
Simple Portfolio Optimization That Works!

Magnus Erik Hvass Pedersen

October 14, 2021

Abstract

We first show that the common “mean-variance” portfolio method fails because variance is a horrible risk-measure for investing, and also because estimation errors may cause that method to concentrate the portfolio in losing assets that are highly correlated. We then present a new so-called “filter-diversify” method for portfolio optimization. The filtering process is trivial as it only allows assets into the portfolio if they have sufficiently high estimated returns. The diversification process is based on a new algorithm with several benefits: The algorithm is fairly simple. It allows both positive and negative portfolio weights. It is extremely fast and only takes a few milli-seconds to compute for a portfolio of 1000 assets. It is guaranteed to converge to the optimal solution. It is very robust to estimation errors, because it will only decrease the portfolio weights, so the worst that can happen is that it moves too much of the portfolio into cash (or another low-risk asset of your choice). We perform numerous tests of the new portfolio method on real-world stock-data from USA, and find that the new method performs extremely well on all performance metrics, and is very robust to estimation errors.

Table of Contents

1 Introduction.....	3
2 Mean-Variance Optimization.....	4
3 Variance is NOT Risk!.....	8
4 Naive Forecasting.....	20
5 Conditional Forecasting.....	24
6 Random Walks.....	29
7 Filtering Methods.....	35
8 Diversification Method.....	38
9 Test Settings.....	67
10 Test A – Full Data Period (Omniscient).....	72
11 Test B – Data Until 2010 (Omniscient).....	91
12 Test C – Data From 2010 (Omniscient).....	102
13 Test D – Noisy Returns (Robustness).....	113
14 Test E – Noisy Correlations (Robustness).....	125
15 Test F – Noisy Returns & Correlations (Robustness).....	140
16 Test G – Parameter Tuning (Omniscient).....	153
17 Future Research.....	160
18 Conclusion.....	161
19 Data & Computer Code.....	162
20 Bibliography.....	164

1 Introduction

In the 1950's a young man named Harry Markowitz invented a new method for optimizing investment portfolios of multiple assets, which is known as "mean-variance" portfolio optimization, because it maximizes the estimated mean return of the portfolio while simultaneously minimizing its variance. Nearly 40 years later in 1990, Markowitz was awarded the Nobel-prize in finance for his work. See [Markowitz 1952] and [Markowitz 1959] for the original paper and book by Markowitz. An easier and more concise description is found in [Luenberger 1998], and there are perhaps more recent descriptions that are even easier to understand.

The "mean-variance" method is still the standard portfolio method used in academia to this day, nearly 70 years after it was first invented. According to Google Scholar, since the beginning of the year 2020 until the time of this writing in early October 2021, there were more than 3000 papers published with the words "portfolio optimization" *and* Markowitz in the title or abstract. Unfortunately there are no good survey papers for getting a quick overview of the state-of-art research in that field. But sampling some of the papers from the most well-known researchers in the field, shows that they are still using the mean-variance method as their basic framework, and they still believe it is fundamentally sound.

In this paper we first show that the mean-variance method is inherently broken because variance is a horrible risk-measure for investing, and because the mean-variance method optimizes both the portfolio's mean and variance simultaneously, so it may concentrate the portfolio in losing assets that are highly correlated, if there are estimation errors in the mean returns and correlations for the assets.

We then present our new "filter-diversify" portfolio method which has two separate phases: First is the filtering process which is almost trivial, as it only allows assets into the portfolio if the estimated future returns are sufficiently high, and the portfolio weights are made higher when the estimated returns are higher. These portfolio weights are then fed into the diversification process, which is based on a new algorithm that we have dubbed "Hvass Diversification" for easy reference, and which has several advantages: The algorithm is fairly simple to describe and implement. It is guaranteed to converge to the optimal solution in just a few iterations. It has quadratic time-complexity so it is very fast and can diversify a portfolio with 1000 assets in just a few milli-seconds on a normal household computer. It greatly improves the portfolio on several performance metrics. And it only allows the portfolio weights to decrease, so the worst that can happen is that it moves too much of the portfolio into cash.

The new portfolio method is tested extensively on many thousands of portfolios of varying sizes, that are selected randomly from nearly 1000 U.S. stocks between the years 2007 and 2021. Although this paper is nearly 170 pages long, most of the pages actually consist of these tests with plots and analysis of the results. The portfolio method is first tested using the actual future 2-3 year average stock-returns and the actual future 10-day stock-correlations. We call this "omniscient" testing and it shows that the new portfolio method works extremely well when given correct input-data for the future stock-returns and correlations. We then add various kinds of heavy noise to the future stock-returns and correlations to test for robustness, and this shows the new portfolio method is very robust to estimation errors. The diversification algorithm can even handle completely malformed correlation matrices without harming the portfolio's performance, probably because it only allows the portfolio weights to decrease.

2 Mean-Variance Optimization

Because academia still considers the mean-variance method to be the standard way of optimizing investment portfolios, it is important to have a basic understanding of how it works. The academic literature is often very dense and mathematical, and that is perhaps why the mean-variance method is still believed to be correct after 70 years, because it is hard for people to connect the abstract mathematics to the real world. In this section we will therefore give a brief explanation of the mean-variance method that is hopefully a bit easier to understand, so we can see its flaws more clearly.

Say we want to invest in a portfolio of different assets, which can be stocks, bonds, currencies, real estate, etc. If we are only considering a few assets, we often denote them by letters such as Asset A and Asset B, or if we are considering many assets we may list them by numbers: Asset 1, Asset 2, etc.

The investment return on an asset is typically measured in percentages. For example, if we buy an asset for \$1000 and sell it for \$1200 we have made \$200 profit, which corresponds to a +20% gain because $\$1200/\$1000 - 1 = +20\%$. If instead we sold the asset for only \$800 we would have incurred a -20% loss because $\$800/\$1000 - 1 = -20\%$. It is sometimes useful to write the returns as “plus one”, so in the two examples above we would write 1.2 instead of +20% and 0.8 instead of -20%.

We are interested in forecasting or predicting the future asset returns, so we can invest in the one asset that has the highest future return. But our predictions about future returns are often inaccurate, and it is possible that we could experience losses instead of gains. If we cannot somehow improve our estimates of the future asset returns so as to make them more accurate, then we would instead like a method that combines multiple assets into a portfolio, so as to maximize the portfolio’s return while minimizing its risk. And that is exactly what the mean-variance method claims to do, as we will see in a moment.

But let us first explain a few basics about so-called random variables. Because the future asset returns are uncertain, we say that they are random variables. Let us denote the future return on Asset A as a random variable named $Return_A$ and sometimes just as R_A . In a simple example, we could have estimated that Asset A could have a future $Return_A$ of either +20% or -20%.

In the real world, asset returns are rarely so simple that they only have two possible outcomes. The outcomes are usually a continuous spectrum of possible returns with varying probabilities. This is called the probability distribution of the random variable. Instead of listing all possible outcomes and each of their probabilities, we usually summarize the probability distribution by only writing its centre-point and the dispersion around this centre-point. This summarizes the whole probability distribution into just two numbers. There are different ways of calculating these two numbers, and perhaps the most common way is to use the mean as the centre-point and the standard deviation as the dispersion.

For a so-called discrete probability distribution, there is only a finite number of possible values for the random variable. If they all have equal probability of occurring, then the mean of the random variable is simply the sum of all its possible values, divided by the number of possible values. This is also called the arithmetic mean and is often denoted $E[Return_A]$ for the random variable $Return_A$, or more shortly as μ_A (pronounced as *mu*).

For example, if the return on Asset A could be either -20%, -10%, 0%, +5%, +15%, or +25%, then its arithmetic mean would be calculated as follows:

$$\mu_A = E[Return_A] = \frac{1}{6} \cdot (-20\% - 10\% + 0\% + 5\% + 15\% + 25\%) = 2.5\% \quad (1)$$

If the possible outcomes have different probabilities of occurring, we would have to modify the above formula to weigh each possible return according to its probability. And for a continuous probability distribution we would have to calculate the arithmetic mean using an integral instead.

Now that we have a centre-point for the probability distribution, let us define a measure of its spread or dispersion around this centre-point. A common measure of dispersion is the so-called variance, which is defined mathematically as follows:

$$Var[Return_A] = E[(Return_A - \mu_A)^2] \quad (2)$$

This formula is very important for the mean-variance portfolio method, because it uses the variance as its risk-measure. It is therefore important that you understand what this formula really means. The inner-part of the formula $(Return_A - \mu_A)^2$ is basically just the difference between each possible return on Asset A minus their average return, and then squared to get a positive number. Then we take the average of all those squared differences, and that is the variance of the random variable $Return_A$. So the variance is a measure of how far from the average all the possible return values are.

The reason that we are calculating the squared differences instead of just taking the absolute value (i.e. removing the sign of the difference), is that it is mathematically convenient in other regards. But it also means that the variance is no longer on the same scale as the original numbers. This is easily corrected by taking the square-root of the variance, which gives us the so-called standard deviation:

$$Std[Return_A] = \sqrt{Var[Return_A]} = \sqrt{E[(Return_A - \mu_A)^2]} \quad (3)$$

We often denote the standard deviation as σ (pronounced *sigma*) and the variance as σ^2 (*sigma squared*).

Consider again the example above where $Return_A$ could be either -20%, -10%, 0%, 5%, 15%, or 25%, all with equal probability, so its mean was $\mu_A = 2.5\%$. We can then calculate the variance as follows:

$$\begin{aligned} \sigma_A^2 &= Var[Return_A] = \frac{1}{6} \cdot ((-20\% - 2.5\%)^2 + (-10\% - 2.5\%)^2 + (0\% - 2.5\%)^2 \\ &\quad + (5\% - 2.5\%)^2 + (15\% - 2.5\%)^2 + (25\% - 2.5\%)^2) \approx 0.02229 \end{aligned} \quad (4)$$

The standard deviation is then the square-root of the variance:

$$\sigma_A = Std[Return_A] = \sqrt{Var[Return_A]} \approx \sqrt{0.02229} \approx 0.1493 \approx 14.93\% \quad (5)$$

Now consider Asset B whose return can be either -8.75%, -3.75%, 1.25%, 3.75%, 8.75%, or 13.75%. The mean is the exact same as for Asset A, namely $\mu_B = \mu_A = 2.5\%$, but now the standard deviation is only $\sigma_B \approx 7.465\%$ which is half that of $\sigma_A \approx 14.93\%$. So Asset B has the same mean as Asset A, but only half the spread of possible return outcomes. So which one is better? This question is central for the mean-variance method, which seeks to maximize the portfolio's mean while minimizing the variance (or equivalently the standard deviation). The entire next Section 3 is dedicated to answering this question. But let us first explain some more relevant formulas for use with entire portfolios of assets.

Let $Weight_i$ denote the portfolio's weight for asset number i . We typically assume that all the portfolio weights sum to 1, so our entire capital is invested in different assets. If the weights sum to more than 1, then we would be investing for borrowed money, and if the weights sum to less than 1, then some of the portfolio would be held in cash (or other low-risk investments such as a short-term bond fund).

The return on the portfolio is then another random variable denoted $Return_p$, which is just defined as the weighted sum of the returns on the N individual assets in the portfolio:

$$Return_p = \sum_{i=1}^N Weight_i \cdot Return_i \quad (6)$$

The portfolio's mean return μ_p is very easy to calculate, as it is just the weighted sum of the mean returns for the individual assets, because of how the arithmetic mean is defined mathematically:

$$\mu_p = E[Return_p] = \sum_{i=1}^N Weight_i \cdot E[Return_i] = \sum_{i=1}^N Weight_i \cdot \mu_i \quad (7)$$

But the portfolio's variance is more complicated to calculate, because the definition of variance from Eq. (2) contains a non-linearity in the form of the squared differences. Although it is not too difficult to derive the variance for a portfolio of only two assets, it is a slightly long derivation so it has been omitted here for brevity, but it would be a very useful exercise for you to do, so you understand why the formula is defined this way. For a portfolio of N assets, the variance of their weighted sum is:

$$\sigma_p^2 = Var[Return_p] = \sum_{i=1}^N \sum_{j=1}^N Weight_i \cdot Weight_j \cdot Cov[Return_i, Return_j] \quad (8)$$

The standard deviation of the portfolio's return is simply the square-root of the variance:

$$\sigma_p = Std[Return_p] = \sqrt{Var[Return_p]} \quad (9)$$

The so-called covariance in Eq. (8) is defined as follows:

$$Cov[Return_i, Return_j] = E[(Return_i - E[Return_i]) \cdot (Return_j - E[Return_j])] \quad (10)$$

So the covariance is defined quite similarly to the variance in Eq. (2), but it uses the returns for two different assets instead of just one. A special case of the covariance formula is when we use it with a single asset, in which case it equals the variance $Cov[Return_i, Return_i] = Var[Return_i]$.

Like the variance, the covariance also has a strange scale and this is difficult to interpret by a human. So we will often use the so-called correlation instead of the covariance. It basically just normalizes the covariance so it ranges between -1 and 1 using this formula:

$$Corr[Return_i, Return_j] = \frac{Cov[Return_i, Return_j]}{Std[Return_i] \cdot Std[Return_j]} \quad (11)$$

The advantage of using the correlation is that it is always a value between -1 and 1, with 1 meaning that the two random variables always move in the same direction (but they may not have the same values), and a correlation of -1 means that the two random variables always move in opposite direction to each other. Note that an asset always has correlation 1 with itself $Corr[Return_i, Return_i] = 1$. And the correlation is symmetrical so that $Corr[Return_i, Return_j] = Corr[Return_j, Return_i]$. If the correlation is zero, then the two random variables are said to be uncorrelated.

Using the correlation instead of the covariance in Eq. (8), and denoting the correlation as $\rho_{i,j} = \text{Corr}[Return_i, Return_j]$ (pronounced *rho*), and using the shorter notation for standard deviation $\sigma_i = \text{Std}[Return_i]$ gives the following formula for the variance of a portfolio of weighted assets:

$$\sigma_p^2 = \text{Var}[Return_p] = \sum_{i=1}^N \sum_{j=1}^N \text{Weight}_i \cdot \text{Weight}_j \cdot \sigma_i \cdot \sigma_j \cdot \rho_{i,j} \quad (12)$$

The inner-part of this formula is $\text{Weight}_i \cdot \text{Weight}_j \cdot \sigma_i \cdot \sigma_j \cdot \rho_{i,j}$ which is just a product of the portfolio weights for the two assets, their standard deviations, and their correlation. And Eq. (12) then calculates the variance of the portfolio by summing over all the possible combinations of assets in the portfolio.

Let us now give an example of how to calculate the variance of a portfolio using Eq. (12). Consider the same example as before in this section, with two assets A and B with equal means $\mu_A = \mu_B = 2.5\%$, and the standard deviations $\sigma_A \approx 14.93\%$ and $\sigma_B \approx 7.465\%$. Let us say that the portfolio weights are $\text{Weight}_A = 0.4$ and $\text{Weight}_B = 0.6$ so they sum to 1. And let us say that the asset returns are positively correlated with a coefficient of 0.5. Because there are only two assets A and B, there are four possible pairs of assets in the summation in Eq. (12): Assets A & B, Assets B & A, Assets A & A, and finally Assets B & B. Let us calculate the inner-part of Eq. (12) for these four asset pairs:

$$\begin{aligned} A,B: \text{Weight}_A \cdot \text{Weight}_B \cdot \sigma_A \cdot \sigma_B \cdot \rho_{A,B} &= 0.4 \cdot 0.6 \cdot 14.93\% \cdot 7.465\% \cdot 0.5 \approx 0.001337 \\ B,A: \text{Weight}_B \cdot \text{Weight}_A \cdot \sigma_B \cdot \sigma_A \cdot \rho_{B,A} &= 0.6 \cdot 0.4 \cdot 7.465\% \cdot 14.93\% \cdot 0.5 \approx 0.001337 \\ A,A: \text{Weight}_A \cdot \text{Weight}_A \cdot \sigma_A \cdot \sigma_A \cdot \rho_{A,A} &= 0.4 \cdot 0.4 \cdot 14.93\% \cdot 14.93\% \cdot 1 \approx 0.003566 \\ B,B: \text{Weight}_B \cdot \text{Weight}_B \cdot \sigma_B \cdot \sigma_B \cdot \rho_{B,B} &= 0.6 \cdot 0.6 \cdot 7.465\% \cdot 7.465\% \cdot 1 \approx 0.002006 \end{aligned} \quad (13)$$

We then sum these four intermediate calculations to get the variance of the portfolio using Eq. (12):

$$\sigma_p^2 = \text{Var}[Return_p] \approx 0.001337 + 0.001337 + 0.003566 + 0.002006 \approx 0.008246 \quad (14)$$

And finally we take the square-root to obtain the standard deviation of the portfolio's return:

$$\sigma_p = \text{Std}[Return_p] = \sqrt{\text{Var}[Return_p]} \approx \sqrt{0.008246} \approx 9.1\% \quad (15)$$

If instead the portfolio weights were $\text{Weight}_A = 0.2$ and $\text{Weight}_B = 0.8$, and we go through all the same calculations as above, then we would get a standard deviation of only 7.9% for the portfolio's return.

The goal of the mean-variance method is to find the portfolio weights that maximize the mean while also minimizing the variance (or standard deviation) of the portfolio's returns. Because these two objectives may be in conflict, there is no single choice of portfolio weights that is optimal on both objectives, and we instead get a so-called Efficient or Pareto Frontier of mutually optimal compromises between these two objectives. The proponents of the mean-variance method claim that this maximizes the portfolio's return while also minimizing the portfolio's risk.

When I first studied the mean-variance portfolio method, it puzzled me that it was using the variance as a risk-measure. If you come from a school of thought where investment risk has to do with the future prospects of a company, a stock's valuation ratio, etc. it seemed very odd to define risk from the spread or dispersion of the return distribution, because that only measures the degree of uncertainty we have about the future returns, and not whether some of those could be big losses. This criticism is formalized in the next section with numerous examples that show the variance is in fact a very bad risk-measure.

3 Variance is NOT Risk!

A dictionary typically defines the word “*risk*” as “*the possibility or chance of injury or harm.*” The mean-variance method from the previous section is based on the notion that variance is a good risk measure for investing, so according to the dictionary definition of risk, the variance should measure the probability and magnitude of possible losses on an investment. But variance is defined mathematically to measure the spread of a distribution of random outcomes, and not specifically if some of those outcomes are negative or positive. So the question is whether the variance somehow implicitly measures the probability and magnitude of loss. In this section we will investigate that question.

It is nearly 70 years ago that Markowitz proposed his mean-variance portfolio method, and it is more than 30 years ago that he was awarded the Nobel prize for that work. So academia has had plenty of time to test whether the mean-variance method is correct. As you will see in this section, it is actually very easy to find counter-examples where the mean-variance method fails completely, but strangely enough, academia has never come up with any substantial criticism of the method, so it is still commonly believed to be correct, and it is the standard portfolio method used in academia to this day.

My own criticism of the mean-variance method goes back to my paper from the year 2014 [Pedersen 2014] (page 16), as well as several short videos.¹ This section assembles and extends that criticism.

In these examples, we will use the standard deviation instead of the variance, but they are equivalent when used as a risk measure, because the variance is just the squared standard deviation.

It is often unclear from the academic literature, how the future mean and variance for the asset returns should be estimated – often it is implicitly assumed that the mean and variance of the past stock-returns are also good predictors for the future. We will dispel of this myth in Section 4 and generally use “omniscient” predictions about the future asset returns in this paper. In this section, it means that we will assume all the return distributions are completely accurate for the future asset returns. Even under such ideal conditions, we will see that the mean-variance method fails completely!

It is also often unclear in the academic literature, whether the mean and variance is for e.g. daily, weekly, monthly or annual returns, and whether these are assumed to be the same values for each time-step, so asset-prices evolve as a completely “random walk”. We will dispel of this myth in Sections 5 and 6. But in the counter-examples here, it does not matter whether the investment periods are daily, weekly, monthly or annual, and it also does not matter if a single or multiple time-steps are considered. The main problem is that variance is a horrible risk-measure for investing, regardless of these aspects.

In my previous counter-examples I often used simple and discrete return distributions, because I thought it made it easier to understand the underlying problems with the mean-variance method. But this has caused some skepticism that perhaps my counter-examples were too simple and contrived, and thus not valid in general. So in this section we will use normal distributions which are continuous and have the typical bell-shaped curves. But it is actually not important which random distribution you use in these counter-examples, because the problem is ultimately that variance is a horrible risk-measure for investing – and this problem is actually magnified by the way the mean-variance method works.

¹ <https://youtu.be/wr8NzThfpAE> – <https://youtu.be/0temN7hAf2c> – <https://youtu.be/DzTlH6ipx98>

3.1 A-B Comparison

In this example, we will test whether the standard deviation is a good measure for the probability that one asset will perform worse than another, and whether the asset returns will be losses.

Figure 1 shows the return distributions for Asset A (the blue bell-curve) and Asset B (the yellow bell-curve). In the top-plot they have the same mean $\mu_A = \mu_B = 10\%$ but different standard deviations $\sigma_A = 4\%$ and $\sigma_B = 8\%$, so the centres of the two return distributions are identical, but Asset B has a much wider spread of possible return outcomes than Asset A. If the two assets are statistically independent, then the probability is 0.5 (or 50%) that Asset A's return is less than the return of Asset B. But it is possible for Asset B to have either higher or lower returns than Asset A, because its standard deviation is twice as high, so the spread of possible return outcomes is much wider. This is the only scenario where it makes sense to say that the standard deviation measures risk in any meaningful way, because if you only have the choice of investing in either Asset A or Asset B, then in order to get the highest possible return, you would have to invest in Asset B, but this would also carry the risk of a much greater loss. The probability of loss is 0.11 (or 11%) for Asset B while it is only 0.01 (or 1%) for Asset A, and because the return distributions are bell-shaped, the possible losses on Asset B would also be much greater than the possible losses on Asset A. So in this example there is a clear relation between the standard deviation and the probability and magnitude of possible losses, but as we will see shortly, this is not generally true. It only makes sense to use the standard deviation as a risk-measure when the two assets have the same mean return.

In the middle plot in Figure 1, the two assets have the same standard deviations as in the top-plot, namely $\sigma_A = 4\%$ and $\sigma_B = 8\%$, but now they have different means $\mu_A = 10\%$ and $\mu_B = 20\%$. They now have exactly the same probability of loss around 0.0062, but the probability is 0.87 that the return on Asset A is less than the return on Asset B. This is a very simple example that the standard deviation does not measure investment risk in any meaningful way, because the standard deviation of Asset A is only half that of Asset B, but the probability is 0.87 so almost 9 out of 10 times it will result in Asset A under-performing Asset B, so it would be absurd to say that Asset A is “less risky” than Asset B!

In the bottom plot in Figure 1, the two assets still have the same standard deviations as in the previous two plots, namely $\sigma_A = 4\%$ and $\sigma_B = 8\%$, but the mean return on Asset A is now negative at $\mu_A = -10\%$ while the mean return on Asset B is positive $\mu_B = 20\%$. The probability of loss is 0.99 for Asset A while it is only 0.01 for Asset B, and the probability that Asset A under-performs Asset B is 0.9996 (which is rounded up to 1.0 in the bottom plot’s title). So Asset A will nearly always have losses and under-perform Asset B. But if you think that the standard deviation measures investment risk, then you would say that Asset A was much less risky than Asset B, because the standard deviation of Asset A is only half the standard deviation of Asset B. Clearly it is a ludicrous claim that the standard deviation (and equivalently the variance) measures investment risk in any meaningful way!

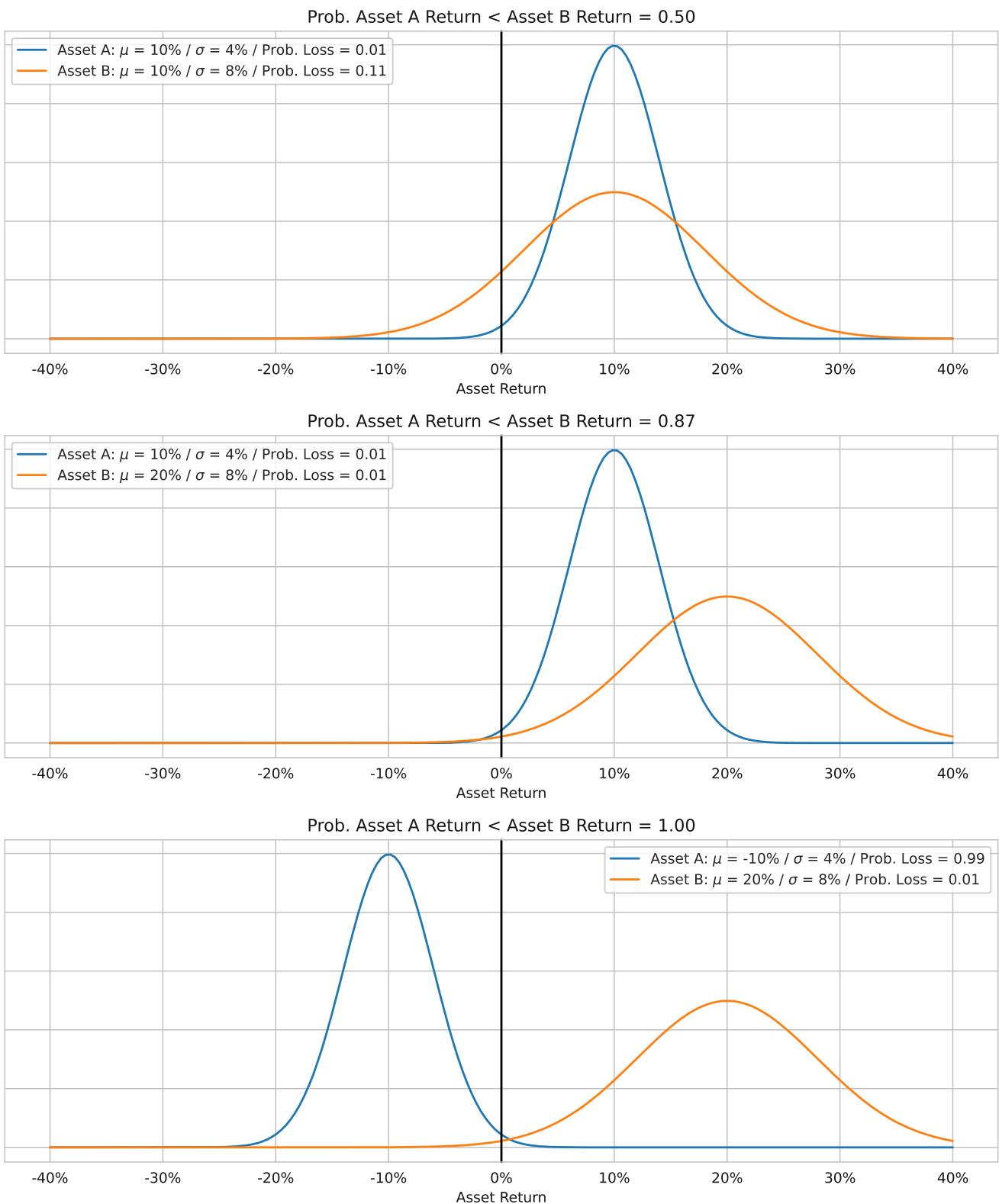


Figure 1: Compare the normal return distributions of two assets.

3.2 Probability of Loss

Let us further investigate the relation between the standard deviation and the probability of loss.

Figure 2 shows four different cases where the mean return is always negative $\mu = -10\%$ but there are four different choices of standard deviation σ which is either 1%, 5%, 10% or 20%. It is obvious from these plots that the standard deviation measures the spread of the distribution, because that is how it was mathematically defined in Eq (3). But notice the relation between the standard deviation and the probability of loss: When $\sigma=1\%$ the probability of loss is 1.0 because all of the return distribution is negative. When $\sigma=5\%$ a small part of the return distribution is now positive so the probability of loss is slightly lower at 0.98. When $\sigma=10\%$ a bit more of the return distribution is positive so the probability of loss is now 0.84. And finally when $\sigma=20\%$ a significant part of the return distribution is positive, so the probability of loss is now only 0.69. Notice the pattern: As the standard deviation increases, the probability of loss goes down. If the standard deviation was a good risk-measure, we would expect to see the opposite. The reason is that the mean return is negative, so return distributions that have a higher standard deviation have a higher spread of the possible return outcomes, and therefore a higher chance of having positive returns, and therefore a lower probability of loss.

Figure 3 shows the opposite four cases where the mean return is now always positive $\mu = 10\%$, and the standard deviation σ is again either 1%, 5%, 10% or 20%. Now the probability of loss does increase with increasing standard deviation: When $\sigma=1\%$ the probability of loss is 0.0. When $\sigma=5\%$ a small part of the return distribution is now negative so the probability of loss is 0.02. When $\sigma=10\%$ a bit more of the return distribution is negative so the probability of loss is now 0.16. And finally when $\sigma=20\%$ a significant part of the return distribution is negative, so the probability of loss is now 0.31. So when the mean return is positive, the standard deviation is directly related to the probability and magnitude of possible losses, because higher standard deviations will cause more of the return distribution to become negative.

Figure 4 is another way of showing this. It shows the probability of loss for normal return distributions that have negative mean return $\mu = -10\%$ (blue curve) and positive mean return $\mu = 10\%$ (yellow curve), when varying the standard deviation σ between 0% and 100%. For negative mean return, the blue curve is decreasing as the standard deviation increases, and vice versa for positive mean return, where the yellow curve is increasing as the standard deviation increases.

So if we require our risk-measure to be directly related to the probability of loss, as a dictionary definition of the word “risk” suggests, then we cannot use the standard deviation (or equivalently the variance), because that is only directly related to the probability of loss when the mean return is positive. When the mean return is negative, the standard deviation is *inversely* related to the probability of loss. It is strange that finance professors have apparently never noticed this problem, and it makes you wonder if it is because they think that return distributions can only have positive means.

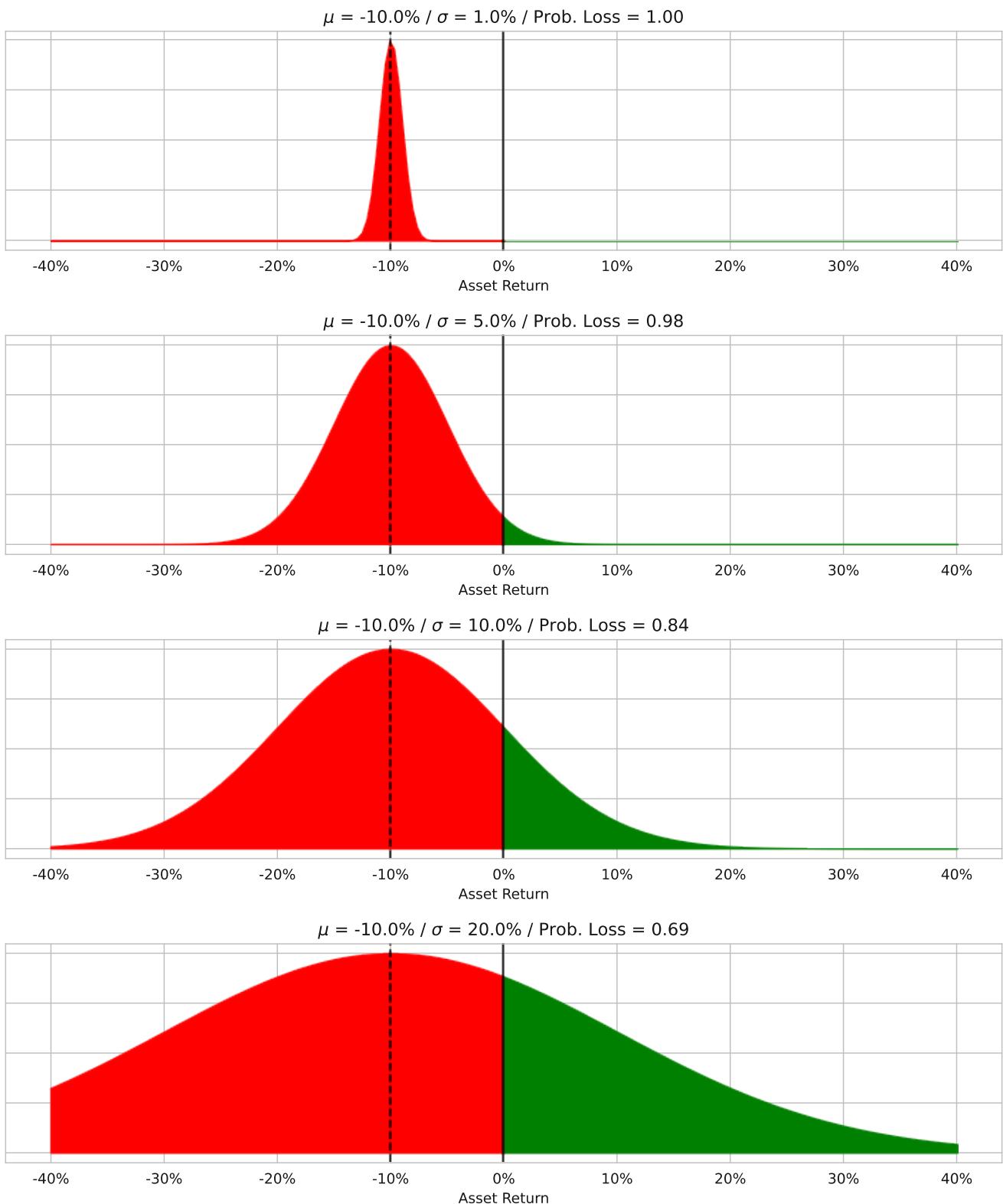


Figure 2: Compare the probability of loss for normal return distributions with **negative mean** and different standard deviations.

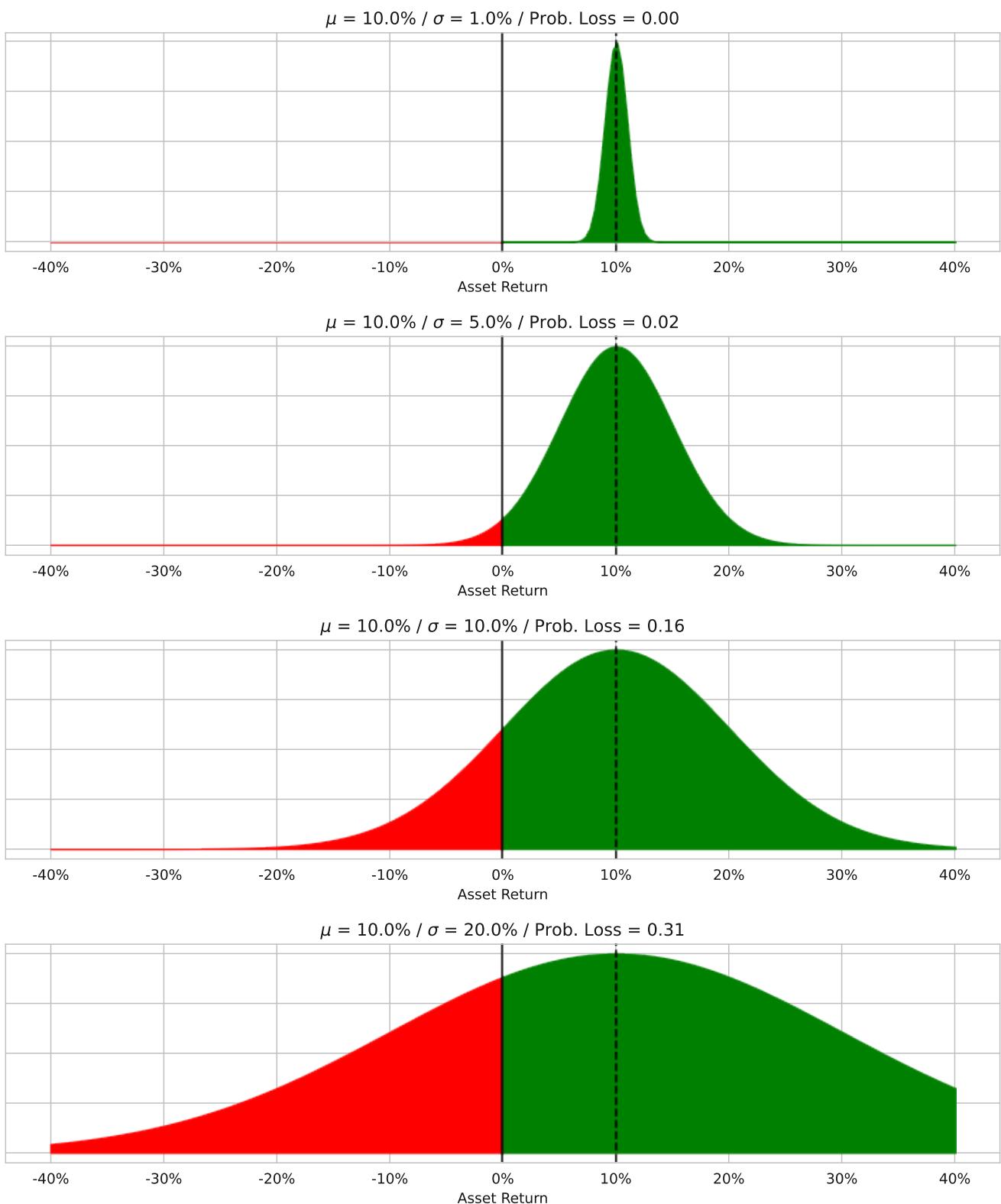


Figure 3: Compare the probability of loss for normal return distributions with **positive mean** and different standard deviations.

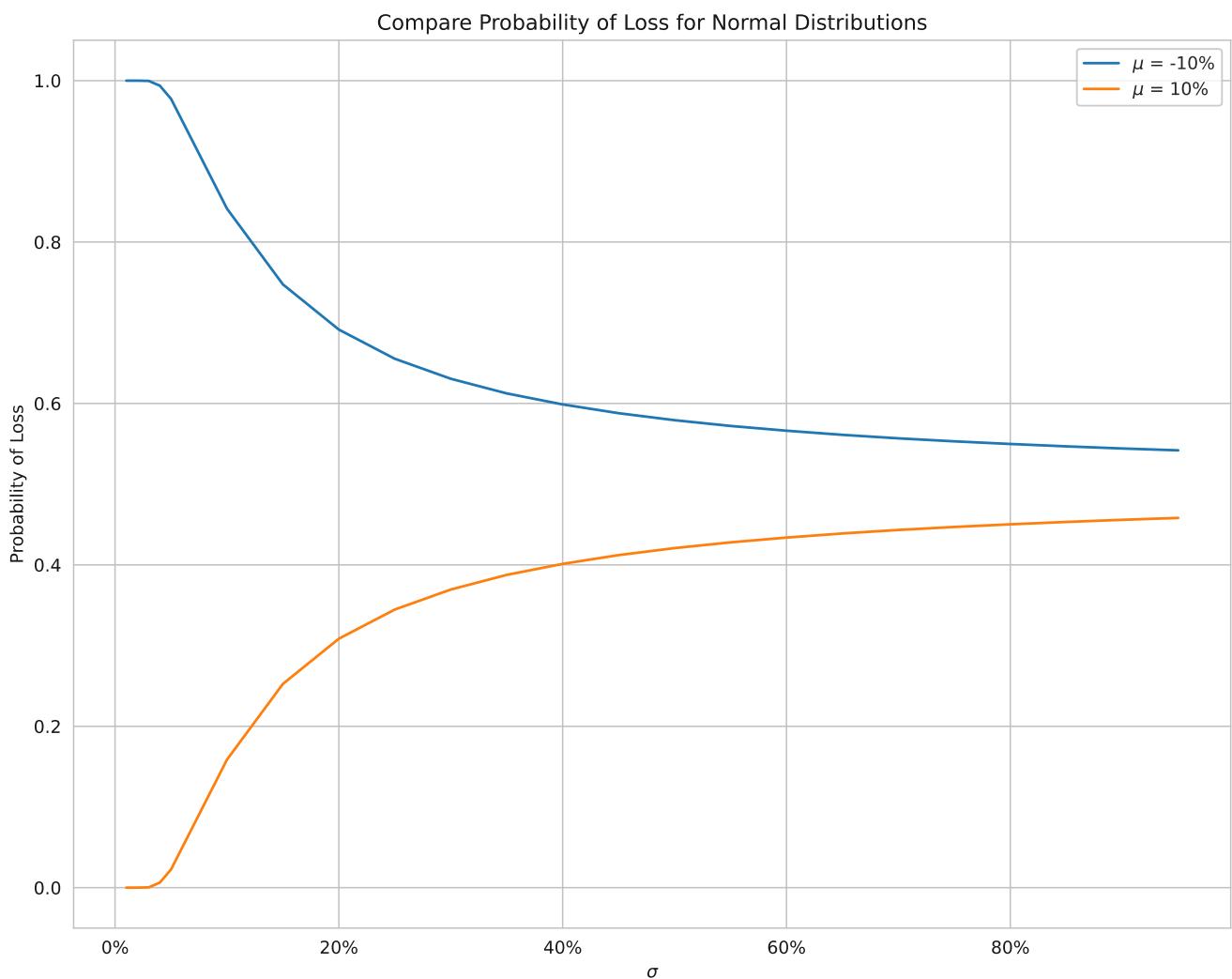


Figure 4: Compare the probability of loss for normal return distributions with **negative mean (blue)** and **positive mean (yellow)** and different standard deviations.

3.3 “Minimum Risk” Portfolio

We have just seen that for individual assets, the standard deviation is a very bad risk measure. What about using the standard deviation (or variance) as a risk-measure for portfolios of multiple assets?

Figure 5 shows an example where two assets A and B are combined into portfolios using different weights. The mean returns for the two assets are $\mu_A = -10\%$ and $\mu_B = 20\%$, the standard deviations are $\sigma_A = 2\%$ and $\sigma_B = 5\%$, and the correlation between the two assets is $\rho_{A,B} = -0.5$ so they tend to move in opposite direction. The top-plot shows the two assets as black dots, and the curve between them shows all possible weighted combinations of the two assets when the weights are positive and sum to 1. The green part of the curve is when the portfolio’s mean return is positive, and the red part is when the portfolio’s mean return is negative. As the portfolio only contains two assets, this curve is also the Pareto or Efficient Frontier, because it represents the combinations of the two assets that are optimal in regards to both the mean and standard deviation for the portfolio return. For portfolios with more than two assets, the Efficient Frontier is much more difficult to find, and this was one of the major challenges in developing the mean-variance portfolio method.

The dashed, black vertical line in the top-plot of Figure 5 is where the standard deviation is at its lowest point. Note how this is even lower than the standard deviations for the two individual assets, which is due to the assets having a negative correlation $\rho_{A,B} = -0.5$ so they tend to move in opposite directions. This means we can achieve a lower standard deviation for the portfolio when combining the two assets, compared to just investing in either asset alone. This is the core idea behind the mean-variance portfolio method. The goal is to maximize the mean return while minimizing the variance (or standard deviation). This is done in the belief that variance measures investment risk, so the portfolio that has the lowest variance is therefore called the “Minimum Risk” portfolio. In this example it is when the weight for Asset A is 0.77 and the weight for Asset B is 0.23, which gives a portfolio mean return of $\mu_p = -3.1\%$, calculated from Eq. (7) as follows:

$$\begin{aligned}\mu_p &= E[Return_p] = Weight_A \cdot E[Return_A] + Weight_B \cdot E[Return_B] \\ &= 0.77 \cdot (-10\%) + 0.23 \cdot 20\% = -3.1\%\end{aligned}\tag{16}$$

The portfolio’s standard deviation is $\sigma_p = 1.4\%$, which is calculated similarly to Eq. (13)-(15).

The second plot in Figure 5 shows the return distribution of this so-called “Minimum Risk” portfolio with $\mu_p = -3.1\%$ and $\sigma_p = 1.4\%$, where most of the return distribution is negative and the probability of loss is about 0.99. So the “Minimum Risk” portfolio is almost guaranteed to result in an investment loss! But the third and fourth plots in Figure 5 show that it is only Asset A that has a negative return with mean -10%, while Asset B always has a large, positive return with mean 20%. Obviously Asset B would be a much better and less risky investment than the “Minimum Risk” portfolio which combines both Asset A and Asset B. The reason is of course that the variance (or standard deviation) is a horrible risk-measure for investing, as we also showed in the previous sections. The variance measures the spread of the return distribution, not whether the returns are losses, and how big those losses are. So when we minimize the variance (or standard deviation), we are minimizing the spread of the return distribution, and not the probability and magnitude of potential losses!

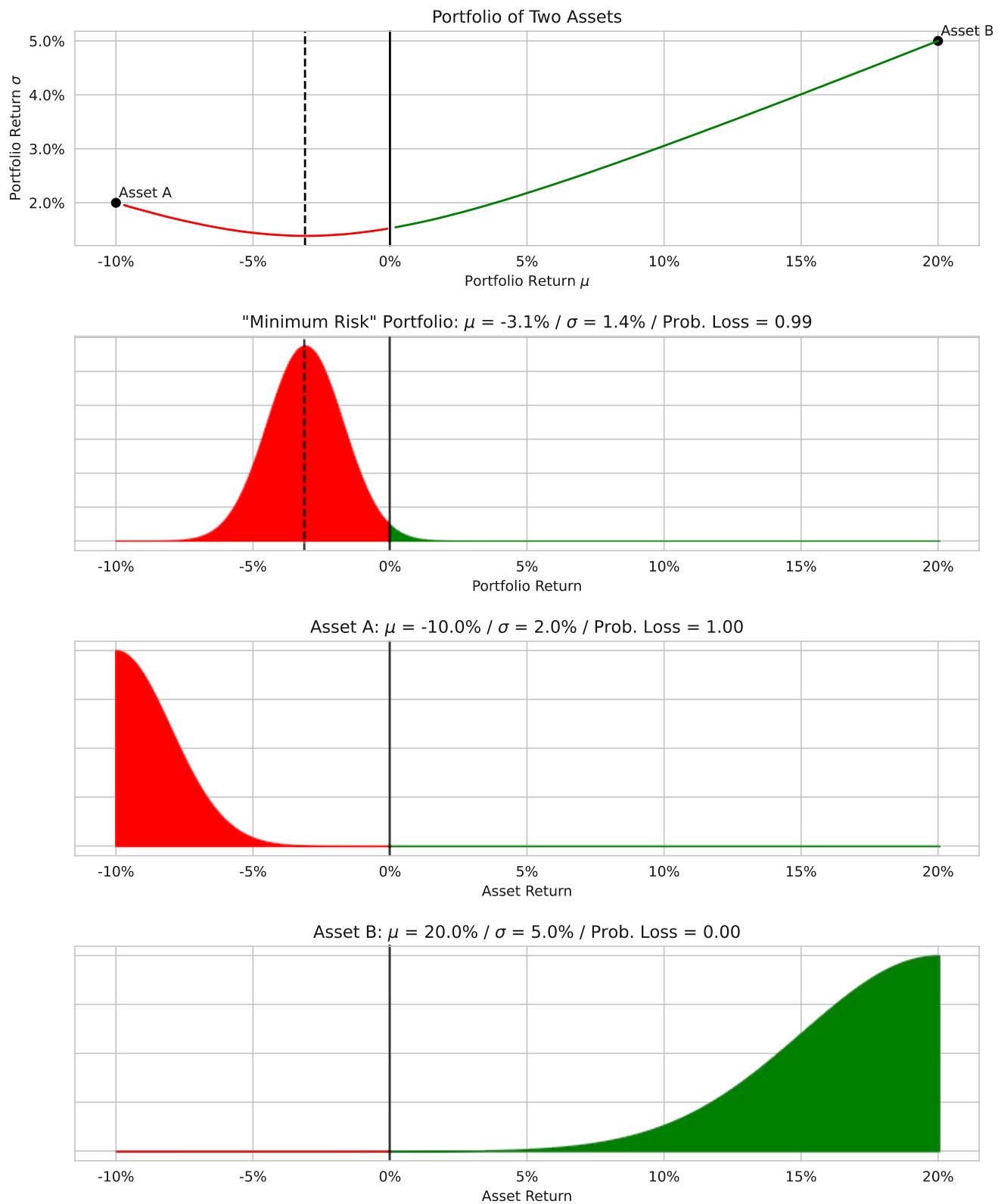


Figure 5: (1) The Efficient / Pareto Front for two assets. (2) Return distribution for the "Minimum Risk" portfolio. (3) Return distribution for Asset A. (4) Return distribution for Asset B.

3.4 Estimation Errors

In the previous examples, we assumed that we had “omniscient” knowledge about the future, so our estimates for the future return distributions were completely correct. Even under such ideal conditions, we saw that the standard deviation (or variance) was actually a horrible risk measure for investing. But what if there were also estimation errors in the future return distributions and their correlations, would that magnify the problems in using the standard deviation as a risk measure?

The mean-variance method briefly described in Section 2 works by increasing the weight of an asset in the portfolio, if that will either increase the overall portfolio’s mean return, or if it will decrease the portfolio’s variance; and if this does not worsen the portfolio’s performance on either objective. This is called a Pareto-optimal or an Efficient portfolio, because there is no other possible choice of asset weights that can make the portfolio perform better on both objectives simultaneously. As we saw in Section 3.3, this may cause the mean-variance method to include assets in the portfolio that are virtually guaranteed to result in losses, merely because those assets have negative correlations with other assets in the portfolio, which therefore lowers the portfolio’s overall variance.

Now assume that some Asset X is included in the portfolio because it is estimated to have a decent mean return as well as negative correlations with other assets in the portfolio, so it can lower the portfolio’s variance without harming the portfolio’s mean return. Then imagine that those estimates about the future returns are actually incorrect, and in fact Asset X has a negative mean return and positive correlations with the other assets. Then Asset X will most likely result in a loss and it is highly correlated with the other assets in the portfolio, so they all experience losses at the same time.

Because the mean-variance method optimizes the two objectives simultaneously, if we make estimation errors in the future mean returns and correlations, the mean-variance method may concentrate the entire portfolio in assets that are likely to result in losses *and* are highly correlated with each other, so all the portfolio’s assets experience losses at the same time. Thus the mean-variance portfolio method is actually a “double whammy” of risk-taking, which is ironic because the method was intended to lower the portfolio’s overall risk while maximizing its mean return.

The two main problems with the mean-variance method is that it uses variance as a risk-measure, and it optimizes the two objectives simultaneously. But even if we were to use another risk-measure, we would likely have similar problems if we still optimize both objectives simultaneously. For example, if we use the probability of loss as risk measure, and we make estimation errors so that we thought Asset X had a low probability of loss and negative correlations with other assets in the portfolio, when in fact it had a very high probability of loss and positive correlations with other assets, then we would again have a “double whammy” of risk-taking, because the portfolio would be concentrated in assets with high probability of loss that were also highly correlated.

The portfolio method presented in this paper does not explicitly use a risk-measure, and it separates the optimization of the two objectives into two phases: First it maximizes the mean return, and then it minimizes the correlation between the portfolio’s assets – and this is done in a way that is much more robust to estimation errors than the mean-variance method, because the process of minimizing the correlations between assets is only allowed to lower the portfolio weights and not increase them.

3.5 Return Formulas

It was shown in [Pedersen 2020] how to decompose the stock return into its three basic components: The Dividend Yield, the change in Earnings or Sales Per Share, and the change in the P/E or P/Sales ratio. If you know the exact future change of these three components, then you can predict the future stock-return with complete accuracy. Usually we cannot predict the exact future values for these three components, but we can sometimes predict a reasonable range of possible future values, so we can instead consider the mean and standard deviation for the future stock-returns.

The following is the formula for calculating the mean annualized return when using the current P/Sales ratio as the predictor variable. The value a is calculated from a range of your best guesses for the future P/Sales ratio, the future growth in Sales Per Share, and the future Dividend Yield for the given number of investment years, so the forecasting formula is:

$$E[\text{Ann Return}] = \frac{a}{P/\text{Sales}^{1/\text{Years}}} - 1 \quad (17)$$

The following is the formula for calculating the standard deviation for the future annualized return. The value b is again calculated from a range of your best guesses for the future P/Sales ratio, the future growth in Sales Per Share, and the future Dividend Yield for the given number of investment years:

$$\text{Std}[\text{Ann Return}] = \frac{b}{P/\text{Sales}^{1/\text{Years}}} \quad (18)$$

It is not important for this brief discussion exactly how the values a and b are defined, as it is explained in much more detail in [Pedersen 2020]. Note that we could also have denoted these formulas as μ instead of $E[\text{Ann Return}]$ and σ instead of $\text{Std}[\text{Ann Return}]$ to use the shorter notation in this paper. The meaning is the same, namely that the mean indicates the centre of the return distribution and the standard deviation measures the spread of the distribution.

Because the two formulas above are derived from the mathematical definition of annualized return, they are actually valid for all investment periods, whether it is a single day, a whole year, or perhaps even 10 years. But for short periods, the formulas are effectively reduced to just predicting the change in share-price, because there is typically no change in the Sales Per Share from one day to the next. So the above formulas are best used for longer investment periods of several years or more, where we can split the stock-return into these three components, that are sometimes more easy to predict individually.

Let us now consider how the two formulas above behave in their extremes. First assume the values a and b remain fixed, so the future range of P/Sales ratios, growth in Sales Per Share, and Dividend Yield do not change. If the current P/Sales ratio approaches zero, then both the mean and standard deviation for the future annualized return both approach infinity. That is, when the current P/Sales ratio becomes very low, the future return becomes very high, and both the mean and standard deviation for the annualized return becomes very large.

In the other extreme, we again assume that the values a and b remain fixed, so the future range of P/Sales ratios, growth in Sales Per Share, and Dividend Yield do not change. But now the current P/Sales ratio is assumed to be very large and approaching infinity. Then both the mean and standard deviation for the future annualized return will approach zero, so the stock-return will be nearly a complete loss and with very low standard deviation.

This again shows that the standard deviation is a horrible risk measure, because it is inversely proportional to the probability and magnitude of loss. According to the formulas above, when the future stock-return is very large, the standard deviation is also very large, and conversely, when the future stock-return is nearly a complete loss, the standard deviation is very small and approaches zero. This is the opposite of how the standard deviation should behave if it was a good risk measure for investing.

