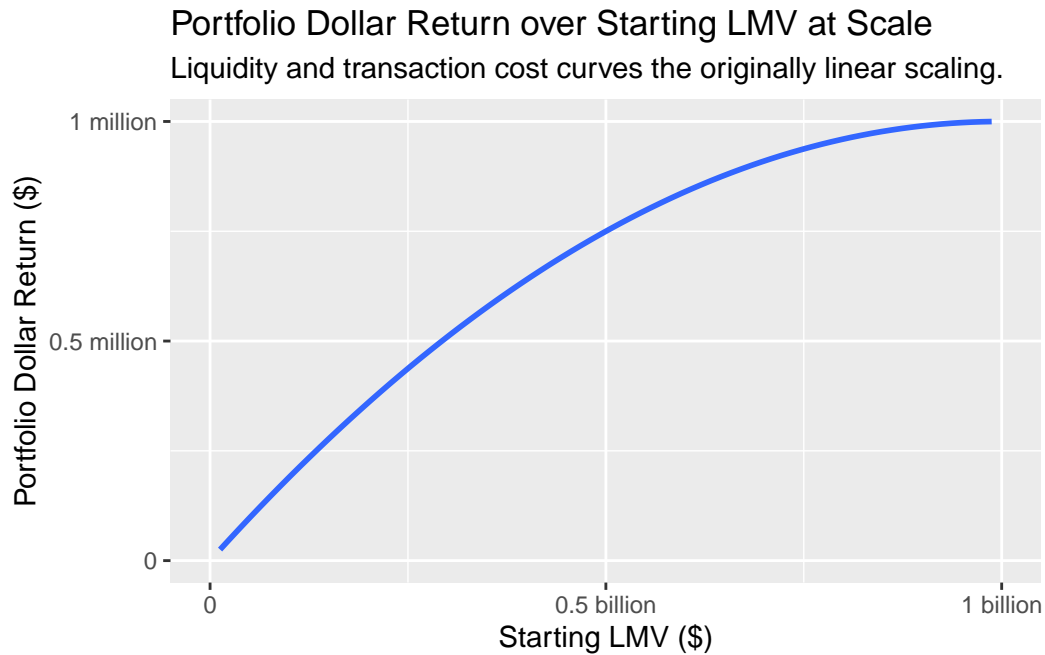
We compare different additional portfolios building upon the same or



On this diagram, an extra initial investment shifts the point to the right, and each of the additional portfolios create a new combined portfolio from the original portfolio, which extends from the original investment and return.

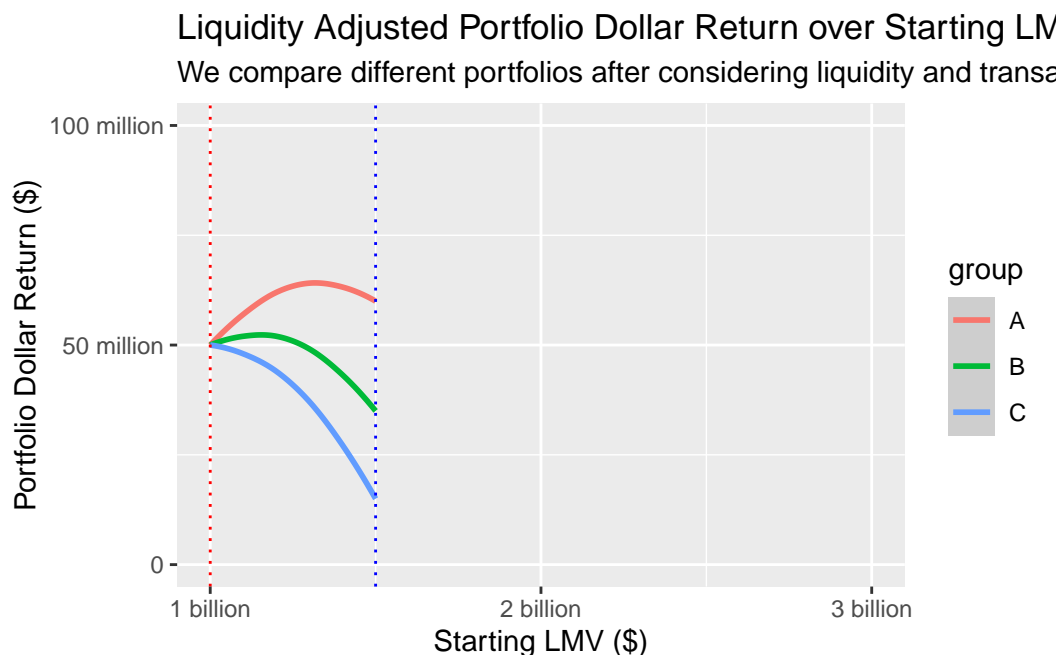
An additional portfolio with a positive return rate would earn positive returns from the extra investment, and this return would be added onto the original portfolio returns. However, a negative return rate would create negative returns and eat into the original portfolio returns.