3.6 Summary

In this section we saw numerous examples that the variance (or standard deviation) is a horrible risk measure for investing. The variance often measures the exact opposite of what we are interested in. The variance does not measure if one asset is likely to perform better or worse than another (Section 3.1). The variance does not measure the probability and magnitude of losses (Section 3.2). And the "Minimum Risk" portfolio found with the mean-variance method, can result in losses for all possible outcomes, even though some of the individual assets may have only positive returns for all outcomes (Section 3.3). All of these problems exist for "omniscient" return distributions where we know the future return distributions with complete accuracy. In practice we will also make estimation errors when trying to predict the future mean, variance and correlation of asset returns, and this may cause the mean-variance method to concentrate the portfolio in losing assets that are highly correlated, so in reality the mean-variance method is a "double-whammy" of risk-taking (Section 3.4).

The mean-variance method is horrible at optimizing investment portfolios, and it is absurd that after it has existed for 70 years, academia has not only failed at properly analyzing the method and rectifying its flaws – instead they have awarded its author with the highest academic accolades and prizes.

If you are still not convinced after having read this section, and you insist the mean-variance method is a robust investment tool, then please contact my trusted business partner [Dr. Augustus Kwembe](#) as we would love to offer you some "zero risk" investments!

4 Naive Forecasting

In the academic research literature, the performance of portfolio methods are often tested using “naive” forecasts of the future stock-returns and their correlations, where the recent past is merely assumed to continue into the future. We now show that such naive forecasts are highly inaccurate and makes it impossible to determine which portfolio method actually works best, because some methods may be more robust than others to noisy estimates of the future stock-returns and their correlations.

Figure 6 shows the mean daily returns for three stocks. The top-plot shows it for AAPL, the middle-plot shows it for BBBY, and the bottom-plot shows it for FL. In each of these sub-plots, the mean daily returns are shown for rolling periods of 20, 60 and 250 days, which roughly correspond to 1, 3 and 12 months. The shortest periods of 20 days are shown in blue and are the most erratic. The periods of 60 days are shown in orange and are a bit more smooth. The periods of 250 days are shown in green and are even more smooth. The important thing to note here, is that the moving averages for the daily stock-returns are unstable for all of these three different window-lengths, which means that we cannot simply make a naive forecast which tries to predict the future mean daily stock-return from the recent past, as it will be highly inaccurate. If it were possible, these plots would have been much more stable.

Figure 7 shows it for the same three stocks and window-lengths, only it shows it for the standard deviations of the daily stock-returns. For 20 day periods these are also highly unstable so the recent past cannot be used to predict the near future. For 60 day periods and especially for 250 day periods the lines are quite smooth, but they still fluctuate significantly over longer periods of time. Perhaps these could be used as a rough forecast for the near future, but it depends on the portfolio method and how robust it is to estimation errors. The method presented in this paper does not use the standard deviation.

Figure 8 shows the correlations between pairs of stocks: AAPL vs. BBBY, AAPL vs. FL, and FL vs. BBBY. For 20 day periods the correlations are highly unstable and may even change sign in a short period of time, so two stocks may be highly positively correlated in one 20 day period, but highly negatively correlated shortly thereafter. For 60 day periods and especially 250 day periods the correlations are a bit more stable, but they do still fluctuate significantly over longer periods of time.

Because the mean-variance portfolio method adjusts stock-weights based on your estimates for the future mean stock-returns, their variance, and their correlations all at once, it is particularly vulnerable to estimation errors. If your estimates for both the mean stock-returns and their correlations are grossly incorrect, then the mean-variance portfolio may become concentrated in losing stocks that are also highly correlated, which would likely perform much worse than a simple equal-weighted portfolio.

The portfolio method presented in this paper is much more robust, because it separates the calculation of portfolio weights into two steps: 1) The filtering method which only allows stocks into the portfolio if their estimated future returns are sufficiently high; and 2) the diversification method which lowers those stock-weights to try and minimize the correlation between stocks. The diversification method will never increase the stock-weights, so the worst that can happen, is that it causes the portfolio to under-invest in some stocks because they were falsely estimated to be highly correlated with other stocks in the portfolio.

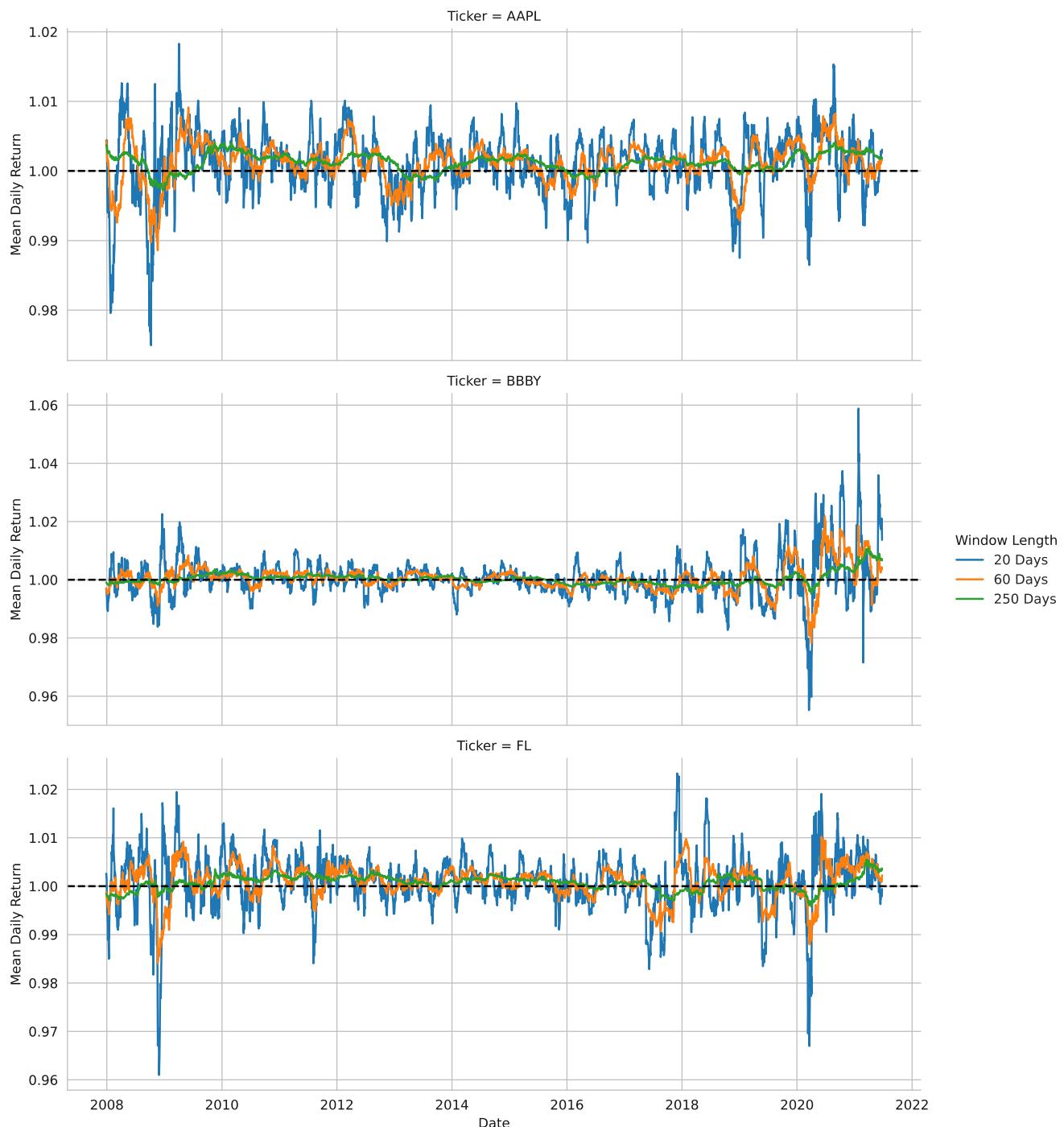


Figure 6: Rolling windows of different lengths for the **mean** of daily returns. These are also known as “moving averages”.

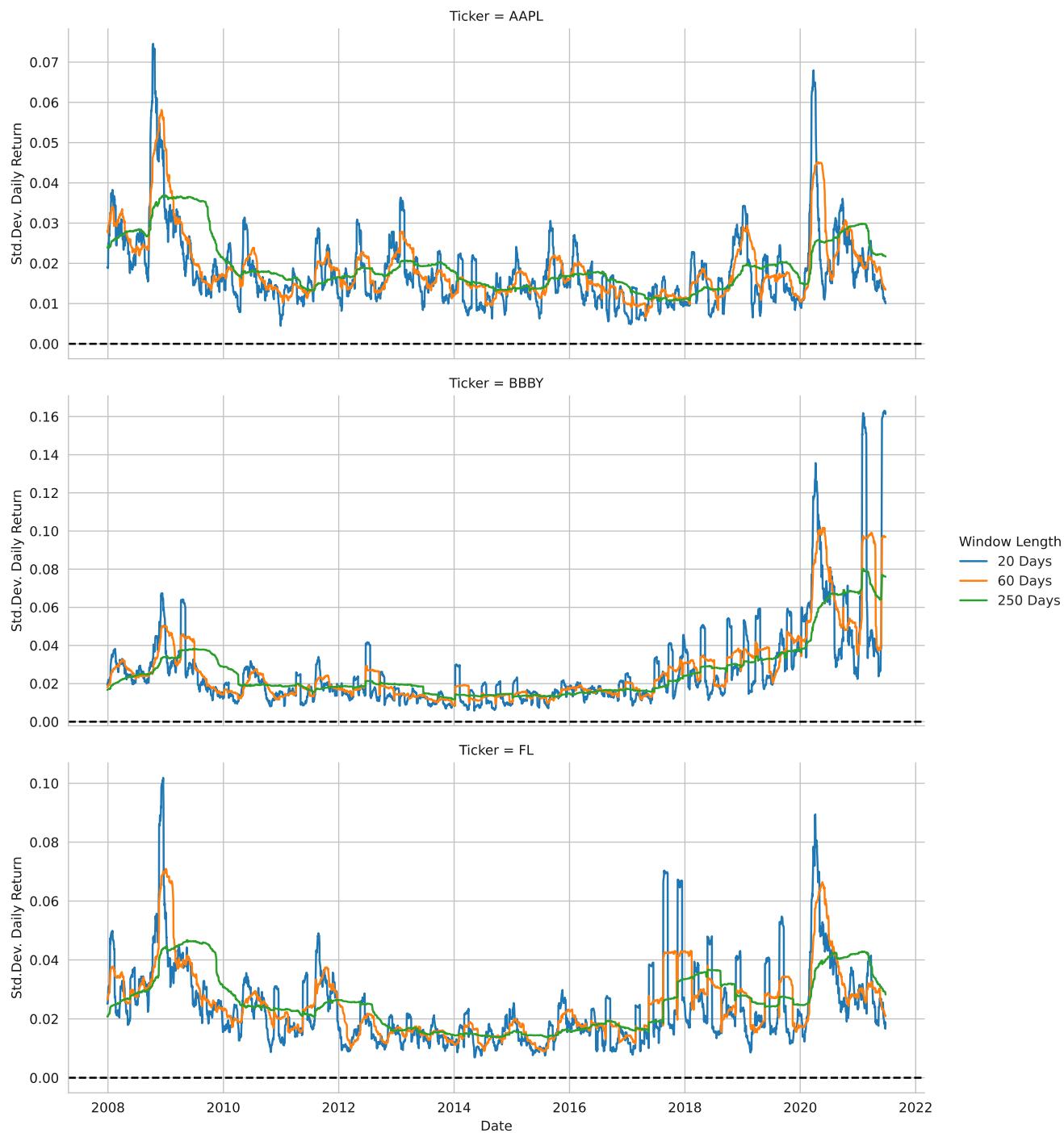


Figure 7: Rolling windows of different lengths for the **standard deviation** of daily returns.

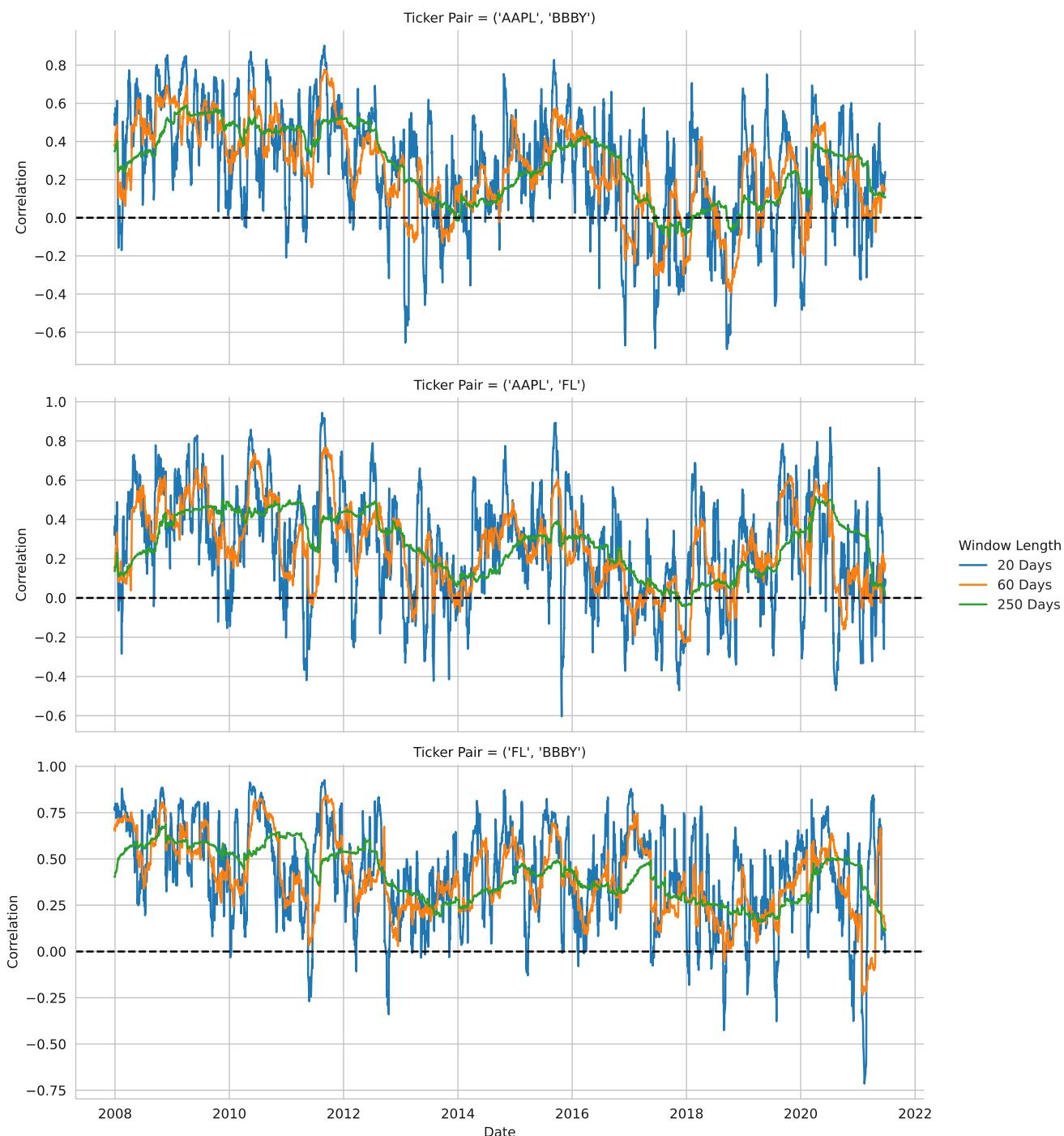


Figure 8: Rolling windows of different lengths for the **correlations** of daily returns.

5 Conditional Forecasting

In the previous section we saw that “naive” forecasts of the mean daily stock-returns and their correlations were highly inaccurate, because the recent past does not simply repeat into the future. You need to have some kind of method that can make reasonable forecasts for the future, based on so-called conditional probability distributions, so the future stock-returns and their correlations are estimated from current observations of some predictive signals and variables. This is a very hard problem and if there are errors in the forecasts, we cannot distinguish whether the portfolios performed poorly because the portfolio method is deficient or because the stock-forecasts are inaccurate.

In this paper we will bypass this problem by using the actual future stock-returns averaged for 2-3 year investment periods to determine the weights of stocks in the portfolio. This is of course a form of cheating which we call “omniscient testing” because we now have perfect knowledge about the future stock-returns. But this allows us to clearly test whether the portfolio method is working correctly, when we are using the actual future returns to determine the stock-weights.

We could use even longer prediction periods than just 2-3 years of stock-returns, but this would make our dataset quite limited for testing purposes, because it only contains about 14 years of data for most stocks. We could also use shorter forecasting periods, maybe even just a few days into the future, which would make the portfolios perform much better. But we will now show that short-term stock forecasting is extremely difficult and probably impossible, while it is sometimes possible to make reasonable forecasts for e.g. 2-3 year stock-returns by using certain predictor variables.

5.1 Case-Study: Apple (AAPL)

The top-plot in Figure 9 shows the so-called Price-To-Sales ratio (P/Sales) as the predictor variable versus the future 2-3 year average return on the AAPL stock. The data covers the years between 2007 and 2021. There is a clear downwards-sloping pattern in the scatter-plot, where higher P/Sales ratios usually correspond to lower future 2-3 year average stock-returns. Although the relation is somewhat noisy there is a clear relation, which is studied and explained in much greater detail in [Pedersen 2020].

Note that the y-axis for all three sub-plots in Figure 9 show the daily returns, so the returns in the top-plot have been converted from annualized into daily returns. For example, if the annualized return is 20% then the daily return is about 0.073%, calculated as follows when assuming there are roughly 250 trading-days in a year:

$$(1+20\%)^{(1/250)} - 1 \approx 0.00073 = 0.073\% \quad (19)$$

We often work with stock-returns that are “plus one” so instead of writing 0.2 for 20% we often write the number 1.2, and similarly instead of writing 0.00073 for 0.073% we often write it “plus one” so it is 1.00073. The daily returns on the y-axis in Figure 9 are written “plus one” like this. The reason is that it avoids the need to add and subtract 1 in many of the calculations, e.g. as done in Eq. (19) above.

The middle plot in Figure 9 shows the actual daily returns on the AAPL stock between the years 2007 and 2021. The x-axis again shows the P/Sales ratio and there is clearly no relation between the P/Sales ratio and the daily returns, as they seem to be completely independent of each other. But the top-plot showed that there was such a clear relation when considering 2-3 year average stock-returns, so why does it not show up in the middle plot for the daily returns? The reason is that the daily returns are much more volatile, often the AAPL stock goes up or down several percent in a single day, so the 2-3 year returns become tiny in comparison when they are converted into daily returns. You can see this by the different scales on the y-axis for the top and middle plots in Figure 9. The bottom plot compares the daily and 2-3 year returns in the same plot, where you can clearly see that the daily returns completely dominate the 2-3 year annualized returns that have been converted into daily returns.

5.2 Case-Study: Foot Locker (FL)

Another example is shown in Figure 10 for the company Foot Locker with stock-ticker FL. Again we see a clear relation between the P/Sales ratio and the future 2-3 year average stock returns, but when this is converted into daily returns, the relation is again completely hidden and dominated by the very volatile daily returns, as shown in the middle and bottom plots in Figure 10.

5.3 Case-Study: Bed, Bath & Beyond (BBBY)

In the previous two case-studies for the stocks AAPL and FL, there was a clear relation between their P/Sales ratio and their future 2-3 year average stock returns. As explained in [Pedersen 2020], the noise in those plots arises from differences in the P/Sales ratio and the growth in Sales Per Share in some of the 2-3 year investment periods being considered. For some stocks the scatter-plots get much smoother when considering longer investment periods of e.g. 6-8 years, and for some stocks it gets worse.

Figure 11 shows an example where the relation is more unclear between the P/Sales ratio and the future 2-3 year average stock returns. The top-plot in Figure 11 now shows a very complex relation between the P/Sales ratio and 2-3 year average stock returns. So the P/Sales would not have been a very good predictor for the returns on this particular stock. The reason is that this particular company was experiencing trouble in their business during several years, which resulted in very low share-prices.

5.4 Summary

The purpose of this section is to demonstrate that for some stocks it is much easier to predict their long-term returns than their daily returns. It is still not *easy* to predict long-term stock returns – but it is certainly *easier* than predicting short-term returns. When you are trying to predict what a stock will do tomorrow, you are essentially trying to predict what other people and computers will do next in the stock-market, while they are trying to predict what you are going to do next. This is a very silly game, like a group of near-sighted people inside a “house of mirrors”. When doing long-term investing, we want to make investments that perform well over several years, while allocating our portfolio so we can take advantage of short-term volatility. That is what a portfolio method should help us achieve.

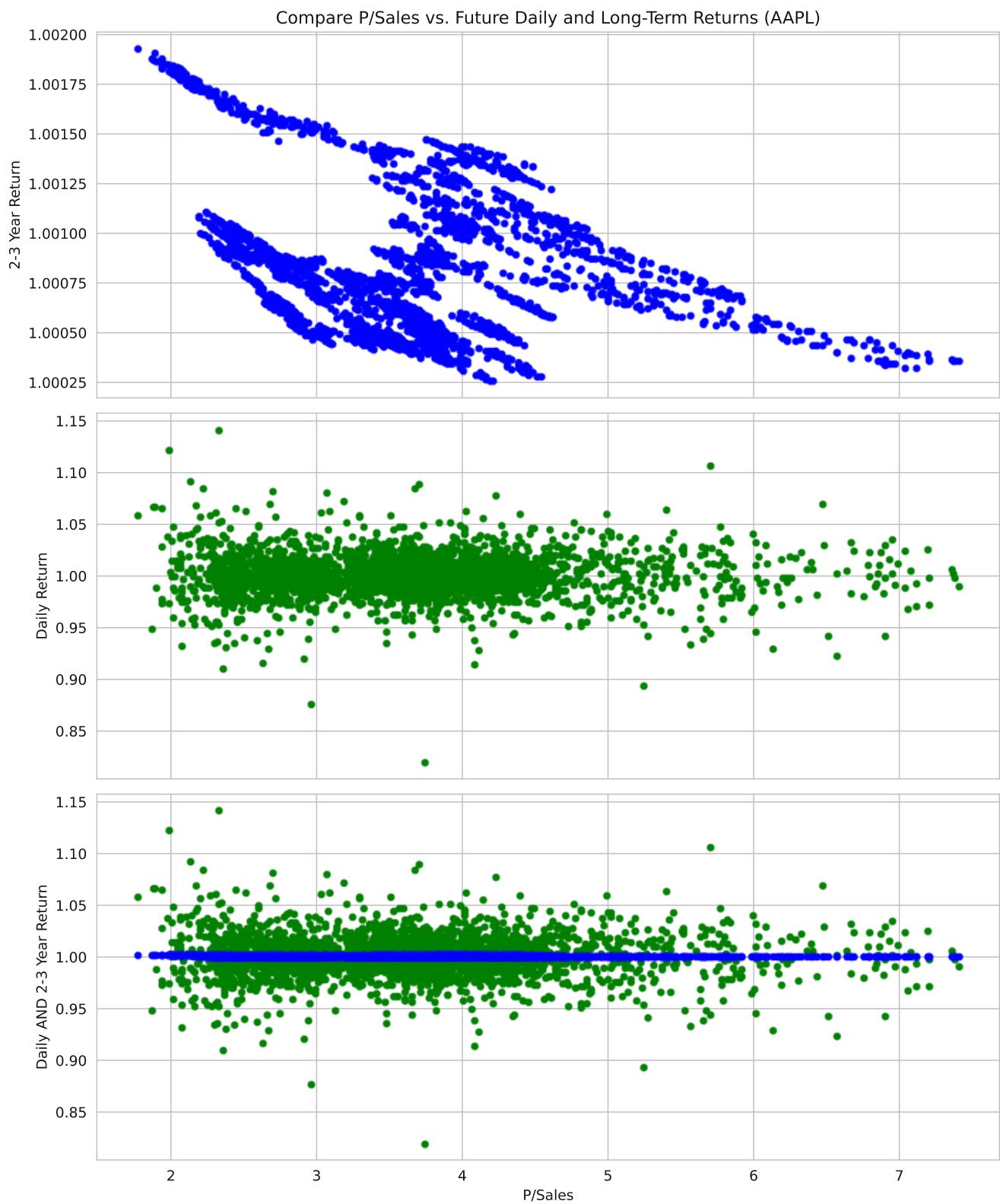


Figure 9: The P/Sales ratio versus daily and 2-3 year future returns for the AAPL stock.

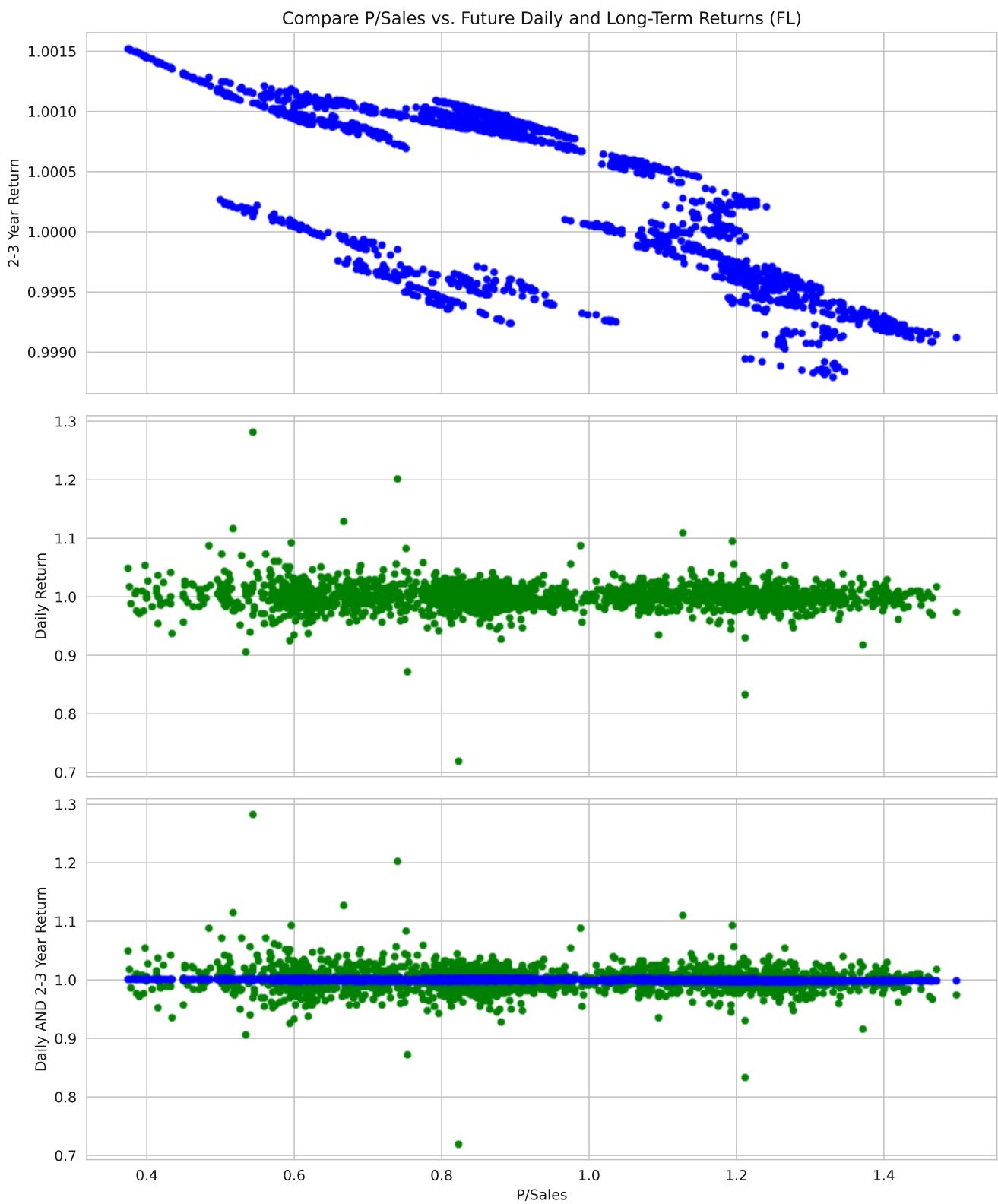


Figure 10: The P/Sales ratio versus daily and 2-3 year future returns for the FL stock.

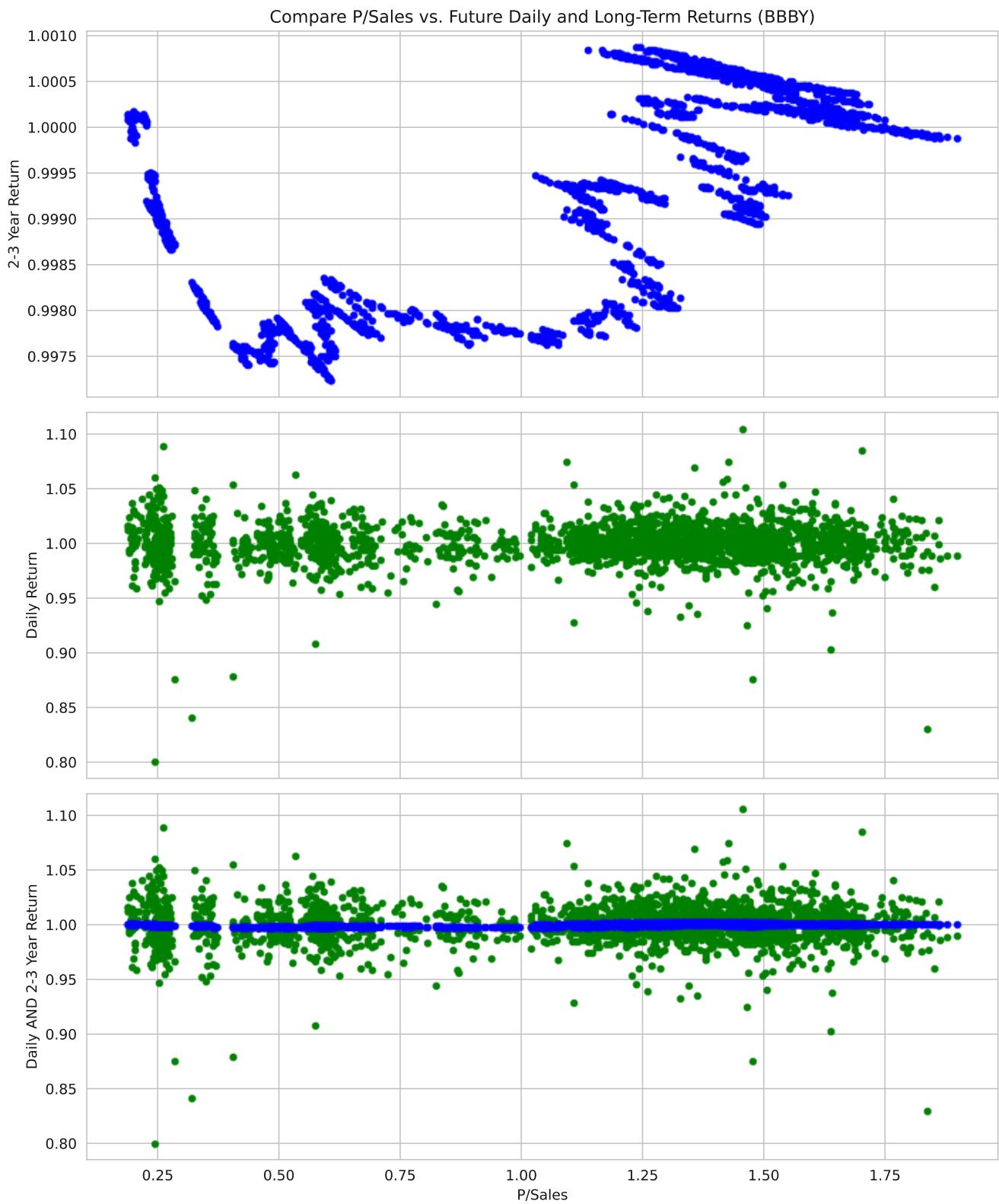


Figure 11: The P/Sales ratio versus daily and 2-3 year future returns for the BBBY stock.

6 Random Walks

In the academic research literature, it is common to assume that stock-returns are so-called IID, which means that they are Independent and Identically Distributed random variables, so for each time-step the random stock-returns would be drawn from the exact same distribution. This is a convenient assumption that allows for some elegant mathematical theory. Unfortunately, the previous section proved that this assumption is completely incorrect. Although it seems impossible to predict the daily stock-returns from a predictor variable such as the P/Sales ratio, we can sometimes predict more long-term stock-returns when considering investment periods of a few years or more.

It follows from the belief that stock-returns are IID random variables, that the stock-prices are so-called “random walks”. Finance professors often seem to believe that stocks don’t have any connection to the real world, but are merely random variables that follow their own random course and are also somehow correlated with each other. Once again, this is incorrect when considering long-term investing, where the stock returns tend to follow the growth of the company’s Sales and Earnings Per Share, while the short-term fluctuations do indeed seem to be random, caused by the daily “tug-of-war” between speculators who think they can outsmart each other.

If stock-prices were completely random walks then they would be unbounded both towards zero and infinity. This means that you would sometimes be able to buy shares in great companies such as Coca-Cola or Microsoft for a tiny fraction of their annual Earnings Per Share, and at other times the stock-prices might be higher than the total amount of money in circulation world-wide. We do not see this in practice, because stock-prices tend to fluctuate within certain ranges of valuation ratios.

A much more useful way of thinking about the progression of stock-prices is as a “semi-random walk” where the short-term volatility is indeed random, but over time the stock-prices tend to converge to their “intrinsic value”, as determined by their change in Sales or Earnings Per Share and their change in valuation ratio. The goal of a long-term investor is to make a good estimate of the stock’s “intrinsic value”, and then allocate the portfolio to maximize expected future returns, while also being able to take advantage of short-term market fluctuations.

6.1 Random Walk Formulas

There are different methods for generating random walks of stock-prices. We will use the following method. Let $Price_t$ denote the original, historical stock-price and let $Random Price_t$ denote the random stock-price, both at time-step t . The random stock-price starts with the value $Random Price_1=1$ and the price is then updated using a random stock-return for each time-step as follows:

$$Random Price_{t+1} = Random Price_t \cdot Random Return_t \quad (20)$$

The random stock-return for time-step t is assumed to be normal-distributed with mean μ and standard deviation σ , denoted as follows:

$$Random Return_t \sim N(\mu, \sigma^2) \quad (21)$$

Note that the random stock-returns in Eq. (21) above are IID because they are always drawn from the same normal distribution with mean μ and standard deviation σ .

In the “semi-random walk” we instead use a new mean μ_t for each time-step t :

$$\text{Random Return}_t \sim N(\mu_t, \sigma^2) \quad (22)$$

We then calculate the mean μ_t from the future “intrinsic value” of the stock at time-step $t+K$, divided by the current random stock-price at time-step t , and take this fraction to the power of $1/K$ so as to get the so-called geometric mean, which is the return required in each time-step, in order to go from the current *Random Price* _{t} to the future *Intrinsic Value* _{$t+K$} in K time-steps, through compounded returns.

$$\mu_t = \left(\frac{\text{Intrinsic Value}_{t+K}}{\text{Random Price}_t} \right)^{1/K} \quad (23)$$

In practice we would have to estimate what the future *Intrinsic Value* _{$t+K$} is, but we can “cheat” in these examples by simply using the actual future stock-price at that time-step, so the formula for μ_t becomes:

$$\mu_t = \left(\frac{\text{Price}_{t+K}}{\text{Random Price}_t} \right)^{1/K} \quad (24)$$

We can use random numbers for the stock-returns drawn from the standard-normal distribution $N(0,1)$, and then merely scale and shift those random numbers differently for the random walks and the semi-random walks, by using the mean μ for the random walks and using μ_t for the semi-random walks, and using the same σ for both. This allows for a completely fair comparison between the two methods, because their random stock-prices are generated from the same underlying random numbers. See the computer code in Section 19.

6.2 Case Study: Apple (AAPL)

Figure 12 shows the actual stock-price for AAPL as a red line normalized to begin at the value 1. This is actually the Total Return which includes the reinvestment of dividends. There are also 100 blue lines showing the random walks generated from Eq. (20) and (21) above, when using the same mean μ in all time-steps. The mean μ and standard deviation σ are calculated from the historical daily stock-returns for the AAPL stock. There are also 100 green lines showing the semi-random walks generated from Eq. (20) and Eq. (22) above, when using a new mean μ_t in each time-step calculated using Eq. (24), and the same standard deviation σ as used in the random walks, which was calculated from the historical stock-returns.

The completely random walks which are shown as blue lines in Figure 12, are very erratic and sometimes achieve incredibly high and incredibly low values. Note that the y-axis is logarithmic so the difference between the random walks and the actual stock-price becomes 2 orders of magnitude (i.e. factors 100 and 1/100) towards the end of these roughly 14 years of simulation data. The actual market-cap of AAPL was more than USD 1.5 trillion at that time, so some of the random walks suggest the market-cap could have been more than USD 150 trillion (a factor 100). But in the year 2019, the Gross Domestic Product (GDP) was only around USD 21 trillion for the entire USA, so it would be absurd if the total market-cap of AAPL was more than 7 times higher than all of USA’s GDP.

In the opposite end, the random walks also suggest the market-cap of AAPL could have been only 1/100 of its actual market-cap around USD 1.5 trillion, so around USD 15 billion, which would also be absurd considering the company had earnings of around USD 57 billion in the year 2020. This again shows how absurd the notion of completely random walks are in the stock-market.

The semi-random walks which are shown as green lines in Figure 12, are much closer to the actual stock-prices. This is a far more reasonable way of thinking about stock-market randomness, namely that the stock-prices are indeed random in the short-term, because they arise from the daily “tug-of-war” between speculators, but in the long-term the stock-prices tend to converge to their “intrinsic value”. This results in the range of the semi-random walks being much more reasonable than the completely random walks.

Of course, we are “cheating” in these examples by using the actual future stock-prices to generate the semi-random walks. In reality you would have to estimate the future “intrinsic value” of the stock. But the point is that you cannot just calculate the mean and standard deviation from the historical stock-returns and assume the future stock-price is a completely random walk with the same parameters. You need to make a reasonably good estimate of the future stock-returns based on one or more predictor variables, such as demonstrated in Section 5 and studied in great detail in [Pedersen 2020]. And every time the predictor variables change, you need to recalculate the estimates for the future stock-returns, and then you also need to change the stock-weights in your portfolio.

6.3 Other Case Studies

Figure 13 shows the random and semi-random walks for the FL stock, and Figure 14 shows it for the BBBY stock. Just as we saw for the AAPL stock, the completely random walks for the FL and BBBY stocks have a very wide range, reaching about 2 orders of magnitude (factors 100 and 1/100) above and below the actual stock-prices, while the semi-random walks are much closer to the actual stock-prices.

6.4 Summary

In this section we saw that completely random walks where stock-returns for each time-step are IID, generate random stock-prices that are often absurd, because they are several orders of magnitude above or below the actual stock-prices. One might think that the problem is that real-world stock-returns are actually not normal-distributed as assumed here, but the main problem is in fact that the random stock-returns are assumed to be IID, as if the stock-prices don’t have any relation to the company in which the stocks represent part-ownership. That is why a semi-random walk is much better at modelling the short-term randomness that arises from the daily “tug-of-war” between speculators, combined with the long-term reality of the actual company and how its sales and earnings grow or shrink over time.

Using this understanding of short-term randomness and long-term predictability, we want a portfolio method that will only invest in stocks whose long-term returns are estimated to be sufficiently high, while allocating the portfolio between multiple stocks, so as to take advantage of short-term volatility. If stock A goes down tomorrow, we would ideally like to have another stock B that goes up, so we can rebalance the portfolio by selling some of stock B and buy more of stock A, to take advantage of the lower price of stock A. We cannot do this if both stocks A and B go down at the same time.

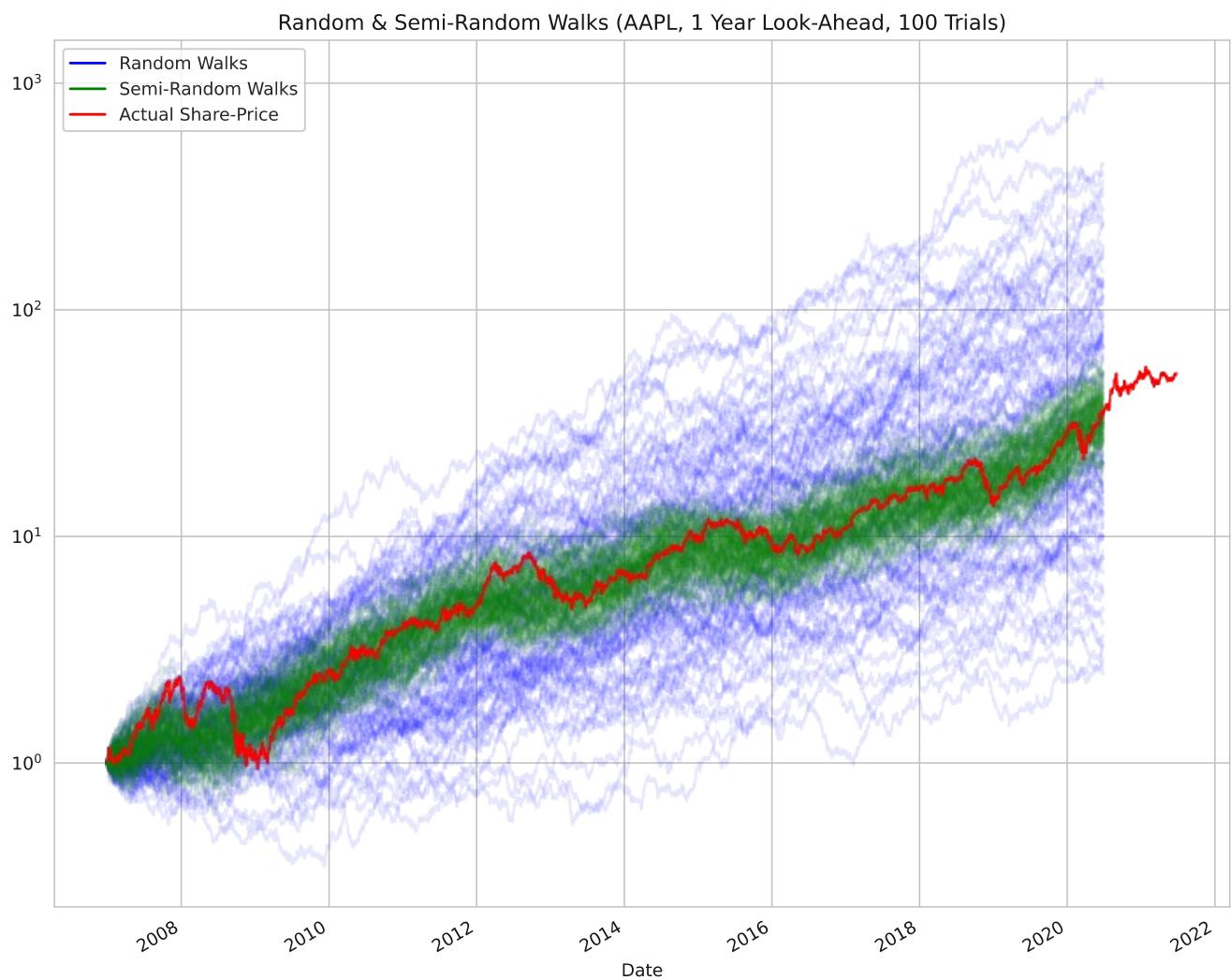


Figure 12: Random and Semi-Random Walks for the AAPL stock.

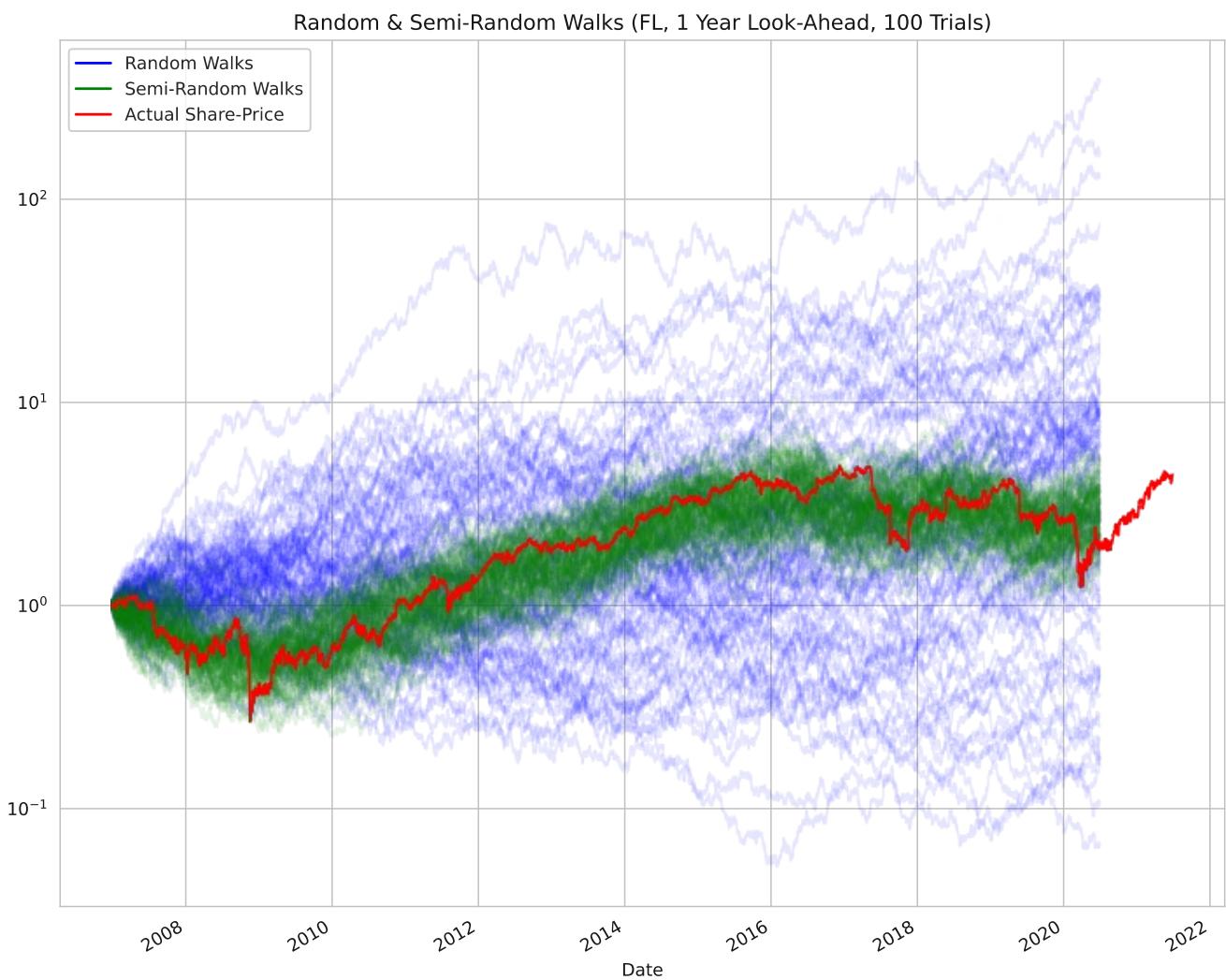


Figure 13: Random and Semi-Random Walks for the FL stock.

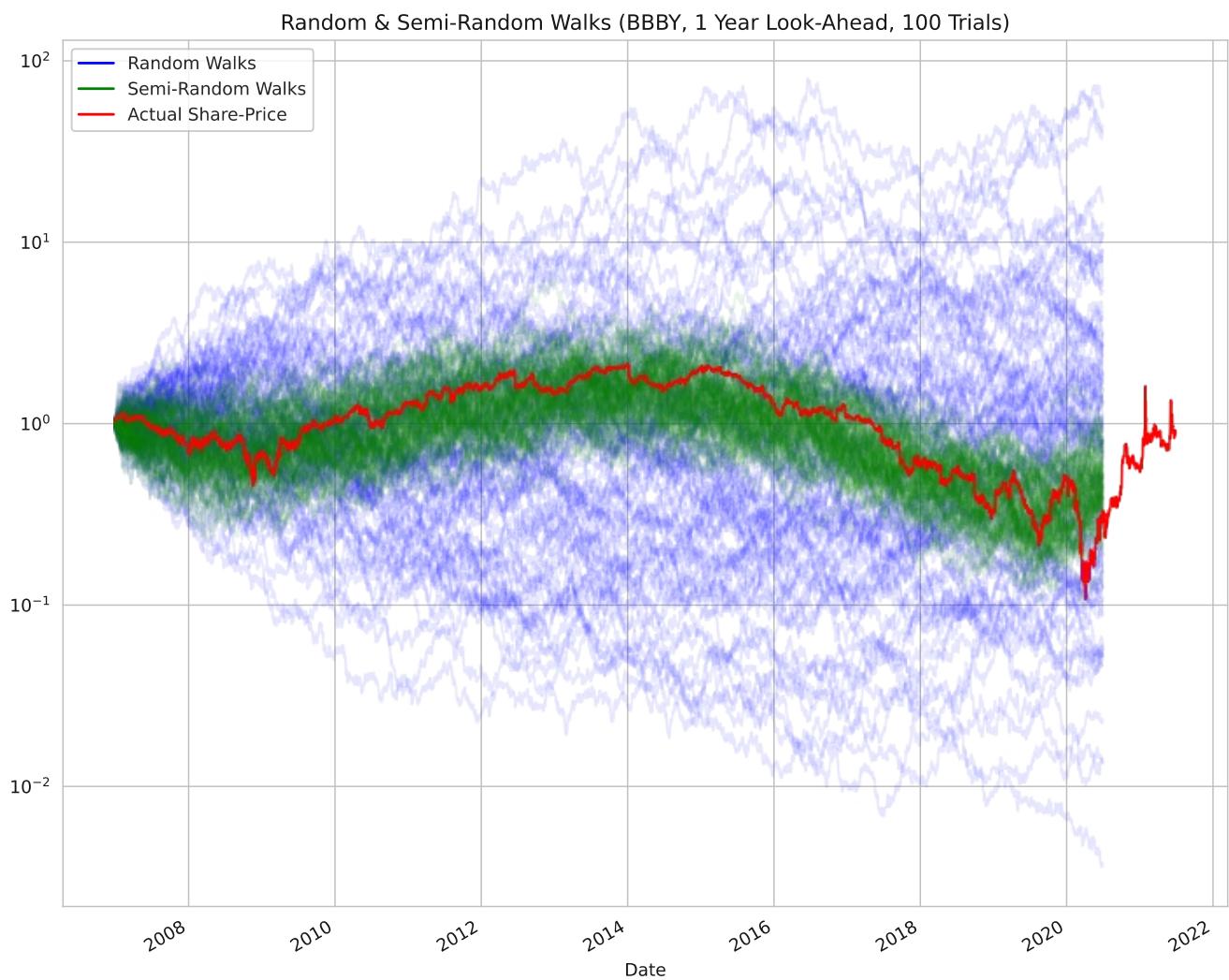


Figure 14: Random and Semi-Random Walks for the BBBY stock.

7 Filtering Methods

In this section we present the first part of our new portfolio method, which is a very basic and almost trivial filtering process, so we only allow assets into the portfolio, if the assets are estimated to have sufficiently high future returns. This should be an obvious requirement, because why would you want to include assets in your portfolio, if you have estimated they will result in a loss? That seems moronic!

But we saw in Section 3.3 that the mean-variance method can do just that, as it may include assets in the portfolio that are guaranteed to result in a loss, if the assets have negative correlations with other assets in the portfolio, because that would lower the variance (or standard deviation) of the portfolio's return. We avoid this by splitting our new portfolio method into two parts: The filtering part that we describe in this section, and the diversification part that we describe in the next Section 8.

7.1 Threshold Filter

The simplest filtering method only allows an asset into the portfolio, if the estimated future mean return μ_i is sufficiently high so it exceeds a given threshold μ_{\min} , then it sets the asset's weight to $Weight_{\max}$. This is a trivial if-else statement in a computer program and we can write it mathematically as follows:

$$Weight_i = \begin{cases} Weight_{\max} & \text{if } \mu_i \geq \mu_{\min} \\ 0 & \text{else} \end{cases} \quad (25)$$

For example, we might set the threshold to $\mu_{\min}=10\%$ so if we have estimated that Asset 1 has a future mean return of $\mu_1=17.5\%$ then it is sufficiently high to be included in the portfolio. We then set its portfolio weight to $Weight_{\max}$ which could be e.g. 5%, in which case we can have a maximum of 20 assets in the portfolio before the sum of their weights would exceed 100% of the portfolio's capacity, unless we invest for borrowed money, which we will not consider in this paper. So we need to ensure the portfolio weights sum to 100% or less. We do this by first calculating their sum:

$$Weight_{\sum} = \sum_{i=1}^N Weight_i \quad (26)$$

If the sum is greater than 1 (or 100%), then we divide all the portfolio weights with the sum. This is also a simple if-else statement in a computer program, which can be written mathematically as follows:

$$Weight_i = \begin{cases} Weight_i / Weight_{\sum} & \text{if } Weight_{\sum} > 1 \\ Weight_i & \text{else} \end{cases} \quad (27)$$

Once we have estimated the future mean returns for all assets, and we have used the above formulas to determine their portfolio weights, we have a portfolio of assets that have all been estimated to have sufficiently high future returns. As we will see in the real-world experiments later in this paper, if you can make a reasonably good estimate of the future 2-3 year returns on stocks, then you can do extremely well with this simple filtering method. But we can do even better with a slightly more sophisticated filtering method.

7.2 Adaptive Filter

In the simple threshold filter above, we assign equal portfolio weights to all assets that exceed the given threshold for their future mean returns. But what if one asset is estimated to return 10%, and another asset is estimated to return 20%, and yet another asset is estimated to return 30%? If we have the same confidence in all three of these return estimates being reasonably accurate, then clearly we would want to invest more of the portfolio in the assets that are estimated to have higher returns. We still cannot invest the entire portfolio in the asset with the highest estimated return, because that estimate could be wrong, so we still need to invest in different assets to guard ourselves from estimation errors. But surely we would want to invest more in the assets that are estimated to have a higher future return.

One method for doing this is called the “Kelly Criterion” which was used in one of the earliest quantitative investment funds by a mathematics professor named Ed Thorp, who wrote several papers on how to use the Kelly Criterion for determining portfolio weights, see e.g. [Thorp 1975]. But the papers can be hard to understand for non-mathematicians, and even more importantly is that the Kelly Criterion is difficult to calculate for entire portfolios, and it is extremely sensitive to estimation errors, so it can severely over-invest in assets that have been incorrectly estimated to have high future returns.

Fortunately there is an effective and robust method that is so simple it is almost trivial: We calculate the portfolio weight using a linear scale on the estimated future mean return for each asset. This is shown in Figure 15 where the portfolio weight of an asset $Weight_i$ is zero when the asset’s future mean return μ_i is below the minimum required return μ_{min} , and the portfolio weight increases linearly until the asset’s mean return μ_i reaches the upper limit μ_{max} , where the asset’s portfolio weight is limited to $Weight_{max}$.

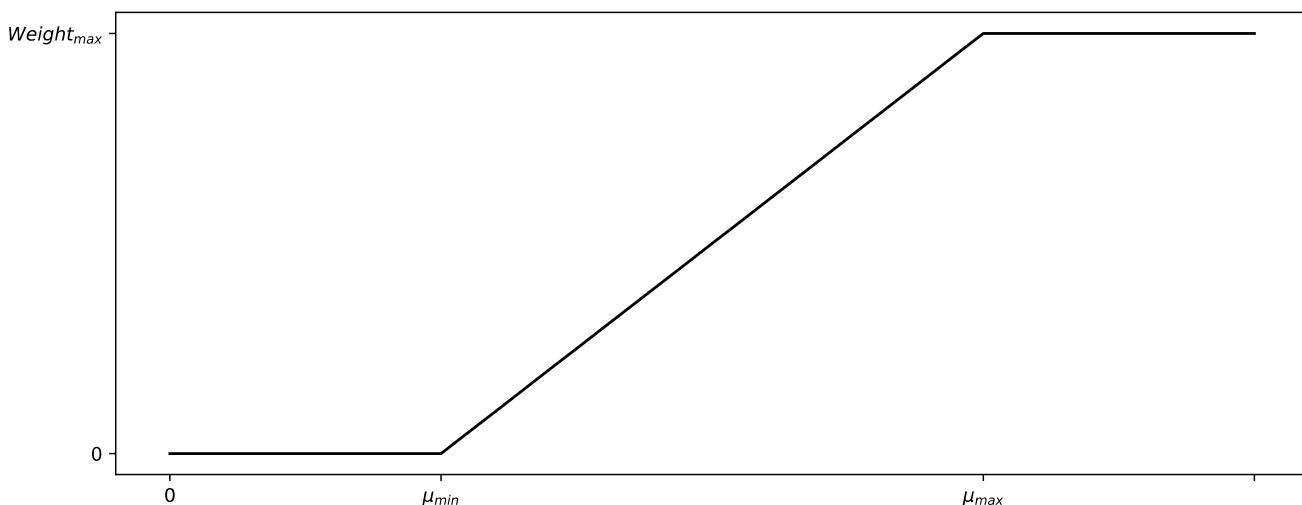


Figure 15: Adaptive filter for calculating portfolio weights from estimated mean returns.

The linear filter adapts to the estimated mean return of each asset, so assets that are expected to have a higher future return will get a larger portfolio weight. It can be written mathematically as follows:

$$Weight_i = \begin{cases} 0 & \text{if } \mu_i < \mu_{\min} \\ \mu_i \cdot a + b & \text{if } \mu_{\min} \leq \mu_i \leq \mu_{\max} \\ Weight_{\max} & \text{if } \mu_i > \mu_{\max} \end{cases} \quad (28)$$

Where the parameters a and b for the linear function are:

$$a = \frac{Weight_{\max}}{\mu_{\max} - \mu_{\min}} \quad b = -a \cdot \mu_{\min} \quad (29)$$

For example, if an asset is estimated to have a future mean return of $\mu_i = 15\%$ and we need at least $\mu_{\min} = 10\%$ to allow an asset into the portfolio, and when it reaches $\mu_{\max} = 30\%$ the portfolio weight is $Weight_{\max} = 5\%$, then using these numbers in Eq. (28) and (29) gives $Weight_i = 1.25\%$ for the asset.

As with the simple threshold filter in Section 7.1, we can also have the portfolio weights sum to more than 1 for this adaptive filter, which would require us to invest for borrowed money. So if the weights sum to more than 1 we again need to normalize them so they only sum to 1 using Eq. (26) and (27).

7.3 Other Ideas

In this paper we will only use the two simple filtering methods above. But they can easily be extended to adjust portfolio weights according to many other aspects you find relevant. Your imagination is the limit for what you could include in your filtering method. But you should test that your filter actually has the intended effect on your portfolio's performance e.g. using the same testing procedures as we use in Sections 10-15. Following are just a few ideas to get you started developing your own filters.

Bond yields are currently very low, but if bond yields return to more historically normal levels, then it would be a good idea to include these in the filter. This could be as simple as subtracting the yield on e.g. 3-year low-risk bonds from each asset's estimated future mean return. If the bond yield is e.g. 7% then you might not be satisfied with an estimated 10% return on stock investments, so you should adjust the filter methods to take the current bond yields into account.

Another idea would be to adjust the portfolio weights according to your personal view of each asset's quality. For example, you might have estimated that you can make a 15% return on the stocks of both Microsoft and Acme Corp, but you have much higher confidence in the future of Microsoft. You can adjust the portfolio weights with your personal view of each asset's quality, simply by setting a number between 0 and 1 for the quality of each asset, where 1 means the asset is top quality and you have very high confidence in your estimate for the asset's future return, and a quality number of 0 means that you have no confidence in the future of the asset. Typically the numbers would be somewhere between 0 and 1. Then you adjust each asset's portfolio weight simply by multiplying it with the quality number.

You can also make adjustments for various quantitative aspects. For example, you may consider various ratios for how much debt each company has and its ability to repay the debt. You convert these debt ratios into numbers between 0 and 1, which can then be multiplied with the asset's portfolio weights, so you would invest less in companies that have higher debt-burdens.

8 Diversification Method

In this section we present the second part of our new portfolio method, which is a diversification method that takes the portfolio weights that were calculated by the filtering method in Section 7, and adjusts those weights to create a more diversified portfolio where the assets have lower correlation.

The diversification method is only allowed to lower the portfolio weights. This is an important distinction from the mean-variance portfolio method, which may include assets in the portfolio that are guaranteed to result in a loss, if those assets have negative correlations with other assets in the portfolio, because that would lower the portfolio's overall variance, as shown in Section 3.3.

Because our new diversification method only allows the portfolio weights to decrease, the worst that can happen is that too much of the portfolio is being held in cash with zero investment return. The diversification method will not over-invest in some assets just because they have low correlations.

Compared to the various algorithms for optimizing mean-variance portfolios, this new diversification algorithm is also much simpler, it can be computed much faster, and it works much better in practice.

8.1 Motivation

There are mainly two reasons why we want to diversify an investment portfolio: The first reason is to protect ourselves from making estimation errors in the future returns of the assets, so we hopefully aren't wrong about the future prospects of all the assets in our portfolio. The second reason is to try and avoid that all the assets in our portfolio experience losses at the same time. We would prefer that some assets go up while others go down in price, so we can take advantage of short-term volatility to rebalance our portfolio, by selling some of the assets that have increased in price, and buy more of the assets that have decreased in price, if we still believe those assets have good long-term prospects.

A good example of such a situation was the “Corona-Virus Panic” in early 2020, where the stock-markets were extremely volatile, and some stocks were highly correlated while others were not. If you had invested your entire portfolio in stocks that all lost half or more of their market-value at the same time, then you would not be able to take advantage of the lower prices and buy more of those assets, because your entire portfolio had suffered big losses. There were several days during the spring of 2020, where many stocks went up or down 10-20% and some stocks even more. So we would like to diversify our portfolio to be able to take advantage of such short-term market volatility.

8.2 Basic Idea

We can explain the basic idea of the diversification method with a simple example. Let us say the filtering process in Section 7 suggests that we should invest $Weight_A = 9\%$ of our portfolio in Asset A, and we should invest $Weight_B = 12\%$ in Asset B. Let us further assume that the short-term correlation between these two assets is $\rho_{A,B} = 0.5$, so the two assets have a tendency to move up or down in price together in the short-term. It is not a perfectly synchronous movement of the two assets' prices as their correlation is only 0.5 and not 1. But because they do have a tendency to move up or down together, it is as if we have invested more in both Asset A and Asset B than just their own portfolio weights.

We define the “Full Exposure” of each asset to be its own portfolio weight plus the entire portfolio’s indirect exposure to that same asset through its correlation with the other assets in the portfolio. A simple and intuitive (but also slightly incorrect) way of calculating the Full Exposure of Asset A is to take the weight of Asset A plus the correlated weight of Asset B:

$$\text{Full Exposure}_A = \text{Weight}_A + \rho_{A,B} \cdot \text{Weight}_B = 9\% + 0.5 \cdot 12\% = 15\% \quad (30)$$

And similarly for the Full Exposure of Asset B, where the correlation is symmetric $\rho_{A,B}=\rho_{B,A}$:

$$\text{Full Exposure}_B = \text{Weight}_B + \rho_{B,A} \cdot \text{Weight}_A = 12\% + 0.5 \cdot 9\% = 16.5\% \quad (31)$$

So we thought we had only invested 9% of the portfolio in Asset A, but through its correlation with Asset B, the portfolio’s Full Exposure to Asset A is in fact 15% of the portfolio. Similarly, we thought we had only invested 12% of the portfolio in Asset B, but through its correlation with Asset A, the portfolio’s Full Exposure to Asset B is in fact 16.5% of the portfolio.

We want to find new weights for the two assets that are denoted Weight_A^* and Weight_B^* (marked with an asterisk * to indicate they are the new or adjusted weights), so that both of the Full Exposures that are calculated using these new weights, will be equal to the originally desired Weight_A and Weight_B . That is, we want to find Weight_A^* and Weight_B^* that solve these two equations:

$$\begin{aligned} \text{Weight}_A &= \text{Weight}_A^* + \rho_{A,B} \cdot \text{Weight}_B^* \\ \text{Weight}_B &= \text{Weight}_B^* + \rho_{B,A} \cdot \text{Weight}_A^* \end{aligned} \quad (32)$$

The solution is $\text{Weight}_A^*=4\%$ and $\text{Weight}_B^*=10\%$, so that is how much of the portfolio we should actually invest in Assets A and B, if we want their Full Exposures to equal 9% and 12% respectively.

This is fairly easy to solve for a portfolio of only two assets. But it is much more difficult for larger portfolios, especially if we also allow negative portfolio weights and correlations. Furthermore, the definition of Full Exposure that was used above is actually over-simplified and slightly incorrect.

8.3 When To Adjust Portfolio Weights

Before we give a proper definition of the Full Exposure, let us first consider which asset correlations should be included in its calculation. Positive weights are also known as “long” investments and negative weights are known as “short” investments. We would like our diversification method to be able to handle both positive and negative portfolio weights, even though it is more complicated.

Let us start by only considering positive (or “long”) portfolio weights. If Assets A and B have positive weights and also positive correlation, it means that the two assets tend to have either a positive or negative effect on the portfolio at the same time. This tendency gets stronger the closer the correlation coefficient is to 1. So in this case we need to include the correlation in the calculation of the Full Exposure, so the two asset weights can be adjusted. This case is shown in the first row of Table 1.

If the portfolio weights for the two assets are still both positive, but their correlation is now negative, it means that the two asset prices tend to move in opposite direction of each other, so their combined effect on the portfolio is to neutralize each other’s movements somewhat. This is desirable because it means that as one asset goes up in price the other asset goes down, so we can take advantage of short-

term price-volatility and rebalance the portfolio to our advantage. It also happens to lower the variance of the portfolio's returns. So we should not include this negative correlation in the calculation of the Full Exposure, because we want both these assets to be included in the portfolio at their originally desired weights. This case is shown in the second row of Table 1.

Another reason that we should not include such negative correlations in the calculation of the Full Exposure, is that the Full Exposure would then be lower than the original portfolio weights. To see this you should try and calculate Eq. (30) and (31) with a negative correlation. This would require for the new portfolio weights that solve Eq. (32) to be greater than the originally desired portfolio weights, so we would increase the portfolio weights simply because the two assets have negative correlations. This is one of the major flaws of the mean-variance method as shown in Section 3.3.

For negative (or “short”) portfolio weights, the two cases for positive and negative correlations are actually the same as for positive portfolio weights. The reason is that even though the portfolio weights are now negative, if the correlation is still positive, then the two assets will tend to move up or down in price at the same time, and therefore have a similar effect on the portfolio. And conversely if the correlation is negative then the two asset prices will tend to move in opposite directions and have opposite effects on the overall portfolio, even though both asset weights are negative. These two cases are shown in rows 5 and 6 in Table 1.

In case one portfolio weight is positive and the other is negative, and the correlation between the assets is also negative, then the two negatives cancel each other out, so the two assets tend to have a similar effect on the overall portfolio. Although one asset is “long” and the other is “short”, because they are negatively correlated, this is effectively the same as two “long” positions or two “short” positions with regard to their correlation, so the correlation needs to be included in the calculation of the Full Exposure, so the portfolio weights can be adjusted accordingly. This case is shown in row 3 of Table 1.

In case one portfolio weight is positive and the other is negative, but the correlation between the assets is now positive, then the two asset prices will tend to move up or down together, but because one asset weight is “long” and the other asset weight is “short”, the two assets will have opposite effects on the overall portfolio. So in this case the correlation should not be included in the calculation of the Full Exposure. This case is shown in row 4 of Table 1.

Weight Types	$\text{sign}(\text{Weight}_i)$	$\text{sign}(\text{Weight}_j)$	$\text{sign}(\rho_{i,j})$	$\text{sign}(W_i \cdot W_j \cdot \rho_{i,j})$	Adjust Weights?
Long	+	+	+	+	Yes
	+	+	-	-	No
Long & Short	+	-	-	+	Yes
	+	-	+	-	No
Short	-	-	+	+	Yes
	-	-	-	-	No

Table 1: Summary of the different cases for signs of portfolio weights and correlations, and whether each case should be included in the calculation of the Full Exposure to adjust the portfolio weights.

Also shown in Table 1 is a column with the sign of the product of the portfolio weights and correlation. This turns out to be a plus whenever the correlation should be included in the calculation of the Full Exposure, and it turns out to be a minus whenever the correlation should not be included in the Full Exposure. This means we can make a simple mathematical formula to decide whether or not to include the correlation between Assets i and j in the calculation of the Full Exposure. It is denoted $Use_{i,j}$ which is just short for “Use in calculation of the Full Exposure”:

$$Use_{i,j} = \begin{cases} 1 & \text{if } \text{sign}(Weight_i \cdot Weight_j \cdot \rho_{i,j}) \text{ is +} \\ 0 & \text{else} \end{cases} \quad (33)$$

Note that both Eq. (33) and Table 1 are symmetrical because the correlation is symmetrical $\rho_{i,j} = \rho_{j,i}$.

8.4 Full Exposure

We are now ready to formalize the definition of the Full Exposure that measures how much a portfolio is really exposed to each asset through correlations with other assets in the portfolio. Figure 16 shows the basic concept where the Full Exposure for Asset i is denoted $Full\ Exposure_i$ and it is the sum of the asset's originally desired portfolio weight (e.g. resulting from the filtering methods in Section 7), plus the Correlated Exposure which is denoted $Corr\ Exposure_i$ and measures how much the portfolio is exposed to Asset i through correlations with other assets in the portfolio. So we have:

$$Full\ Exposure_i = Weight_i + Corr\ Exposure_i \quad (34)$$

The adjusted portfolio weights are denoted $Weight_i^*$ and the Correlated and Full Exposure calculated using these new portfolio weights are denoted $Corr\ Exposure_i^*$ and $Full\ Exposure_i^*$ with an asterisk * to indicate they are different from those calculated with the original portfolio weights. So we have:

$$Full\ Exposure_i^* = Weight_i^* + Corr\ Exposure_i^* \quad (35)$$

The goal is to find $Weight_i^*$ for all Assets i so their Full Exposure equals the original $Weight_i$:

$$Full\ Exposure_i^* = Weight_i \quad (36)$$

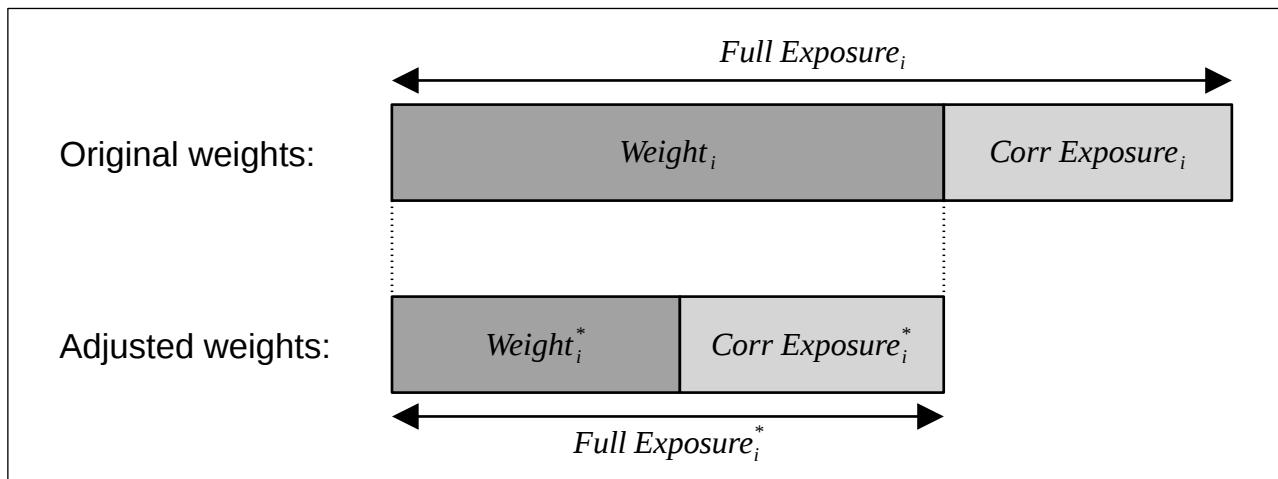


Figure 16: The concept of Full and Correlated Exposure, and how to adjust the portfolio weights.

My first definition of the Full Exposure was the simple and intuitive one used in Section 8.2, but that resulted in problems if some of the portfolio weights were zero. After some more attempts at making a meaningful definition of the Full Exposure, it became clear that it should at least satisfy these criteria:

- (1) $\text{Full Exposure}_i = 0 \quad \text{if } \text{Weight}_i = 0$
 - (2) $\text{Full Exposure}_i = \text{Weight}_i \quad \text{if } \text{Weight}_j = 0 \text{ or } \rho_{i,j} = 0 \text{ for all } j \neq i$
 - (3) $\text{Full Exposure}_i \geq \text{Weight}_i \quad \text{if } \text{Weight}_i > 0$
 - (4) $\text{Full Exposure}_i \leq \text{Weight}_i \quad \text{if } \text{Weight}_i < 0$
- (37)

Let us explain each of these criteria and why they are necessary:

- (1) If an asset's portfolio weight is zero $\text{Weight}_i=0$ then we must also have $\text{Full Exposure}_i=0$, because we cannot decrease the updated portfolio weight Weight_i^* any further than zero, but we need Full Exposure_i^* to equal the original $\text{Weight}_i=0$, so the only way to achieve that would be to lower the weights of the other assets in the portfolio to get $\text{Full Exposure}_i^*=\text{Weight}_i=0$. So without this criterion it would cause many of the other portfolio weights to become zero.
- (2) If an asset is either not correlated with any other asset in the portfolio $\rho_{i,j}=0$ for all $j \neq i$, or if the other portfolio weights are all zero $\text{Weight}_j=0$ for all $j \neq i$, or a mix of these two cases, then Full Exposure_i should equal Weight_i , because the asset has no correlated exposure to other assets in the portfolio, so we don't want the portfolio weight to change $\text{Weight}_i^*=\text{Weight}_i$.
- (3) If a portfolio weight is positive, then its Full Exposure should be greater than the weight.
- (4) If a portfolio weight is negative, then its Full Exposure should be less than the weight. Together with criterion (3) this ensures the Full Exposure is always greater in magnitude than the weight, while having the same sign as the portfolio weight.

There are other necessary requirements for the definition of Full Exposure to be sensible, such as it being a strictly increasing function when one or more of the portfolio weights are increased. And when changing Weight_i^* it should have a much greater impact on Full Exposure_i^* for that particular Asset i compared to the Full Exposure for all the other assets. These requirements will be clarified in the convergence proof in Section 8.12.

Through experimentation it was found that the following definition of the Full Exposure works very well in practice, and you should check that it satisfies all the criteria in Eq. (37) above:

$$\text{Full Exposure}_i = \text{sign}(\text{Weight}_i) \cdot \sqrt{\sum_{j=1}^N |\text{Weight}_i \cdot \text{Weight}_j \cdot \rho_{i,j}^2 \cdot \text{Use}_{i,j}|} \quad (38)$$

Let us explain the reasoning behind the different parts of this formula. We of course want the Full Exposure to somehow include the weights of the portfolio's other correlated assets. So we sum over all other Assets j . Inside the summation, we first multiply with Weight_i so the Full Exposure is also zero if $\text{Weight}_i=0$. We then multiply with Weight_j for the other Asset j and the correlation between the two assets $\rho_{i,j}$, but we use the squared correlation $\rho_{i,j}^2$ instead, because the surrounding square-root would

otherwise amplify the correlation. We then multiply with $Use_{i,j}$ which is either 0 or 1 to only include the “bad” correlations that require adjustments to their portfolio weights, as explained in Section 8.3.

From its definition in Eq. (33), $Use_{i,j}$ ensures the product of $Weight_i$, $Weight_j$ and $\rho_{i,j}$ is positive, but because we are using the squared correlation $\rho_{i,j}^2$ instead, it is possible that the overall product is negative, so to ensure the product is positive, we simply take its absolute value. We calculate this for all combinations of Assets i and j , sum the results, take the square-root, and finally multiply with the sign of the original portfolio weight, to ensure the Full Exposure is also negative if $Weight_i$ is negative.

You may note that the way we have defined the Full Exposure in Eq. (38) is different from how we originally defined the Full Exposure from the Correlated Exposure in Eq. (34). We can also define the Correlated Exposure somewhat similar to Eq. (38) where the summation just excludes Asset i :

$$Corr\ Exposure_i = \text{sign}(Weight_i) \cdot \sqrt{\sum_{j \neq i} |Weight_i \cdot Weight_j \cdot \rho_{i,j}^2 \cdot Use_{i,j}|} \quad (39)$$

If we add this to $Weight_i$ then we get the following definition of the Full Exposure:

$$Full\ Exposure_i = Weight_i + \text{sign}(Weight_i) \cdot \sqrt{\sum_{j \neq i} |Weight_i \cdot Weight_j \cdot \rho_{i,j}^2 \cdot Use_{i,j}|} \quad (40)$$

But that is not exactly the same as the Full Exposure defined in Eq. (38), which can be rewritten slightly to separate $Weight_i$ from the summation:

$$Full\ Exposure_i = \text{sign}(Weight_i) \cdot \sqrt{Weight_i^2 + \sum_{j \neq i} |Weight_i \cdot Weight_j \cdot \rho_{i,j}^2 \cdot Use_{i,j}|} \quad (41)$$

Which one of these two definitions of the Full Exposure is the correct one? They both satisfy all the criteria in Eq. (37) so they are both valid in that sense. But they are in fact different and Eq. (40) generally results in higher values for the Full Exposure compared to the ones calculated in Eq. (41), because $Weight_i$ is inside the square-root in Eq. (41). This means that Eq. (40) will generally result in lower adjusted portfolio weights $Weight_i^*$ and therefore a more conservative portfolio allocation.

In this paper we will use the Full Exposure defined in Eq. (41), which is the same as Eq. (38). But it is possible that Eq. (40) works even better, or perhaps that you can find a completely different definition of the Full Exposure that works better still. For example, you may note that these definitions have some similarity to the definition of a portfolio’s standard deviation as defined in Eq. (12), but we don’t use the standard deviations of the assets σ_i and σ_j in the above definitions of the Full Exposure. Perhaps it would improve the diversification even more if we include these in the definition of the Full Exposure? It would be interesting to see a performance comparison of many different definitions of the Full Exposure, but it is beyond the scope of this paper, so hopefully you feel inspired to write such a paper.

8.5 Classic Optimization

We now have a mathematical definition of the Full Exposure in Eq. (38), and we want to find new portfolio weights $Weight_i^*$ that make their $Full\ Exposure_i^*$ equal to the original $Weight_i$, which we stated as our goal back in Eq. (36). We can pose this as a classic optimization problem, by defining the Mean Squared Error (MSE) between the original $Weight_i$ and the adjusted $Full\ Exposure_i^*$ as follows:

$$MSE = \frac{1}{N} \cdot \sum_{i=1}^N (Weight_i - Full\ Exposure_i^*)^2 \quad (42)$$

The goal is to find $Weight_i^*$ that minimize the MSE. Because the MSE is a continuous function it can be minimized by common methods such as the L-BFGS-B method which is implemented in many software packages including the SciPy package for the Python programming language. It is important that the boundaries are set correctly to ensure the sign of $Weight_i^*$ is the same as the sign of $Weight_i$.

For portfolios of only 100 assets, minimizing the MSE using L-BFGS-B takes about half a second, which is 1000x slower than the custom algorithm in Section 8.7 below. For larger portfolios of 1000 assets it becomes very slow and impractical to minimize the MSE using L-BFGS-B, while the custom algorithm in Section 8.7 still only needs 20 milli-seconds to adjust the portfolio weights of 1000 assets!

It might be possible to improve the computation speed by deriving the gradient of the MSE, but this depends on the exact definition of the Full Exposure, so we would need to derive the gradient for each variant of the Full Exposure that we might want to experiment with, while the custom algorithm in Section 8.7 should be able to handle all reasonable definitions of the Full Exposure.

8.6 Inverse Algorithm

Another way of finding the new $Weight_i^*$ that makes $Full\ Exposure_i^*$ equal to the original $Weight_i$ is to derive the mathematical inverse of $Full\ Exposure_i^*$. Using the definition of $Full\ Exposure_i^*$ from Eq. (38) that has been slightly rewritten in Eq. (41), we obtain the following:

$$\begin{aligned} Full\ Exposure_i^* &= Weight_i \\ \Leftrightarrow \\ \text{sign}(Weight_i) \cdot \sqrt{(Weight_i^*)^2 + \sum_{j \neq i} |Weight_j^* \cdot Weight_j^* \cdot \rho_{i,j}^2 \cdot Use_{i,j}|} &= Weight_i \\ \Leftrightarrow \\ (Weight_i^*)^2 + |Weight_i| \cdot \sum_{j \neq i} |Weight_j^* \cdot \rho_{i,j}^2 \cdot Use_{i,j}| - Weight_i^2 &= 0 \end{aligned} \quad (43)$$

Let us denote the summation inside the last equation as:

$$s_i = \sum_{j \neq i} |Weight_j^* \cdot \rho_{i,j}^2 \cdot Use_{i,j}| \quad (44)$$

So the equation from the last line of Eq. (43) can be written as:

$$|Weight_i^*|^2 + |Weight_i^*| \cdot s_i - Weight_i^2 = 0 \quad (45)$$

We then want to find $|Weight_i^*|$ that satisfies this equation, as it will give us the solution to the original equation where the Full Exposure equals the originally desired weight $|Full\ Exposure_i^*| = |Weight_i|$ and the sign can easily be copied from the original $Weight_i$. To solve this note that Eq. (45) is actually a 2nd degree polynomial in the variable $|Weight_i^*|$ whose solution is well-known:

$$|Weight_i^*| = \frac{-s_i \pm \sqrt{s_i + 4 \cdot Weight_i^2}}{2} \quad (46)$$

Because the variable $|Weight_i^*|$ is an absolute value it is always positive, so the solution to the 2nd degree polynomial must also be positive, so we only use the + case of the \pm operator in Eq. (46).

When we have found the new portfolio weight $|Weight_i^*|$ that makes $|FullExposure_i^*|=|Weight_i^*|$ it also impacts the Full Exposure of the other assets that use $Weight_i^*$ in their calculations, and this causes $|FullExposure_j^*|\neq|Weight_j^*|$ for those other assets. So we may need to perform several iterations of the weight-update in Eq. (46) before $|FullExposure_i^*|$ converges to $|Weight_i^*|$ for all Assets i .

So that we can distinguish the updated portfolio weights for different iterations of the algorithm, let us denote the portfolio weight for Asset i in the k 'th iteration of the algorithm as $Weight_{i,k}^*$, and the starting weight for iteration $k=1$ is denoted $Weight_{i,1}^*$.

The algorithm is then:

- Initialize the new weights by setting $Weight_{i,1}^*=Weight_i$
- Repeat the following for a given number of iterations from $k=1$ to some upper limit:
 - For each Asset i do the following:
 - Calculate the sum in Eq. (44) for the k 'th iteration:

$$s_{i,k} = \sum_{j \neq i} |Weight_{j,k}^* \cdot \rho_{i,j}^2 \cdot Use_{i,j}| \quad (47)$$

- Calculate the new weight using the original sign and the positive case of Eq. (46):

$$Weight_{i,k+1}^* = \text{sign}(Weight_i) \cdot \frac{-s_{i,k} + \sqrt{s_{i,k}^2 + 4 \cdot Weight_i^2}}{2} \quad (48)$$

This algorithm works extremely well and converges to the correct solution in just a few iterations, even for portfolios of 1000 assets. The only problem is that it is specifically made for the definition of Full Exposure in Eq. (38) (and slightly rewritten in Eq. (41)), which means that a completely new algorithm has to be implemented if you want to use another definition of the Full Exposure.

8.7 General Algorithm

Let us now present an algorithm that works for more general definitions of the Full Exposure, and is almost as efficient as the specialized algorithm in the previous section. This algorithm also performs a number of iterations to gradually improve $Weight_i^*$ to make $FullExposure_i^*$ converge to the originally desired $Weight_i$. The algorithm assumes that $FullExposure_i^*$ changes in almost direct proportion to changes made to $Weight_i^*$, which is indeed the case for the definition of $FullExposure_i^*$ in Eq. (38).

The algorithm is quite simple. The only minor complication is that once we have calculated the difference $DifWeight_i^* = FullExposure_i^* - Weight_i$, we cannot use this to update $Weight_i^*$ directly. This would over-adjust $Weight_i^*$ because $FullExposure_i^*$ is calculated using $Weight_j^*$ for all the other correlated Assets j in the portfolio, and this would cause $Weight_i^*$ to fluctuate erratically and eventually

diverge towards infinity. So we should only adjust Weight_i^* with the part of Dif Weight_i^* that can be directly attributed to the influence of Weight_i^* on Full Exposure_i^* , which is the fraction in Eq. (50).

Because Full Exposure_i^* changes when the other portfolio weights Weight_j^* change, it is still possible that Full Exposure_i^* will be slightly over-adjusted. That is why Eq. (51) allows a smaller step-size.

So that we can distinguish the updated portfolio weights and their corresponding Full Exposure for different iterations of the algorithm, let us again denote the portfolio weight for Asset i in the k 'th iteration of the algorithm as $\text{Weight}_{i,k}^*$, and the starting weight for iteration $k=1$ is denoted $\text{Weight}_{i,1}^*$. The Full Exposure that is calculated using $\text{Weight}_{i,k}^*$ is denoted $\text{Full Exposure}_{i,k}^*$.

The algorithm is then:

- Initialize the new weights by setting $\text{Weight}_{i,1}^* = \text{Weight}_i$
- Repeat the following for a given number of iterations from $k=1$ to some upper limit:
 - Calculate $\text{Full Exposure}_{i,k}^*$ using Eq. (38) with $\text{Weight}_{i,k}^*$.
 - Calculate the difference between the new $\text{Full Exposure}_{i,k}^*$ and the original Weight_i :

$$\text{Dif Weight}_{i,k}^* = \text{Full Exposure}_{i,k}^* - \text{Weight}_i \quad (49)$$

- Calculate the weight adjustment (see the discussion in the text above):

$$\text{Adj Weight}_{i,k}^* = \text{Dif Weight}_{i,k}^* \cdot \frac{\text{Weight}_{i,k}^*}{\text{Full Exposure}_{i,k}^*} \quad (50)$$

- Calculate the new $\text{Weight}_{i,k+1}^*$. You can use a *StepSize* less than 1 for slower convergence:

$$\text{Weight}_{i,k+1}^* = \text{Weight}_{i,k}^* - \text{Adj Weight}_{i,k}^* \cdot \text{StepSize} \quad (51)$$

- Terminate the computation early if the updated weights are already sufficiently precise:

$$\text{Max}(|\text{Dif Weight}_{i,k}^*|) < \text{Required Precision} \quad (52)$$

Omitted from the above algorithm is the inner for-loop that iterates over all Assets i . This can be done in two different ways. The implementation described in Section 19 implements both these versions of the algorithm, and they both result in the same adjusted portfolio weights.

In the so-called “element-wise” way of iterating over all Assets i , the steps inside the loop of the algorithm are calculated for each Asset i in turn, so we first calculate the Full Exposure for just one Asset i using Eq. (38), and then we use Eq. (49), (50) and (51) to update the portfolio weight for just Asset i . Then we do the same for the next asset, etc.

In the “vectorized” way, we iterate over all Assets i in each step of the algorithm, so we first calculate the Full Exposure for all Assets i using Eq. (38), then we calculate Eq. (49) for all Assets i , and then we calculate Eq. (50) for all Assets i , and finally we calculate Eq. (51) for all Assets i .

8.8 Simpler Algorithm

The algorithm in Section 8.7 is how it was originally conceived, but it turns out that it can be made even simpler by setting $\text{StepSize}=1$ in Eq. (51), and Section 8.12 shows that this simpler algorithm still converges, so there is really no need for a step-size in the algorithm. We simplify the weight-update by using the definitions from Eq. (49), (50) and (51) with $\text{StepSize}=1$ as follows:

$$\begin{aligned}
 \text{Weight}_{i,k+1}^* &= \text{Weight}_{i,k}^* - \text{Adj Weight}_{i,k}^* \\
 &= \text{Weight}_{i,k}^* - \text{Dif Weight}_{i,k}^* \cdot \frac{\text{Weight}_{i,k}^*}{\text{Full Exposure}_{i,k}^*} \\
 &= \text{Weight}_{i,k}^* - (\text{Full Exposure}_{i,k}^* - \text{Weight}_i) \cdot \frac{\text{Weight}_{i,k}^*}{\text{Full Exposure}_{i,k}^*} \\
 &= \text{Weight}_{i,k}^* \cdot \left(1 - \frac{\text{Full Exposure}_{i,k}^* - \text{Weight}_i}{\text{Full Exposure}_{i,k}^*}\right) \\
 &= \text{Weight}_{i,k}^* \cdot \frac{\text{Weight}_i}{\text{Full Exposure}_{i,k}^*}
 \end{aligned} \tag{53}$$

It means that we can update the portfolio weight for Asset i with this simple formula:

$$\text{Weight}_{i,k+1}^* = \text{Weight}_{i,k}^* \cdot \frac{\text{Weight}_i}{\text{Full Exposure}_{i,k}^*} \tag{54}$$

So the updated portfolio weight $\text{Weight}_{i,k+1}^*$ is simply the portfolio weight from the previous iteration $\text{Weight}_{i,k}^*$ multiplied by the ratio between the originally desired Weight_i and $\text{Full Exposure}_{i,k}^*$. We are merely changing $\text{Weight}_{i,k}^*$ by how far $\text{Full Exposure}_{i,k}^*$ is from the desired Weight_i .

The simplified algorithm is:

- Initialize the new weights by setting $\text{Weight}_{i,1}^* = \text{Weight}_i$
- Repeat the following for a given number of iterations from $k=1$ to some upper limit:
 - Calculate $\text{Full Exposure}_{i,k}^*$ using Eq. (38) with $\text{Weight}_{i,k}^*$.
 - Calculate the new $\text{Weight}_{i,k+1}^*$ using Eq. (54):

$$\text{Weight}_{i,k+1}^* = \text{Weight}_{i,k}^* \cdot \frac{\text{Weight}_i}{\text{Full Exposure}_{i,k}^*} \tag{55}$$

- Terminate the computation early due to convergence if the following condition is met:

$$\text{Max}(|\text{Full Exposure}_{i,k}^* - \text{Weight}_i|) < \text{Required Precision} \tag{56}$$

Another loop is required inside the above algorithm for iterating over all the Assets i . The inner-loop could either iterate over the assets “element-wise” or “vectorized”, as we also discussed in Section 8.7.

8.9 Weight Initialization

The two algorithms in Sections 8.7 and 8.8 both initialize the new weights from the originally desired weights: $\text{Weight}_{i,1}^* = \text{Weight}_i$. But if you run the algorithm on nearly the same portfolio weights and correlation-matrix many times, then you may be able to improve the algorithm's time-usage, by using the previous output of the algorithm as the initial weights of the algorithm when you run it again. This may help the algorithm converge even faster, if the previous output is already close to the solution.

But it creates a complication if the sign is different for some Asset i : $\text{sign}(\text{Weight}_{i,1}^*) \neq \text{sign}(\text{Weight}_i)$ because we require that their signs are always equal, so the Full Exposure also has the same sign as Weight_i . Fortunately, the simple weight-update in Eq. (54) corrects the sign in the first iteration of the algorithm so that: $\text{sign}(\text{Weight}_{i,2}^*) = \text{sign}(\text{Weight}_i)$. This is because Eq. (38) ensures $\text{sign}(\text{Weight}_{i,1}^*)$ is the same as $\text{sign}(\text{Full Exposure}_{i,1}^*)$, so their ratio is always positive, and the updated $\text{Weight}_{i,2}^*$ then gets the same sign as Weight_i from the update in Eq. (54), even though $\text{Weight}_{i,1}^*$ has the wrong sign.

8.10 Basic Example

Let us again consider the basic example from Section 8.2, where we only have two Assets A and B with the originally desired portfolio weights $\text{Weight}_A = 9\%$ and $\text{Weight}_B = 12\%$. And let us again assume that their correlation is $\rho_{A,B} = 0.5$. We now want to find new portfolio weights using the simplified algorithm in Section 8.8, so their Full Exposure is equal to their originally desired portfolio weights.

We initialize the new portfolio weights to equal the original weights as the starting point for the search:

$$\begin{aligned}\text{Weight}_{A,1}^* &= \text{Weight}_A = 9\% \\ \text{Weight}_{B,1}^* &= \text{Weight}_B = 12\%\end{aligned}\tag{57}$$

We then calculate the Full Exposure for these two weights using Eq. (38). Because the two weights and their correlation are all positive, we know from Eq. (33) that $\text{Use}_{A,B} = \text{Use}_{B,A} = 1$, so we have:

$$\begin{aligned}\text{Full Exposure}_{A,1}^* &= \sqrt{9\% \cdot 9\% + 9\% \cdot 12\% \cdot 0.5^2} \approx 10.4\% \\ \text{Full Exposure}_{B,1}^* &= \sqrt{12\% \cdot 12\% + 12\% \cdot 9\% \cdot 0.5^2} \approx 13.1\%\end{aligned}\tag{58}$$

We then update the portfolio weights using Eq. (55) from the simplified algorithm in Section 8.8:

$$\begin{aligned}\text{Weight}_{A,2}^* &= \text{Weight}_{A,1}^* \cdot \text{Weight}_A / \text{Full Exposure}_{A,1}^* = 9\% \cdot 9\% / 10.4\% \approx 7.8\% \\ \text{Weight}_{B,2}^* &= \text{Weight}_{B,1}^* \cdot \text{Weight}_B / \text{Full Exposure}_{B,1}^* = 12\% \cdot 12\% / 13.1\% \approx 11.0\%\end{aligned}\tag{59}$$

We then calculate the Full Exposure for the adjusted portfolio weights, which are now very close to the originally desired portfolio weights of $\text{Weight}_A = 9\%$ and $\text{Weight}_B = 12\%$:

$$\begin{aligned}\text{Full Exposure}_{A,2}^* &= \sqrt{7.8\% \cdot 7.8\% + 7.8\% \cdot 11.0\% \cdot 0.5^2} \approx 9.1\% \\ \text{Full Exposure}_{B,2}^* &= \sqrt{11.0\% \cdot 11.0\% + 11.0\% \cdot 7.8\% \cdot 0.5^2} \approx 11.9\%\end{aligned}\tag{60}$$

Let us try another iteration of the algorithm by inserting the updated weights into Eq. (55) again:

$$\begin{aligned} \text{Weight}_{A,3}^* &= \text{Weight}_{A,2}^* \cdot \text{Weight}_A / \text{Full Exposure}_{A,2}^* = 7.8\% \cdot 9\% / 9.1\% \approx 7.7\% \\ \text{Weight}_{B,3}^* &= \text{Weight}_{B,2}^* \cdot \text{Weight}_B / \text{Full Exposure}_{B,2}^* = 11.0\% \cdot 12\% / 11.9\% \approx 11.1\% \end{aligned} \quad (61)$$

These portfolio weights are very close to the weights in Eq. (59) so the algorithm has nearly converged. We can repeat the above steps of the algorithm, to find portfolio weights whose Full Exposure get arbitrarily close to the originally desired portfolio weights. After only a few more iterations, the algorithm converges to approximately $\text{Weight}_A^* \approx 7.72\%$ and $\text{Weight}_B^* \approx 11.07\%$ whose Full Exposure is very close to the originally desired portfolio weights of $\text{Weight}_A = 9\%$ and $\text{Weight}_B = 12\%$.

This means that if the two assets are correlated with a coefficient of 0.5, and we want to invest 9% of the portfolio in Asset A and 12% of the portfolio in Asset B, then we should actually only invest about 7.72% of the portfolio in Asset A and only about 11.07% of the portfolio in Asset B, in order for the Full Exposure of Asset A to be 9%, and the Full Exposure of Asset B to be 12%, as originally desired. This is because the two assets are positively correlated, so an investment in one asset is also an indirect investment in the other asset through this correlation.

In this basic example, there is only a small difference between the original and adjusted portfolio weights. But for larger portfolios where some assets are highly correlated and other assets are perhaps negatively correlated, the adjusted portfolio weights can be very different from the original portfolio weights. And it would be very difficult to adjust the portfolio weights for large portfolios by hand without a computer algorithm such as the one above.

8.11 More Examples

Let us consider a few more small examples using the simple diversification method from Section 8.8.

Example 1

In the first example, the portfolio has two assets whose desired portfolio weights and correlation are:

$$\begin{aligned} \text{Weight}_1 &= -9\%, \quad \text{Weight}_2 = +12\% \\ \rho_{1,2} &= +0.5 \end{aligned} \quad (62)$$

Because the two weights have different signs (one is negative and the other is positive), and because their correlation is positive, we know from Eq. (33) that $\text{Use}_{1,2}=0$ so this is a “good” correlation and the two portfolio weights should not be adjusted. As the Full Exposure already equals the desired portfolio weights, the diversification method terminates immediately, as shown in Table 2.

Iteration k	$\text{Weight}_{1,k}^*$	$\text{FullExp}_{1,k}^*$	$\text{Weight}_{2,k}^*$	$\text{FullExp}_{2,k}^*$	MSE
1	-9.000%	-9.000%	12.000%	12.000%	0.0
2	-9.000%	-9.000%	12.000%	12.000%	0.0

Table 2: Iterations of the diversification algorithm for the portfolio weights in Eq. (62).

Example 2

In the second example, the portfolio still has two assets whose desired portfolio weights are the same as before, but now their correlation is negative instead of positive, so we have:

$$\begin{aligned} \text{Weight}_1 &= -9\%, \quad \text{Weight}_2 = +12\% \\ \rho_{1,2} &= -0.5 \end{aligned} \tag{63}$$

From Eq. (33) we know that $\text{Use}_{1,2}=1$ because this is a “bad” correlation so we need to adjust the two portfolio weights. Table 3 shows the iterations of the diversification method, which converges very quickly to new portfolio weights, thus making their new Full Exposure almost exactly equal to the originally desired portfolio weights. These numbers are shown in bold for Iteration 1, because we initialized the diversification method with the originally desired weights from Eq. (63). Compare these to the final values for the Full Exposure, which are shown in bold in the row of the last iteration. Note how the Mean Squared Error (MSE) between $\text{FullExp}_{i,k}^*$ and Weight_i decreases by several orders of magnitude in each iteration. For such a small portfolio of only two assets, the diversification method appears to be very efficient, as it only requires a few iterations to converge to the correct solution.

Iteration k	$\text{Weight}_{1,k}^*$	$\text{FullExp}_{1,k}^*$	$\text{Weight}_{2,k}^*$	$\text{FullExp}_{2,k}^*$	MSE
1	-9.000%	-10.392%	12.000%	13.077%	1.5E-04
2	-7.794%	-9.067%	11.012%	11.947%	3.7E-07
3	-7.737%	-9.014%	11.061%	11.989%	1.5E-08
4	-7.725%	-9.003%	11.071%	11.998%	6.6E-10
5	-7.722%	-9.001%	11.073%	12.000%	2.8E-11

Table 3: Iterations of the diversification algorithm for the portfolio weights in Eq. (63).

Example 3

The next example has 3 assets instead of only 2. Their weights and correlations are all positive:

$$\begin{aligned} \text{Weight}_1 &= +10\%, \quad \text{Weight}_2 = +15\%, \quad \text{Weight}_3 = +20\% \\ \rho_{1,2} &= +0.8, \quad \rho_{1,3} = +0.5, \quad \rho_{2,3} = +0.2 \end{aligned} \tag{64}$$

Table 4 shows the iterations of the diversification method, which converges after only a few iterations, so the new portfolio weights are very close to the originally desired portfolio weights from Eq. (64). Note how $\text{FullExp}_{1,k}^*$ decreases in every iteration and converges from above towards its target value of 10%, while both $\text{FullExp}_{2,k}^*$ and $\text{FullExp}_{3,k}^*$ get over-adjusted in the first iteration and then converges from below towards their target values of 15% and 20%. But they all converge very quickly. Also note that $\text{Weight}_{1,k}^*$ converges to around 5.4% which is nearly half of the original weight of 10%, while $\text{Weight}_{3,k}^*$ converges to around 19.1% which is much closer to the original weight of 20%. This is because Asset 1 has comparatively much higher correlation with both of the other assets.

Iteration k	$\text{Weight}_{1,k}^*$	$\text{FullExp}_{1,k}^*$	$\text{Weight}_{2,k}^*$	$\text{FullExp}_{2,k}^*$	$\text{Weight}_{3,k}^*$	$\text{FullExp}_{3,k}^*$	MSE
1	10.000%	15.684%	15.000%	18.248%	20.000%	21.494%	1.5E-03
2	6.376%	10.983%	12.330%	14.544%	18.610%	19.626%	4.4E-05
3	5.805%	10.415%	12.717%	14.786%	18.965%	19.921%	7.5E-06
4	5.574%	10.180%	12.901%	14.909%	19.040%	19.972%	1.4E-06
5	5.476%	10.078%	12.980%	14.962%	19.067%	19.989%	2.6E-07
6	5.433%	10.034%	13.013%	14.984%	19.078%	19.995%	4.8E-08
7	5.415%	10.015%	13.027%	14.993%	19.082%	19.998%	9.1E-09
8	5.407%	10.006%	13.033%	14.997%	19.084%	19.999%	1.7E-09
9	5.403%	10.003%	13.036%	14.999%	19.085%	20.000%	3.2E-10

Table 4: Iterations of the diversification algorithm for the portfolio weights in Eq. (64).

Example 4

This example has the same weights and correlations as the previous example, except one of the weights and one of the correlations are now negative:

$$\begin{aligned} \text{Weight}_1 &= -10\%, \quad \text{Weight}_2 = +15\%, \quad \text{Weight}_3 = +20\% \\ \rho_{1,2} &= -0.8, \quad \rho_{1,3} = +0.5, \quad \rho_{2,3} = +0.2 \end{aligned} \tag{65}$$

Table 5 shows the iterations of the diversification method, which requires a few iterations less than when all the weights and correlations were positive in the previous example. This is probably because the single negative weight and correlation means that not all the correlations are “bad” according to Table 1 and Eq. (33), so they are not all included in the calculation of the Full Exposure. The diversification method can therefore converge faster to the new portfolio weights whose Full Exposure is equal to the originally desired portfolio weights from Eq. (65).

Note that the adjusted portfolio weights from the last row in Table 5 are not the same as the weights from the last row in Table 4. Although the original portfolio weights and correlations are all the same in magnitude, the ones in Eq. (64) are all positive while the ones in Eq. (65) are both positive and negative. This causes them to have different Full Exposure so the adjusted weights are also different.

Iteration k	$\text{Weight}_{1,k}^*$	$\text{FullExp}_{1,k}^*$	$\text{Weight}_{2,k}^*$	$\text{FullExp}_{2,k}^*$	$\text{Weight}_{3,k}^*$	$\text{FullExp}_{3,k}^*$	MSE
1	-10.000%	-14.000%	15.000%	18.248%	20.000%	20.298%	8.9E-04
2	-7.143%	-10.363%	12.330%	14.769%	19.707%	19.952%	6.2E-06
3	-6.893%	-10.137%	12.523%	14.899%	19.754%	20.003%	9.7E-07
4	-6.800%	-10.055%	12.608%	14.960%	19.751%	20.002%	1.5E-07
5	-6.762%	-10.022%	12.642%	14.984%	19.749%	20.001%	2.5E-08
6	-6.747%	-10.009%	12.655%	14.994%	19.749%	20.000%	4.0E-09
7	-6.741%	-10.004%	12.661%	14.997%	19.748%	20.000%	6.5E-10

Table 5: Iterations of the diversification algorithm for the portfolio weights in Eq. (65).

Example 5

This example has the same weights and correlations as the previous example, except two of the weights and two of the correlations are now negative:

$$\begin{aligned} \text{Weight}_1 &= -10\%, \quad \text{Weight}_2 = +15\%, \quad \text{Weight}_3 = -20\% \\ \rho_{1,2} &= -0.8, \quad \rho_{1,3} = +0.5, \quad \rho_{2,3} = -0.2 \end{aligned} \tag{66}$$

Table 6 shows the iterations of the diversification method which again converges very quickly, so the Full Exposure of the adjusted portfolio weights are close to the originally desired weights in Eq. (66). Compare the last row in Table 6 to the last row in Table 5 and note how the final adjusted weights have slightly different magnitudes. Then compare the last row in Table 6 to the last row in Table 4 and note that the final adjusted portfolio weights are the same except for their signs. This is because all the correlations in both Eq. (64) and Eq. (66) are considered “bad” according to Eq. (33), so all correlations are included in the calculation of the Full Exposure using Eq. (38), and therefore the adjusted portfolio weights converge to the same values – they just have different signs to match the original weights.

Iteration k	$\text{Weight}_{1,k}^*$	$\text{FullExp}_{1,k}^*$	$\text{Weight}_{2,k}^*$	$\text{FullExp}_{2,k}^*$	$\text{Weight}_{3,k}^*$	$\text{FullExp}_{3,k}^*$	MSE
1	-10.000%	-15.684%	15.000%	18.248%	-20.000%	-21.494%	1.5E-03
2	-6.376%	-10.983%	12.330%	14.544%	-18.610%	-19.626%	4.4E-05
3	-5.805%	-10.415%	12.717%	14.786%	-18.965%	-19.921%	7.5E-06
4	-5.574%	-10.180%	12.901%	14.909%	-19.040%	-19.972%	1.4E-06
5	-5.476%	-10.078%	12.980%	14.962%	-19.067%	-19.989%	2.6E-07
6	-5.433%	-10.034%	13.013%	14.984%	-19.078%	-19.995%	4.8E-08
7	-5.415%	-10.015%	13.027%	14.993%	-19.082%	-19.998%	9.1E-09
8	-5.407%	-10.006%	13.033%	14.997%	-19.084%	-19.999%	1.7E-09
9	-5.403%	-10.003%	13.036%	14.999%	-19.085%	-20.000%	3.2E-10

Table 6: Iterations of the diversification algorithm for the portfolio weights in Eq. (66).

Example 6

Let us repeat Example 3 from above with all positive weights and correlations. But the starting point for the diversification method is not the originally desired portfolio weights, but instead their negation:

$$\begin{aligned} \text{Weight}_1 &= +10\%, \quad \text{Weight}_2 = +15\%, \quad \text{Weight}_3 = +20\% \\ \text{Weight}_{1,1}^* &= -10\%, \quad \text{Weight}_{2,1}^* = -15\%, \quad \text{Weight}_{3,1}^* = -20\% \\ \rho_{1,2} &= +0.8, \quad \rho_{1,3} = +0.5, \quad \rho_{2,3} = +0.2 \end{aligned} \quad (67)$$

Table 7 shows the diversification method still converges very quickly even though its starting point had incorrect signs. This is because in the first iteration of the diversification algorithm, the weight-update in Eq. (54) flips the sign of the adjusted weight to match the originally desired portfolio weight. Comparing Table 7 and Table 4 shows that it is only the weights in the initial iteration 1 that have the wrong signs, and the adjusted weights are otherwise completely identical for all other iterations.

Iteration k	$\text{Weight}_{1,k}^*$	$\text{FullExp}_{1,k}^*$	$\text{Weight}_{2,k}^*$	$\text{FullExp}_{2,k}^*$	$\text{Weight}_{3,k}^*$	$\text{FullExp}_{3,k}^*$	MSE
1	-10.000%	-15.684%	-15.000%	-18.248%	-20.000%	-21.494%	1.2E-01
2	6.376%	10.983%	12.330%	14.544%	18.610%	19.626%	4.4E-05
3	5.805%	10.415%	12.717%	14.786%	18.965%	19.921%	7.5E-06
4	5.574%	10.180%	12.901%	14.909%	19.040%	19.972%	1.4E-06
5	5.476%	10.078%	12.980%	14.962%	19.067%	19.989%	2.6E-07
6	5.433%	10.034%	13.013%	14.984%	19.078%	19.995%	4.8E-08
7	5.415%	10.015%	13.027%	14.993%	19.082%	19.998%	9.1E-09
8	5.407%	10.006%	13.033%	14.997%	19.084%	19.999%	1.7E-09
9	5.403%	10.003%	13.036%	14.999%	19.085%	20.000%	3.2E-10

Table 7: Iterations of the diversification algorithm for the portfolio weights in Eq. (67).