Until this point, all diagrams and calculations are under the assumption of no transaction costs. In practice, the larger the investment, the higher the transaction costs due to the restriction of stock liquidity caps. The rate at which transaction cost increases relative to dollars invested is based on the liquidity capacity size of a stock.



As the initial investment grows larger, the transaction costs to execute a set of portfolio weights grows exponentially due to its liquidity capacity. Thus the actual return of a set of portfolio weights at different levels of initial investment would look more like the diagram above where the return does not scale linearly with the investment. In fact, the combination of the specific weights in the additional portfolio and the additional dollar amount to be invested determines whether the linear contribution assumption is reasonable.

We apply this transaction cost scaling relationship to our previous diagram of different additional portfolio.



This diagram shows a more realistic representation of how different additional portfolios would perform over different amounts of additional investment. This diagram shows that if the new portfolio weights are invested in a lot of stocks with small liquidity cap, then each marginal dollar invested curves the marginal return by applying a larger transaction cost.

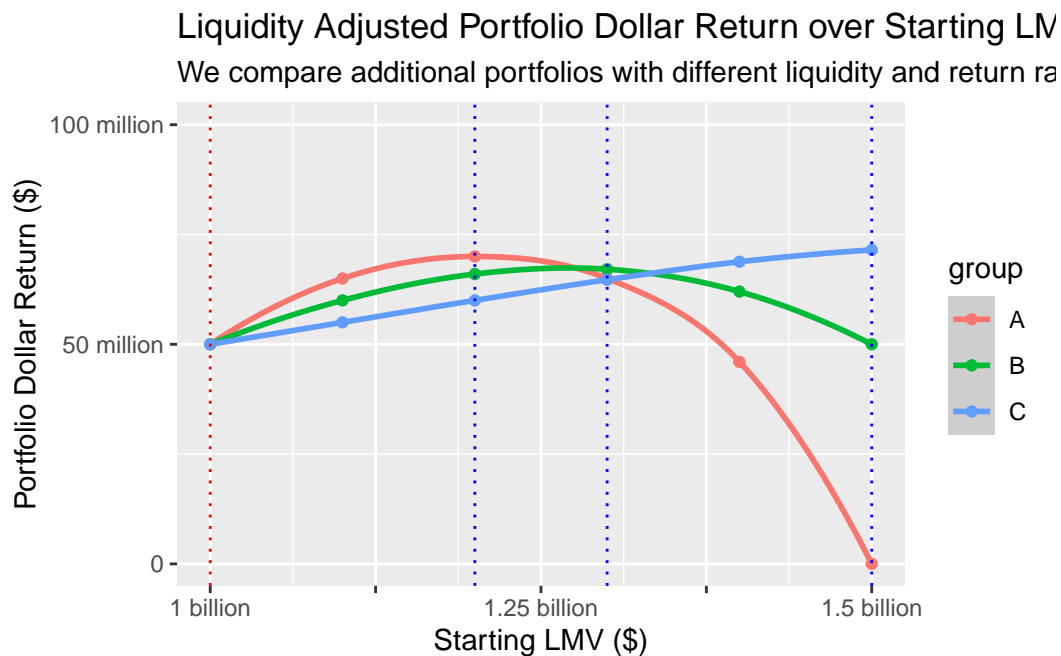
See for example portfolio B. The initial portfolio return rate is 2%, however with the exponential increase in transaction costs, the marginal return can become negative and eventually eat into the original portfolio returns.

It is important to note that the point at which a portfolio's marginal return curve hits the total investment line (blue dotted line) is the actual return of the portfolio. Therefore, although the marginal return of portfolio A is negative when hitting the total investment line, the total return from portfolio A is still positive.

In this diagram, the liquidity is consistent over the different additional portfolios, therefore it is easy to see which portfolio would perform the best. However, additional portfolios vary both in terms of liquidity and return rates. Therefore, we demonstrate a more difficult choice of additional portfolios.

For the next diagram, we compare three portfolios with different initial return rates and liquidity.

- Portfolio A: return rate of 15%, low liquidity stocks
- Portfolio B: return rate of 10%, mid liquidity stocks
- Portfolio C: return rate of 5%, high liquidity stocks



In this diagram, each portfolio has a different initial return rate and liquidity. Comparing the two extremes, Portfolio A has an initial return rate of 15% but invests in low liquidity stocks

and therefore is restricted by higher transaction costs while Portfolio C has an initial return rate of 5% but invests in high liquidity stocks and therefore transaction costs increase at a slow rate.

The optimal portfolio in this case really depends on how much is being invested into this portfolio. If an additional \$200 million are invested, Portfolio A is optimal. However, if an additional \$500 million are invested, Portfolio C is the best choice.

To summarize, we show that the true value of a portfolio depends on how much money is being invested. Additionally, while the dollar return from the initial return rate of a portfolio scales linearly with the dollars invested, the transaction cost scales exponentially due to liquidity. Therefore, the initial return rate of a portfolio at any point cannot solely determine which portfolio is the best option to invest in.