Example 7

Let us repeat Example 5 above, but with initial guesses for the adjusted weights that are just “crazy”:

$$\begin{aligned} \text{Weight}_1 &= +10\%, \quad \text{Weight}_2 = -15\%, \quad \text{Weight}_3 = +20\% \\ \text{Weight}_{1,1}^* &= -123456\%, \quad \text{Weight}_{2,1}^* = +567890\%, \quad \text{Weight}_{3,1}^* = -912345\% \\ \rho_{1,2} &= -0.8, \quad \rho_{1,3} = +0.5, \quad \rho_{2,3} = -0.2 \end{aligned} \quad (68)$$

Table 8 shows the iterations of the diversification method. The first row shows the initial “crazy” weights. The second row shows the results of adjusting the weights just once using the weight-update from Eq. (54), whose Full Exposure is already much closer to the originally desired portfolio weights. After just a few more iterations the Full Exposure converges to the original weights in Eq. (68). So it seems that the diversification method can easily handle “crazy” wrong guesses for the initial weights.

However, there is one problematic case: The weight-update in Eq. (54) cannot handle an initial weight of zero, because all further adjustments to the weight would also be zero. The implementation in Section 19 takes care of this and uses the original Weight_i as the starting-point for the diversification method, in case the user has erroneously supplied an initial guess that is zero $\text{Weight}_{i,1}^* = 0$.

Iteration k	$\text{Weight}_{1,k}^*$	$\text{FullExp}_{1,k}^*$	$\text{Weight}_{2,k}^*$	$\text{FullExp}_{2,k}^*$	$\text{Weight}_{3,k}^*$	$\text{FullExp}_{3,k}^*$	MSE
1	-123456%	-297103%	567890%	622972%	-912345%	-938753%	4.5E+07
2	4.155%	8.592%	-13.674%	-15.296%	19.437%	20.215%	7.1E-05
3	4.836%	9.389%	-13.409%	-15.219%	19.231%	20.085%	1.4E-05
4	5.151%	9.735%	-13.216%	-15.111%	19.150%	20.037%	2.8E-06
5	5.292%	9.885%	-13.119%	-15.052%	19.114%	20.017%	5.4E-07
6	5.353%	9.950%	-13.073%	-15.023%	19.098%	20.007%	1.0E-07
7	5.380%	9.978%	-13.053%	-15.010%	19.091%	20.003%	1.9E-08
8	5.392%	9.991%	-13.045%	-15.004%	19.088%	20.001%	3.7E-09
9	5.397%	9.996%	-13.041%	-15.002%	19.087%	20.001%	6.9E-10

Table 8: Iterations of the diversification algorithm for the portfolio weights in Eq. (68).

8.12 Convergence

We have now seen several small examples of using the diversification algorithm, which all converged to the optimal solution in just a few iterations of updating the portfolio weights. The question is whether we can be certain that the algorithm will always converge to the optimal solution?

This section is more mathematical and although it can be skipped completely, it is recommended that you at least try to grasp the ideas for proving that the diversification method converges.

Bounds for the Full Exposure

Let us first shorten the notation so the formulas aren't so long. Let $W_i = \text{Weight}_i$ denote the originally desired portfolio weight, and let $W_{i,k}^* = \text{Weight}_{i,k}^*$ denote the adjusted portfolio weight for the k 'th iteration, and let $FE_{i,k}^* = \text{Full Exposure}_{i,k}^*$ denote the Full Exposure for the k 'th iteration.

We want to prove that the Full Exposure $FE_{i,k}^*$ converges to the originally desired weight W_i if we just calculate enough iterations of the adjusted portfolio weights $W_{i,k}^*$ which are used to calculate the Full Exposure $FE_{i,k}^*$. The weights are updated using Eq. (54) from the simple algorithm in Section 8.8.

The Full Exposure from Eq. (38) can be written as follows using the short notation:

$$FE_{i,k}^* = \text{sign}(W_{i,k}^*) \cdot \sqrt{\sum_{j=1}^N |W_{i,k}^* \cdot W_{j,k}^* \cdot \rho_{i,j}^2 \cdot Use_{i,j}|} \quad (69)$$

For simplicity, we consider the absolute value of the Full Exposure $|FE_{i,k}^*|$ which removes the sign:

$$|FE_{i,k}^*| = \sqrt{\sum_{j=1}^N |W_{i,k}^* \cdot W_{j,k}^* \cdot \rho_{i,j}^2 \cdot Use_{i,j}|} \quad (70)$$

Using the absolute value makes the convergence proof easier, as we would otherwise have to keep track of the sign and flip the inequalities for the boundaries when the sign is negative. We can easily restore the sign of $|FE_{i,k}^*|$ when needed, because we know that the signs are always identical (except for the very first iteration, if we have initialized the new weights $W_{i,1}^*$ with other values than the original weights W_i and they have different signs, as explained in Section 8.9):

$$\text{sign}(W_i) = \text{sign}(W_{i,k}^*) = \text{sign}(FE_{i,k}^*) \quad (71)$$

The simple weight-update from Eq. (54) which calculates the new portfolio weight $W_{i,k+1}^*$ using the weight $W_{i,k}^*$ from the previous iteration, its Full Exposure $FE_{i,k}^*$, and the original weight W_i , is then:

$$W_{i,k+1}^* = W_{i,k}^* \cdot \frac{W_i}{FE_{i,k}^*} \quad (72)$$

We are interested in knowing how the Full Exposure for the next iteration $|FE_{i,k+1}^*|$ changes when all of the portfolio weights change from $W_{i,k}^*$ to $W_{i,k+1}^*$, so inserting Eq. (72) into Eq. (70) gives:

$$|FE_{i,k+1}^*| = \sqrt{\sum_{j=1}^N \left| W_{i,k}^* \cdot \frac{W_i}{FE_{i,k}^*} \cdot W_{j,k}^* \cdot \frac{W_j}{FE_{j,k}^*} \cdot \rho_{i,j}^2 \cdot Use_{i,j} \right|} \quad (73)$$

The fraction $W_i/FE_{i,k}^*$ can be pulled out of the summation because it is the exact same value for all the summation indices j . The fraction can also be pulled further outside the square-root, so we get:

$$|FE_{i,k+1}^*| = \sqrt{\left| \frac{W_i}{FE_{i,k}^*} \right|} \cdot \sqrt{\sum_{j=1}^N \left| W_{i,k}^* \cdot W_{j,k}^* \cdot \frac{W_j}{FE_{j,k}^*} \cdot \rho_{i,j}^2 \cdot Use_{i,j} \right|} \quad (74)$$

Now find the index L that minimizes the fraction $W_j/FE_{j,k}^*$ so that for all indices j we have:

$$\left| \frac{W_j}{FE_{j,k}^*} \right| \geq \left| \frac{W_L}{FE_{L,k}^*} \right| \quad (75)$$

And similarly find the index U that maximizes the fraction $W_j/FE_{j,k}^*$ so that for all indices j we have:

$$\left| \frac{W_j}{FE_{j,k}^*} \right| \leq \left| \frac{W_U}{FE_{U,k}^*} \right| \quad (76)$$

Then first using the lower-bound from Eq. (75) to replace the fraction $W_j/FE_{j,k}^*$ in Eq. (74) we get:

$$|FE_{i,k+1}^*| \geq \sqrt{\left| \frac{W_L}{FE_{L,k}^*} \right|} \cdot \sqrt{\left| \frac{W_i}{FE_{i,k}^*} \right|} \cdot \sqrt{\sum_{j=1}^N \left| W_{i,k}^* \cdot W_{j,k}^* \cdot \rho_{i,j}^2 \cdot Use_{i,j} \right|} = \sqrt{\left| \frac{W_L}{FE_{L,k}^*} \right|} \cdot \sqrt{\left| \frac{W_i}{FE_{i,k}^*} \right|} \cdot |FE_{i,k}^*| \quad (77)$$

And then using the upper-bound from Eq. (76) to replace the fraction $W_j/FE_{j,k}^*$ in Eq. (74) we get:

$$|FE_{i,k+1}^*| \leq \sqrt{\left| \frac{W_U}{FE_{U,k}^*} \right|} \cdot \sqrt{\left| \frac{W_i}{FE_{i,k}^*} \right|} \cdot \sqrt{\sum_{j=1}^N \left| W_{i,k}^* \cdot W_{j,k}^* \cdot \rho_{i,j}^2 \cdot Use_{i,j} \right|} = \sqrt{\left| \frac{W_U}{FE_{U,k}^*} \right|} \cdot \sqrt{\left| \frac{W_i}{FE_{i,k}^*} \right|} \cdot |FE_{i,k}^*| \quad (78)$$

Combining these two bounds from Eq. (77) and Eq. (78) we get:

$$\sqrt{\left| \frac{W_L}{FE_{L,k}^*} \right|} \cdot \sqrt{\left| \frac{W_i}{FE_{i,k}^*} \right|} \cdot |FE_{i,k}^*| \leq |FE_{i,k+1}^*| \leq \sqrt{\left| \frac{W_U}{FE_{U,k}^*} \right|} \cdot \sqrt{\left| \frac{W_i}{FE_{i,k}^*} \right|} \cdot |FE_{i,k}^*| \quad (79)$$

Which can be reduced to the following:

$$\sqrt{\left| \frac{W_L}{FE_{L,k}^*} \right|} \cdot \sqrt{|W_i \cdot FE_{i,k}^*|} \leq |FE_{i,k+1}^*| \leq \sqrt{\left| \frac{W_U}{FE_{U,k}^*} \right|} \cdot \sqrt{|W_i \cdot FE_{i,k}^*|} \quad (80)$$

So the Full Exposure $|FE_{i,k+1}^*|$ for the updated portfolio weight $W_{i,k+1}^*$ is in a neighbourhood of $\sqrt{|W_i \cdot FE_{i,k}^*|}$ which is roughly a mid-point between the originally desired portfolio weight $|W_i|$ and the Full Exposure from the previous iteration $|FE_{i,k}^*|$. Figure 17 shows an example of these boundaries.

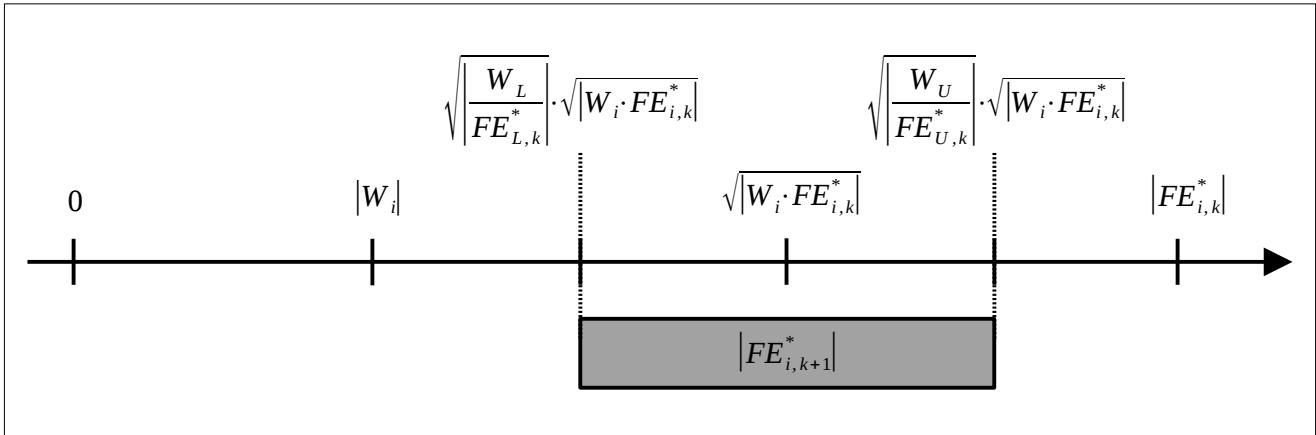


Figure 17: Boundaries for the updated Full Exposure $|FE_{i,k+1}^*$ from Eq. (80).

There are several other possibilities for the boundaries of $|FE_{i,k+1}^*$ than those shown in Figure 17, such as both boundaries being either below or above $\sqrt{|W_i \cdot FE_{i,k}^*|}$ if the boundary ratios $\sqrt{|W_L/FE_{L,k}^*|}$ and $\sqrt{|W_U/FE_{U,k}^*|}$ are either both below or above 1. It is also possible that the boundaries for $|FE_{i,k+1}^*$ can exceed $|W_i|$ and $|FE_{i,k}^*$ if the ratios $\sqrt{|W_L/FE_{L,k}^*|}$ and $\sqrt{|W_U/FE_{U,k}^*|}$ are sufficiently extreme.

Explanation

What does all this mean? We can think of this as the Full Exposure $|FE_{i,k+1}^*$ for the updated portfolio weight $W_{i,k+1}^*$ being the value $\sqrt{|W_i \cdot FE_{i,k}^*|}$ instead of the value $|FE_{i,k}^*$ from the previous iteration. The value $\sqrt{|W_i \cdot FE_{i,k}^*|}$ is a mid-point somewhere between the previous value $|FE_{i,k}^*$ and the originally desired portfolio weight $|W_i|$, so the update from the previous weight $W_{i,k}$ to the new portfolio weight $W_{i,k+1}^*$ using the update from Eq. (72) has indeed brought the Full Exposure $|FE_{i,k+1}^*$ much closer to the originally desired weight $|W_i|$ compared to the previous Full Exposure $|FE_{i,k}^*$.

However, all the other portfolio weights $W_{j,k+1}^*$ for assets $j \neq i$ have also been updated using Eq. (72), which may pull the new Full Exposure $|FE_{i,k+1}^*$ away from the value $\sqrt{|W_i \cdot FE_{i,k}^*|}$. The impact from this is bounded by the two ratios $\sqrt{|W_L/FE_{L,k}^*|}$ and $\sqrt{|W_U/FE_{U,k}^*|}$ as shown in Eq. (80). Because these two ratios are square-roots, they will usually only have a very minor effect on $\sqrt{|W_i \cdot FE_{i,k}^*|}$. Even in case they pull $|FE_{i,k+1}^*$ further away from its goal $|W_i|$ than the previous $|FE_{i,k}^*$, in the next iteration $|FE_{i,k+2}^*$ will be moved back towards the goal $|W_i|$ again, because $\sqrt{|W_i \cdot FE_{i,k+1}^*|}$ is now somewhere between $|FE_{i,k+1}^*$ and the goal $|W_i|$. As the move of the Full Exposure towards the goal $|W_i|$ is always roughly a mid-point between the Full Exposure and the goal, and the pull from the adjustments to the other portfolio weights is at maximum the square-root of the biggest adjustment made to the other portfolio weights; over a few iterations the portfolio weights will all be moved much closer to their individual goals than their mutual pulling on each other.

Normal Initialization

If all the adjusted portfolio weights are initialized with the originally desired weights $W_{i,1}^* = W_i$, then their Full Exposure all exceed the originally desired weight $|FE_{i,1}^*| \geq |W_i|$ according to the criteria from Eq. (37) that the Full Exposure must satisfy. So the weight-updates calculated using Eq. (72) will decrease all the portfolio weights after the first iteration so that $|W_{i,2}^*| \leq |W_{i,1}^*|$. Because the portfolio weights are all being adjusted in the same direction, the Full Exposure $|FE_{i,2}^*|$ will get close to the goal $|W_i|$ for most assets i . Some $|FE_{i,2}^*|$ may be a bit too low and others may be a bit too high compared to their goal $|W_i|$, but they will be close already after one weight-update. In the next iteration they move even closer around $\sqrt{|W_i \cdot FE_{i,2}^*|}$ while only being pulled slightly away from this point, because the two boundary ratios $\sqrt{|W_L / FE_{L,2}^*|}$ and $\sqrt{|W_U / FE_{U,2}^*|}$ in Eq. (80) are already close to 1 as the weights are already close to their goal. So when we initialize the new portfolio weights with the original weights $W_{i,1}^* = W_i$, we get rapid convergence of $|FE_{i,k}^*|$ towards $|W_i|$ in just a few iterations.

“Crazy” Initialization

What if we initialize the adjusted portfolio weights $W_{i,1}^*$ with some “crazy” values as we did in some of the examples above? It actually only takes a single update of the portfolio weights using Eq. (72) to bring the “crazy” values back into the normal range again. To see this first note that the adjusted portfolio weight $|W_{i,k}^*|$ is always less than the Full Exposure $|FE_{i,k}^*|$, because we have defined the Full Exposure to satisfy this criterion from Eq. (37). This means that their ratio is always less than one:

$$|W_{i,k}^*| \leq |FE_{i,k}^*| \Leftrightarrow \left| \frac{W_{i,k}^*}{FE_{i,k}^*} \right| \leq 1 \quad (81)$$

Using this fact in the weight-update formula from Eq. (72), we see that the updated weight $|W_{i,k+1}^*|$ is upper-bounded by the originally desired portfolio weight $|W_i|$ as follows:

$$|W_{i,k+1}^*| = \left| W_i \cdot \frac{W_{i,k}^*}{FE_{i,k}^*} \right| \leq |W_i| \quad (82)$$

The lower bound for the updated weight is zero, and it can only be equal to zero if either the previous weight was zero $|W_{i,k}^*|=0$ or if the originally desired weight is zero $|W_i|=0$. So the updated weight is lower-bounded by zero and upper-bounded by the originally desired weight:

$$0 \leq |W_{i,k+1}^*| \leq |W_i| \quad (83)$$

If we update all the portfolio weights using Eq. (72) so they are all bounded as in Eq. (83), then their Full Exposure is also lower-bounded by zero and upper-bounded by the Full Exposure of the originally desired portfolio weights:

$$0 \leq |FE_{i,k+1}^*| \leq |FE_i| \quad (84)$$

So no matter how “crazy” the initial portfolio weights were, we just need to perform a single weight-update using Eq. (72) to bring the weights and their Full Exposure back into a reasonable range again.

“Crazy” Example 1

Let us consider some more examples with “crazy” initial values for the new portfolio weights. The portfolio has only two assets. The first asset weight is initialized to +4.876% which is the value that it will ultimately converge to, so this initial guess is actually the correct value. But the second portfolio weight is initialized to a “crazy” value of +100000%, which would of course require a completely unrealistically leveraged investment, but we merely want to see what happens with the diversification algorithm when using such a “crazy” value for the initial portfolio weight. The desired and initial portfolio weights, and the correlations between the two assets, are summarized as follows:

$$\begin{aligned} \text{Weight}_1 &= +8\%, \quad \text{Weight}_2 = +12\% \\ \text{Weight}_{1,1}^* &= +4.876\%, \quad \text{Weight}_{2,1}^* = +100000\% \\ \rho_{1,2} &= 0.9 \end{aligned} \tag{85}$$

Table 9 shows the iterations of the diversification method. Note that the Full Exposure in the 1st initial iteration is about +628% for the first asset and about +100000% for the second asset, because the initial guess for the second asset weight is +100000%. Both of these Full Exposures are just “crazy” wrong compared to the originally desired portfolio weights of +8% and +12%, respectively.

But already in the 2nd iteration we get much more reasonable values, so the Full Exposure is around +12% for the second asset, which is the same as the originally desired portfolio weight, as shown in Eq. (85), so the second asset has already reached its goal. But now the first asset has a much too low Full Exposure of only around +0.088% where its goal is +8%. So even though the initial guess for the portfolio weight of the first asset was actually the correct value, because the weight of the second asset was “crazy” wrong, it has pulled the first weight far away from its goal. So we need to increase the weight for the first asset and decrease the weight for the second asset, which is done in the 3rd iteration.

How do we avoid moving back and forth forever between “crazy” wrong portfolio weights? That is because the most “crazy” wrong portfolio weights get moved much closer to their correct values when updating the weights using Eq. (72). This also impacts the portfolio weights of the other assets as we have just seen, but it has a relatively minor effect on the other asset weights, because it is at most the square-root of the weight-adjustments that are propagated to the other assets, which is what Eq. (80) shows. In the next iteration those assets have their portfolio weights moved back again, and after a few more iterations, all the portfolio weights start to converge to their correct values, as shown in Table 9.

Iteration k	$\text{Weight}_{1,k}^*$	$\text{FullExp}_{1,k}^*$	$\text{Weight}_{2,k}^*$	$\text{FullExp}_{2,k}^*$	MSE
1	4.876%	628.448%	100000.000%	100001.975%	5.0E+05
2	0.001%	0.088%	12.000%	12.000%	3.1E-03
3	4.501%	7.606%	10.315%	12.000%	7.7E-06
4	4.848%	7.971%	10.196%	12.000%	4.3E-08
5	4.873%	7.998%	10.187%	12.000%	2.4E-10
6	4.875%	8.000%	10.187%	12.000%	1.3E-12

Table 9: Iterations of the diversification algorithm for the portfolio weights in Eq. (85).

“Crazy” Example 2

In the second “crazy” example, the portfolio still only has two assets whose desired portfolio weights are the same as before, and the initial guess for the first portfolio weight is still +4.876% as before, but now the initial guess for the second portfolio weight is only +0.01%, so we have:

$$\begin{aligned} \text{Weight}_1 &= +8\%, \quad \text{Weight}_2 = +12\% \\ \text{Weight}_{1,1}^* &= +4.876\%, \quad \text{Weight}_{2,1}^* = +0.01\% \\ \rho_{1,2} &= 0.9 \end{aligned} \tag{86}$$

Table 10 shows the iterations of the diversification method. In the 1st initial iteration, the Full Exposure of the first asset is around 4.88% which is much below its goal of 8%, while the second asset has an even lower Full Exposure of only around 0.2% with a goal of 12%. In the 2nd iteration the Full Exposure of the second asset has increased to its goal of 12%, while the Full Exposure of the first asset has also increased to about 11.1% but that is now significantly higher than its goal of only 8%.

The weight of the second asset has increased about 60-fold while the weight of the first asset has only roughly doubled. This is again because the weight-update in Eq. (72) moves each portfolio weight much closer to its correct value, and while this weight-change also propagates to the Full Exposure of the other assets in the portfolio, it only has a minor impact on those, which is at most the square-root of the weight-adjustment, as shown in Eq. (80).

The result is that the diversification algorithm takes some major jumps in the first few iterations to correct the initial “crazy” wrong guesses for the portfolio weights, and then the algorithm quickly converges to the correct values.

Iteration k	$\text{Weight}_{1,k}^*$	$\text{FullExp}_{1,k}^*$	$\text{Weight}_{2,k}^*$	$\text{FullExp}_{2,k}^*$	MSE
1	4.876%	4.880%	0.010%	0.199%	7.5E-03
2	7.996%	11.111%	9.191%	12.000%	4.8E-04
3	5.101%	8.234%	10.111%	12.000%	2.7E-06
4	4.892%	8.017%	10.181%	12.000%	1.5E-08
5	4.877%	8.001%	10.186%	12.000%	8.4E-11
6	4.876%	8.000%	10.187%	12.000%	4.7E-13

Table 10: Iterations of the diversification algorithm for the portfolio weights in Eq. (86).

“Crazy” Example 3

In the third “crazy” example, the portfolio still only has two assets whose desired portfolio weights are the same as before, but now both the initial guesses for the adjusted weights are “crazy” wrong. The first portfolio weight is guessed to be +100000% and the second weight is guessed to be only +0.01%:

$$\begin{aligned} \text{Weight}_1 &= +8\%, \quad \text{Weight}_2 = +12\% \\ \text{Weight}_{1,1}^* &= +100000\%, \quad \text{Weight}_{2,1}^* = +0.01\% \\ \rho_{1,2} &= 0.9 \end{aligned} \tag{87}$$

Table 11 shows the iterations of the diversification method. Because the initial guess for the first weight is “crazy” high, and the initial guess for the second weight is “crazy” low, when we update these two weights using Eq. (72), we expect the weights to pull each other in opposite directions. So the first weight that is “crazy” high gets moved down towards its correct value by its own weight-update, but it also gets pulled upwards to compensate for the other weight being “crazy” low. We expect the opposite for the second weight, which gets moved up towards its correct value by its own weight-update, but at the same time it gets pulled downwards to compensate for the other weight being “crazy” high.

We might imagine that these weight-updates and their pulling on one another would cancel each other out, but because each weight-adjustment only has a comparatively minor impact on the other weights, the combined effect is that the weight-update still brings the weights much closer to their correct values, even though all of the portfolio weights were “crazy” to begin with. This is again what Eq. (80) shows, that the most important effect of a weight-update using Eq. (72) is that the weight is moved much closer to its own goal, while only having a negligible impact on the other portfolio weights.

As usual, we see that after a few iterations of correcting the “crazy” wrong initial guesses for the portfolio weights, the diversification algorithm quickly converges to the correct values that make the Full Exposure of each asset equal to its originally desired portfolio weight, as shown in Table 11.

Iteration k	$\text{Weight}_{1,k}^*$	$\text{FullExp}_{1,k}^*$	$\text{Weight}_{2,k}^*$	$\text{FullExp}_{2,k}^*$	MSE
1	100000.000%	100000.004%	0.010%	28.461%	5.0E+05
2	7.996%	11.111%	9.191%	12.000%	4.8E-04
3	5.101%	8.234%	10.111%	12.000%	2.7E-06
4	4.892%	8.017%	10.181%	12.000%	1.5E-08
5	4.877%	8.001%	10.186%	12.000%	8.4E-11
6	4.876%	8.000%	10.187%	12.000%	4.7E-13

Table 11: Iterations of the diversification algorithm for the portfolio weights in Eq. (87).

General Convergence Criteria

The convergence proof above was made specifically for the definition of Full Exposure from Eq. (38). If you want to use another definition of the Full Exposure with the same diversification algorithm from Section 8.8, then the Full Exposure must satisfy a few more criteria in addition to those in Eq. (37).

Firstly, the Full Exposure of Asset i must change roughly in proportion to a change in the portfolio weight for Asset i . Secondly, changing the weight of Asset i must have a relatively minor impact on the Full Exposure of all the other assets in the portfolio. If these criteria are satisfied along with the other criteria in Eq. (37), then the diversification algorithm from Section 8.8 should still work.

You may find all these arguments a bit too “hand-wavy” and would prefer to see a typical convergence proof where the Full Exposure is shown to converge to the originally desired portfolio weights like this:

$$\lim_{k \rightarrow \infty} |Full\ Exposure_{i,k}^* - Weight_i| = 0 \quad (88)$$

One difficulty in making such a convergence proof, is that the difference $|FE_{i,k}^* - W_i|$ for an individual Asset i can increase for a few iterations before it starts to converge again, as we saw in some of the previous examples. So it may be easier to prove convergence for the Mean Absolute Error of all assets:

$$\lim_{k \rightarrow \infty} \frac{1}{N} \cdot \sum_{i=1}^N |Full\ Exposure_{i,k}^* - Weight_i| = 0 \quad (89)$$

But due to the self-referential (or recursive) nature of how the Full Exposure is defined, this is very challenging to prove directly, without resorting to slightly “hand-wavy” arguments similar to the ones we used in the convergence proof above.

Perhaps it is possible to use more sophisticated mathematics, such as the Banach fixed-point theorem or Cauchy sequences to prove that if the Full Exposure satisfies certain criteria, then the diversification algorithm is guaranteed to converge, and the solution always exists and it is unique. If you are able to make such a proof, then it would be a very valuable contribution to this work.

8.13 Time & Space Complexity

From the above, we know that the diversification algorithm converges to the correct solution, and we now want to find out how much time that takes. The simple diversification algorithm from Section 8.8 has two main loops: The outer-loop is for the iterations that were denoted k in the above, so let us say there are a total of K such iterations of the outer-loop. The inner-loop is for each asset and there are N assets in total. Furthermore, the inner-loop calculates the Full Exposure using Eq. (38) which also has a loop over all N assets. So the time-complexity for the algorithm is:

$$O(K \cdot N^2) \quad (90)$$

When running the diversification algorithm, we typically set a max-value such as 100 for the number of outer-loop iterations K , but the algorithm checks for convergence after each iteration and is usually finished after only a small number of iterations, when a sufficiently precise solution has been found.

Remarkably, the number of iterations required to reach a sufficiently precise solution does not depend on the number of portfolio assets N , but only depends on the level of precision that is required, and the difference between the initial Full Exposure $|FE_{i,1}^*|$ and the originally desired portfolio weight $|W_i|$.

We know from Eq. (80) that for each iteration of the algorithm, the updated portfolio weight $W_{i,k+1}^*$ causes the Full Exposure $|FE_{i,k+1}^*|$ to be approximately $\sqrt{|W_i \cdot FE_{i,k}^*|}$ which is much closer to the goal of $|W_i|$ compared to $|FE_{i,k}^*|$. For simplicity, let us say that $\sqrt{|W_i \cdot FE_{i,k}^*|}$ is about half-way between $|W_i|$ and $|FE_{i,k}^*|$. So after each weight-update the Full Exposure is brought about half-way closer to its goal. The required number of iterations K is therefore related to the required precision as follows:

$$K \simeq \log_2 \left(\frac{\max |FE_{i,1}^* - Weight_i|}{Required Precision} \right) \quad (91)$$

For example, if $\max |FE_{i,1}^* - W_i| = 1000$ and $Required Precision = 0.001$ as used in all the experiments in this paper, then we only need $K \approx 20$ iterations of the algorithm to find portfolio weights with that precision. But this is a quite conservative estimate because it is calculated from the boundaries in Eq. (80). In practice the algorithm converges much faster and usually only needs around 7 iterations to achieve a precision of 0.001 (or 0.1%) for the portfolio weights, regardless of the number of assets.

The factor K in the time-complexity is therefore negligible and remains a constant low number if you always use the same precision requirement. So the time-complexity of the diversification algorithm is dominated by the quadratic number of assets N in the portfolio, that is, the time-complexity is $O(N^2)$.

Figure 18 shows the time-usage of the simple diversification algorithm from Section 8.8 for portfolios of different sizes and with a required precision of 0.001 (or 0.1%) for the adjusted portfolio weights. As can be seen, the time-usage is indeed quadratic in the number of portfolio assets N . For example, the algorithm uses about 0.5 second for a portfolio of $N=5,000$ assets, while it uses about 2 seconds for a portfolio of $N=10,000$ assets, that is, a 4-fold increase in time-usage for a 2-fold increase in portfolio size. There is also roughly a quadratic difference in time-usage between portfolios with $N=10,000$ assets and portfolios with $N=20,000$ assets. And similarly between portfolios with $N=15,000$ and $N=30,000$ assets. So these experiments confirm that the diversification algorithm has quadratic time-complexity in the number of assets N , provided we hold constant the required precision.

The algorithm is extremely efficient and only takes 20 (twenty) milli-seconds to compute for a portfolio of 1000 assets! The time-usage can be further improved by enabling parallelism (this is a simple toggle in the computer code), and removing all of the research options and error-checking to simplify the computer code. The implementation of the Full Exposure in Eq. (38) has been optimized for speed, because it is the most expensive part of the algorithm. The diversification algorithm is already so fast that it can easily be used in back-testing of investment strategies, as well as real-time High-Frequency Trading. Perhaps a highly optimized C++ implementation can be made even faster.

In Figure 18 the original portfolio weights and their correlations are randomly generated using various normal-distributions. For the smaller portfolio sizes, many thousands of random portfolios are generated. For the larger portfolio sizes only 10 random portfolios are generated. See the computer code in Section 19 for details. It is run on a laptop computer with a 2.6 GHz CPU (boost 3.5 GHz).

The space-complexity for the diversification algorithm is linear in the number of portfolio assets N , provided it has been properly implemented so it does not need to allocate memory for extra matrices. However, the correlation matrix itself requires quadratic storage in computer memory, which becomes very large for portfolios with many assets. For example, a portfolio with $N=10,000$ assets requires a correlation matrix with $N^2=100,000,000$ elements to be stored in memory. If you have a very sparse correlation matrix for very large portfolios with many assets, or if the portfolio weights are zero for many assets, then it is possible to improve the diversification algorithm to take advantage of this sparsity and achieve both lower time and space complexity.

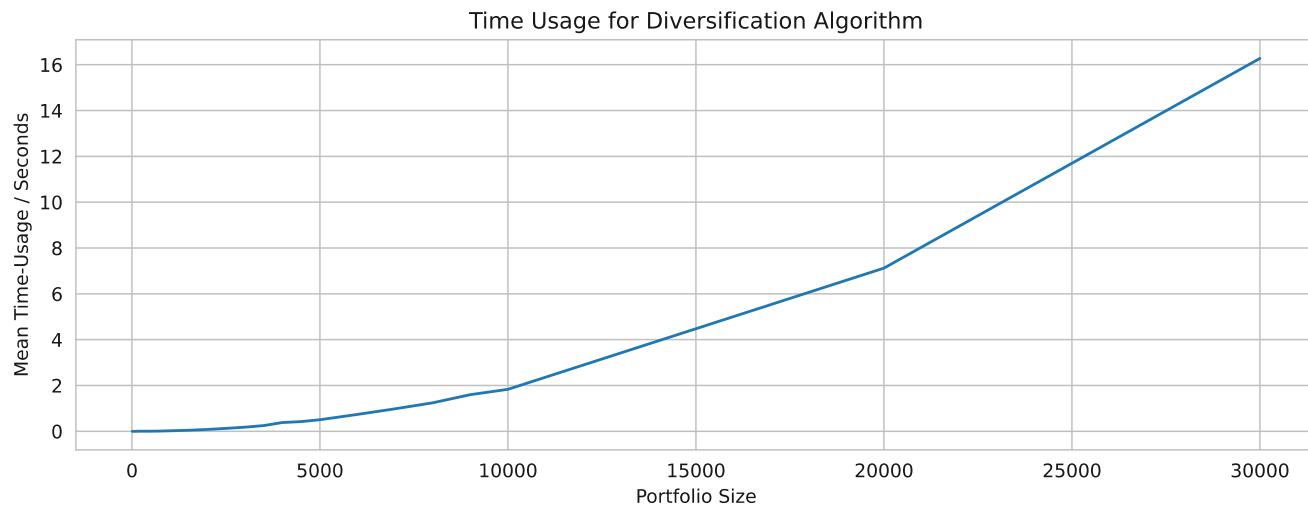


Figure 18: Time usage for the diversification method from Section 8.8 with different portfolio sizes.

8.14 Correlation Forecasting

In order to use the diversification method, you need to provide a reasonable estimate of the future correlations of the asset returns. We know from Section 4 that correlations can change dramatically over time, so a “naive” forecasting of the future correlations from the recent past is going to be quite imprecise. Fortunately the diversification method is very robust to such estimation errors as we will see in the robustness tests on real-world data in Sections 14 and 15. This is because the diversification method only allows the portfolio weights to decrease, so the worst that can happen is that too much of the portfolio is being moved into cash or another low-risk asset of your choice.

Here are a few suggestions for estimating the future correlations. The first thing to consider is how often you intend to rebalance your portfolio. If you only plan on rebalancing your portfolio a few times per year, and you honestly do not care about short-term volatility, then you might want to use the correlations of e.g. 3-month returns instead of daily returns. You can also use correlations from specific historic periods, such as the big stock-market crashes in the years 2009 and 2020. You can also use other correlation measures than the typical Pearson coefficient, for example if you want to lessen the impact of extreme outliers – or perhaps you want to do the opposite and use a correlation measure that focuses more on the extreme outliers. You can also average many different estimates for the future correlations and scale them according to how important you think they are. As long as your correlation estimates are values between -1 and 1, the diversification method from Section 8.8 should still work.

8.15 Summary

This section presented a new method for diversifying an investment portfolio, which takes as input the desired portfolio weights that were created by another process, such as the filtering methods from Section 7. The diversification method also needs an estimate of the future correlations of the asset returns. Then the diversification method adjusts the original portfolio weights downwards, so that the so-called Full Exposure of each asset becomes equal to the originally desired portfolio weights.

The Full Exposure measures how much the portfolio is exposed to each asset, both directly through the investment in that particular asset, but also indirectly through its correlations with other assets in the portfolio. The mathematical formula for the Full Exposure must satisfy certain criteria so it is sensible.

Because of how the Full Exposure is defined, the portfolio weights are only allowed to decrease. This makes the diversification method very robust to estimation errors in the correlation matrix, because the worst that can happen, is that too much of the portfolio is moved into cash with zero returns.

A few algorithms were also presented in this section, for adjusting the portfolio weights to find their correct values. The simplest algorithm was proven to always converge, no matter how “crazy” the initial guesses for the portfolio weights are. The time-complexity of the algorithm is roughly quadratic in the number of assets, and a portfolio of 1000 assets can be optimized in just a few milli-seconds.

9 Test Settings

This section gives an overview of how the tests are performed in the following sections.

9.1 Stock-Data

We use daily share-price data for 949 U.S. stocks between the years 2007 and 2021. The stock-returns are calculated from the so-called Total Return, which is the daily closing share-price adjusted for both stock-splits and reinvestment of dividends, and assuming there were no taxes.

The stock-data is processed and cleaned before it is being used, in order to remove problematic data-points and make the analysis easier and hopefully more reliable. The full data-set contains nearly 3000 stocks, but we remove stocks where the median daily trading-volume is less than USD 1 million, or the max daily return is greater than 100%, or more than 20% of the days have missing data. This results in only 949 stocks remaining in the data-set, which are listed in Section 19.2.

Figure 19 shows the number of stocks available on each day between the years 2007 and 2021. To ensure all stocks have price-data available for the same days, we could either shorten the data-period, or we could remove the stocks that don't have data for the entire period. Either way, this would remove a large part of the data-set, so we instead fill in the missing share-prices using the nearest value. This means that an investment in a stock with missing data simply corresponds to a cash-position during that period. This should not be a problem for comparing the portfolio methods, because it just means that those particular stocks have zero returns during the periods with missing data.

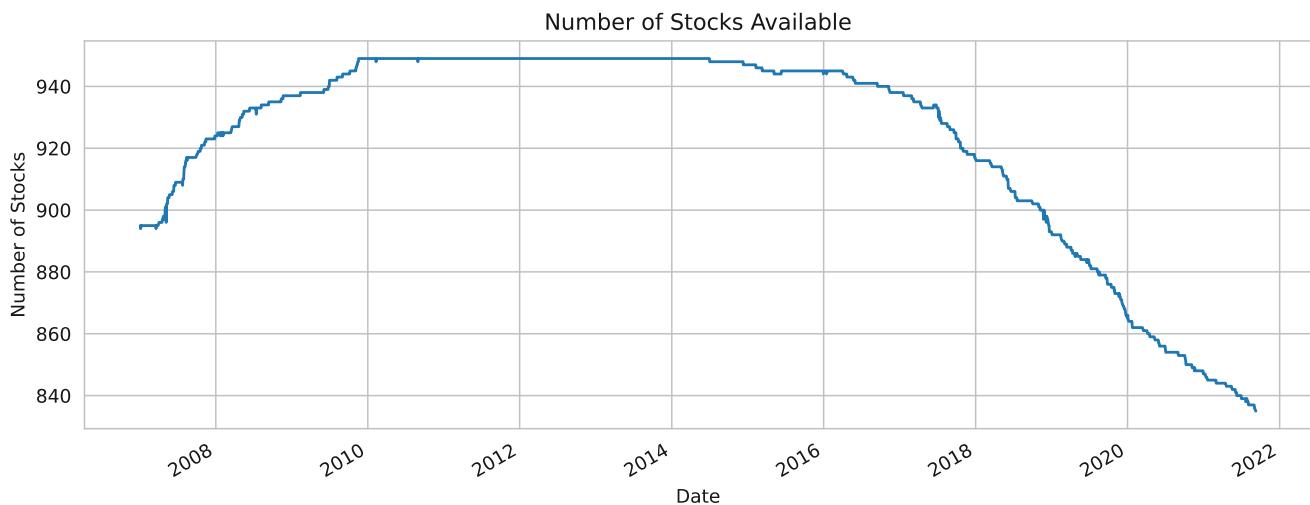


Figure 19: Number of stocks available in the data-set for each day between 2007 and 2021.

9.2 Log-Returns

Whenever we use the future 2-3 year average stock-returns in this paper, it is actually the average log-returns because they can be calculated more efficiently by the computer, and they are fairly close to the returns for values between $\pm 30\%$. For example, $\ln(1 - 20\%) \approx -22.3\%$ and $\ln(1 + 20\%) \approx +18.2\%$.

9.3 Portfolio Methods

We will compare the following portfolio methods against each other:

- **Buy & Hold:** Buys the stocks at the beginning of the data-period and holds them until the end.
- **Rebalanced:** Rebalances the portfolio to equal stock-weights every day. For a portfolio of N stocks in total, each stock is given a portfolio weight of $1/N$ every day.
- **Rebalanced+:** Rebalances the portfolio every day, and also adjusts the stock-weights using the diversification method from Section 8.8. This is different from the Rebalanced portfolio which ensures all stock-weights are always equal. Because the diversification method only allows stock-weights to decrease, it would cause a big part of the portfolio to be held in cash, if the stock-weights were not allowed to be greater than $1/N$ where N is the number of assets in the portfolio. So all the stock-weights are initialized to 10%, and then the diversification method lowers the stock-weights to improve diversification. Finally the weights are normalized so they sum to max 1. Because of this process some of the stock-weights may be greater than $1/N$.
- **Threshold:** This only allows stocks into the portfolio when their estimated future return is above a given threshold, which is set to 10% in all the tests in this paper unless stated otherwise. These stocks all have equal weight in the portfolio for a given day. Each stock-weight is allowed to be max 10% of the portfolio. This is the filtering method from Section 7.1.
- **Adaptive:** Like the Threshold method but this adapts the portfolio-weights so stocks that are estimated to have higher future returns are given higher portfolio weights. This is the filtering method from Section 7.2. It uses Eq. (28) with the parameters $\mu_{min}=10\%$, $\mu_{max}=50\%$ and $Weight_{max}=10\%$, which means that the stock-weight is zero if the future mean stock-return is less than 10%, and the portfolio-weight then increases linearly until the future mean stock-return is 50%, at which point the portfolio-weight is set to 10% which is the max allowed.
- **Adaptive+:** This is the same as the Adaptive portfolio above, but it also processes the portfolio weights using the diversification method from Section 8.8.

Note that the portfolio weights are always ensured to sum to max 1 so investments cannot be made from borrowed money. And although the diversification method can easily handle negative portfolio weights, we only consider positive portfolio weights here (so-called “long” portfolios).

9.4 Omnicient & Robustness Testing

There are essentially three basic parts of using a portfolio method, which are also shown in Figure 20:

- A) Forecast the future financial data that is needed by the portfolio method, such as the future asset-returns, their variance, and their covariance or correlation matrix.
- B) Use the portfolio method to convert the forecasted financial data into portfolio weights.
- C) Buy and sell assets in the financial markets so the portfolio gets the desired composition.

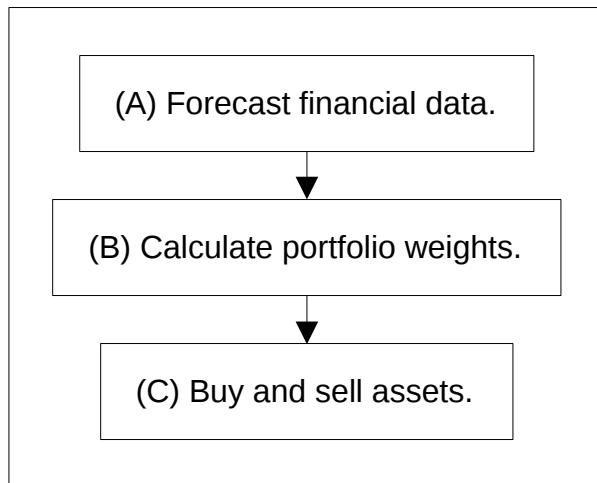


Figure 20: The three basic parts of using a portfolio method for investing.

These three parts are often conflated in academic research papers, where it is assumed that previous financial data is predictive of the future. For example, the mean, variance and covariance for the preceding year may be used to forecast the future when testing the mean-variance portfolio method. Often the assumptions are poorly described so you have to guess exactly how the historical data is being used to forecast the future. This is probably because academia generally has very strong beliefs about the efficiency and randomness of financial markets, so the academic researchers tend to use the same assumptions and therefore do not describe them so carefully. But we saw in Sections 4, 5 and 6 that their beliefs about “naive” forecasting and random walks in the financial markets are incorrect.

Furthermore, some portfolio methods are very vulnerable to estimation errors, while other methods are very robust. The mean-variance method is very vulnerable because it tries to maximize the mean return while simultaneously minimizing the variance, and this can greatly amplify portfolio-weights if you make estimation errors in both the mean return of an asset and its correlation with other assets in the portfolio. This may cause the portfolio to get concentrated in losing assets that are highly correlated.

When we are using some kind of forecasting model for step (A) in Figure 20, we are actually testing both the forecasting model and the portfolio method at the same time, thus conflating the testing of two separate parts of the investment system. So in this paper we will first test the portfolio method using the actual future stock-data. We call this “omniscient” testing which is of course a form of cheating.² This allows us to properly test if the portfolio method works as intended when using the correct predictions about the future stock-returns and their correlations. The omniscient testing is done in Sections 10-12.

Then we test the robustness of the portfolio methods, by adding noise to the omniscient stock-data, to see how well the methods cope with estimation errors in the data. This is done in Sections 13-15.

The last part of the investment system is the buying and selling of assets in the financial markets. This can also interfere with the testing of the portfolio method, if for example one portfolio method is better suited than another method for using stop-loss orders. So to keep this testing as fair as possible, all the portfolio methods are tested by simply buying and selling the stocks at their daily closing prices.

² The academic term for testing with the actual historical data is “ex-post” testing, but I prefer the term “omniscient” which is also Latin and means “all-knowing”.

9.5 Computation Flowchart

Figure 21 shows a flowchart of how we test the new portfolio methods. The different parts are labelled with both letters and numbers, where the letters correspond to the ones used in Figure 20, as follows:

- A) Compute the actual future stock-returns and their correlations. (A.1) is for the actual future 2-3 year average stock-returns. (A.2) is for the actual future correlations of daily stock-returns for a 10-day moving window. (A.3) and (A.4) are omitted in the omniscient testing, but in the robustness testing they add noise to the actual future stock-returns and correlations.
- B) Compute the portfolio weights using the filtering and diversification methods. (B.1) is one of the filtering methods from Section 7, which creates portfolio weights from the actual future 2-3 year average stock-returns. The resulting portfolio weights are all positive and can sum to more than 1 at this stage. (B.2) is the diversification method from Section 8.8, which adjusts the portfolio weights downwards to make their Full Exposure equal to the originally desired weights that were output from the filtering method. (B.3) normalizes the portfolio weights to sum to max 1. If the sum is less than 1, then it means that a part of the portfolio is held in cash.
- C) Simulate the buying and selling of stocks so the portfolio gets the desired composition.

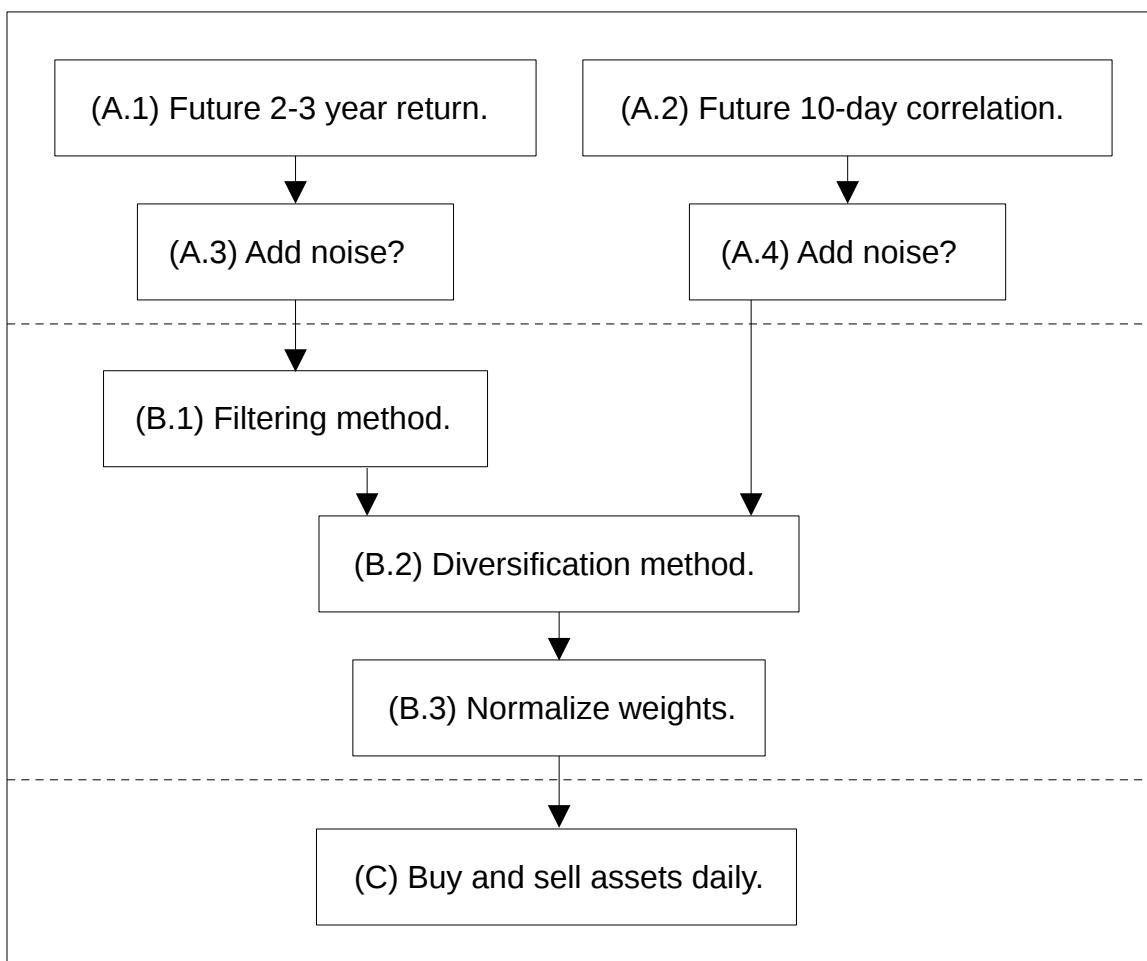


Figure 21: Computation flowchart used for testing the portfolio methods.

9.6 Test Procedure

In the following sections, we will test various aspects of the portfolio methods on many different portfolio sizes. For each test configuration we perform 128 trials where the stock-tickers are chosen at random. For example, in the first omniscient test where we use the actual future stock-returns and their actual future correlations, as described above, we want to test the portfolio methods on different portfolio sizes, so we first create 128 different portfolios each having 5 randomly chosen stocks, then we create another 128 portfolios each having 10 randomly chosen stocks, etc. For each of these randomly chosen portfolios we run all the different portfolio methods. And finally we calculate various performance statistics and make plots for easily comparing the results. So the test procedure is:

- Repeat the following for all the different choices of portfolio size: 5, 10, 30, 50, 100, etc.
 - Repeat the following 128 times to generate the random trials.
 - Select random stocks from the data-set according to the current portfolio size.
 - Simulate the different portfolio methods using the flowchart from Figure 21.
 - Calculate various performance statistics and log the results for later use.
- Make plots with the portfolio values through time and plots with the performance statistics.

Note that we need a fairly high number of trials to make the results statistically significant. The reason we perform exactly 128 trials is because that number is divisible with 8, which is the number of CPU cores / threads available in the computer where these tests were run. The computer code in Section 19 supports parallel execution which makes the tests run much faster. It typically takes around 45 minutes to run a test in parallel on a 4-core (8 thread) CPU with 2.6 GHz (3.5 GHz boost).

10 Test A – Full Data Period (Omniscient)

This is the first “omniscient” test which uses the actual future 2-3 year stock-returns, as well as the actual future 10-day stock-correlations. We use the entire period available in the data-set between the years 2007 and 2021, but since we are using the future 2-3 year average stock-returns in some of the portfolio methods, the portfolio returns can only be calculated between the years 2007 and 2018. The test procedure is shown in Section 9.6, and all the plots are gathered towards the end of this section.

10.1 Test A – Portfolio Values

We first compare the portfolio values when using the different portfolio methods, to see which method performed best over a decade of investing. We show the ratio between the portfolio values of two portfolio methods to see which method was best. A ratio of 1 means that the two portfolio methods had equal portfolio value at that time, and ratios either below or above 1 means that one or the other portfolio method had performed better up until that point in time.

The computer code in Section 19 generates the plots with portfolio values for all these portfolio sizes: 5, 10, 30, 50, 70, 100, 150, 200, and 300, but to save space in this paper, we only show the plots for 5, 30, 100 and 300 stocks here, and in the following tests we only show them for 30, 100 and 300 stocks.

Test A – Portfolio Values for 5 Stocks

Figure 22 compares the portfolio values when each portfolio has 5 random stocks. There are 128 trials where we generate a portfolio with 5 random stocks. The same portfolios of 5 random stocks are used for all the portfolio methods, so the comparison is fair because it is made on the same stock-data. The different portfolio methods were listed and explained in Section 9.3.

The top-plot in Figure 22 compares the Rebalanced and the Buy&Hold portfolios. The dashed black line shows when the ratio is 1 so the two portfolio values were equal at that point in time. When the ratio is above 1 it means that the Rebalanced portfolio had the highest value at that point in time, and vice versa when the ratio is below 1 it means that the Buy&Hold portfolio had the highest value at that point in time. As can be seen from the plot, the ratio of the portfolio values are about 1.3 on average in the final year 2018, which means that the Rebalanced portfolios were on average roughly 30% better than the Buy&Hold portfolios in the year 2018. However, most of this advantage probably comes from the stock-market crash and recovery around year 2009. As demonstrated in my previous paper [Pedersen 2021], it appears that Rebalancing is beneficial in stock-market recoveries. After the year 2009, the ratio of the portfolio values appears to be roughly horizontal, which means that the two portfolio methods performed roughly equal after the stock-market crash and recovery in 2009.

The 2nd plot in Figure 22 shows the ratio between the Rebalanced+ and Rebalanced portfolios. Recall from Section 9.3 that the Rebalanced+ portfolio also uses the diversification method from Section 8.8. It appears from the ratio of the portfolio values that the Rebalanced+ portfolios performed much better than the normal Rebalanced portfolios during the stock-market crash in 2009, but then performed much worse in the following years. As we will discuss in more detail below, the reason can be found in

Figure 30 and is that the Rebalanced+ portfolios had on average more than 60% of the portfolio placed in cash. This is typical for small portfolios because the diversification method needs quite large portfolios to function properly. So the under-performance of the diversified Rebalanced+ portfolios did not arise from a bad methodology, but simply from having placed a majority of the portfolio in cash.

The 3rd and middle plot in Figure 22 compares the portfolio values of the Threshold and Rebalanced portfolio methods. We see a similar pattern as before, namely that the Threshold method performed much better during the 2009 stock-market crash, but in the following years it started to perform worse than the simple Rebalanced portfolios. The reason is also similar, namely that the Threshold portfolio only invests when the future 2-3 year average stock-returns would give an annualized return of at least 10%. Because the portfolio only contains 5 stocks, there are long periods where the Threshold method simply cannot invest in any stocks and therefore holds a lot of the portfolio in cash, which is again shown in Figure 30.

The 4th plot in Figure 22 compares the portfolio values of the Adaptive and Threshold portfolio methods, neither of which employs the diversification method. The only difference between these two portfolio methods, is that the Adaptive method adjusts or adapts the portfolio weights to the magnitude of the future 2-3 year stock-returns, while the Threshold method simply gives the stocks equal weights in the portfolio if their future 2-3 year stock-returns exceed 10%. We once again see from the plot, that the Adaptive method performed much better during the 2009 crash, but under-performed the Threshold method in the following years. Figure 30 shows that the Adaptive portfolios had on average more than 90% of their portfolios in cash. So a portfolio with only 5 stocks is simply too small for the Adaptive portfolio method to work properly and keep the portfolio fully invested.

The 5th and final plot in Figure 22 compares the portfolio values of the Adaptive+ and Adaptive portfolios. The only difference between these two portfolio methods is that the Adaptive+ also uses the diversification method from Section 8.8, which gave a small advantage in the 2009 crash, but in the following years the Adaptive+ method significantly under-performed the Adaptive method. Figure 30 shows the likely cause was again that the Adaptive+ portfolios had even bigger cash-positions.

Test A – Portfolio Values for 30 Stocks

Figure 23 compares the portfolio methods when each portfolio contains 30 random stocks. Now the Rebalanced+ portfolios perform significantly better than the Rebalanced portfolios. And the Threshold portfolios perform much better than the Rebalanced portfolios. But the Adaptive portfolios are sometimes better and sometimes worse than the Threshold portfolios. And the Adaptive+ portfolios are only better than the Adaptive portfolios during particular periods such as the 2009 crash, and otherwise the Adaptive+ portfolios mostly under-perform the Adaptive portfolios. So portfolios with only 30 stocks are also insufficient for the Adaptive+ portfolio method to function properly.

Test A – Portfolio Values for 100 Stocks

Figure 24 compares the portfolio methods when each portfolio contains 100 random stocks. The Rebalanced+ portfolios were sometimes better and sometimes worse than the Rebalanced portfolios. But the Threshold portfolios always performed much better than the Rebalanced portfolios, and after

nearly 12 years the Threshold portfolios had roughly 5-9 times higher portfolio values. The Adaptive portfolios also performed much better than the Threshold portfolios, except during the 2009 crash, and after nearly 12 years the Adaptive portfolios had roughly 2-3 times higher portfolio values. The Adaptive+ portfolios were generally better than the Adaptive portfolios, but this seems to be mostly between the years 2007 and 2013, and after 2013 the Adaptive+ portfolios seem to have been worse than the Adaptive portfolios, which can be seen from the declining ratios of the portfolio values. So portfolios with 100 stocks are sufficient for the Threshold and Adaptive portfolio methods, but still insufficient for the Adaptive+ method to function properly.

Test A – Portfolio Values for 300 Stocks

Figure 25 compares the portfolio methods when each portfolio contains 300 random stocks. The Rebalanced+ portfolios were sometimes worse and sometimes better than the Rebalanced portfolios. But the Threshold portfolios were consistently much better than the Rebalanced portfolios, and after nearly 12 years the Threshold portfolios had roughly between 6-8 times higher portfolio values (corresponding to about 16-19% excess return per year). The Adaptive portfolios were also consistently better than the Threshold portfolios, achieving roughly 3-4 times higher portfolio values after nearly 12 years (corresponding to about 10-12% excess return per year). And the Adaptive+ portfolios were also consistently better than the Adaptive portfolios, achieving roughly between 4-8 times higher portfolio values after nearly 12 years (corresponding to about 12-19% excess return per year).

We can conclude from these experiments, that once there are enough assets available for investment, the portfolio methods that rely on filtering and diversification work extremely well, at least when they are using the actual future 2-3 year stock-returns and the actual future 10-day stock-correlations.

10.2 Test A – Performance Statistics

The portfolio values are a simple way of comparing the long-term performance of portfolio methods. We will now compare various performance statistics such as the mean and standard deviation for the daily returns of the different portfolios. We use box-plots to show the distributions for the performance statistics, which makes it easy to compare the performance of the portfolio methods against each other.

There is no statistical hypothesis testing done, because there are so many combinations of portfolio methods to compare with e.g. paired t-tests that it would either clutter these plots, or require their own plots. Furthermore, the differences between the portfolio methods are often very large, and as there are 128 random trials for each choice of portfolio size, the large differences are statistically significant.

Each plot has 9 box-plots for the different choices of portfolio sizes from 5 up to 300. Each box-plot compares the 6 different portfolios methods, by showing the distribution of a given performance statistic for each choice of portfolio method and portfolio size. The 128 random portfolios used to calculate these performance statistics are the exact same as the 128 random portfolios that were used to make the previous plots of the portfolio values, so the plots are consistent.

Also note that when we say “portfolio size” we do not mean the number of stocks that the portfolio is actually invested in. The “portfolio size” can be 300 but the portfolio method may not have invested in

any of them. The words “portfolio size” really mean “the size of the universe available for investing”, and when that is e.g. 300 it means that there are 300 random stocks available for investing.

Test A – Arithmetic Mean Return

Figure 26 compares the arithmetic mean of the daily returns for the different portfolio methods. For small portfolio sizes of only 5 or 10 stocks, the Adaptive and Adaptive+ portfolios perform much worse than the trivial Buy&Hold and Rebalanced portfolios. For portfolios of 10 stocks the Threshold portfolios begin to perform better, and especially for portfolios of 30 stocks and above, the Threshold portfolios are much better than the trivial Buy&Hold and Rebalanced portfolios. The Adaptive and Adaptive+ portfolios also start to perform better than the trivial portfolios when there are 30-50 stocks or more, but the Adaptive+ portfolios still under-perform the Threshold portfolios until the portfolios have 70 stocks or more. When the portfolios have 100 stocks, the Adaptive and Adaptive+ portfolios have roughly the same performance, and for portfolios with 150 or more stocks, the Adaptive+ portfolios are much better than the Adaptive portfolios, which are in turn much better than the Threshold portfolios, which are also much better than the trivial Buy&Hold and Rebalanced portfolios.

So the Threshold, Adaptive and Adaptive+ portfolio methods need increasingly larger portfolio sizes to function properly and improve the mean return over the other portfolio methods. The explanation is found in Figure 30 which shows that for smaller portfolio sizes, these three portfolio methods place a very large part of the portfolio in cash. So their lower mean return is not due to a malfunction in the portfolio methods, but it is just a by-product of how they protect the portfolio from losses – they simply move a lot of the portfolio into cash when there are no investments available with sufficiently high returns; and in the case of the Adaptive+ method, it lowers the portfolio weights to limit correlation between stocks. So these portfolio methods are only fully invested when there are sufficiently many assets to choose from, but then they also perform extremely well.

For a portfolio size of 300 (i.e. an investment universe where 300 stocks are available for investment), the trivial Buy&Hold portfolios have a mean daily return of about 1.0005, which corresponds to an annualized return of about $1.0005^{250} - 1 \approx 13.3\%$ when assuming there are 250 trading-days in a year. The Threshold portfolios have a mean daily return of about 1.0013, which corresponds to an annualized return of about 38%. The Adaptive portfolios have a mean daily return of about 1.00175, which corresponds to an annualized return of about 55%. The Adaptive+ portfolios have a mean daily return of about 1.00225, which corresponds to an annualized return of about 75%.

So these portfolio methods work extremely well for larger portfolios of e.g. 300 stocks, if they are given the actual future 2-3 year average stock-returns and the actual future 10-day stock-correlations. This is of course “cheating” but it tells us that the portfolio methods do work as intended when they are given the correct input.

Also note that the Threshold and Adaptive portfolio methods work so well when their input is the future 2-3 year average stock-returns. The portfolio methods don’t know what happens in the interim years, only what will happen 2-3 years into the future, yet the two portfolio methods still perform extremely well, even though they are basically just very simple filtering processes. And perhaps even

more remarkable is that the diversification method can greatly improve on that performance, simply by adjusting the portfolio weights according to the stock-correlations in the next 10 days.

The explanation that the simple filtering methods work so well when using the future 2-3 year returns, is that even though some or all of the stocks might crash in the interim period, we know that they will recover again within a few years, because we are using the actual future 2-3 year stock-returns when making the investments. So any interim losses will only be temporary. And if some stocks crash deeper than others, it will be beneficial to move more of the portfolio into other stocks that will have a higher return in the following 2-3 year period. This is further improved by the diversification method, which tries to avoid that all the stocks in the portfolio crash at the same time (but this also avoids that they all increase at the same time). As we have just seen, this works extremely well when the portfolio methods are using the actual future stock-returns and correlations. In the robustness tests further below, we will see how well the methods cope with noisy estimates of the future stock-returns and correlations.

Test A – Geometric Mean Return

Figure 27 compares the geometric mean of the daily returns, which are almost the exact same as the arithmetic mean returns in Figure 26, so the geometric means are omitted in the other tests below.

Test A – Standard Deviation

Figure 28 compares the standard deviation of the daily returns for the different portfolio methods. For small portfolio sizes, the Threshold, Adaptive and Adaptive+ portfolios have much lower standard deviations than the trivial Buy&Hold and Rebalanced portfolios, but this is simply because a very large part of those portfolios were held in cash, as shown in Figure 30.

For larger portfolio sizes, the Threshold and Adaptive portfolios have slightly higher standard deviations compared to the Buy&Hold and Rebalanced portfolios, while the Adaptive+ portfolios have about the same or perhaps slightly lower standard deviation, perhaps because the Adaptive+ portfolios still have a significant part of their portfolios in cash.

But note that the Rebalanced+ portfolios have significantly lower standard deviation than all the other portfolio methods. This means the diversification method has significantly lowered the daily volatility compared to all the other portfolio methods, when doing simple daily rebalancing of the portfolio.

Test A – Sharpe Ratio

Figure 29 compares the so-called Sharpe Ratio of the daily returns for the different portfolio methods, which is calculated here as the arithmetic mean daily return minus one, divided by the standard deviation. So it measures how high a portfolio's daily return was on average relative to the portfolio's daily volatility, where a higher Sharpe Ratio is considered better.

For all the different portfolio sizes, the portfolio methods have increasingly higher Sharpe Ratios, so that the Adaptive+ portfolios have higher Sharpe Ratios than the Adaptive portfolios, which are in turn higher than the Threshold portfolios, which are higher than the Rebalanced+ portfolios, which are higher than the Rebalanced portfolios, which are slightly higher than the Buy&Hold portfolios.

For small portfolio sizes this is partly because the Threshold, Adaptive and Adaptive+ portfolios have a large part of their portfolios in cash. But it shows that in general these portfolio methods are very effective at improving the mean daily return of the portfolio, without incurring a higher daily volatility.

Test A – Cash Mean

Figure 30 compares the average daily cash position for the different portfolio methods, which has already been referenced several times in the analyses above. This shows that even the Rebalanced+ method needs a portfolio size of at least 30 stocks to be fully invested most of the time. The Threshold method needs a portfolio size of at least 50 to be nearly fully invested. The Adaptive method is nearly fully invested for a portfolio size of 300, while the Adaptive+ method is still less than 90% invested on average for a portfolio size of 300, so the Adaptive+ method might benefit from even larger portfolios.

Test A – Months With Losses

Figure 31 compares the percentages of months with losses for the different portfolio methods. The relation between the portfolio methods seems to be roughly the same regardless of the portfolio size.

For small portfolio sizes, the trivial Buy&Hold and Rebalanced portfolios have roughly 40% of months with losses, while the Threshold, Adaptive and Adaptive+ portfolios have a significantly lower 20-30% of months with losses. This is probably to be expected when considering that those portfolio methods had very large cash positions for smaller portfolio sizes, as shown in Figure 30.

For large portfolio sizes, the trivial Buy&Hold and Rebalanced portfolios had roughly 35% of months with losses, while the Threshold, Adaptive and Adaptive+ portfolios were nearly fully invested, yet they also had much fewer months with losses, at around 25% for the Threshold portfolios, 22% for the Adaptive portfolios, and only 15% of months had losses for the Adaptive+ portfolios.

Also note that Rebalanced+ portfolios had fewer months with losses than the Rebalanced portfolios. So the filtering and diversification methods both help with lowering the number of months with losses, and their combination in the Adaptive+ portfolios is especially effective at lowering the monthly losses.

Test A – Max Drawdown

Figure 32 compares the so-called Max Drawdown for the different portfolio methods. This is the maximum loss incurred from a peak to a subsequent valley of the portfolio values over time.

For smaller portfolio sizes, the trivial Buy&Hold and Rebalanced portfolios have experienced very large losses of 60% or more, while the Threshold, Adaptive and Adaptive+ portfolios only had losses of 10-20%. This is because most of their portfolios were held in cash, as shown in Figure 30.

As the portfolio sizes increase, the box-plots show that the distributions of losses become more narrow. This is because the portfolios are now a larger part of the entire stock-market, so a stock-market crash of 50-60% affects many of the stocks in the market.

For larger portfolio sizes, the Threshold, Adaptive and Adaptive+ portfolios only seem to have slightly better Max Drawdowns than the trivial Buy&Hold and Rebalanced portfolios, which is to be expected, because the filtering process uses the future 2-3 year mean returns, and does not try to predict if there is

a stock-market crash in the interim period. If you can predict a stock-market crash, you can of course improve the Max Drawdown significantly, but in this test we only use the future 2-3 year mean return.

Test A – Max Pullup

Figure 33 compares the so-called Max Pullup for the different portfolio methods. This is the maximum gain experienced within one year from a valley to a subsequent peak in the portfolio values. This is useful for measuring how well a portfolio method performs during a stock-market recovery.

For very small portfolios with only 5 or 10 stocks, the Threshold, Adaptive and Adaptive+ portfolios have much lower Max Pullup than the trivial Buy&Hold and Rebalanced portfolios. But for portfolios with 30 stocks or more, the Threshold and Adaptive portfolios are superior. And for portfolios with 70 stocks or more, the Adaptive+ portfolios are far superior to all the others. For portfolios with 300 stocks, the Adaptive+ portfolios have a median Max Pullup of nearly 500%, that is, a 6-fold gain in portfolio value! This is compared to a median Max Pullup of around 250% for the Adaptive portfolios, and maybe around 180% for the Threshold portfolios, and maybe around 100% for the Buy&Hold portfolios. So the combination of adaptive filtering and portfolio diversification gives a tremendous boost to returns during recoveries from stock-market crashes.

10.3 Summary

These experiments have shown on several different performance metrics, that in order for the filtering and diversification methods to become truly effective, the investment universe needs to contain at least a few hundred assets, so the filtering method can select enough assets for inclusion into the portfolio, and the diversification method can lower their desired portfolio weights without moving a significant part of the portfolio into cash, which is what often happens for portfolios with only a few assets.

Although the tests have shown that there is benefit to using the diversification method alone, it works best in combination with the filtering method, which only allows assets into the portfolio if their future returns are sufficiently high. And this filtering works best when it adapts to the magnitude of the future return estimates, so an asset with a higher estimated return is given a bigger position in the portfolio.

In this section we used so-called omniscient data, which is the actual future 2-3 year mean returns, and the actual future 10-day stock-correlations. We will make a few more tests using omniscient data in the following sections, and then we will make several robustness tests to see how well the portfolio methods work when the estimates of the future stock-returns and correlations are very noisy.

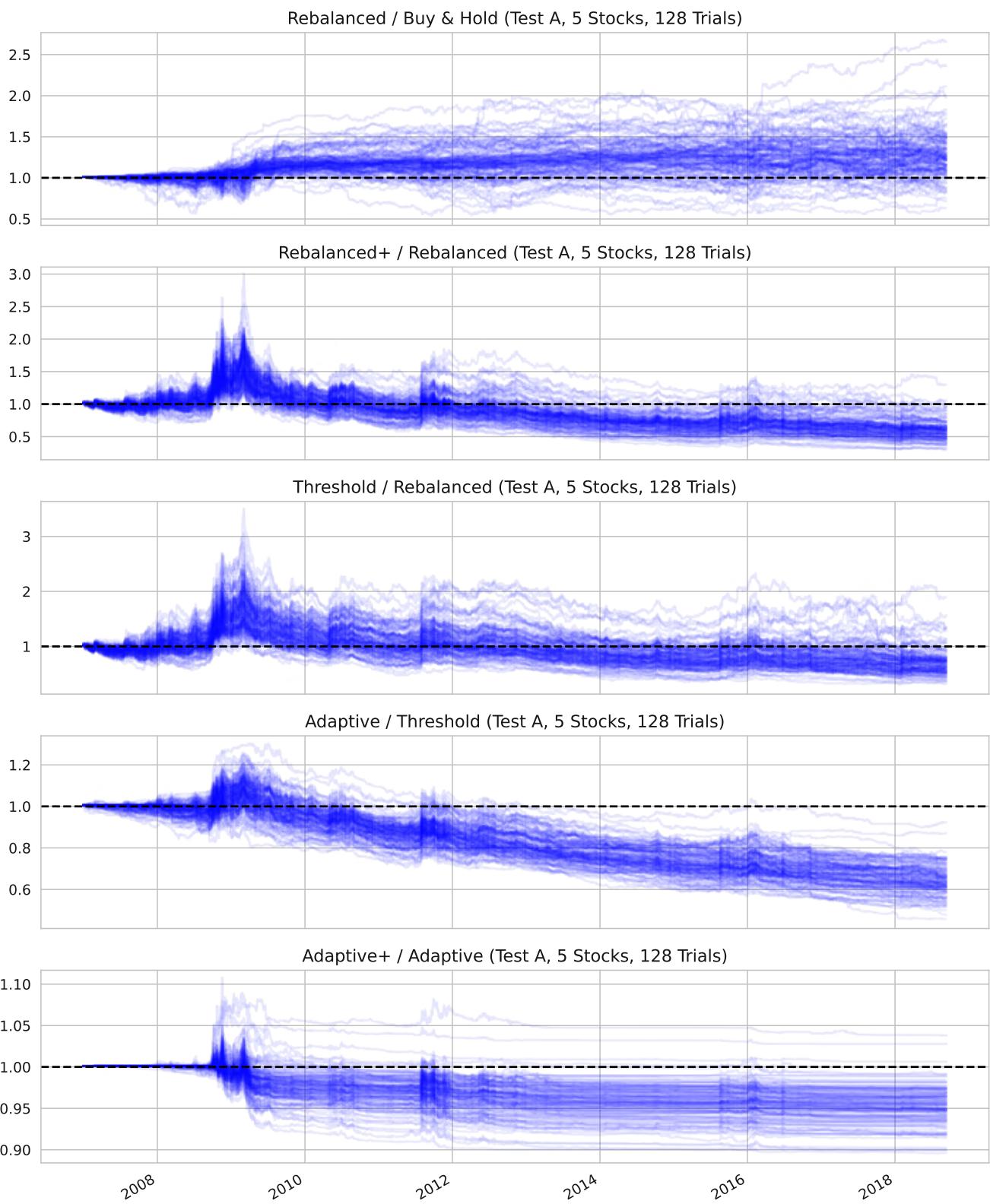


Figure 22: **Test A** – Compare the values of 128 random portfolios with 5 stocks each.

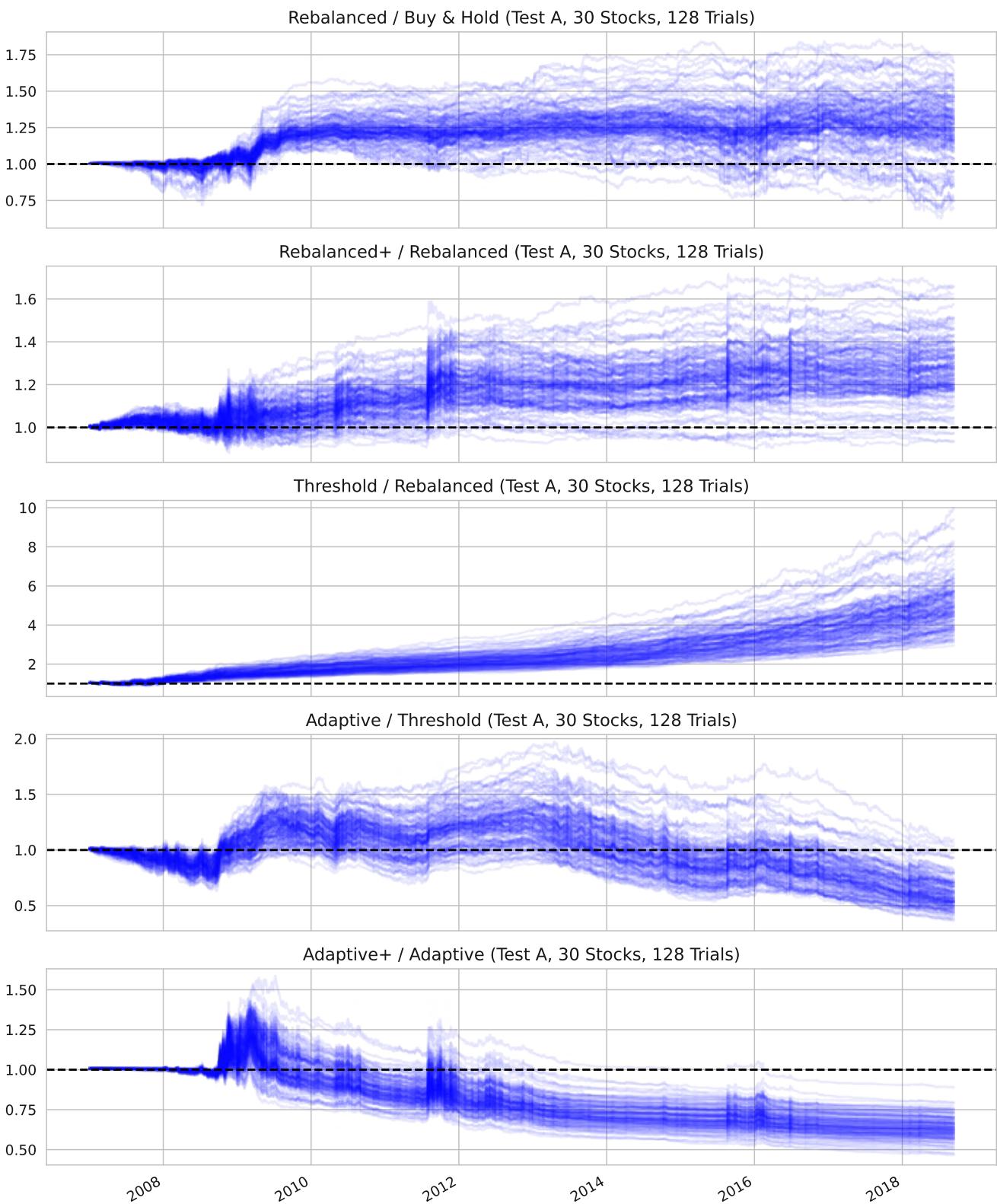


Figure 23: Test A – Compare the values of 128 random portfolios with **30 stocks** each.

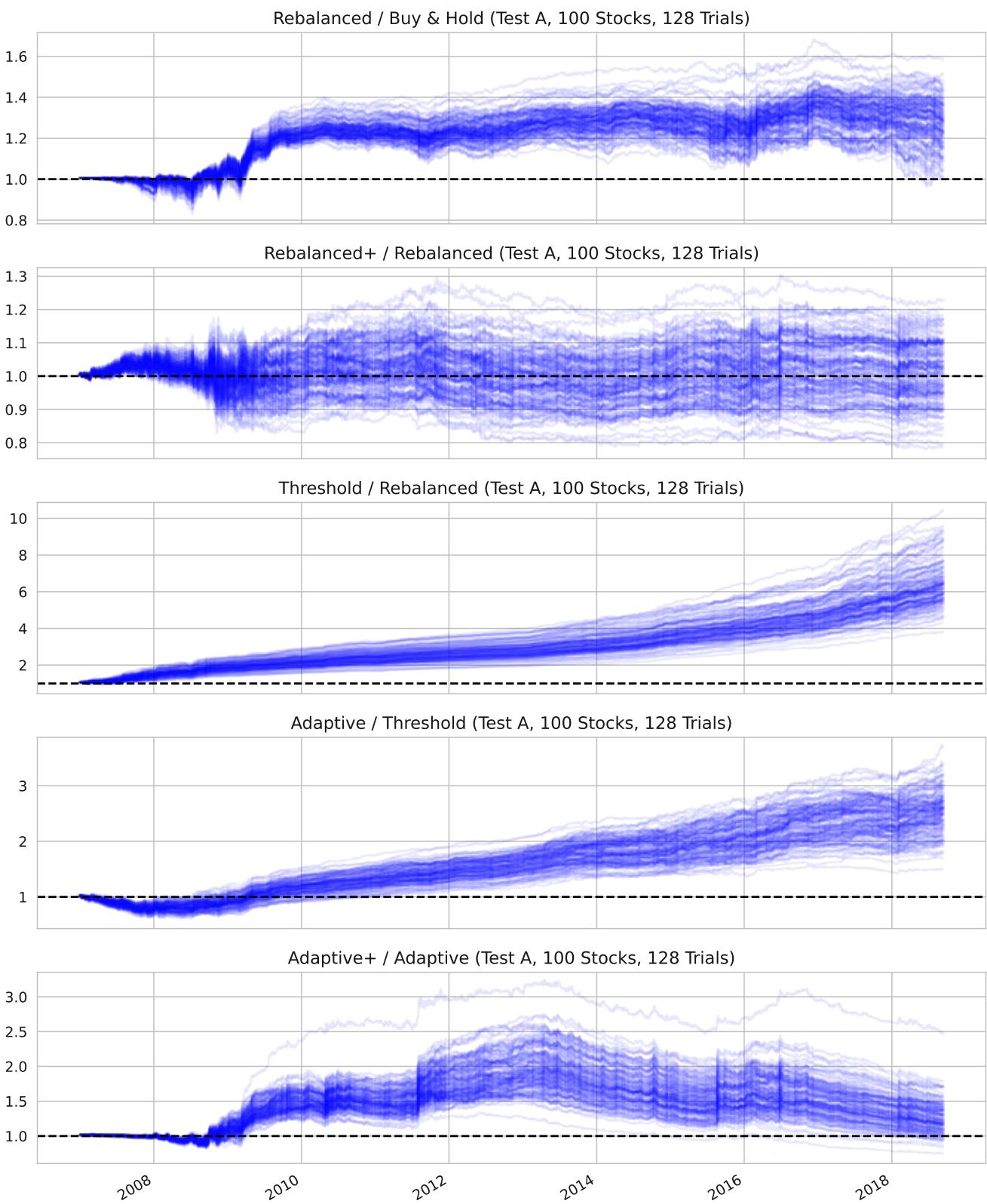


Figure 24: **Test A** – Compare the values of 128 random portfolios with **100 stocks** each.

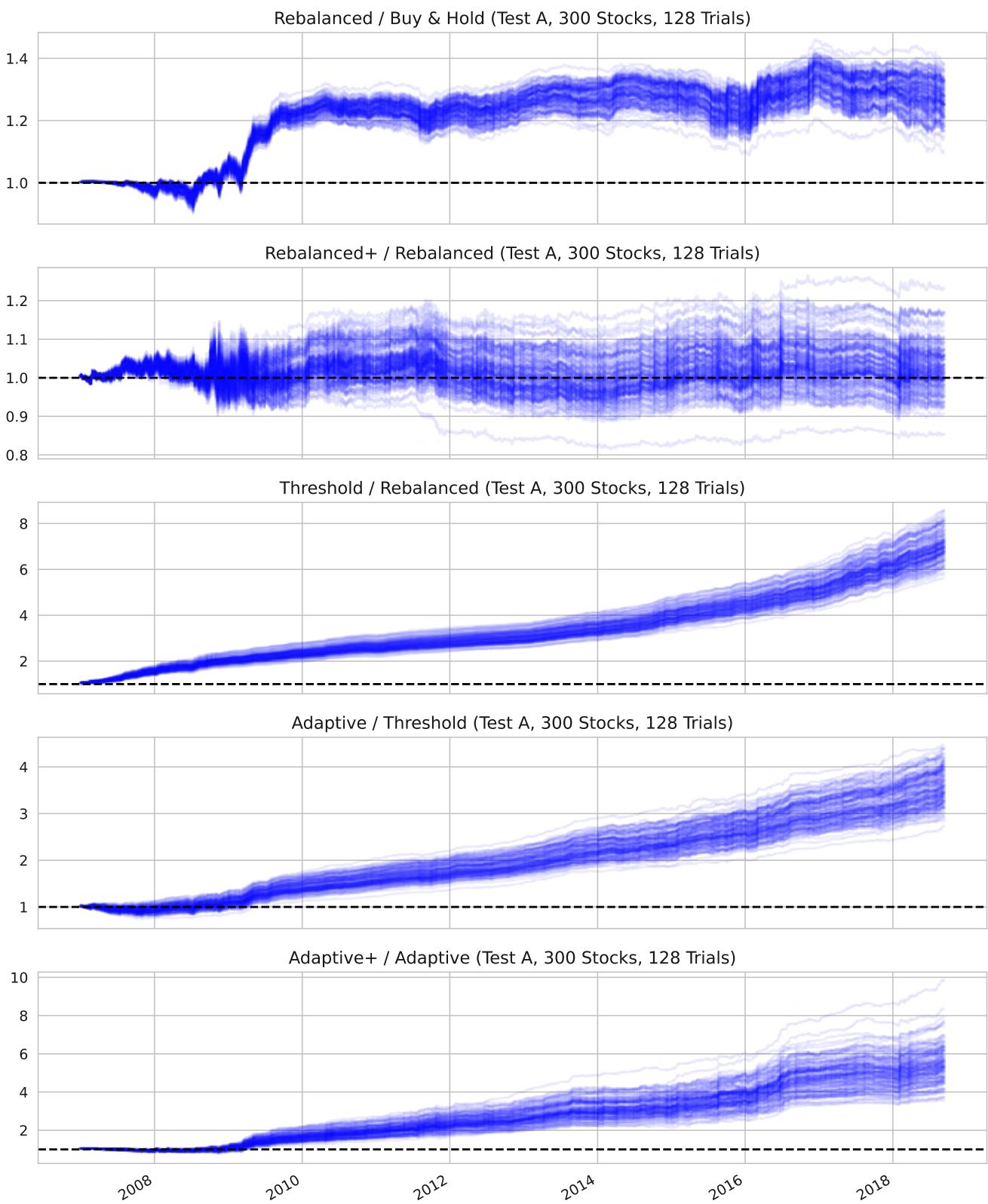


Figure 25: **Test A** – Compare the values of 128 random portfolios with **300 stocks** each.

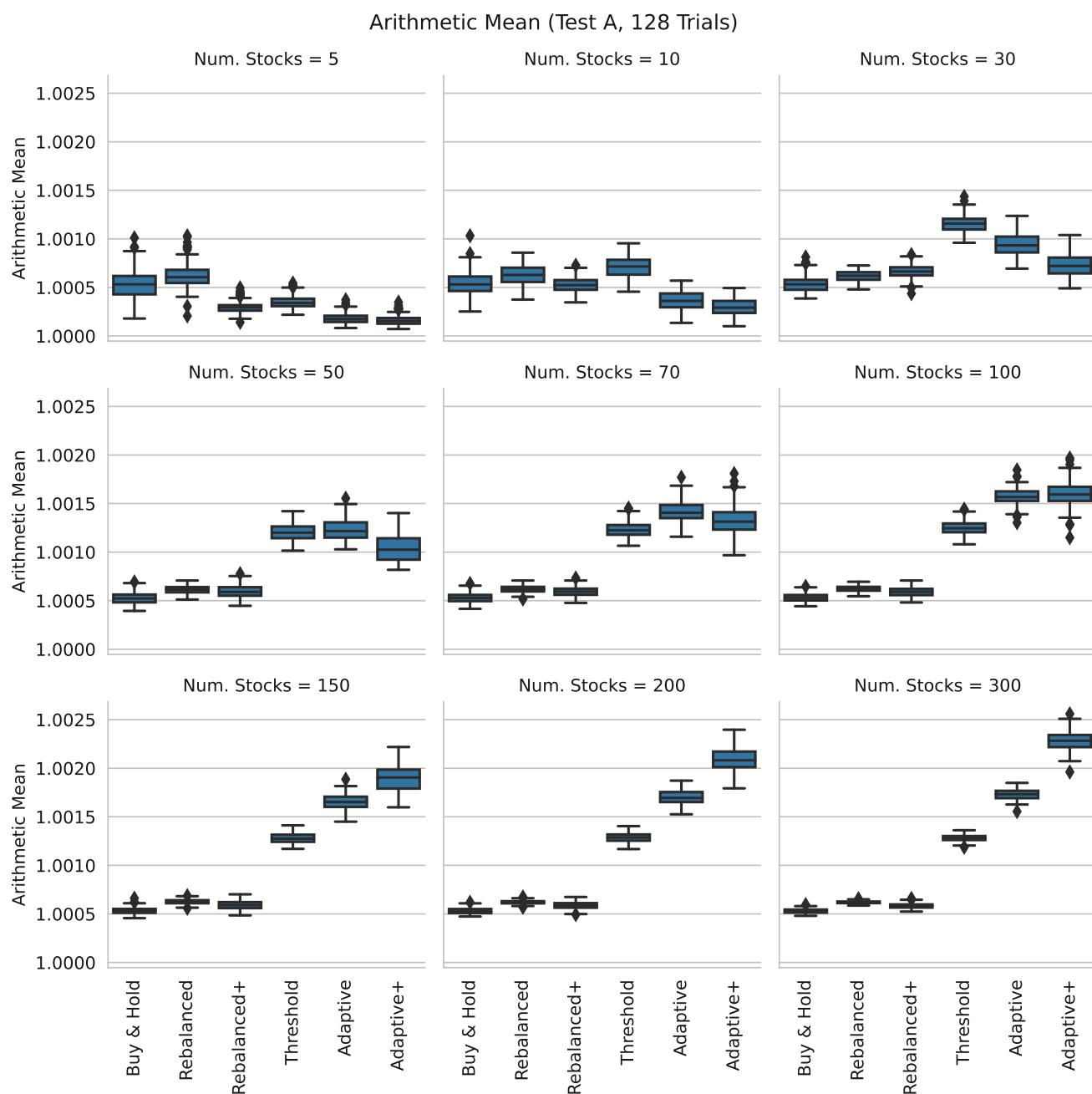


Figure 26: Test A – Compare Arithmetic Mean daily return of different portfolio methods and sizes.

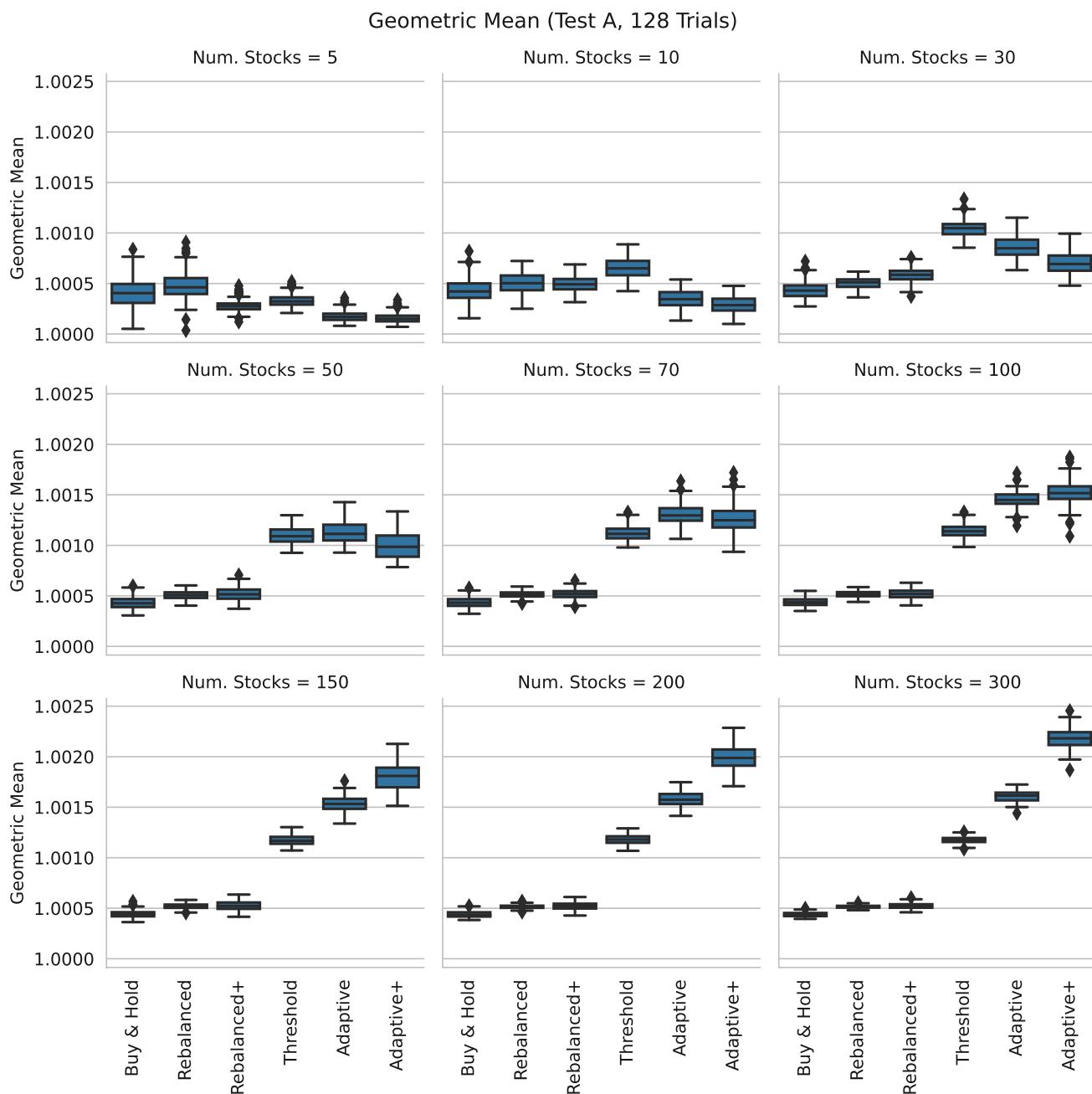


Figure 27: **Test A** – Compare **Geometric Mean** daily return of different portfolio methods and sizes.

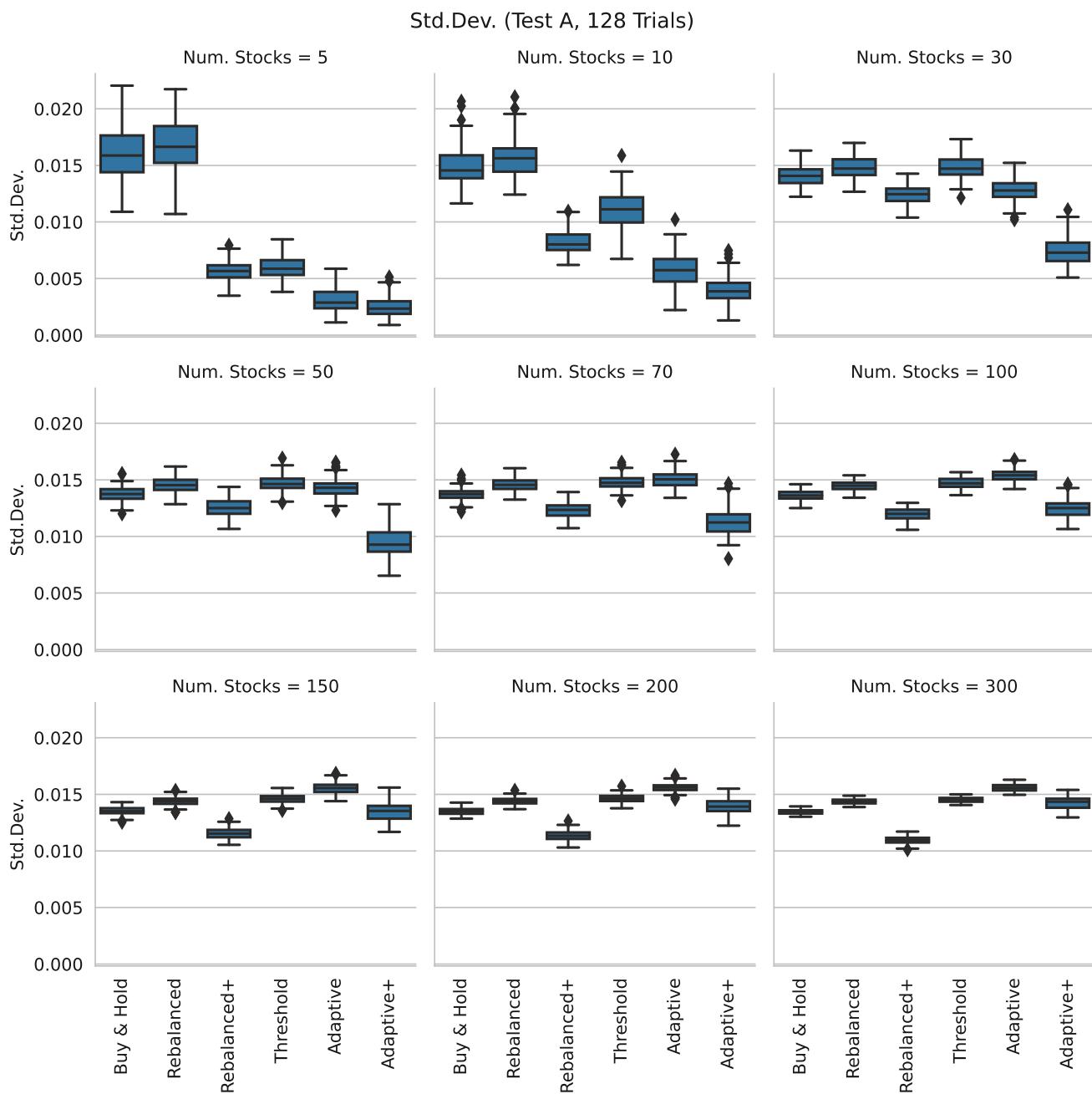


Figure 28: **Test A** – Compare Std.Dev. of the daily return of different portfolio methods and sizes.

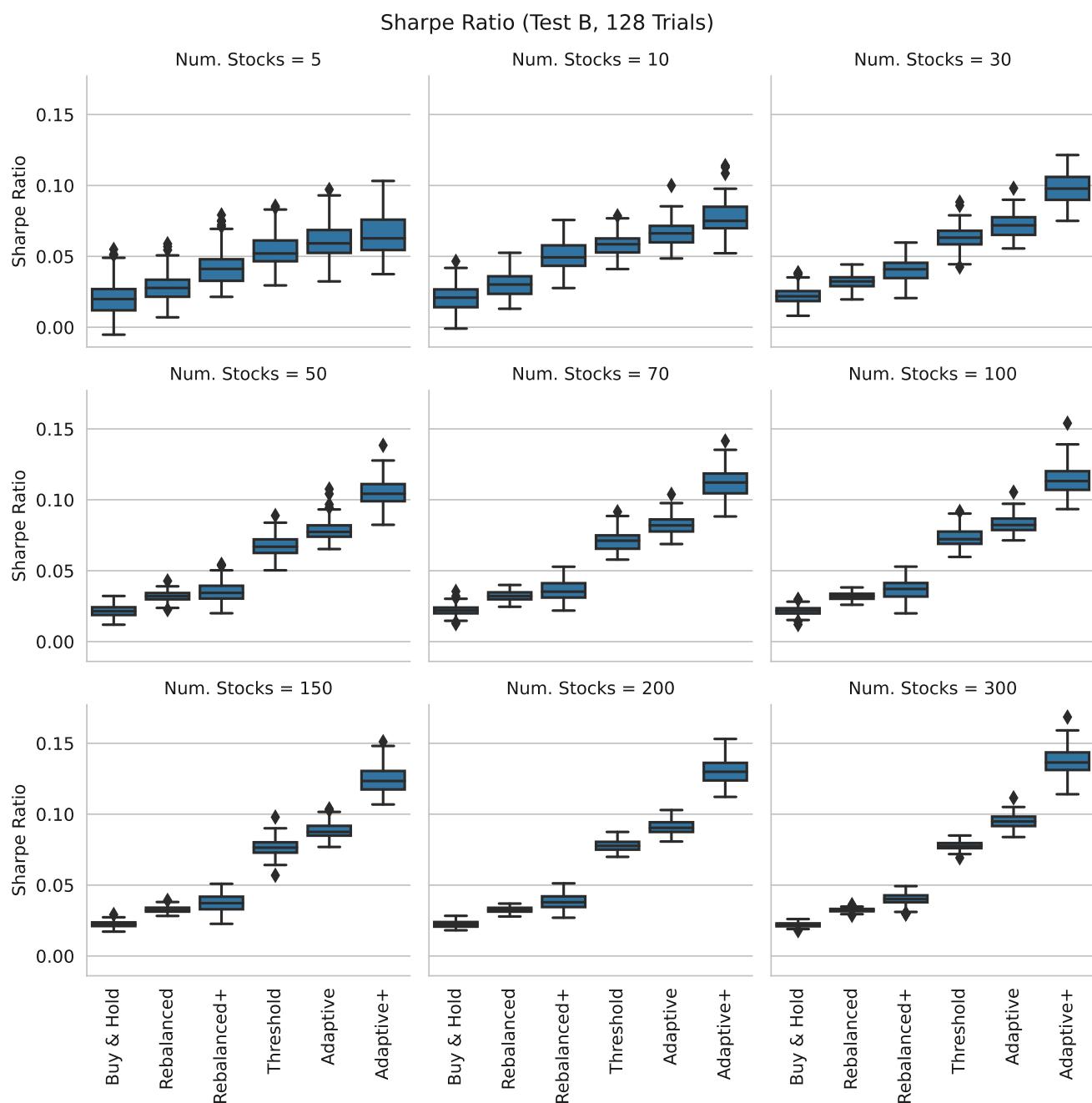


Figure 29: **Test A** – Compare **Sharpe Ratios** for daily returns of different portfolio methods and sizes.

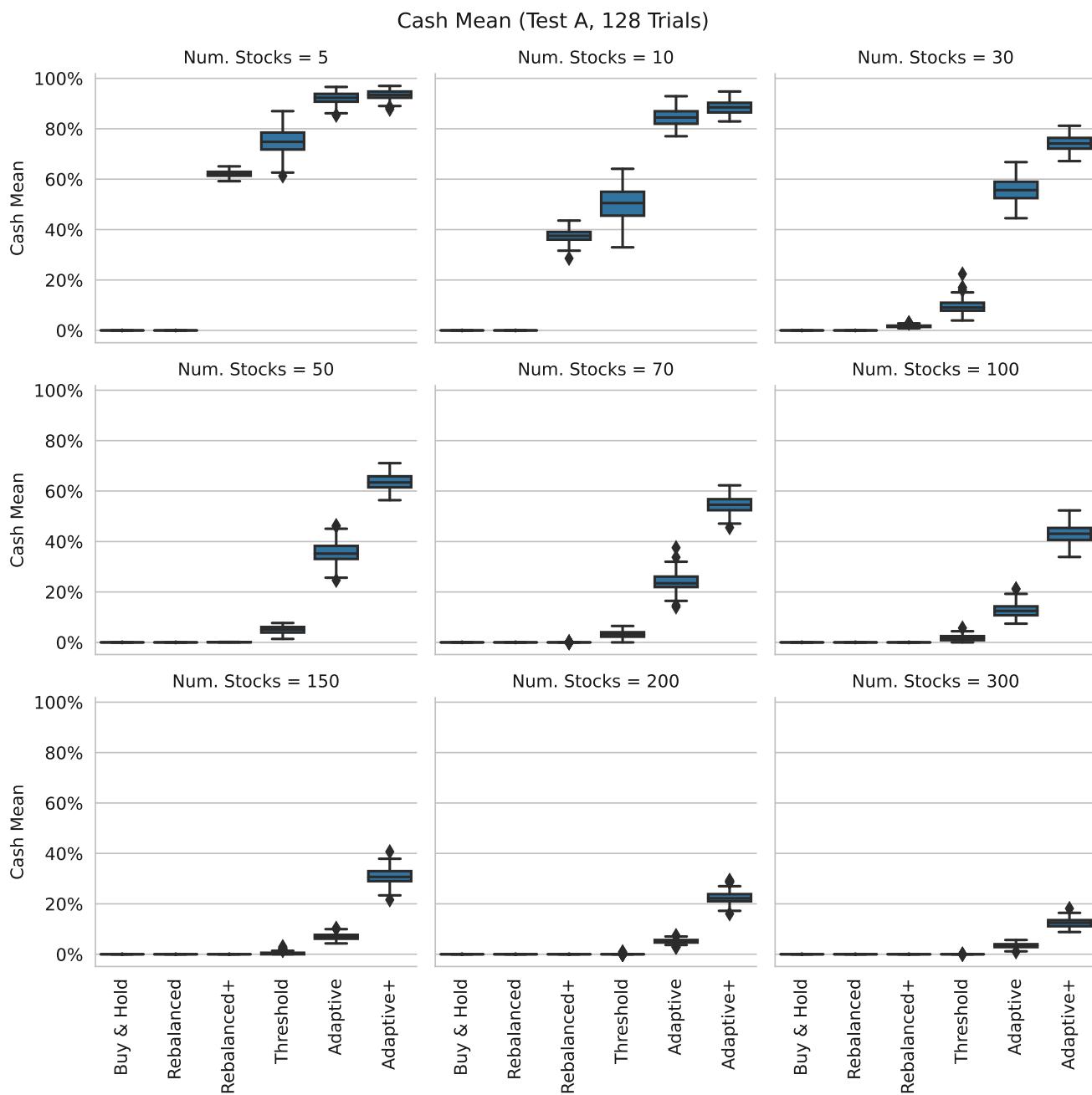


Figure 30: **Test A – Compare Cash Mean Daily Position** of different portfolio methods and sizes.

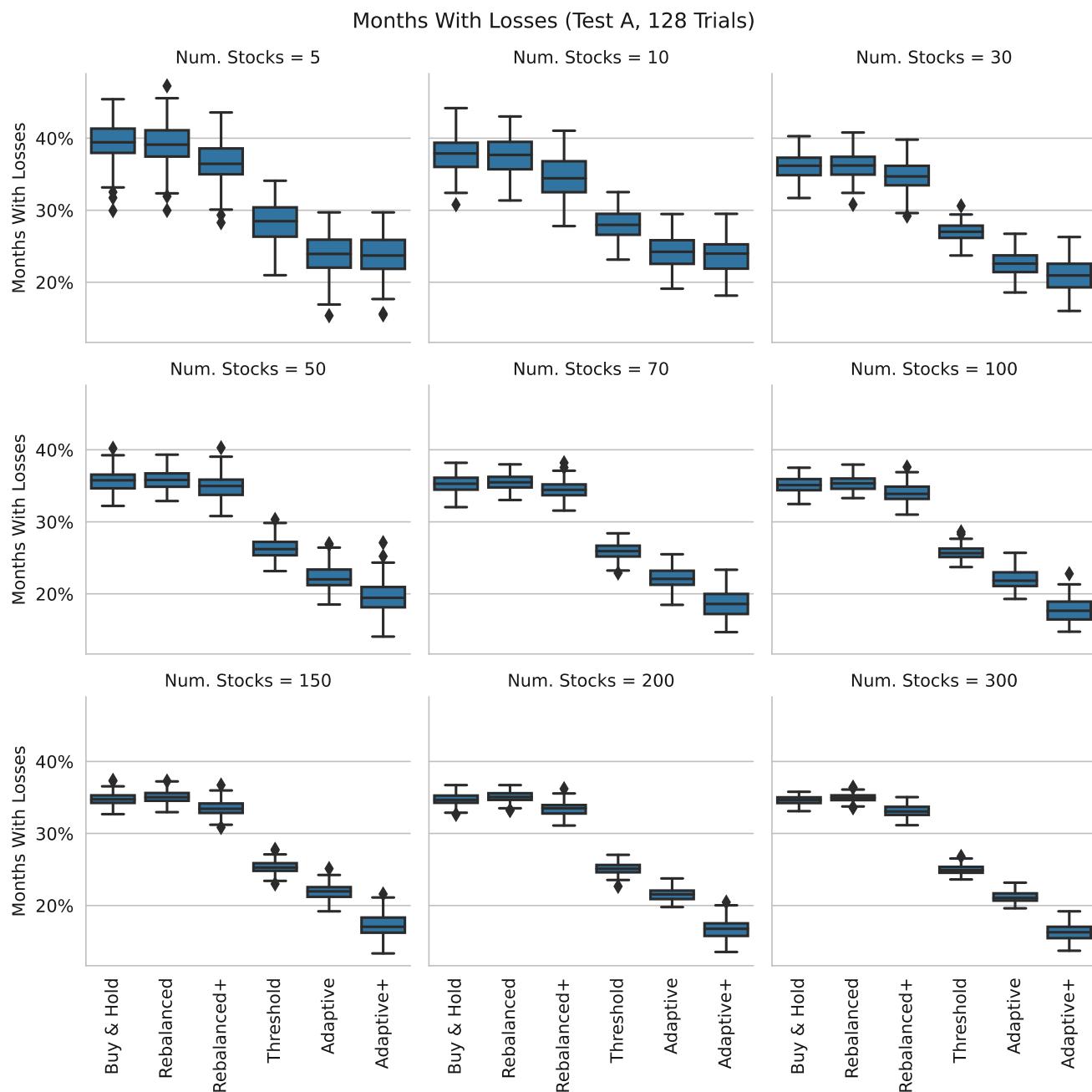


Figure 31: **Test A – Compare Months With Losses** of different portfolio methods and sizes.

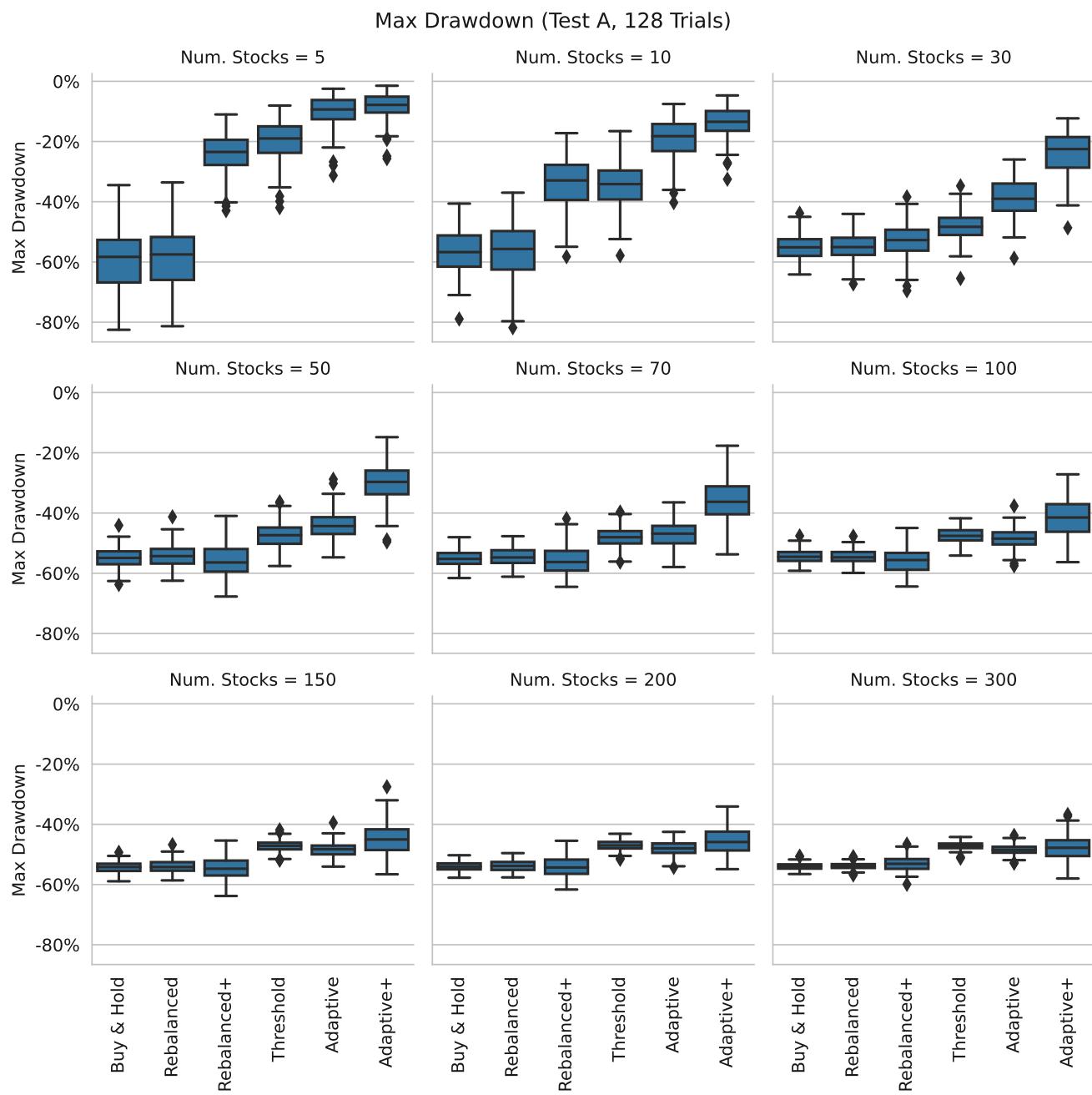


Figure 32: Test A – Compare Max Drawdown of different portfolio methods and sizes.

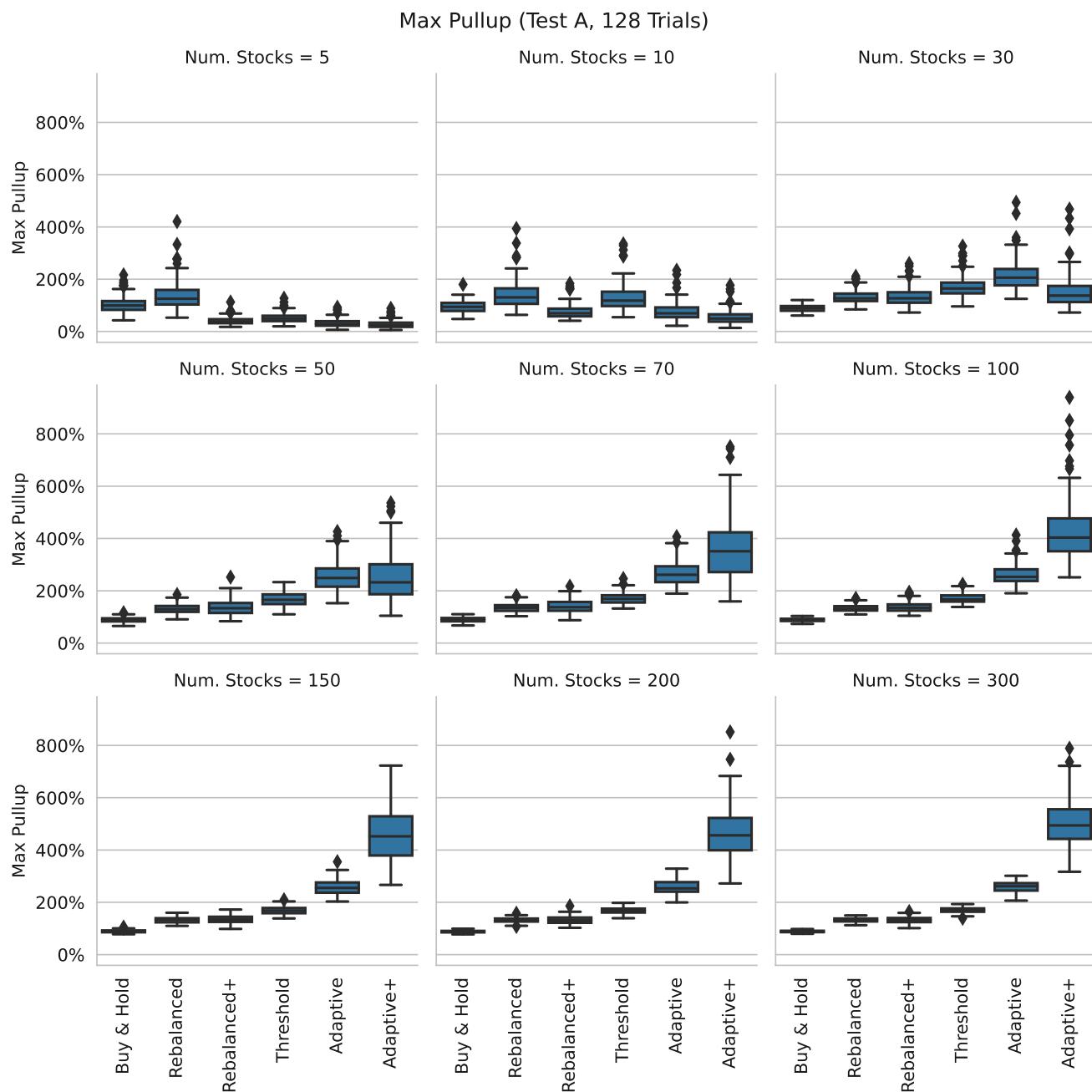


Figure 33: Test A – Compare Max Pullup of different portfolio methods and sizes.

11 Test B – Data Until 2010 (Omniscient)

In the previous section we saw that the Adaptive+ portfolios performed extremely well on all performance metrics when the portfolio sizes were sufficiently large and had a few hundred stocks. Figure 33 showed that the Adaptive+ portfolios were vastly superior on the Max Pullup metric compared to all the other portfolio methods, as the Adaptive+ portfolios had a median gain of 500%, while e.g. the Buy&Hold portfolios only had a median gain around 100%. This raises the question if the overall superiority of the Adaptive+ portfolios were mostly due to extreme out-performance during the recovery-phase of the great stock-market crash in the year 2009, as shown by the large Max Pullup.

To examine this we split the data-set into two periods: Between the years 2007 and 2010 which will be studied in this section, and the period between 2010 and 2018 which will be studied in Section 12. We are still using the same omniscient data as before with actual future stock-returns and correlations.

Figures 34, 35 and 36 compare the portfolio values between the years 2007 and 2010 for the different portfolio methods, again using 128 trials with randomly selected stocks for portfolios of varying sizes. Figures 37, 38, 39, 40, 41, 42 and 43 compare the performance metrics for all those random portfolios. We will not analyse all of these statistics in great detail here, but only draw the main conclusions.

Overall the tendencies in all these plots are very similar to the ones in Section 10 for the full data-period between 2007 and 2018. Once again we see in Figure 37 that for smaller portfolio sizes the Threshold, Adaptive and Adaptive+ portfolios have lower mean daily return than the trivial Buy&Hold and Rebalanced portfolios. But for larger portfolio sizes the Threshold, Adaptive and Adaptive+ portfolios have much higher mean daily return. The reason for this is found in Figure 40 which shows the daily mean cash position of the portfolios, where the trivial Buy&Hold and Rebalanced portfolios are always fully invested by definition, but the Threshold, Adaptive and Adaptive+ portfolios hold very large parts of their portfolios in cash, so that is why they have lower mean returns. Even for portfolios with 300 stocks, the Adaptive+ portfolios still hold an average of about 25% of the portfolios in cash.

Figure 43 shows the Max Pullup for the different portfolio methods between the years 2007 and 2010, which looks almost identical to Figure 33 for the entire period between 2007 and 2018. That is because the biggest stock-market crash occurred around the year 2009, and between 2010 and 2018 there were only minor stock-market “corrections” as they are called. So by only considering the period between 2007 and 2010 we have successfully isolated the period that caused the extreme out-performance of the Adaptive+ portfolios in the Max Pullup statistic. In the next Section 12 we can therefore study what happens to all the performance metrics when the big stock-market crash of 2009 is removed.

An important thing should be noted here. Because the Threshold, Adaptive and Adaptive+ portfolios use the actual future stock-returns for 2-3 years in these omniscient tests, they avoid investing in the period leading up to the stock-market crash in 2009. But because the portfolio methods are completely “blind” to what happens within the next 2 years, they will start to invest as soon as the stock-prices have dropped sufficiently, so they can get a return of at least 10% per year. If the portfolio methods had waited just a few months more to start investing, they would have performed much better. But short-term stock forecasting is extremely difficult in general, and it is still remarkable how well the portfolio methods perform if they can merely predict 2-3 years into the future.

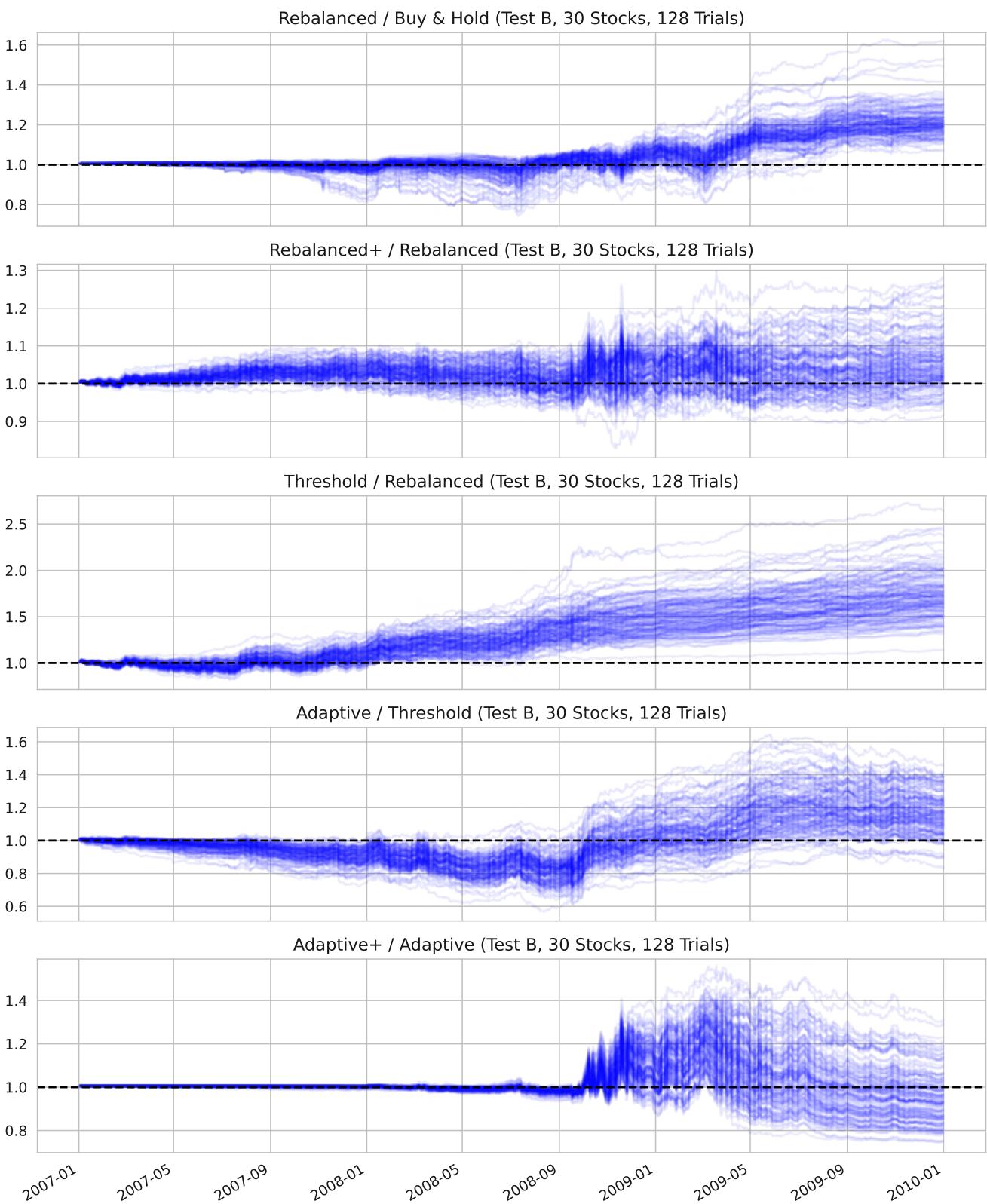


Figure 34: **Test B** – Compare values of 128 random portfolios with **30 stocks** each.

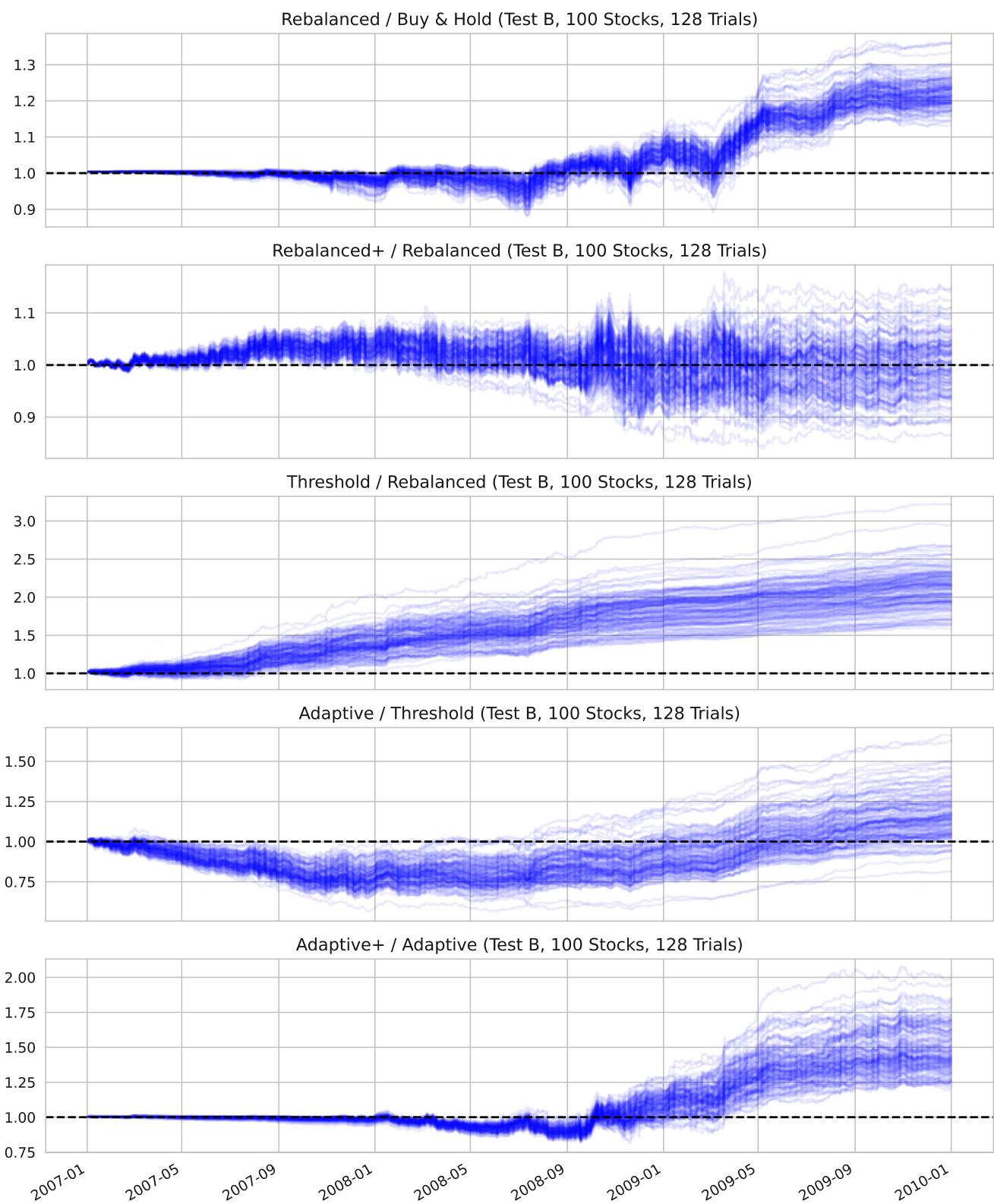


Figure 35: **Test B** – Compare values of 128 random portfolios with **100 stocks** each.

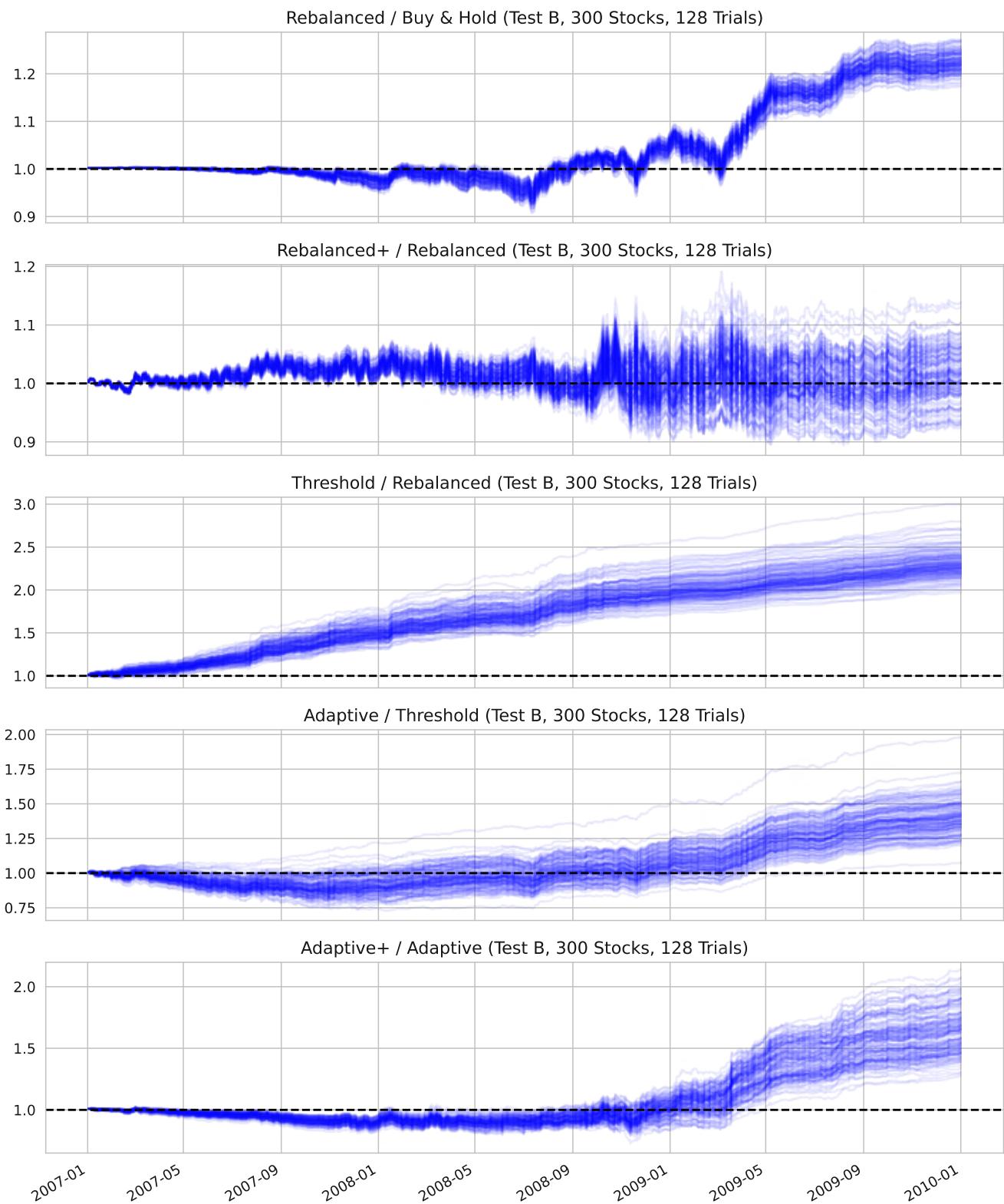


Figure 36: **Test B** – Compare values of 128 random portfolios with **300 stocks** each.

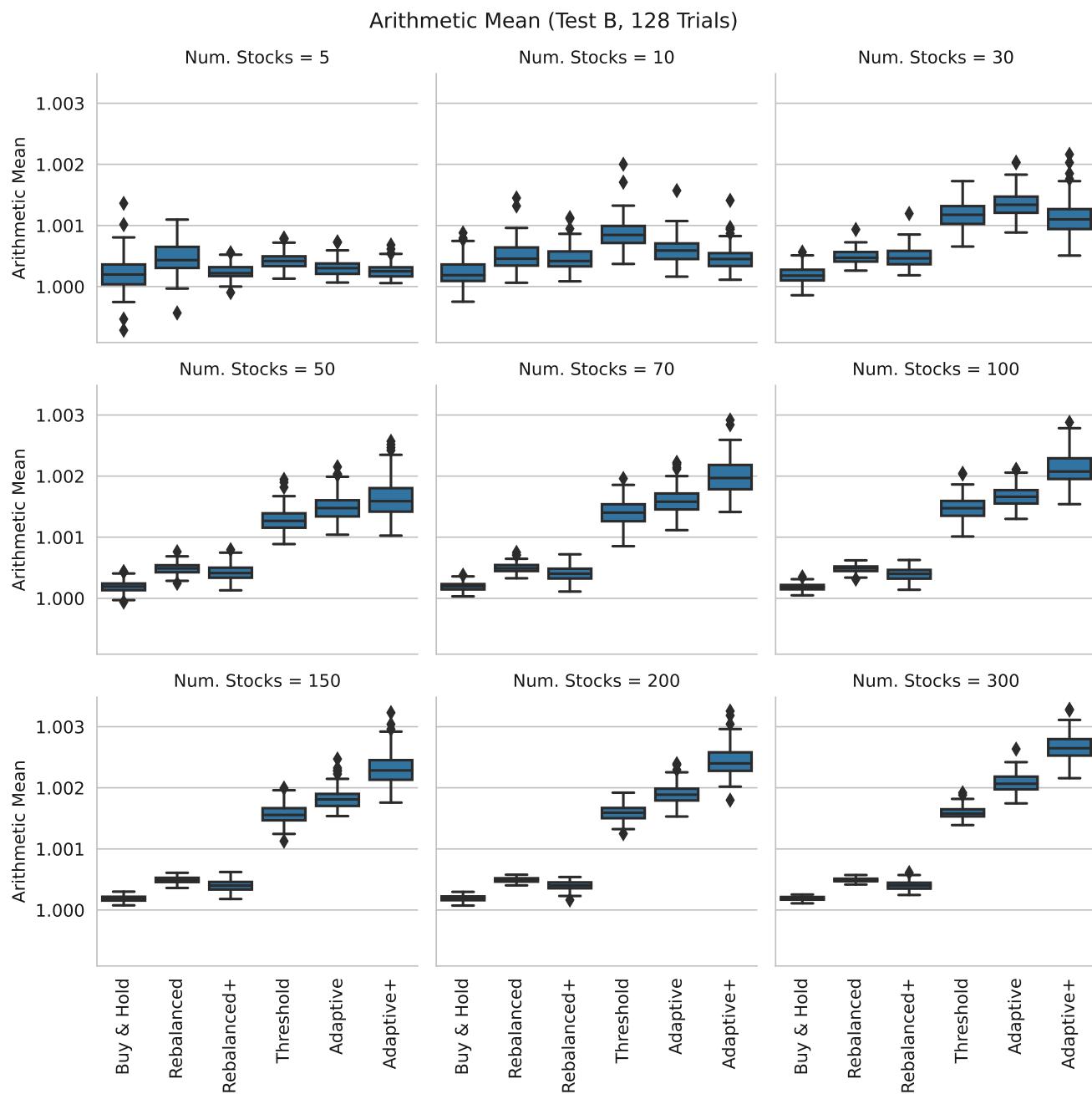


Figure 37: **Test B** – Compare **Arithmetic Mean** daily return of different portfolio methods and sizes.

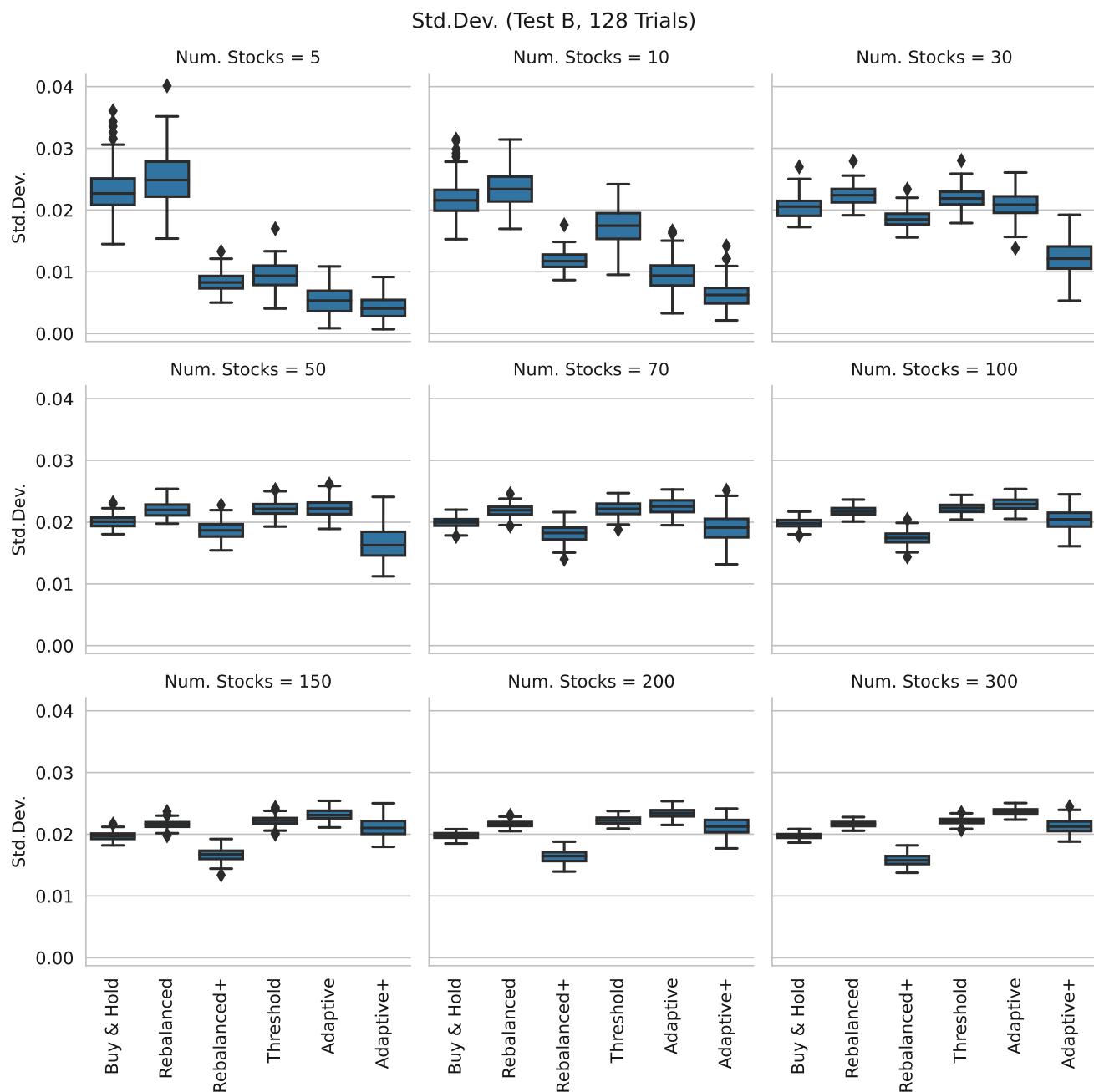


Figure 38: **Test B** – Compare Std.Dev. of the daily return of different portfolio methods and sizes.

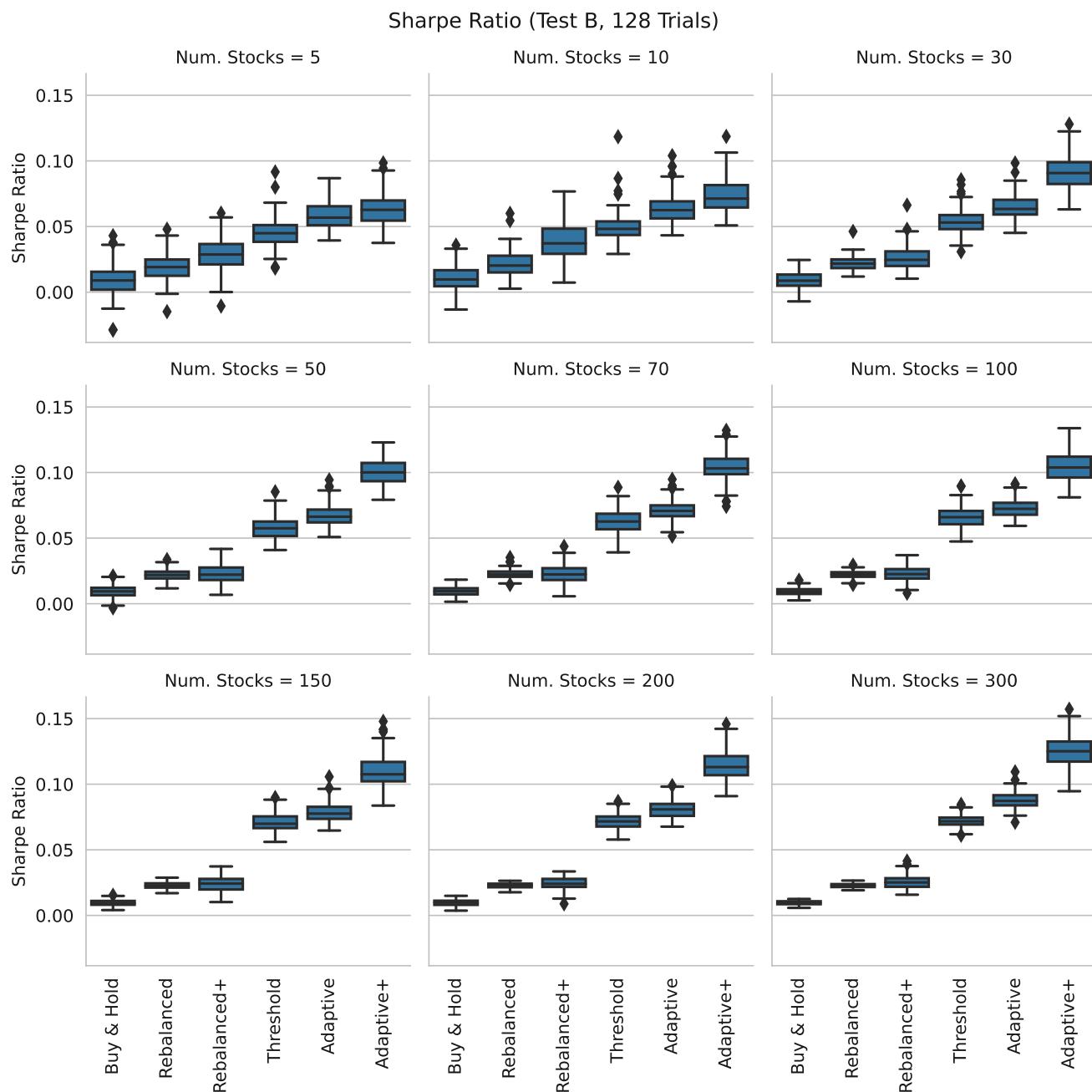


Figure 39: **Test B** – Compare **Sharpe Ratios** for daily returns of different portfolio methods and sizes.

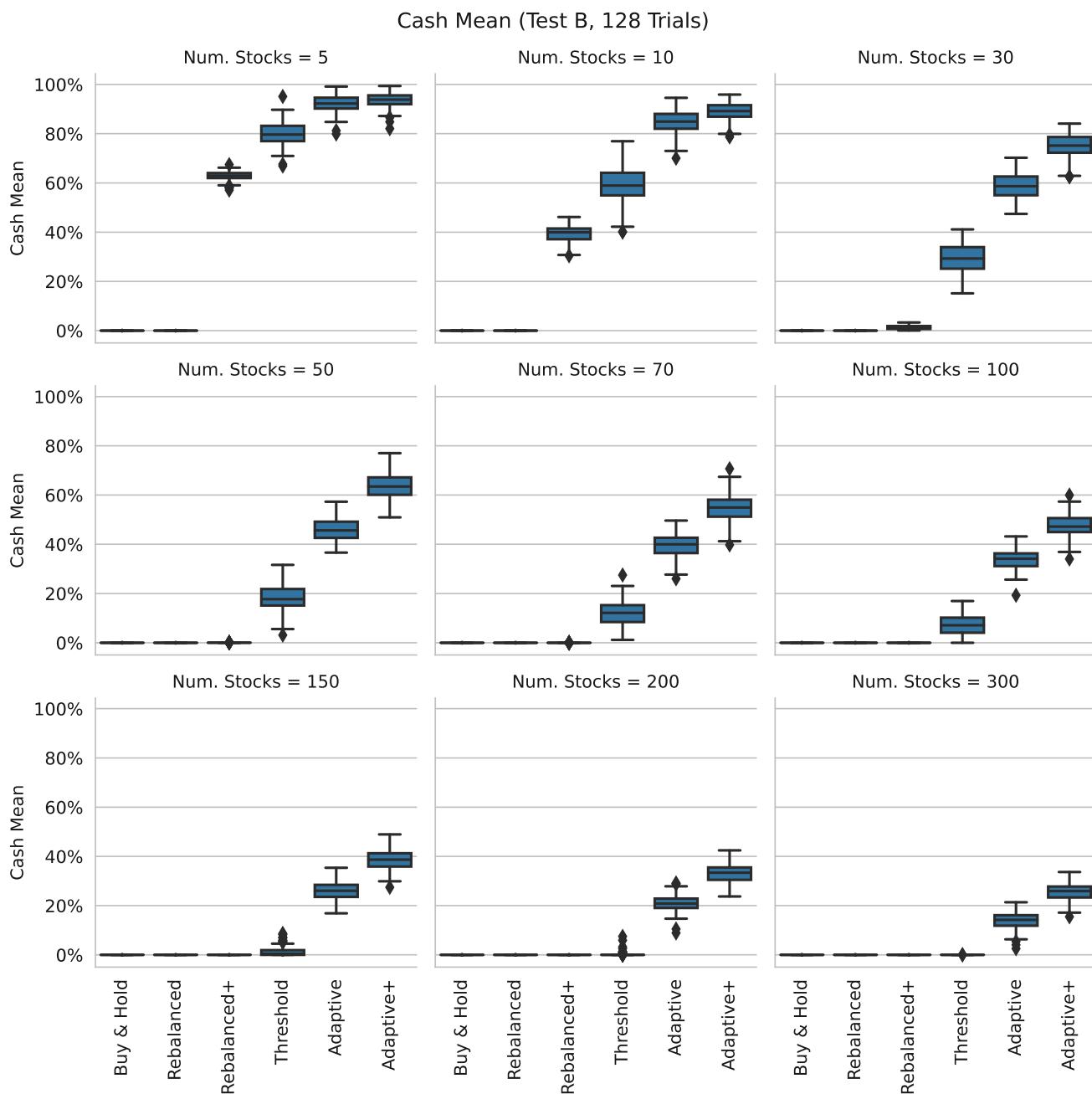


Figure 40: Test B – Compare Cash Mean Daily Position of different portfolio methods and sizes.

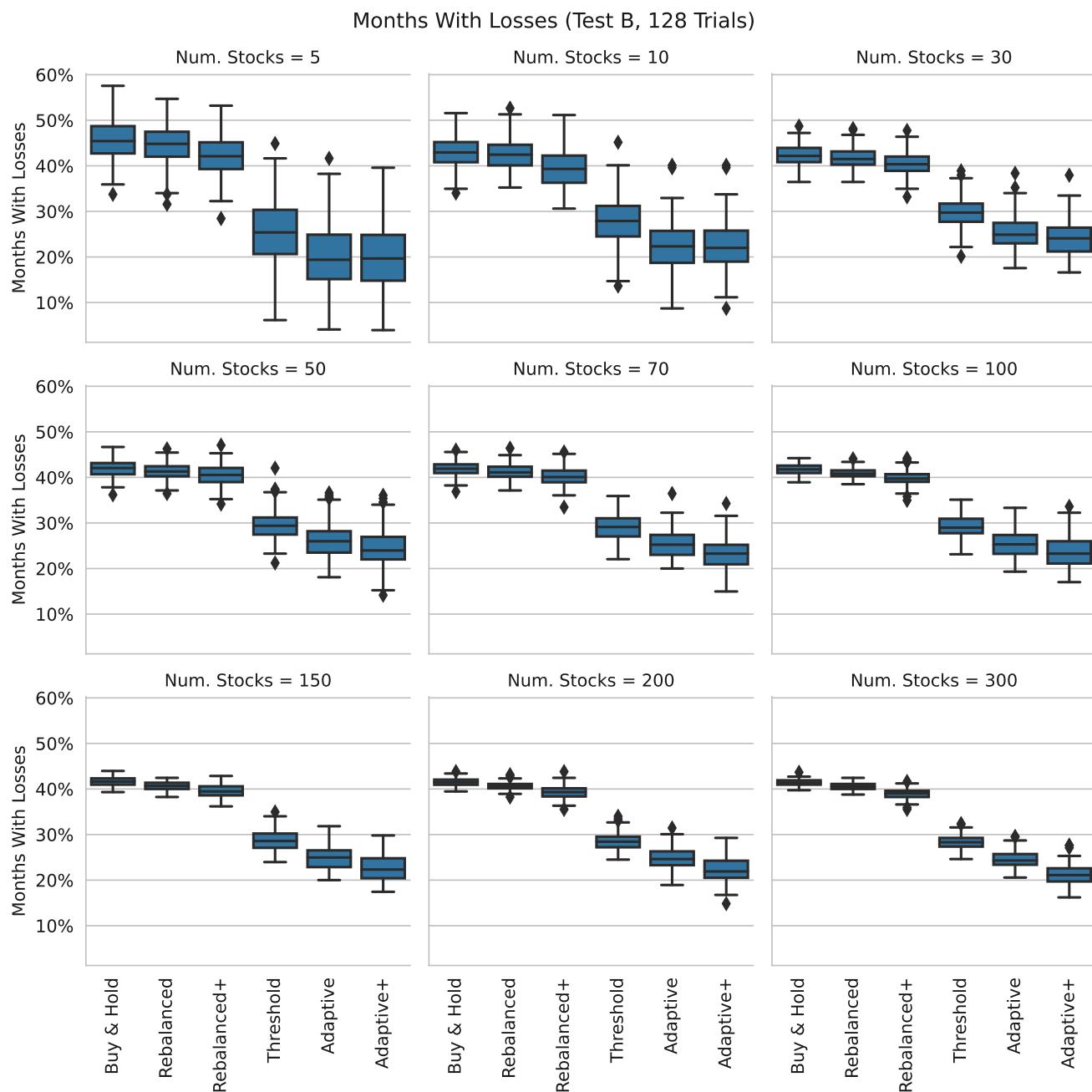


Figure 41: **Test B – Compare Months With Losses** of different portfolio methods and sizes.

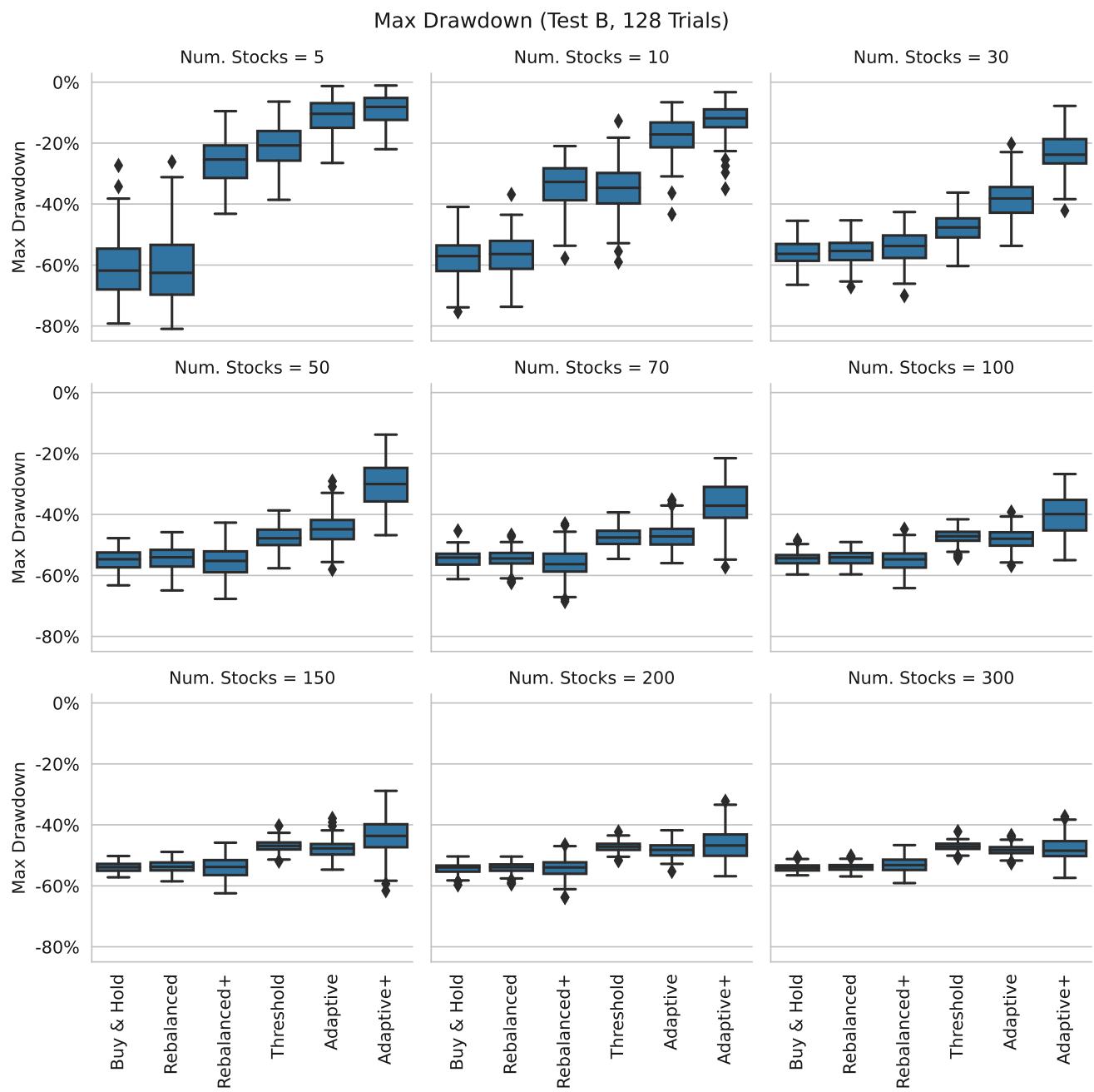


Figure 42: **Test B** – Compare **Max Drawdown** of different portfolio methods and sizes.

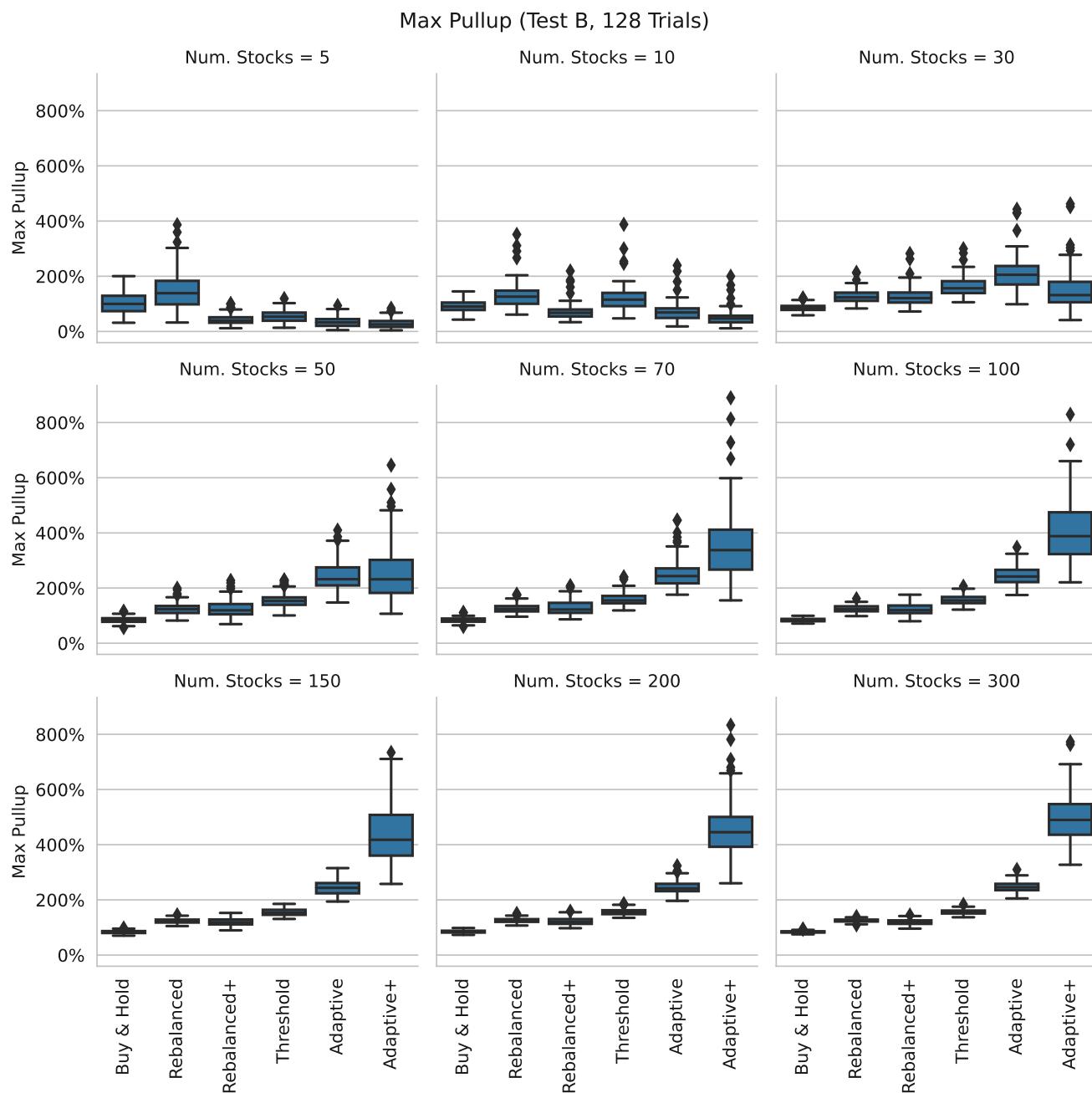


Figure 43: **Test B** – Compare **Max Pullup** of different portfolio methods and sizes.

12 Test C – Data From 2010 (Omniscient)

In the previous Section 11 we studied the years between 2007 and 2010. In this section we will study the years between 2010 and 2018. We are still using the same omniscient data with actual future stock-returns for 2-3 year periods and the actual future stock-correlations for 10-day periods.

Figures 44, 45 and 46 compare the portfolio values between the years 2010 and 2018 for the different portfolio methods, again using 128 trials with randomly selected stocks for portfolios of varying sizes. Figures 47, 48, 49, 50, 51, 52 and 53 compare the performance metrics for all those random portfolios.

Let us first compare the arithmetic mean daily returns. These are shown for Test C in Figure 44, for Test B in Figure 37, and for Test A in Figure 26. These show very similar tendencies, namely that the Threshold, Adaptive and Adaptive+ portfolios have increasingly better performance for larger portfolio sizes when compared to the trivial Buy&Hold and Rebalanced portfolios. The only difference is that the mean returns are slightly higher overall in Test B compared to Test A and Test C. The reason is probably that the Threshold, Adaptive and Adaptive+ portfolios performed extremely well in the recovery-phase of the great stock-market crash in 2009, which is the period covered in Test B and which has been excluded from Test C. This gives slightly higher mean daily returns in Test B. But Figure 44 for the mean daily returns between 2010 and 2018 show that these portfolio methods still performed great in more “normal” times without big stock-market crashes.

Figure 44 shows that the mean daily return for portfolios with 300 stocks was around 1.0022 for the Adaptive+ portfolios, corresponding to about 73% per year, which is calculated as $1.0022^{250} - 1 \approx 73\%$ when assuming there are 250 trading days per year. The mean daily return was around 1.0016 for the Adaptive portfolios, corresponding to about 49% per year. The mean daily return was around 1.0012 for the Threshold portfolios, corresponding to about 35% per year. For the Buy&Hold and Rebalanced portfolios the mean daily return was only about 1.0006, corresponding to about 16% per year.

Figure 52 shows that for portfolios with more than 50 stocks in the years between 2010 and 2018, the worst Max Drawdowns were around -25% for all the different portfolio methods, with the exception being the Adaptive+ portfolios that performed significantly better, probably because they often had a large part of the portfolio in cash, as shown in Figure 50.

Figure 53 shows that the median Max Pullup was nearly 150% for the Adaptive+ portfolios with 300 stocks, while it was a bit below 100% for the Adaptive portfolios, and maybe only around 70% for the Threshold portfolios. The Buy&Hold and Rebalanced portfolios only had Max Pullups around 50%.

So even though the Max Drawdowns were only around -25% as shown in Figure 52 because there were only minor stock-market crashes between the years 2010 and 2018, the Threshold, Adaptive and Adaptive+ portfolios could still take much better advantage of these smaller stock-market recoveries than the trivial Buy&Hold and Rebalanced portfolios.

Overall the Threshold, Adaptive and Adaptive+ portfolios still worked extremely well during the less turbulent market conditions between 2010 and 2018 – provided we were using the actual future stock-returns and correlations to allocate the portfolios. In the following sections we will test the robustness of these portfolio methods when the estimates for the future stock-returns and correlations are noisy.

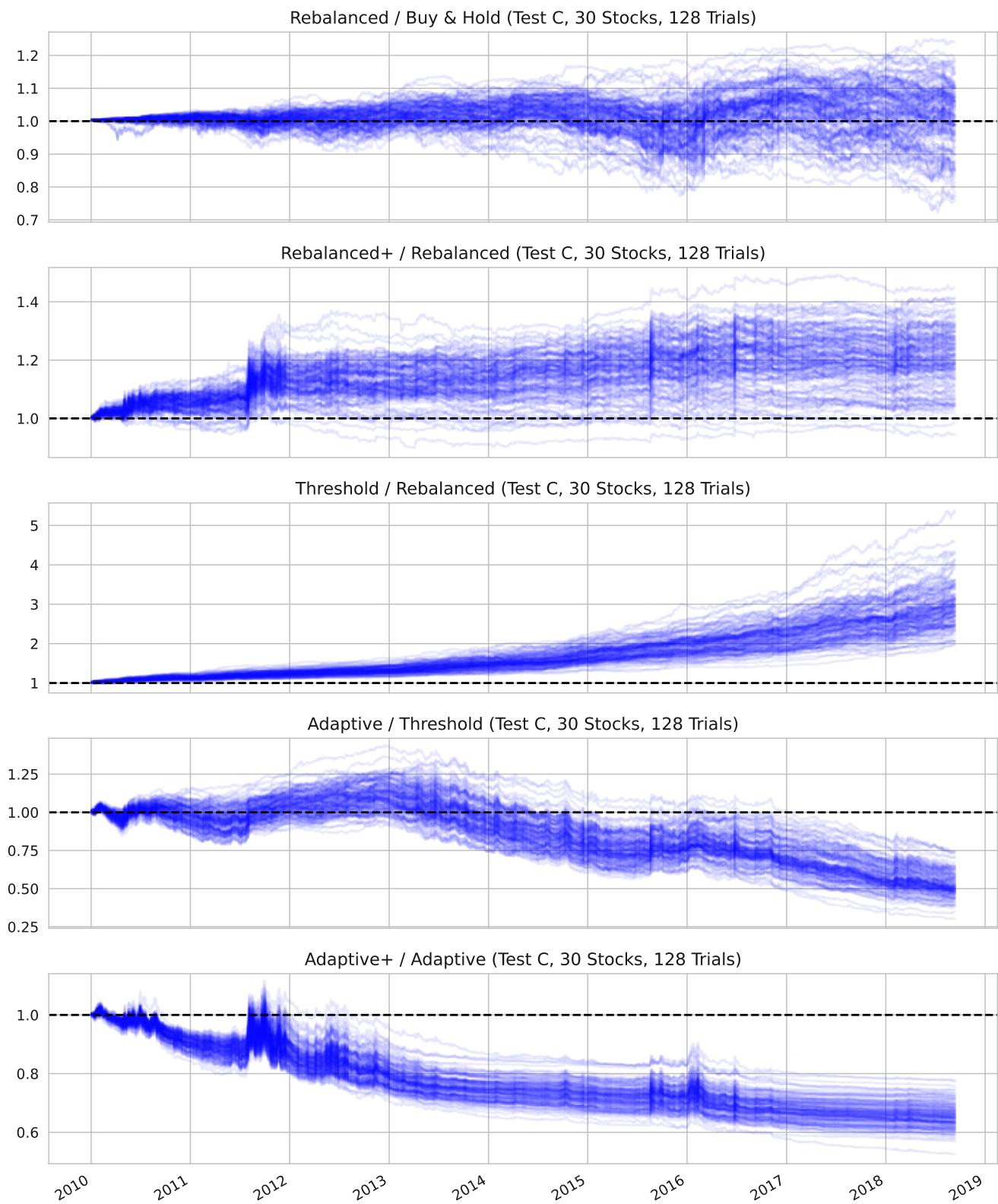


Figure 44: **Test C** – Compare values of 128 random portfolios with **30 stocks** each.

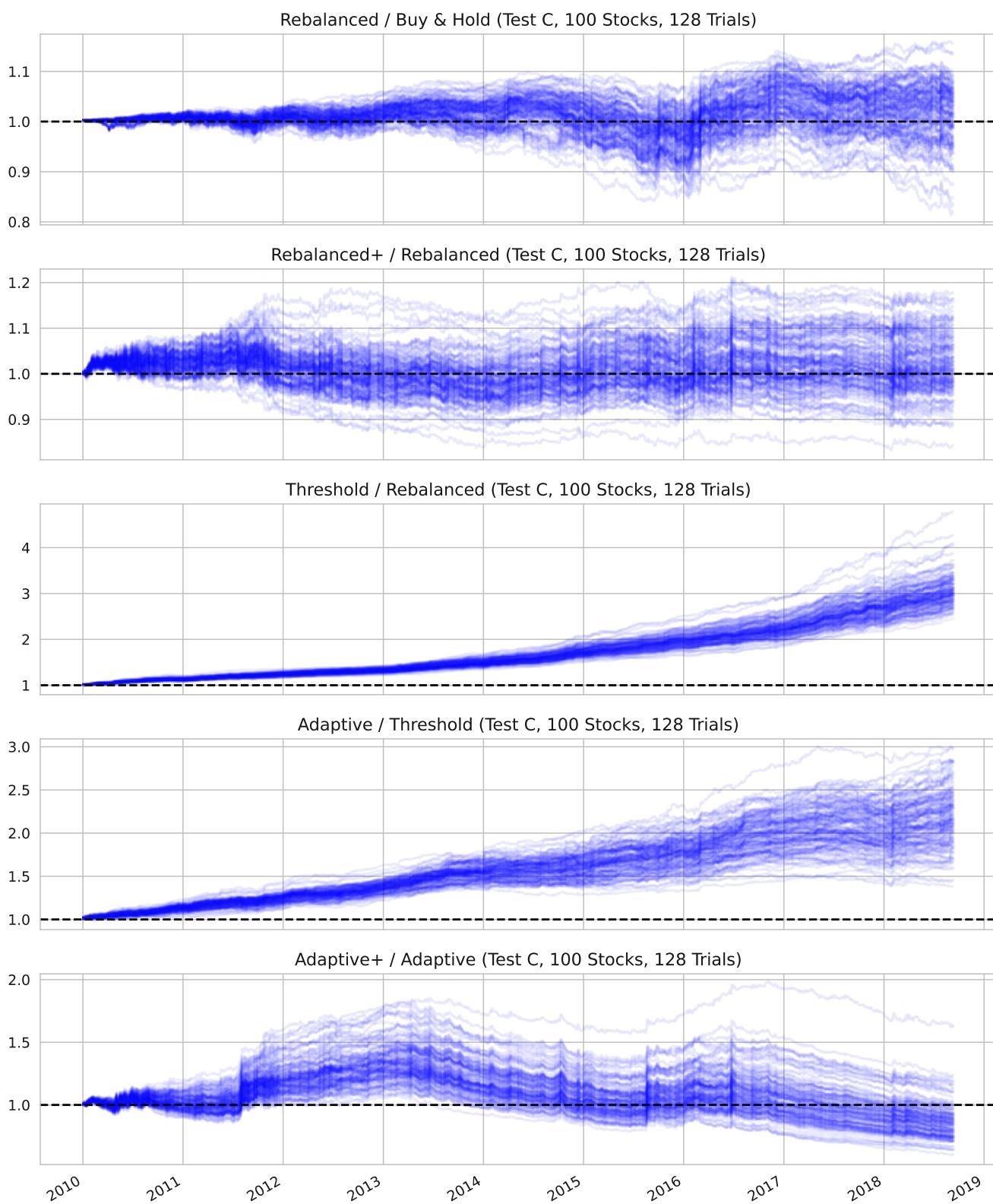


Figure 45: **Test C** – Compare values of 128 random portfolios with **100 stocks** each.

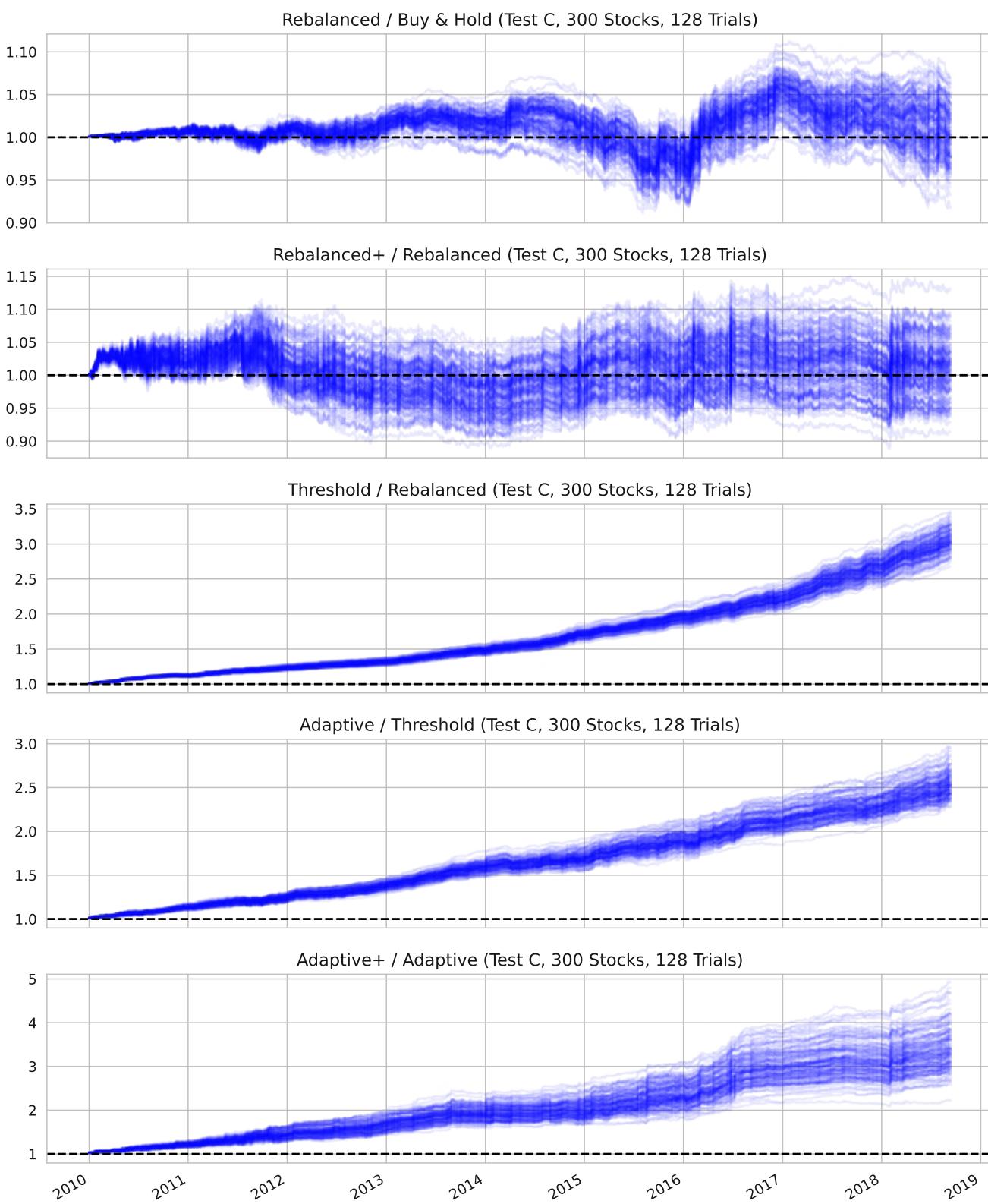


Figure 46: **Test C** – Compare values of 128 random portfolios with **300 stocks** each.

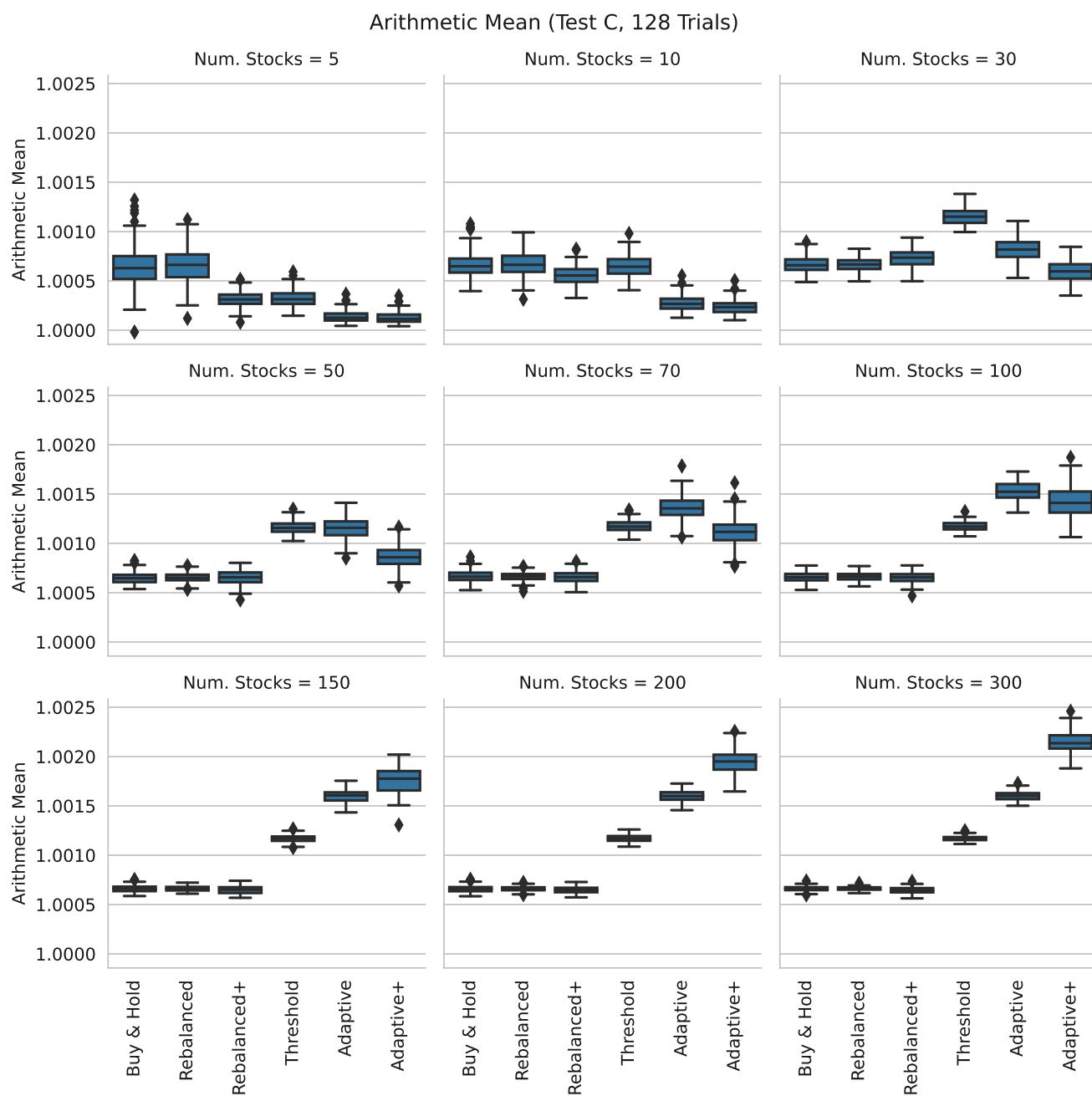


Figure 47: **Test C – Compare Arithmetic Mean** daily return of different portfolio methods and sizes.

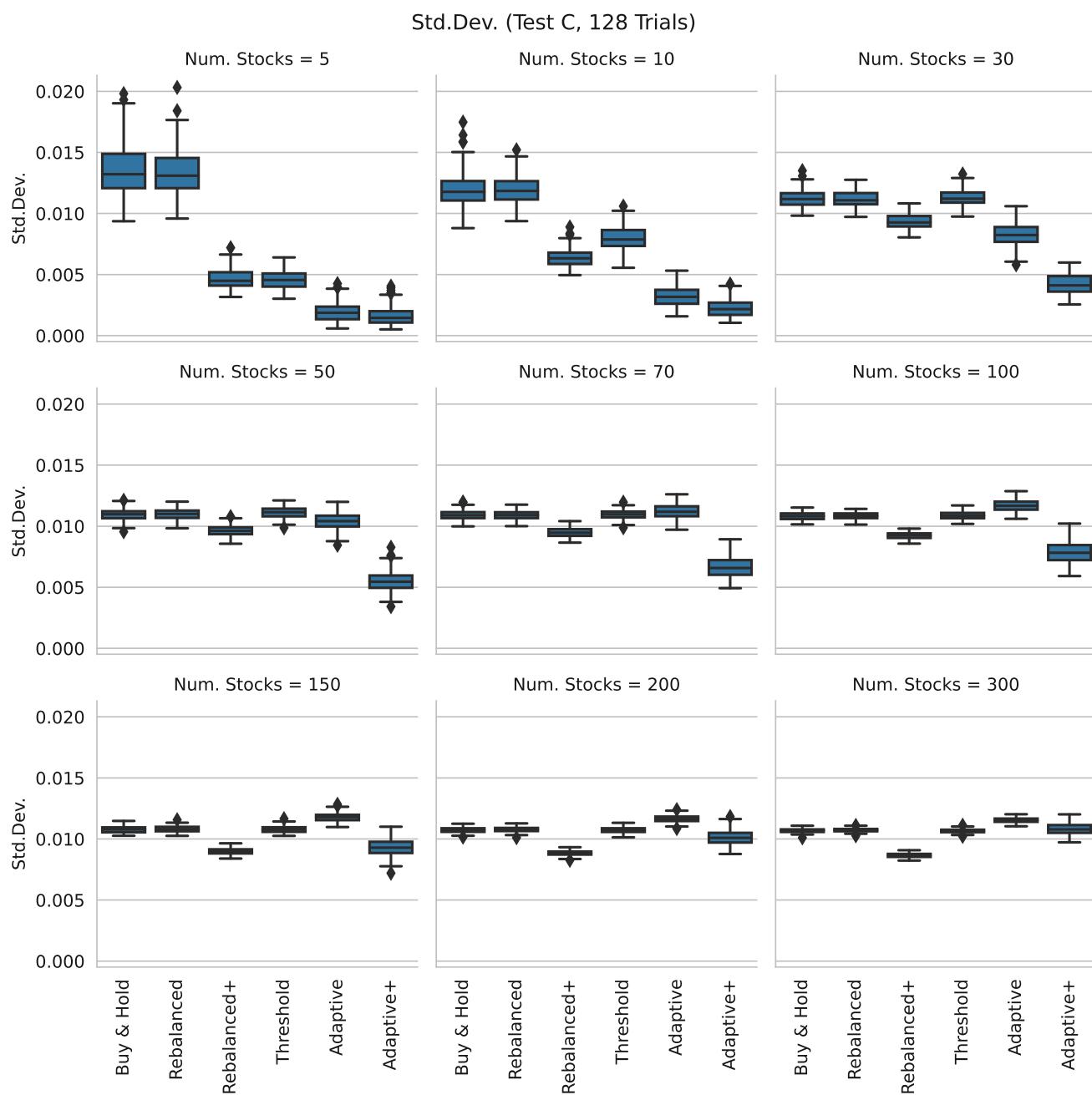


Figure 48: **Test C** – Compare **Std.Dev.** of the daily return of different portfolio methods and sizes.

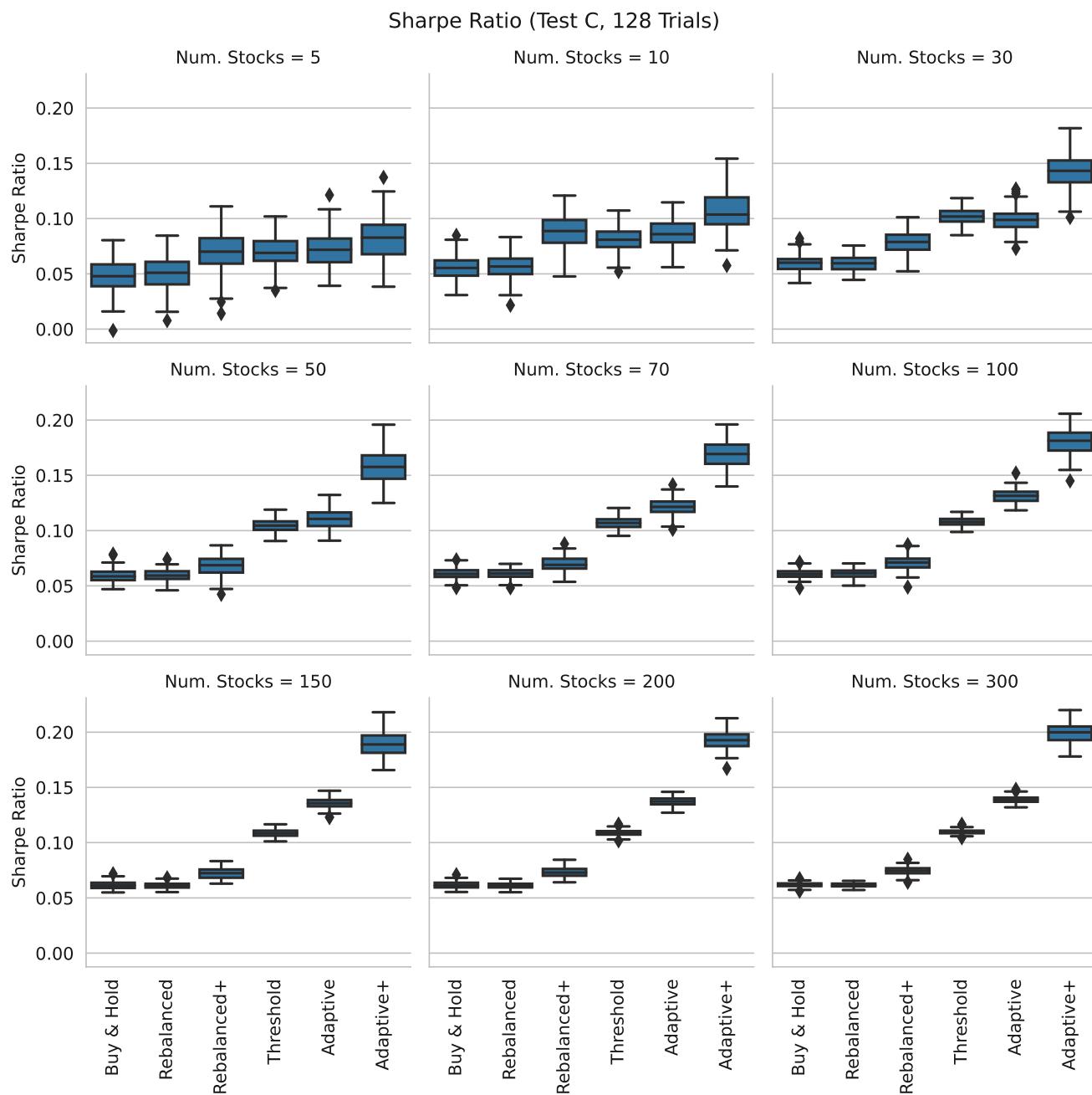


Figure 49: **Test C** – Compare **Sharpe Ratios** for daily returns of different portfolio methods and sizes.

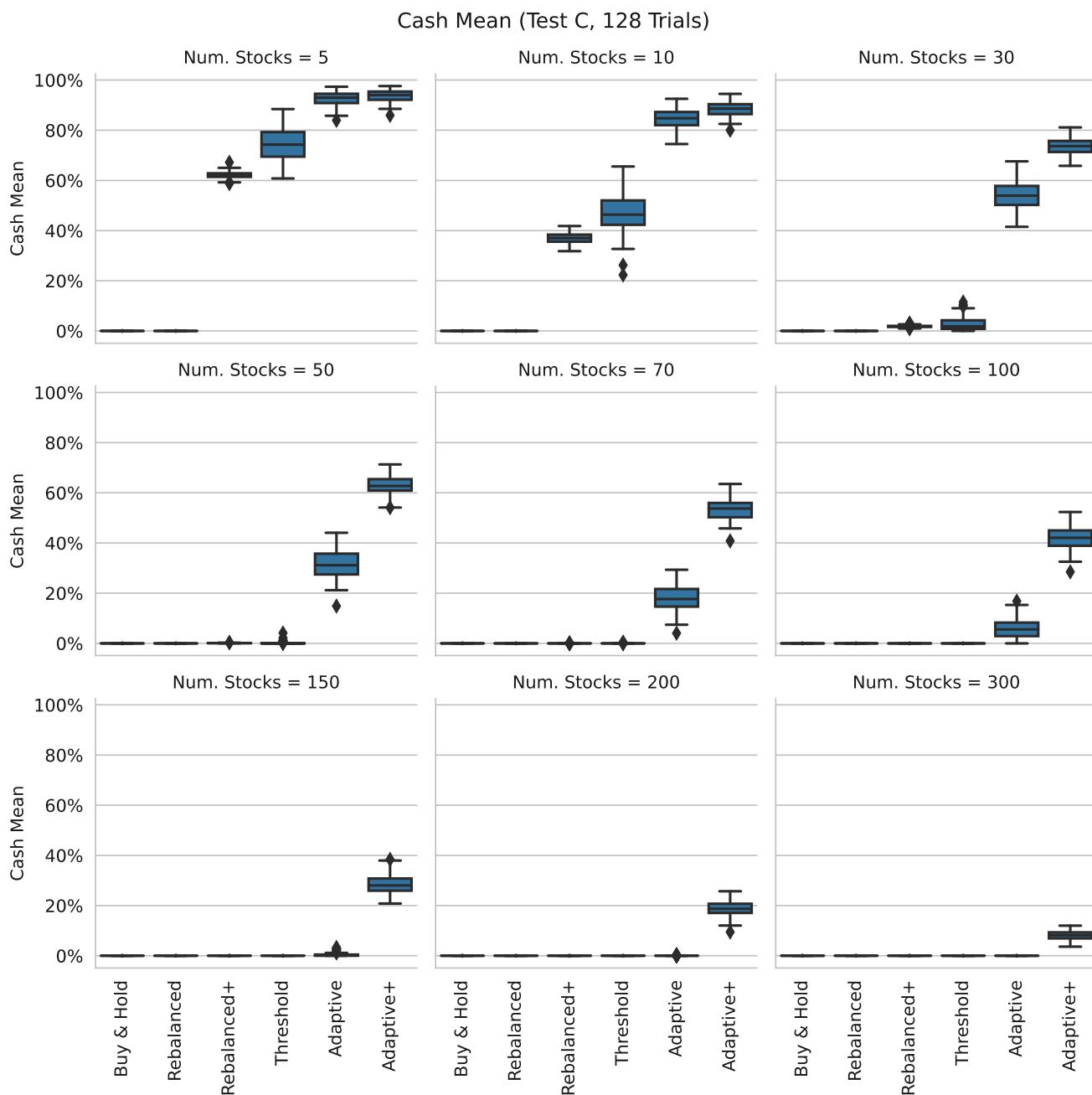


Figure 50: **Test C – Compare Cash Mean Daily Position** of different portfolio methods and sizes.

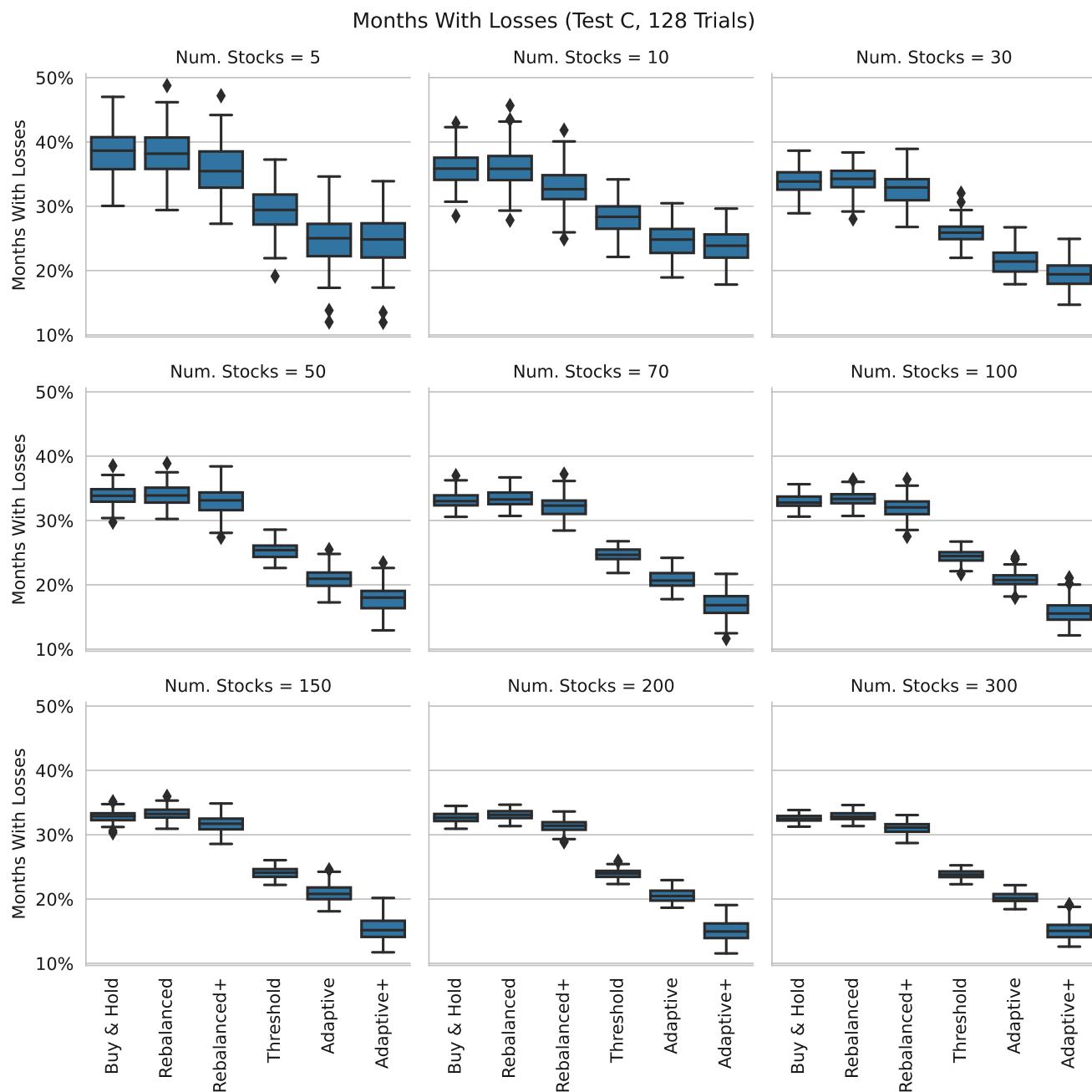


Figure 51: **Test C – Compare Months With Losses** of different portfolio methods and sizes.

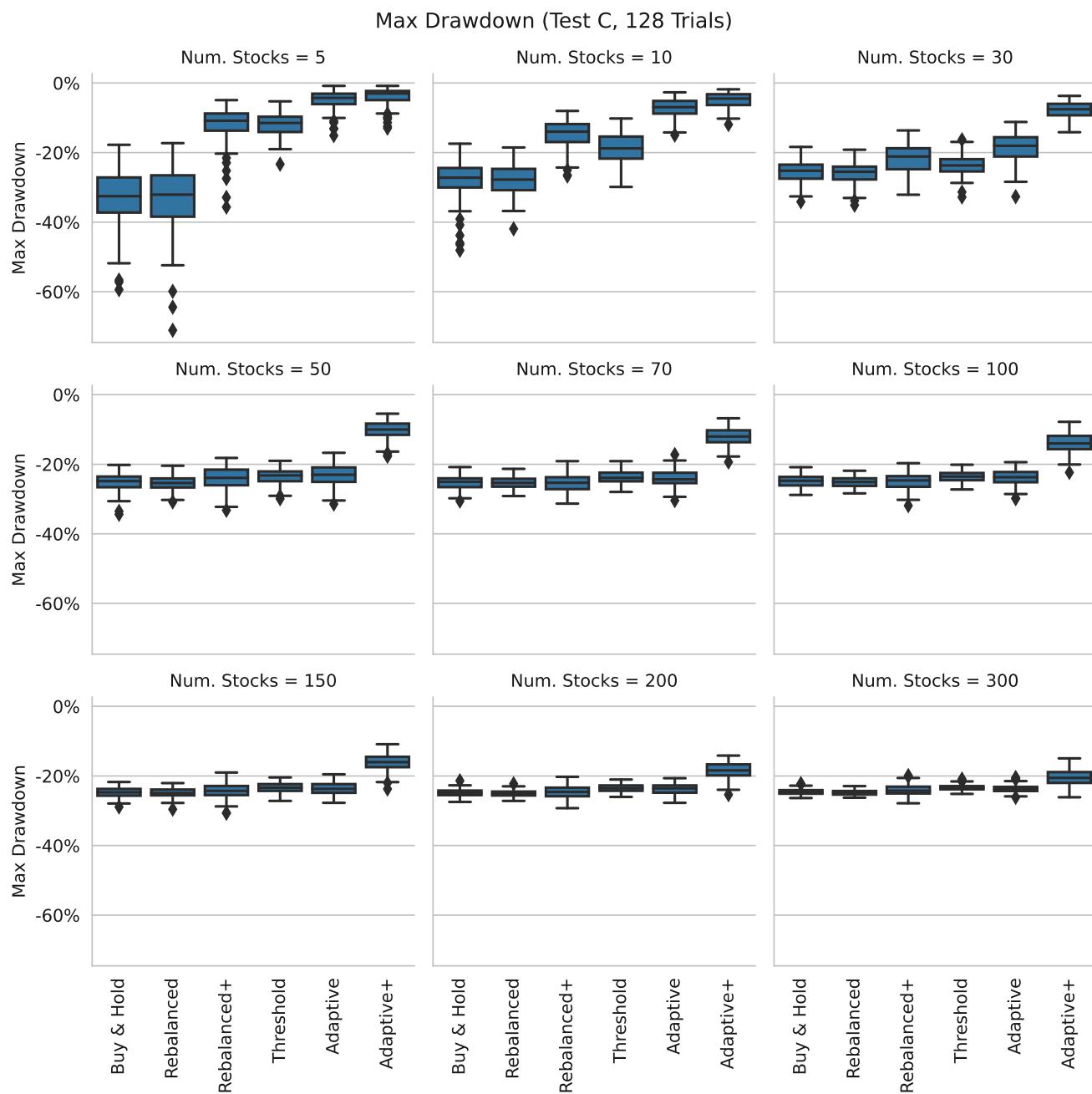


Figure 52: **Test C – Compare Max Drawdown** of different portfolio methods and sizes.

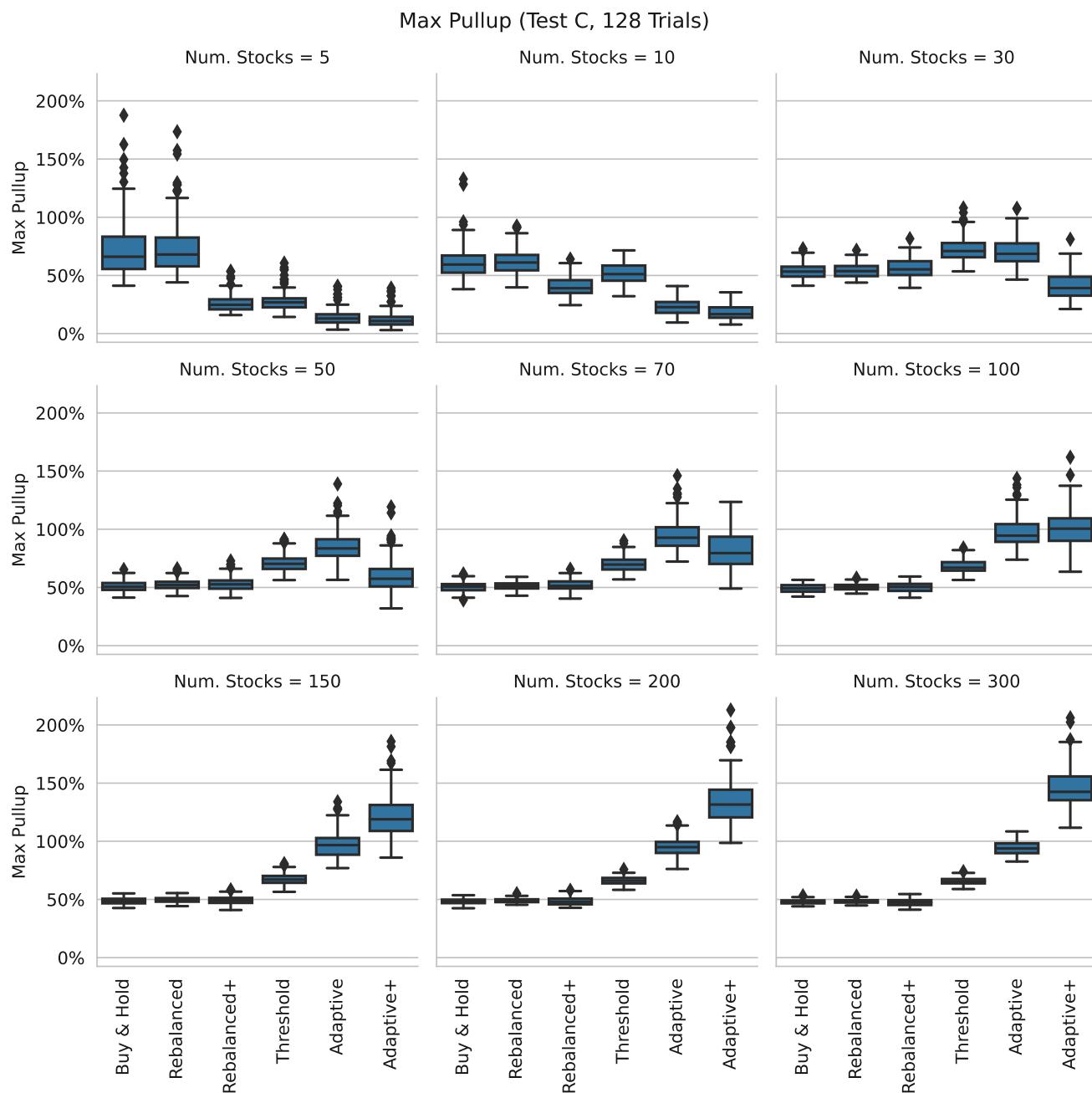


Figure 53: Test C – Compare Max Pullup of different portfolio methods and sizes.

13 Test D – Noisy Returns (Robustness)

The previous tests were so-called omniscient because they used the actual future stock-returns and correlations to allocate the portfolios. That is of course pure cheating, but it allowed us to test if the portfolio methods work if they are supplied with the correct input-data. If we use some forecasting model to predict the future stock-returns and correlations, then we are simultaneously testing both the forecasting model and the portfolio methods, so we cannot tell if bad performance is caused by a bad forecasting model or a bad portfolio method. Using the actual future stock-returns and correlations in the previous sections, we found that the portfolio methods work extremely well with correct input-data.

We will now start to test the robustness of the portfolio methods, to see how they perform when there is heavy noise in the estimated future stock-returns and correlations. In this section we will again be using the actual future stock-returns for 2-3 year periods, as well as the actual future stock-correlations for 10-day periods, but we will now add heavy noise to the future stock-returns. In the following sections we will also add noise to the future stock-correlations.

13.1 Pure Noise

The portfolio methods were first tested when the future stock-returns were pure noise, so they could not be used to predict the future stock-returns at all. In this case the Threshold and Adaptive portfolios had completely random performance, as they would sometimes perform worse and sometimes better than the trivial Rebalanced portfolios, but on average they had about the same performance, which is what we would expect when the portfolios are allocated using completely random estimates for the returns.

13.2 Noise Generation

To make a more realistic test, we use the actual future stock-returns for 2-3 year periods and add some Gaussian or normal-distributed noise to them. The noise has zero mean and 0.5 standard deviation, so that about 68% of the noise is within ± 0.5 , and 95% of the noise is within ± 1 , and 99.7% of the noise is within ± 1.5 . This is very heavy noise, because an actual future stock-return of e.g. -0.2 or -20% could easily be changed to e.g. +0.8 or +80% if the noise is +1, which would mean that an actual -20% loss would be estimated to be a +80% gain when the portfolio is being allocated, so the portfolio would invest in a losing stock because it was estimated to have a very large gain in the future. The question is how well the portfolio methods perform with such heavy noise in the estimated future stock-returns.

The noise-samples are added on a per-stock basis, so the same noise-sample is added to all time-steps for one stock. We do not draw new noise-samples for every time-step. This would be very easy to do in the computer code, but it is not realistic that your forecasting model would radically change the estimated future stock-returns in every time-step. So we are instead simulating a persistent estimation error for each stock through time, by adding the same noise-sample to all time-steps for one stock.

There is no bias in the noise. This would also be very easy to add in the computer code, but it really just corresponds to changing the threshold level where the portfolio methods will start to invest in the assets. In the real-world, of course, if you are persistently over-estimating future stock-returns by e.g.

20% then you are going to get an investment return that is always 20% lower than your forecasts. But for the purpose of robustness testing, it seems adequate to use heavy noise without a bias.

13.3 Performance Comparison

Figures 54, 55 and 56 show the portfolio values between the years 2007 and 2018, and Figures 57, 58, 59, 60, 61, 62 and 63 show the corresponding performance statistics. These should be compared to the figures from Test A in Section 10, which also covered the same period between 2007 and 2018 but used the actual future stock-returns and correlations without any noise.

The overall tendencies are quite similar for Test D and Test A, namely that the Threshold, Adaptive and Adaptive+ portfolios under-perform the trivial Buy&Hold and Rebalanced portfolios when there are only 5 or 10 stocks in each random portfolio. But already from a portfolio size of just 30 stocks, the Threshold, Adaptive and Adaptive+ portfolios start to perform significantly better in comparison.

Figure 57 shows the daily arithmetic mean return for Test D, where e.g. the Adaptive+ portfolios with 300 random stocks had a median daily return slightly above 1.001, corresponding to an annualized return around $1.001^{250} - 1 \approx 28\%$ when assuming there are 250 trading-days per year. Compare this to Test A whose daily mean return is shown in Figure 26, where the Adaptive+ portfolios with 300 stocks had a median daily return around 1.00225, corresponding to an annualized return around 75%. Similarly the Adaptive portfolios in Test D had a median daily return around 1.0009, corresponding to about 25% per year. The Threshold portfolios in Test D had a median daily return around 1.0008, corresponding to about 22% per year. The Rebalanced portfolios are the same in both Test A and D because they do not use the estimated future stock-returns in their portfolio allocations, so they had a median daily return around 1.0006, corresponding to about 16% per year. So even with the heavy noise in the future stock-returns in Test D, the Threshold, Adaptive and Adaptive+ portfolios still significantly out-performed the Rebalanced portfolios. Their returns were obviously not as high as in Test A, which used perfectly accurate predictions for the future stock-returns. But the portfolio methods still worked very well in the presence of heavy noise in the estimated future stock-returns.

Another interesting thing to note is that Figure 60 shows the average daily cash holdings for Test D, and already for portfolios of just 70 stocks the Threshold, Adaptive and Adaptive+ portfolios had invested almost everything and had no cash holdings. This is very different from Test A whose cash holdings are shown in Figure 30, where even for portfolios with 300 stocks, the Adaptive+ portfolios still held an average of 15% of the portfolio in cash every day. So when the estimated future stock-returns are very noisy, these portfolio methods invest much more of the portfolio instead of holding it in cash, presumably because the noise causes a lot of the future return estimates to be highly positive.

13.4 Summary

These tests showed that the portfolio methods still work very well with heavy and unbiased noise in the estimated stock-returns. This means that if you can create a forecasting model that is just broadly correct at predicting 2-3 year stock-returns, and it is also roughly unbiased so the forecasting model neither over-predicts nor under-predicts all the future returns in the same way, then with sufficient diversification in enough stocks, the new portfolio methods should still work really well.

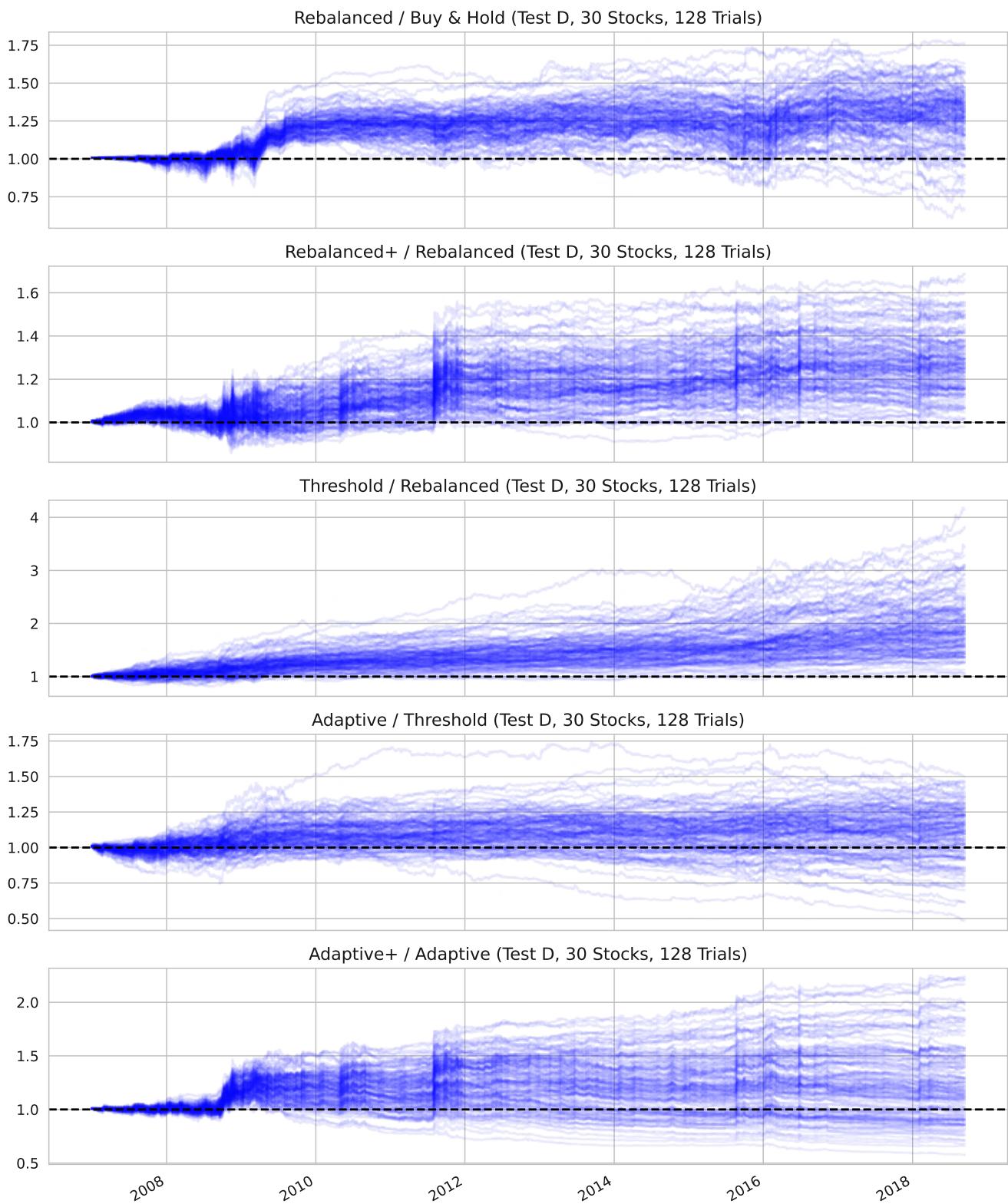


Figure 54: **Test D** – Compare values of 128 random portfolios with **30 stocks** each.

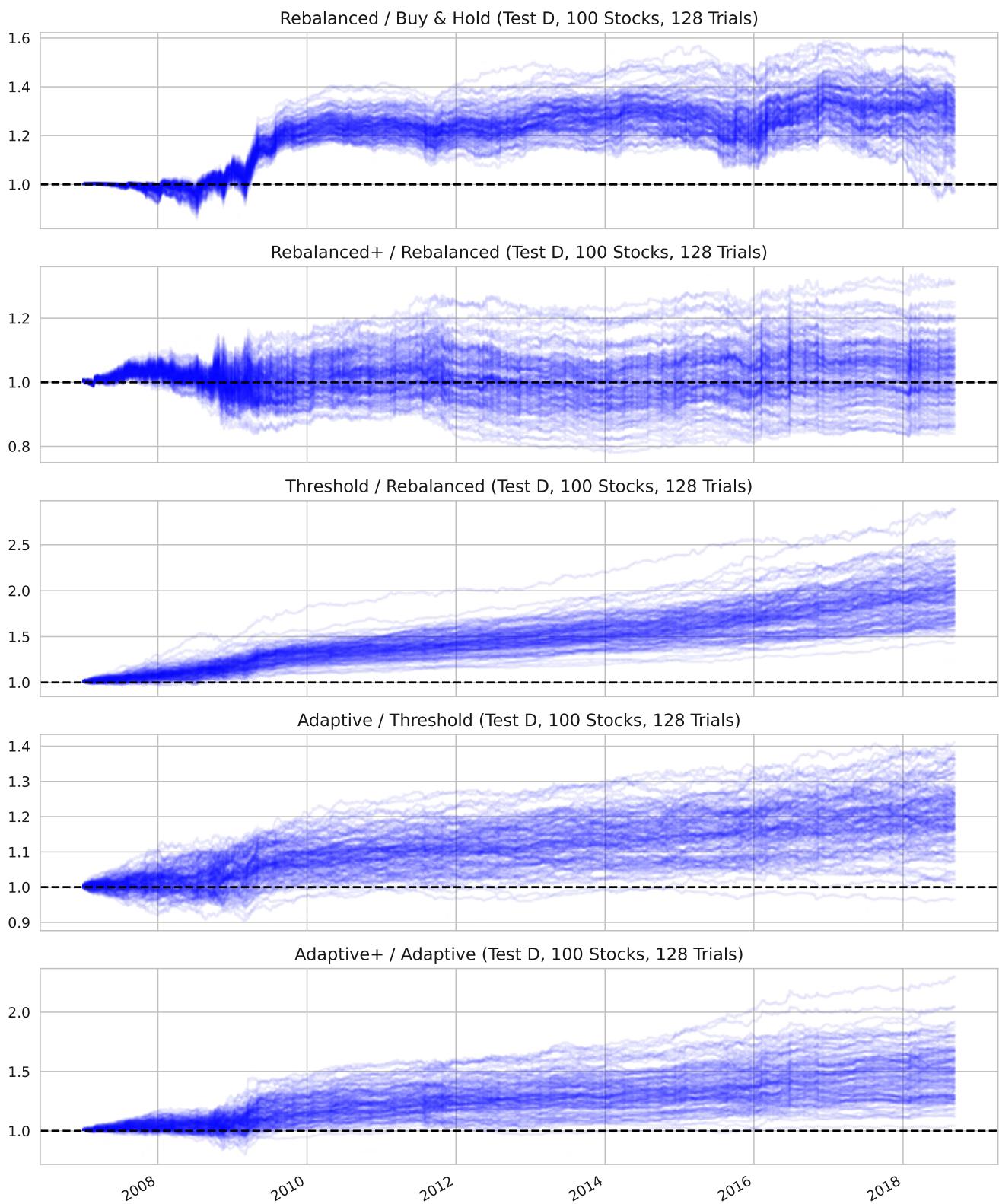


Figure 55: **Test D** – Compare values of 128 random portfolios with **100 stocks** each.

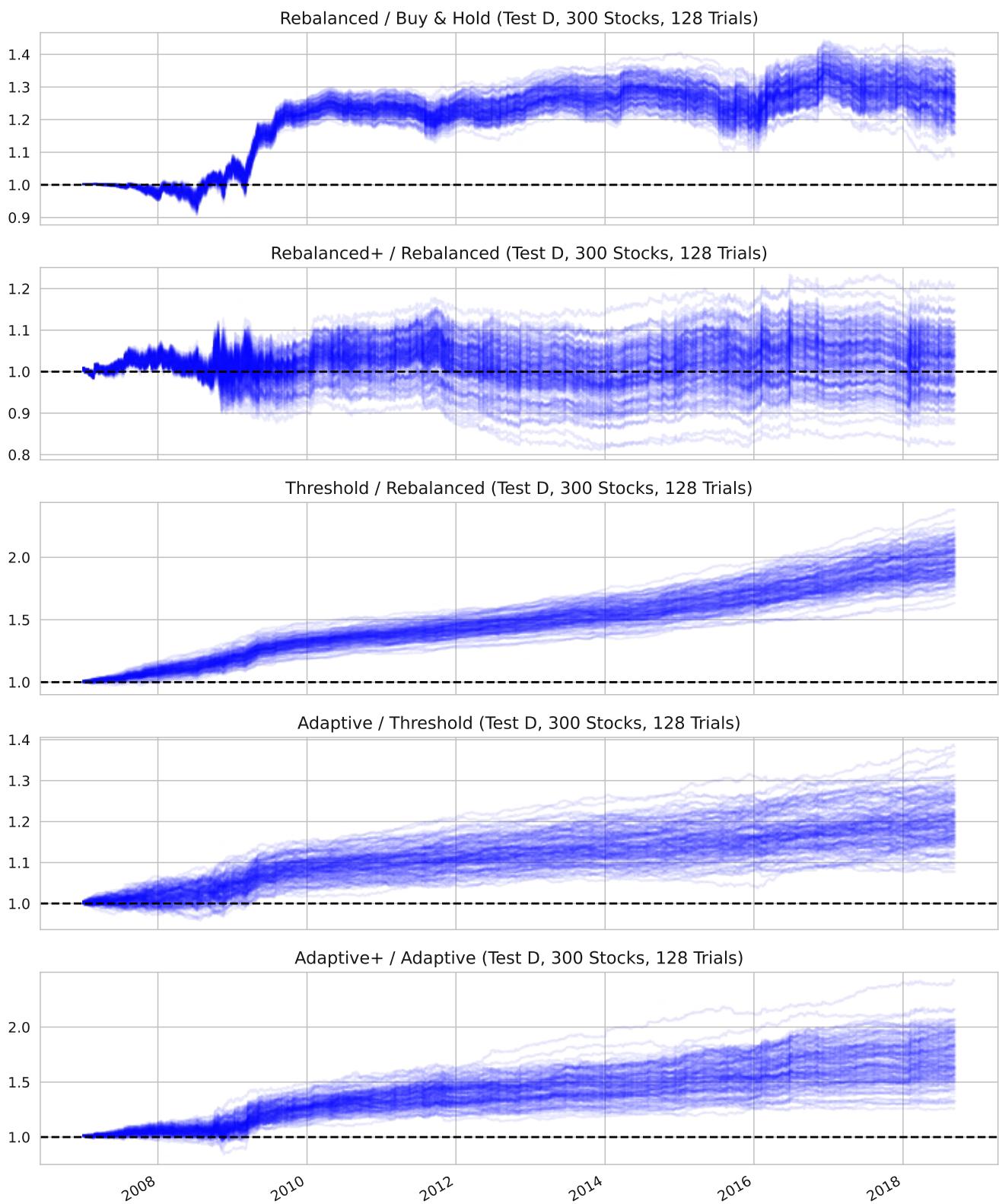


Figure 56: **Test D** – Compare values of 128 random portfolios with **300 stocks** each.

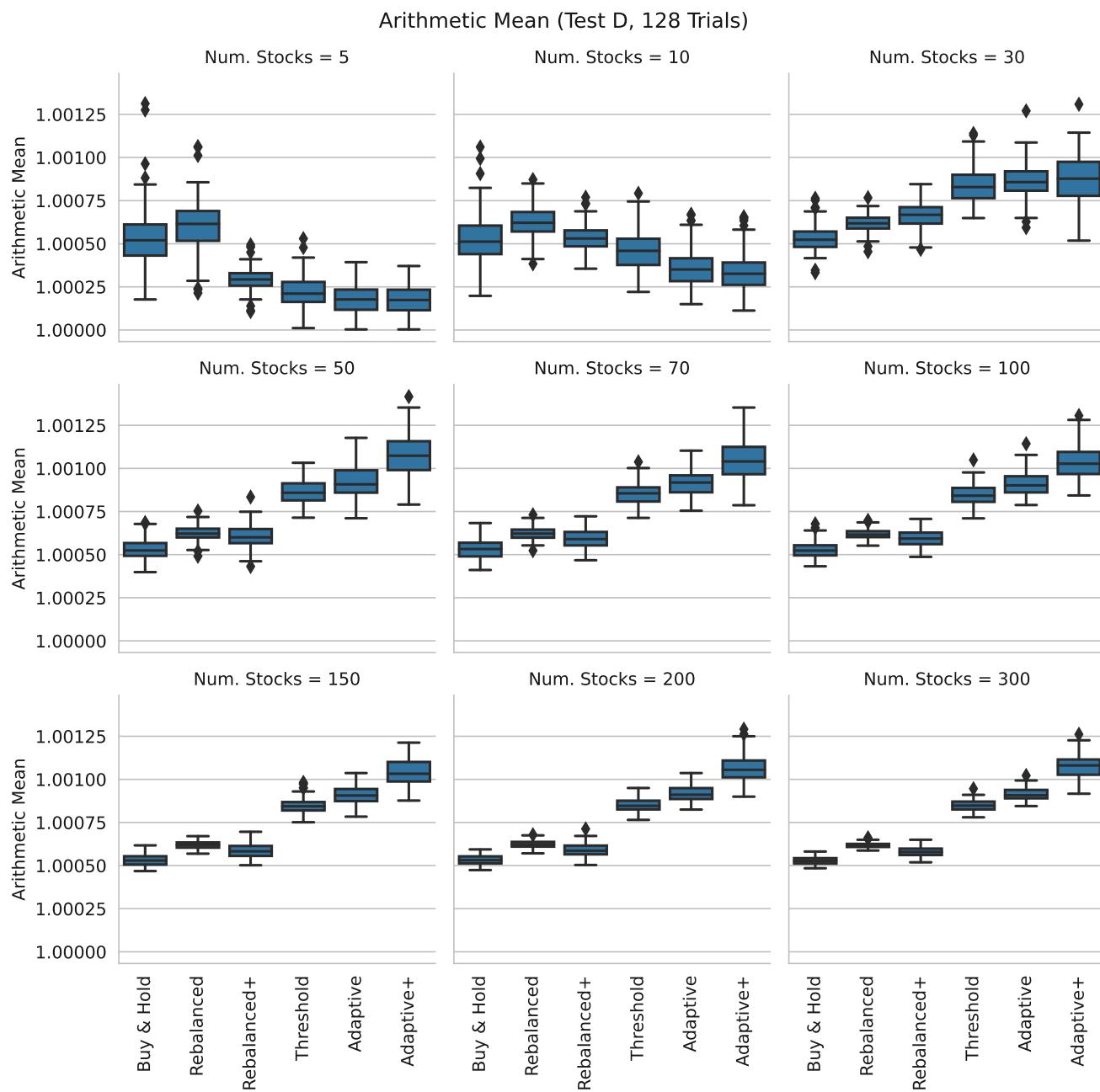


Figure 57: **Test D** – Compare **Arithmetic Mean** daily return of different portfolio methods and sizes.

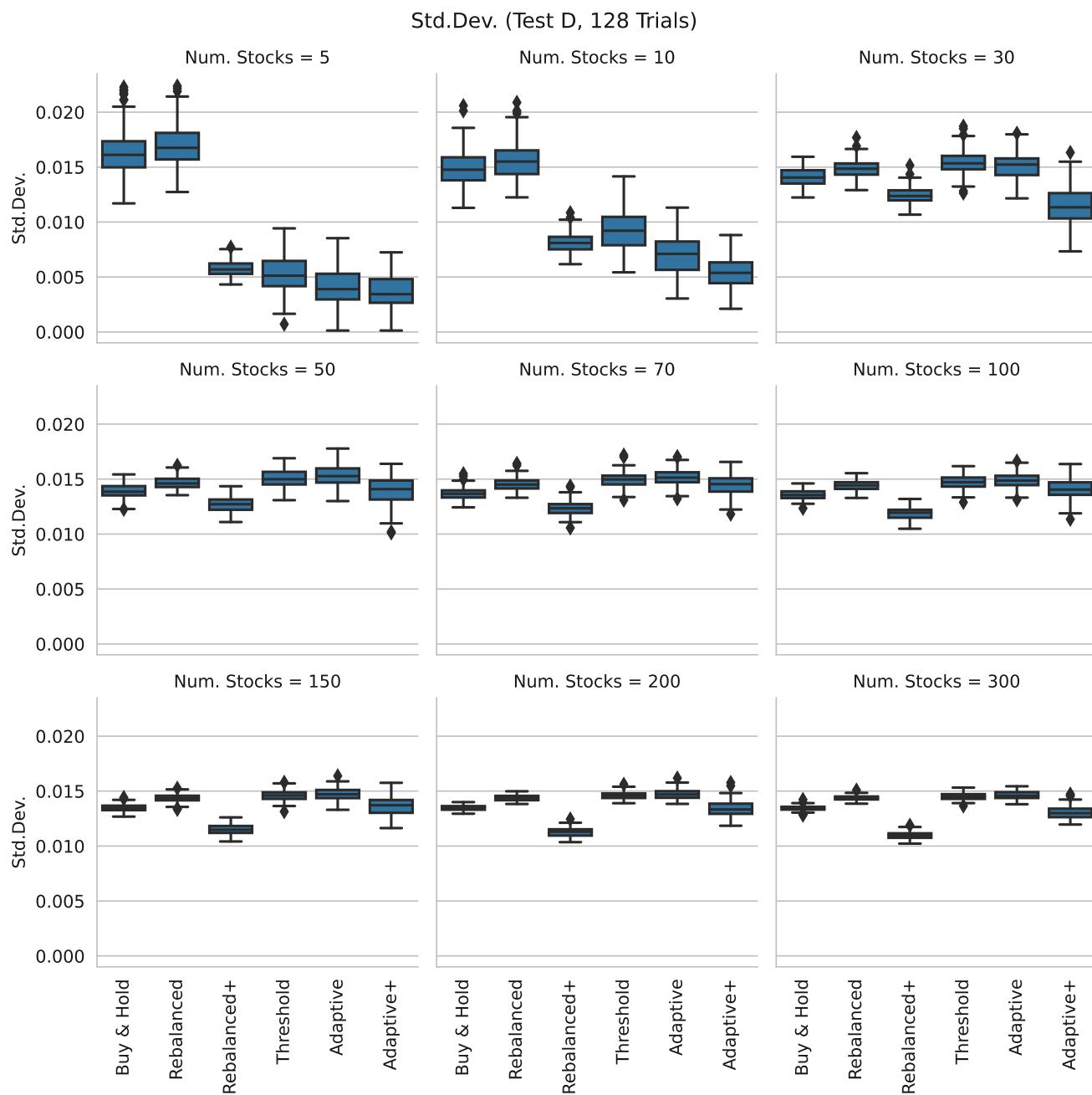


Figure 58: **Test D** – Compare Std.Dev. of the daily return of different portfolio methods and sizes.

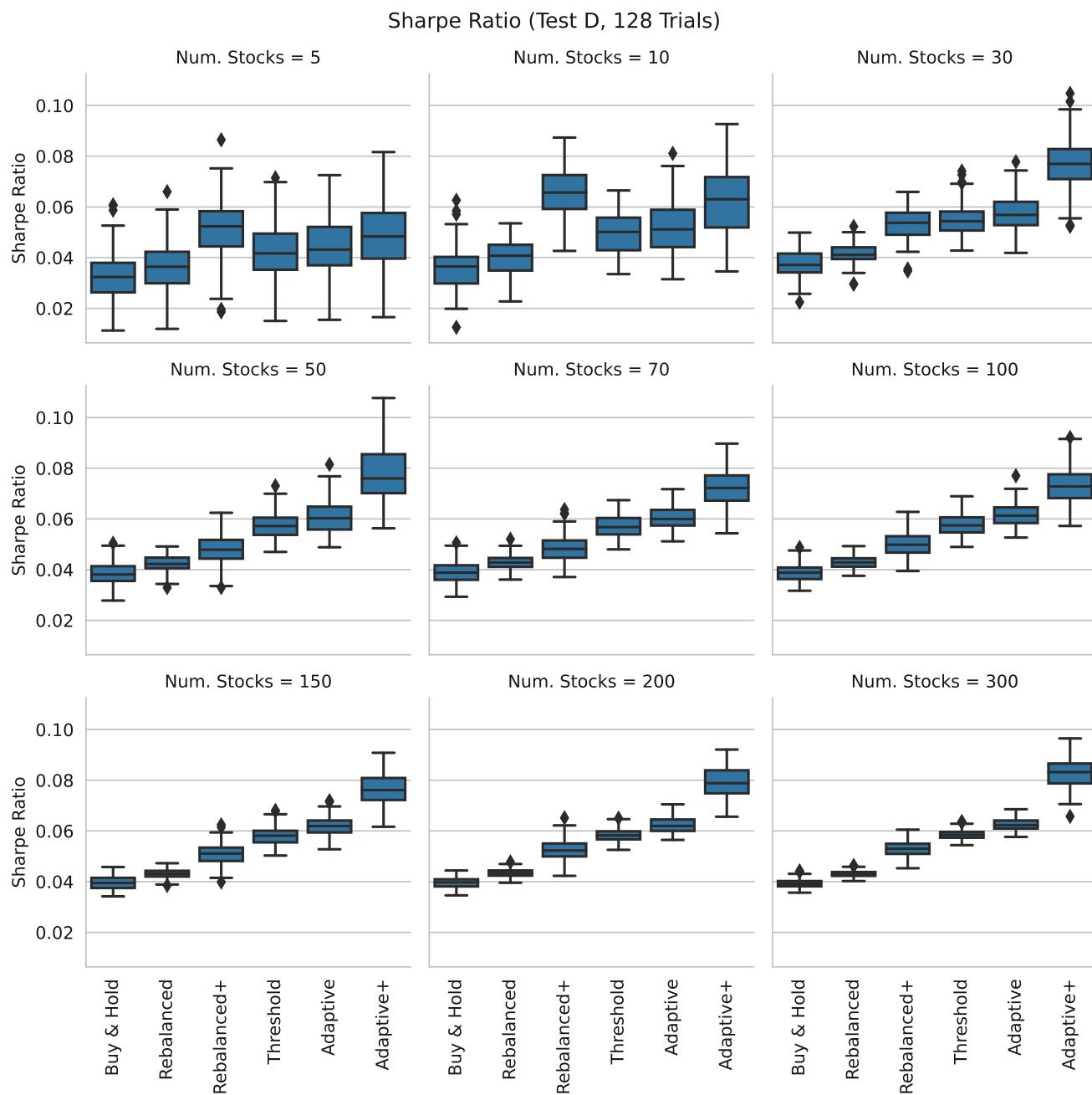


Figure 59: Test D – Compare Sharpe Ratios for daily returns of different portfolio methods and sizes.

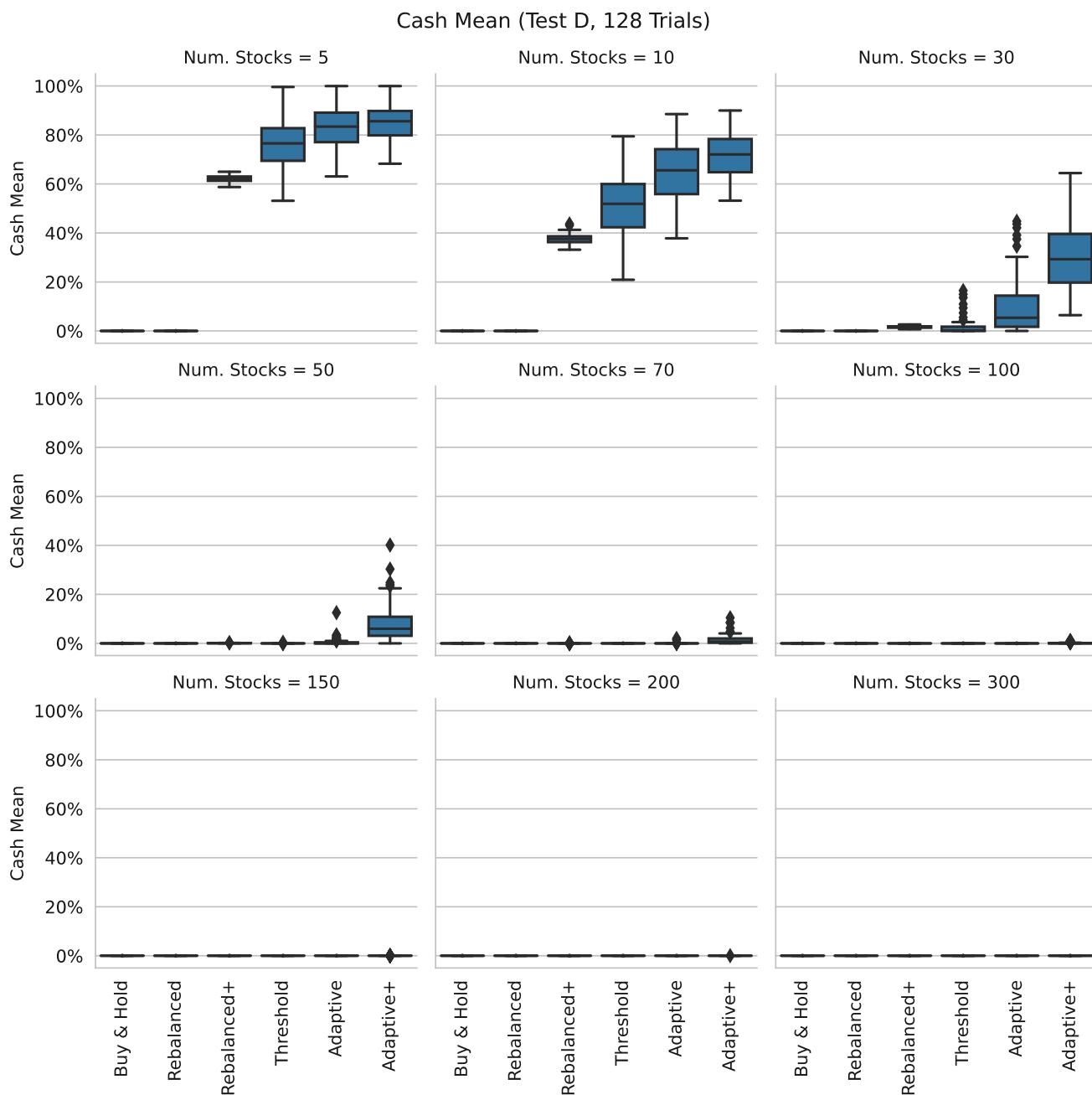


Figure 60: Test D – Compare Cash Mean Daily Position of different portfolio methods and sizes.

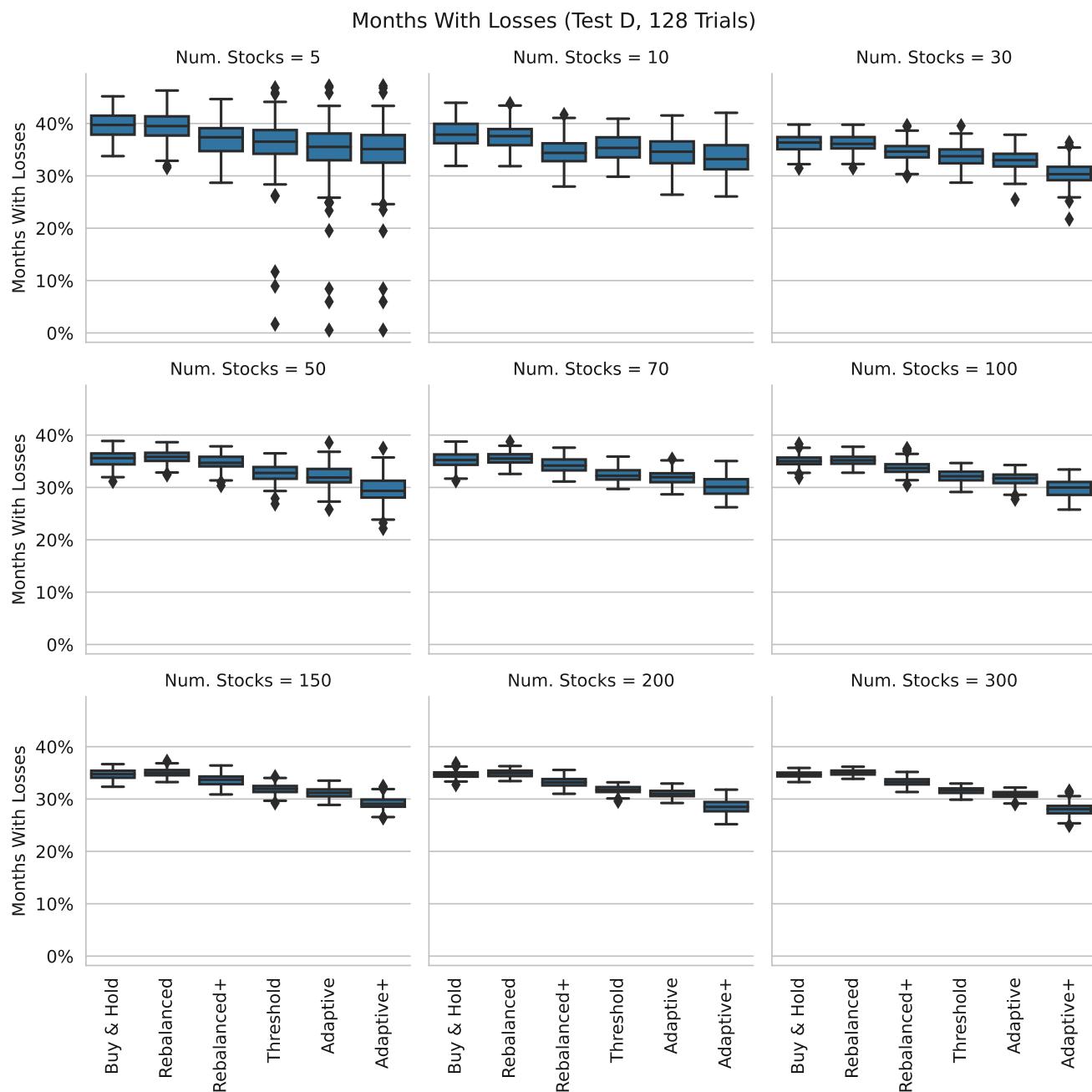


Figure 61: **Test D** – Compare **Months With Losses** of different portfolio methods and sizes.

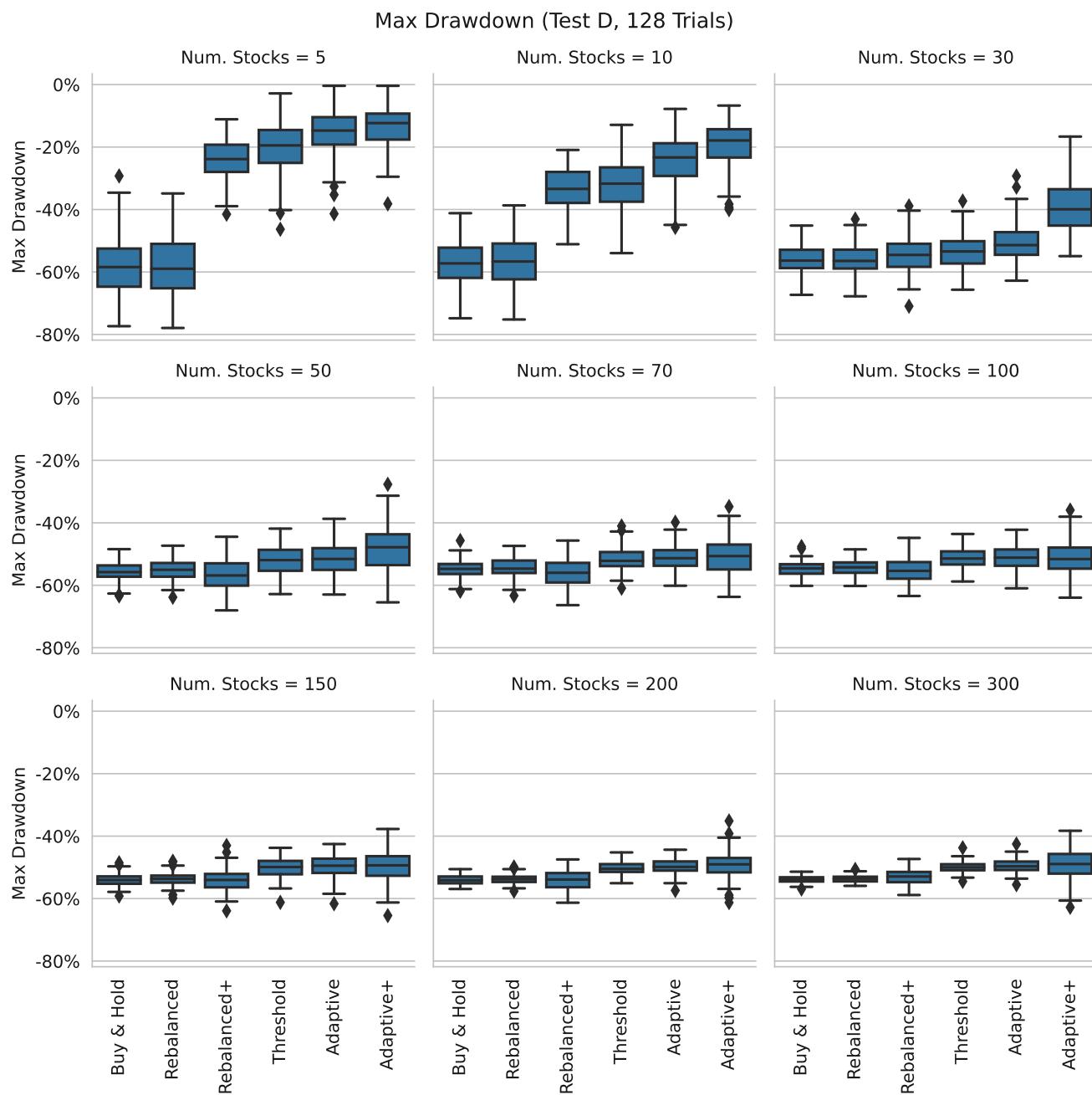


Figure 62: **Test D** – Compare **Max Drawdown** of different portfolio methods and sizes.

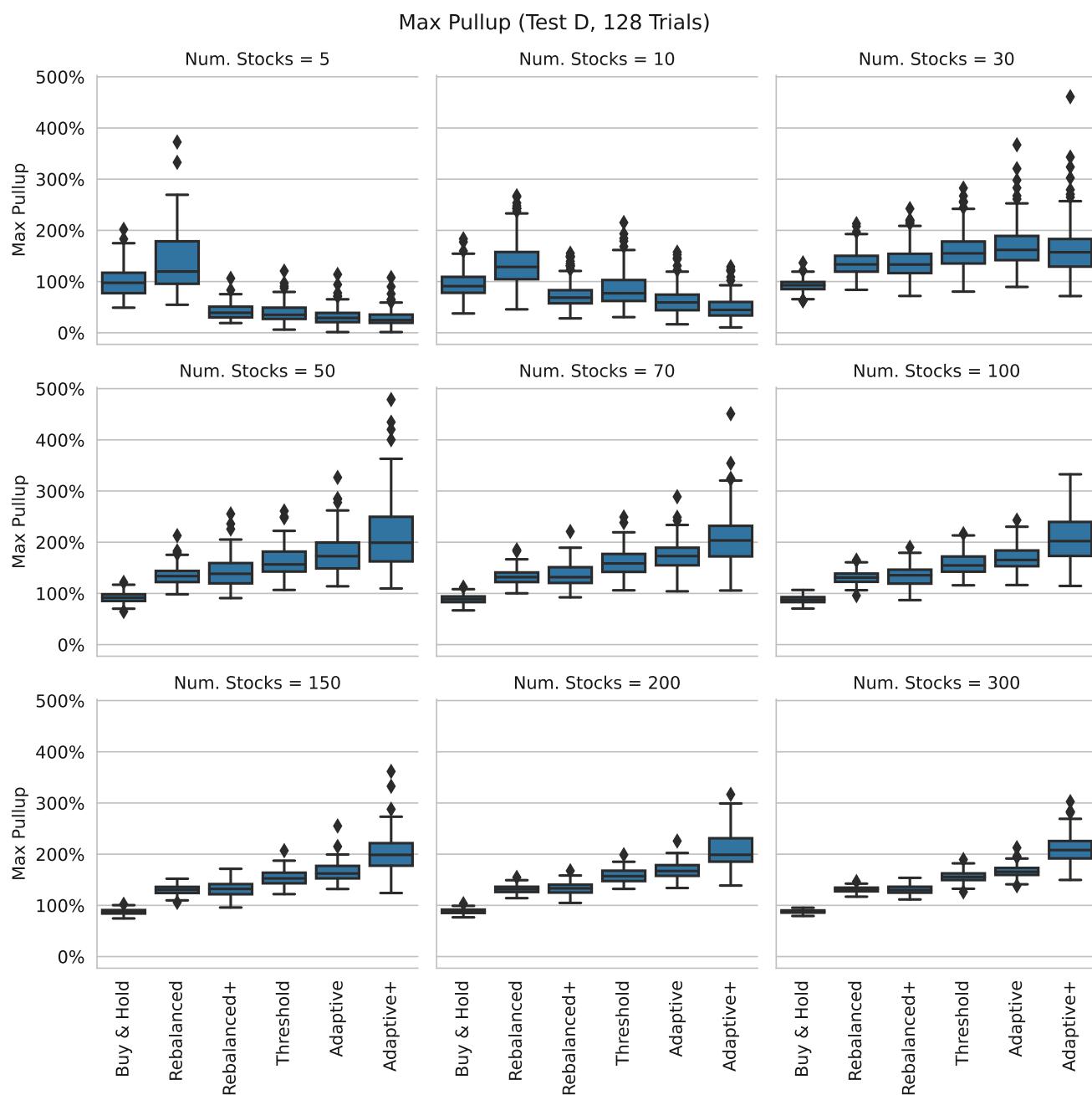


Figure 63: **Test D** – Compare **Max Pullup** of different portfolio methods and sizes.

14 Test E – Noisy Correlations (Robustness)

We now test the robustness of the diversification method from Section 8.8, by using the Adaptive+ portfolio method with different estimates of the correlation matrix. In this section we will use the actual future 2-3 year average stock-returns in the filtering process, but in Section 15 we will use noisy return-estimates in the filtering process as well.

We will compare the following portfolio methods against each other:

- **Rebalanced:** This is the same as in the previous tests. It rebalances the portfolio to equal stock-weights every day. For a portfolio of N stocks in total, each stock is given a portfolio weight of $1/N$ every day. This is used as a trivial base-line portfolio in the performance comparisons.
- **Adaptive:** This is also the same as in the previous tests. It adapts the portfolio-weights so stocks that are estimated to have higher future returns are given higher portfolio weights. But this does not adjust the portfolio weights with regard to the stock-correlations. The Adaptive portfolio method is also used as a base-line in the performance comparisons.
- **Adaptive+:** This is also the same as in the previous tests, and it is just the Adaptive method with adjustment of the portfolio weights using the diversification method from Section 8.8. The correlation matrix is the actual future 10-day correlation of the stock-returns. This gives a base-line in the performance comparisons when we are “cheating” and using future correlation data.

The following variants are the same as the Adaptive+ method but use different correlation estimates:

- **Ad+ Heavy Noise:** This adds heavy noise to the actual future stock-correlations that were also used in the Adaptive+ portfolios. In each time-step, normal-distributed noise with zero mean and 0.5 standard deviation is added to each correlation value. This is very heavy noise that can easily turn a large positive correlation into a large negative correlation, and vice versa.
- **Ad+ Pure Noise:** The correlation matrix is now pure noise and there is no correlation signal from the actual stock-returns. The correlation values are selected in each time-step from a uniform random distribution between -1 and 1.
- **Ad+ Corr. Naive:** This uses a “naive” forecast of the correlations, by using the correlations from the previous 10-day period and assuming they also predict the future correlations.
- **Ad+ Corr. All:** The correlation matrix is calculated for the entire data-period between 2007 and 2018. This is a form of cheating, but the correlations change greatly over time, so these are very imprecise and are more like a kind of average correlation over the entire data-period.
- **Ad+ Corr. Equal:** The correlation coefficients are all set to 0.1.
- **Ad+ Corr. Invert:** The correlation matrix is for the actual future 10-day stock-returns, but the correlations are inverted, so a correlation of e.g. 0.5 becomes -0.5 instead, and vice versa.

Note that the correlation matrix is always “repaired” so it is a valid correlation matrix: Its diagonal is set to 1, the matrix is made symmetrical, and all correlation values are bounded between -1 and 1.

14.1 Test E – Portfolio Values

As in the previous tests, we perform 128 trials for each portfolio-size. For example, if the portfolio size is 30, then we draw 30 random stocks from the data-set and use all the different portfolio methods on those 30 stocks, then we record the portfolio values through time and calculate various performance statistics. We do this 128 times for each choice of portfolio size. The process is described in Section 9.6

Test E – Portfolio Values for 30 Stocks

Figure 64 compares the portfolio values of the different variants of the Adaptive+ portfolios to the Adaptive portfolios, when the portfolios have 30 random stocks. The top-plot in Figure 64 compares the Adaptive+ portfolios to the Adaptive portfolios – the only difference between these two portfolio methods is that the Adaptive+ portfolios also adjust the portfolio weights using the diversification method from Section 8.8, by using the actual future 10-day stock-correlations. The plot shows the ratios of the portfolio values, so a ratio of 1 means that the two portfolios had the same value at that point in time, while a ratio above 1 means that the Adaptive+ portfolio was better, and a ratio below 1 means that the Adaptive portfolio was better at that point in time.

The 2nd plot in Figure 64 compares the “Ad+ Heavy Noise” portfolios to the Adaptive portfolios, which are basically the same as the Adaptive+ portfolios but with heavy noise added to the correlations. The 3rd plot compares the Adaptive+ portfolios where the correlations are pure noise, and so on.

The sub-plots in Figure 64 show that all the variants of the Adaptive+ portfolios performed worse than the Adaptive portfolios for most of this 11-year period. The explanation may be found in Figure 70 which shows that most of the Adaptive+ variants had much larger cash-positions than the Adaptive portfolios, with the exception being the “Ad+ Corr. Equal” and “Ad+ Corr. Invert” variants which use equal and inverted correlation values, respectively. These portfolios had roughly the same average cash-positions as the Adaptive portfolios, and Figure 64 shows they also had the best performance of the Adaptive+ variants. Note that the scales on the y-axis are different in many of these sub-plots.

Test E – Portfolio Values for 100 Stocks

Figure 65 compares the portfolio values when each portfolio contains 100 random stocks. The variants of the Adaptive+ portfolios generally performed better than the Adaptive portfolios, although there were some periods in time where the ratios were declining, which means the Adaptive portfolios were better during those periods. But overall the Adaptive+ portfolios with different correlation variants performed better than the Adaptive portfolios which did not adjust the portfolio weights for correlation. Even the Adaptive+ portfolios whose correlation estimates were pure noise, or all equal to 0.1, or all inverted, often performed better than the Adaptive portfolios which did not adjust for correlation.

Figure 70 once again shows that the various Adaptive+ portfolios usually held a large part of their portfolios in cash. The Adaptive+ Equal and Inverted variants usually held around 20% on average of their portfolios in cash, while the other variants held around 40% on average of their portfolios in cash. The Adaptive portfolios only held around 10% of their portfolios in cash on average. So the

diversification method moves a part of the portfolio into cash as expected, but it is remarkable that even when the correlation matrix is e.g. pure noise or completely inverted, it still improves the performance.

Test E – Portfolio Values for 300 Stocks

Figure 66 compares the portfolio values when each portfolio contains 300 random stocks. All of the Adaptive+ portfolio variants performed much better than the Adaptive portfolios which did not adjust for correlation. Figure 70 shows that all the Adaptive+ variants had less than 20% of their portfolios in cash on average, with the Adaptive+ Equal and Inverted variants having very little of their portfolios in cash. So perhaps the Adaptive+ variants could perform even better with larger portfolio sizes.

Note the scales are different for the plots in Figure 66, so the Adaptive+ variants have quite different effects on the performance, but they are all significant improvements over the Adaptive portfolios. The top-plot in Figure 66 shows that the Adaptive+ portfolios which used the actual future 10-day stock-correlations also performed the best, with the typical improvement being around 5-6 times better than the Adaptive portfolios, but a few Adaptive+ portfolios being as much as 10x better than the Adaptive portfolios. The bottom-plot shows that inverting the future 10-day correlations performed the worst, albeit still being an improvement over the Adaptive portfolios which did not adjust the portfolio weights for stock-correlations. The inverted correlations gave a typical improvement between 1.25 and 1.50, which means a 25-50% improvement over the Adaptive portfolios after 11 years of investing.

14.2 Test E – Performance Statistics

Figure 67 compares the arithmetic mean daily return. For small portfolio sizes of only 5 or 10 stocks, the trivial Rebalanced portfolios were much better than both the Adaptive and all of the Adaptive+ portfolios, but Figure 70 shows that this was just because the Rebalanced portfolios were always fully invested in stocks, while the various Adaptive portfolios usually held more than 90% of their portfolios in cash. For portfolios of 30, 50 or 70 stocks, the Adaptive portfolios performed much better than the Rebalanced portfolios, and the various Adaptive+ portfolios performed slightly worse than the Adaptive portfolios. Once again Figure 70 suggests the explanation is that the Adaptive+ portfolios had much larger cash-positions than the Adaptive portfolios, because the diversification method which is used by all the Adaptive+ variants, can move a lot of the portfolio into cash. For portfolios with 100 stocks, the Adaptive+ portfolios which used the actual future 10-day stock-correlations, performed roughly on par with the Adaptive portfolios, while most of the other Adaptive+ variants performed slightly worse. For portfolios with 150 stocks, and especially with 200 or even 300 stocks, all the Adaptive+ variants had significantly higher mean daily return than the Adaptive portfolios, even the Adaptive+ variants that used completely random or malformed correlation estimates. This is really quite remarkable and a possible reason for this will be discussed further below.

Figure 68 shows the standard deviation for the daily returns of the different portfolio methods. For smaller portfolio sizes the standard deviations of the Adaptive+ variants are generally much lower than for both the Adaptive and Rebalanced portfolios. But the reason is just that large parts of the portfolios for the Adaptive+ variants were held in cash, as shown in Figure 70. For larger portfolios of 150, 200 and 300 stocks, the standard deviations are roughly the same for all portfolio methods.

Figure 69 shows the Sharpe Ratios, where some portfolio methods had consistently higher and therefore better Sharpe Ratios than other portfolio methods. For all portfolio sizes the highest Sharpe Ratios were for the Adaptive+ portfolios which use the actual future 10-day stock-correlations. The 2nd best Sharpe Ratio is for the Adaptive+ portfolios which add heavy noise to the actual future 10-day stock-correlations. And the 3rd best Sharpe Ratio is for the Adaptive+ portfolios which use the previous 10-day stock-correlations as a “naive” forecast of their future correlations. Remarkably, the Adaptive+ portfolios with completely random or malformed correlation matrices still have slightly higher Sharpe Ratios than the Adaptive portfolios which do not use the diversification method at all. This again shows how extremely robust the diversification method is to noise and errors in the correlation estimates.

Figure 71 compares the percentages of months with losses for the different portfolio methods. In all cases the Adaptive and Adaptive+ variants have much better performance than the trivial Rebalanced portfolios. For smaller portfolio sizes all the Adaptive and Adaptive+ variants have roughly the same performance, probably because they all hold so much of their portfolios in cash. For portfolios with 50 stocks and above, the Adaptive+ portfolios had significantly better performance on this metric, and the Adaptive+ portfolios with heavy noise had the second best performance. Even the Adaptive+ variants with purely random or malformed correlation matrices performed roughly the same as the Adaptive portfolios which did not adjust the portfolio weights for diversification, again showing the robustness of the diversification method on this performance metric.

Figure 72 compares the Max Drawdown for the different portfolio methods. In all cases the Adaptive and Adaptive+ variants performed significantly better than the Rebalanced portfolios on this metric, and for smaller portfolio sizes the performance was much better, because the Adaptive and Adaptive+ variants held a large part of their portfolios in cash. For larger portfolio sizes of 150, 200 and 300 stocks, the Adaptive+ portfolios with heavy noise in the correlation matrix performed slightly better than all other portfolios, and the Adaptive+ variants with all correlations either being equal or inverted performed roughly on par with the Adaptive portfolios that did not adjust for correlation at all. Recall that the originally desired portfolio weights from the filtering process were calculated using the actual future 2-3 year average stock-returns, which means that all these portfolio methods are completely “blind” to any impending stock-market crashes, which is the reason all the portfolio methods have fairly high Max Drawdowns. If you can somehow predict an impending stock-market crash, you can incorporate this into the filtering process when calculating the desired portfolio-weights, and thereby dramatically increase the performance. But it is very hard to predict stock-market crashes.

Figure 73 compares the Max Pullup for the different portfolio methods, which measures how well the portfolios performed during the recovery-phases of stock-market crashes. For small portfolio sizes of only 5 or 10 stocks the Adaptive and Adaptive+ variants had worse Max Pullups than the trivial Rebalanced portfolios, but already for portfolios with 30 stocks they start to perform better, and for portfolios with 50 stocks or more, the Adaptive and Adaptive+ portfolios all performed much better than the trivial Rebalanced portfolios. For portfolios with 100 stocks and above, all the Adaptive+ variants performed much better than the Adaptive portfolios which did not adjust for correlation. Even the Adaptive+ variants that use equal or inverted correlations, performed better than the Adaptive portfolios. Interestingly, the best performance was for the Adaptive+ portfolios whose correlations had heavy noise, and the portfolios that used pure noise for their correlations had roughly the same Max

Pullup as the Adaptive+ portfolios that used the actual future 10-day stock-correlations. So the diversification method is often very beneficial for a portfolio's performance during stock-market recoveries, and it is extremely robust to noisy and malformed estimates of the correlation matrix.

14.3 Summary

In this section we tested the diversification method from Section 8.8 when using various kinds of noisy and malformed correlation matrices. For smaller portfolio sizes it performed worse than the portfolios whose weights were not adjusted for correlation, but this was merely because the diversification method moves a lot of the portfolio into cash when there are too few assets available for investing.

For larger portfolio sizes the diversification method works extremely well. It performs significantly better when the correlation matrix accurately represents the future stock-correlations. But it is truly remarkable that the diversification method still works so well in the presence of heavy noise in the correlation matrix – it even works with a correlation matrix that is completely random, or where all the correlation values are set to 0.1, or all the correlations are inverted. That is really quite stunning!

It would require further research to establish the exact reason why the diversification method is so extremely robust to correlation estimates that are completely malformed. But a brief explanation is perhaps that correlations actually change continually over time, and they are merely serving as a rough guide to whether different assets tend to move up or down together in price. And because our new diversification method is only allowed to decrease the portfolio-weights, the worst that can happen is that it moves too much of the portfolio into cash, as we indeed saw for the smaller portfolio-sizes in the tests above. So nothing bad happens when the correlation estimates are wrong, but when the correlations are roughly correct, the diversification method still benefits the portfolio's performance.

When using completely random or malformed correlation estimates, it is almost as if the diversification method is like the proverbial blind hen that sometimes gets lucky and finds a grain of corn. In the next section we will test if the diversification method still manages to “get lucky” when there is also heavy noise in the future stock-returns that are used in the filtering process to calculate the portfolio weights.

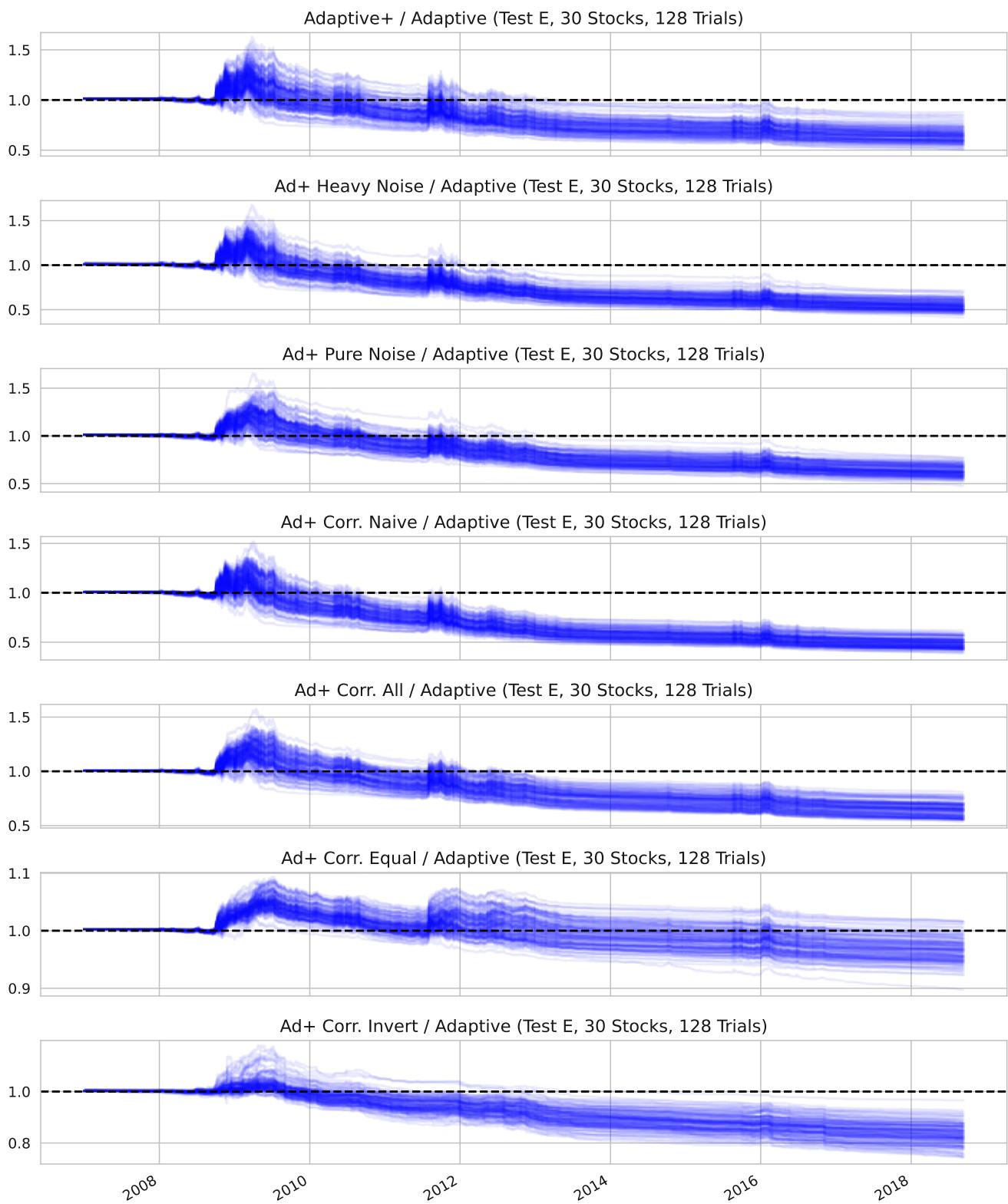


Figure 64: **Test E** – Compare values of 128 random portfolios with **30 stocks** each.

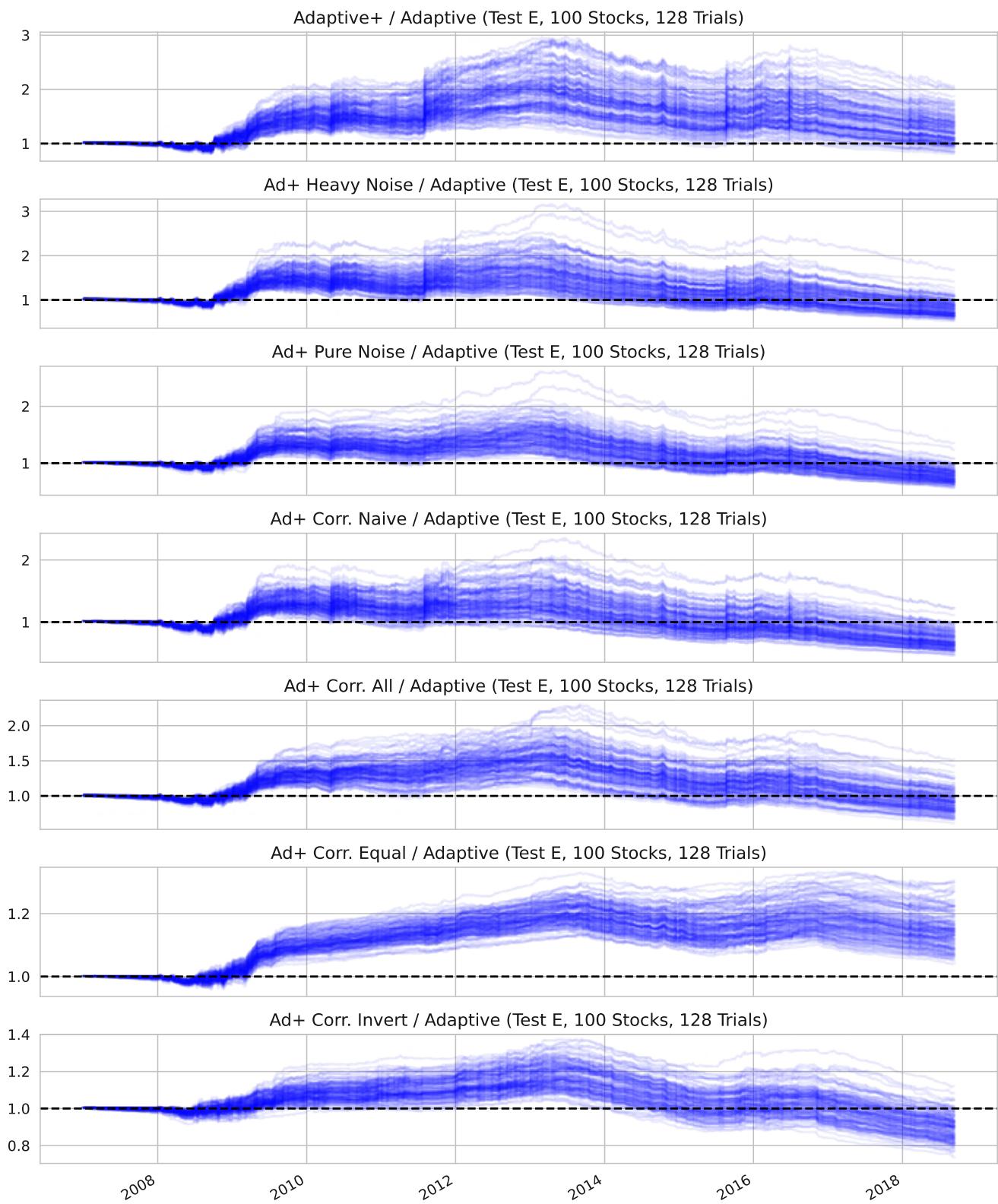


Figure 65: **Test E** – Compare values of 128 random portfolios with **100 stocks** each.

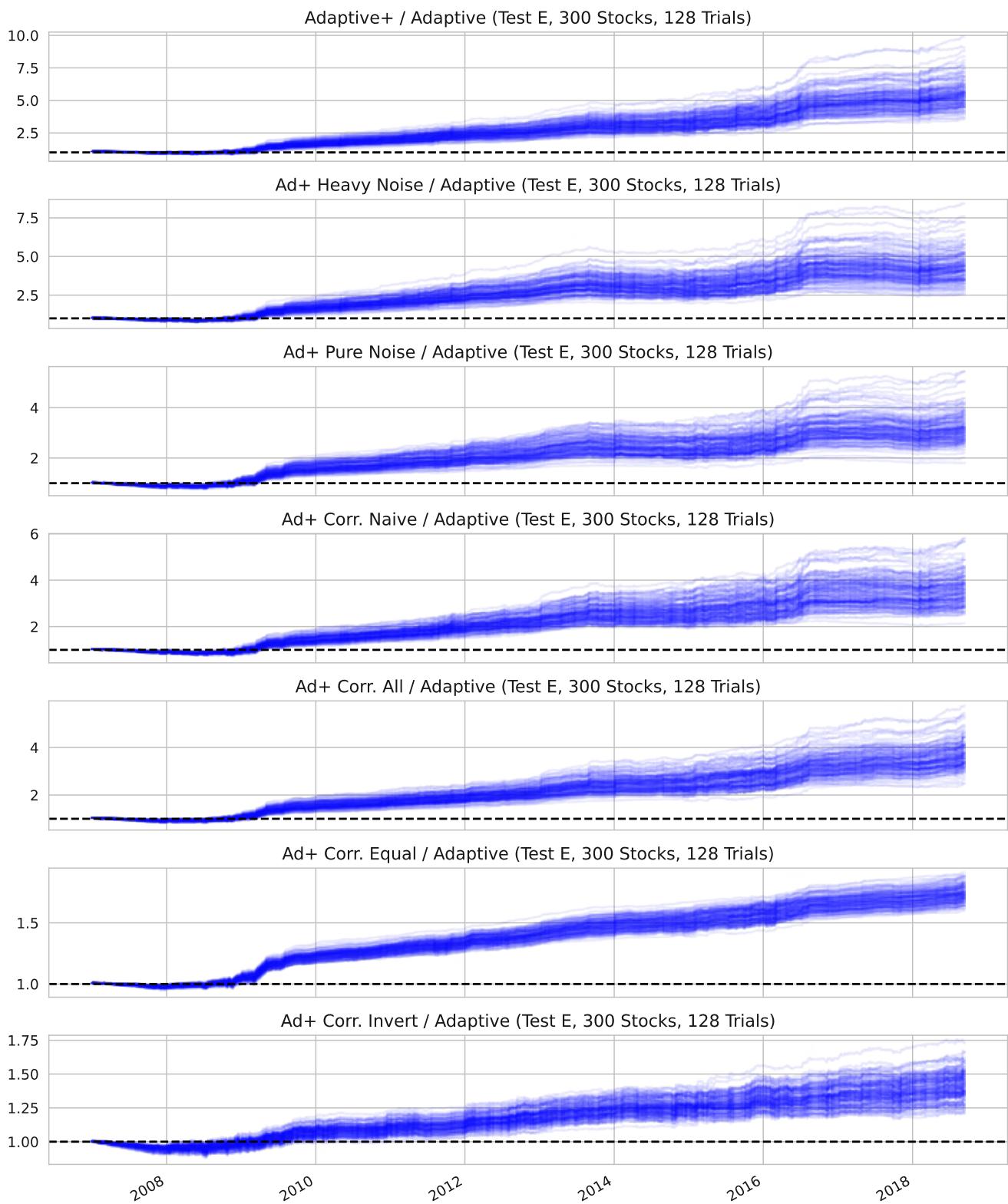


Figure 66: **Test E** – Compare values of 128 random portfolios with **300 stocks** each.

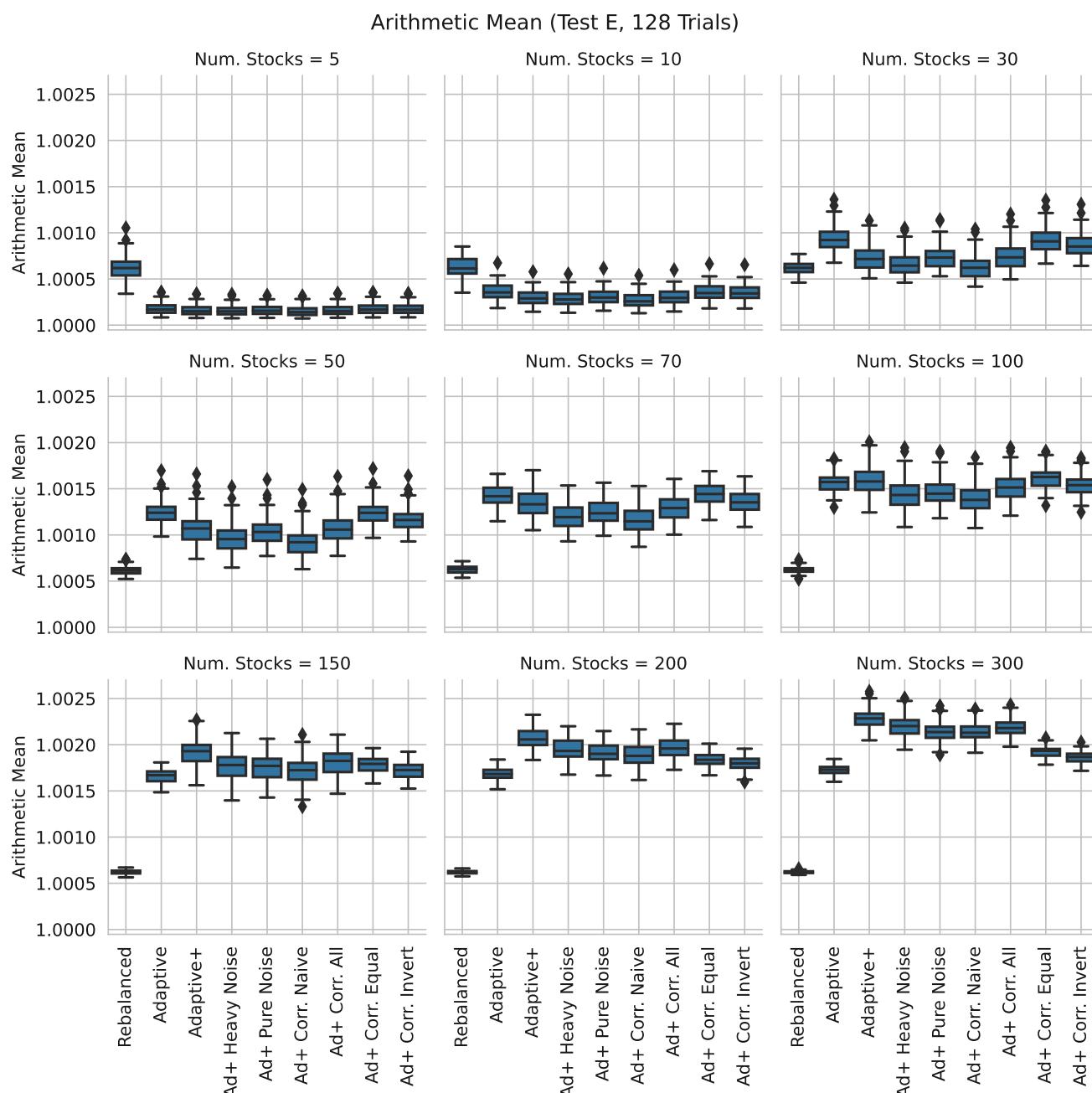


Figure 67: Test E – Compare Arithmetic Mean daily return of different portfolio methods and sizes.

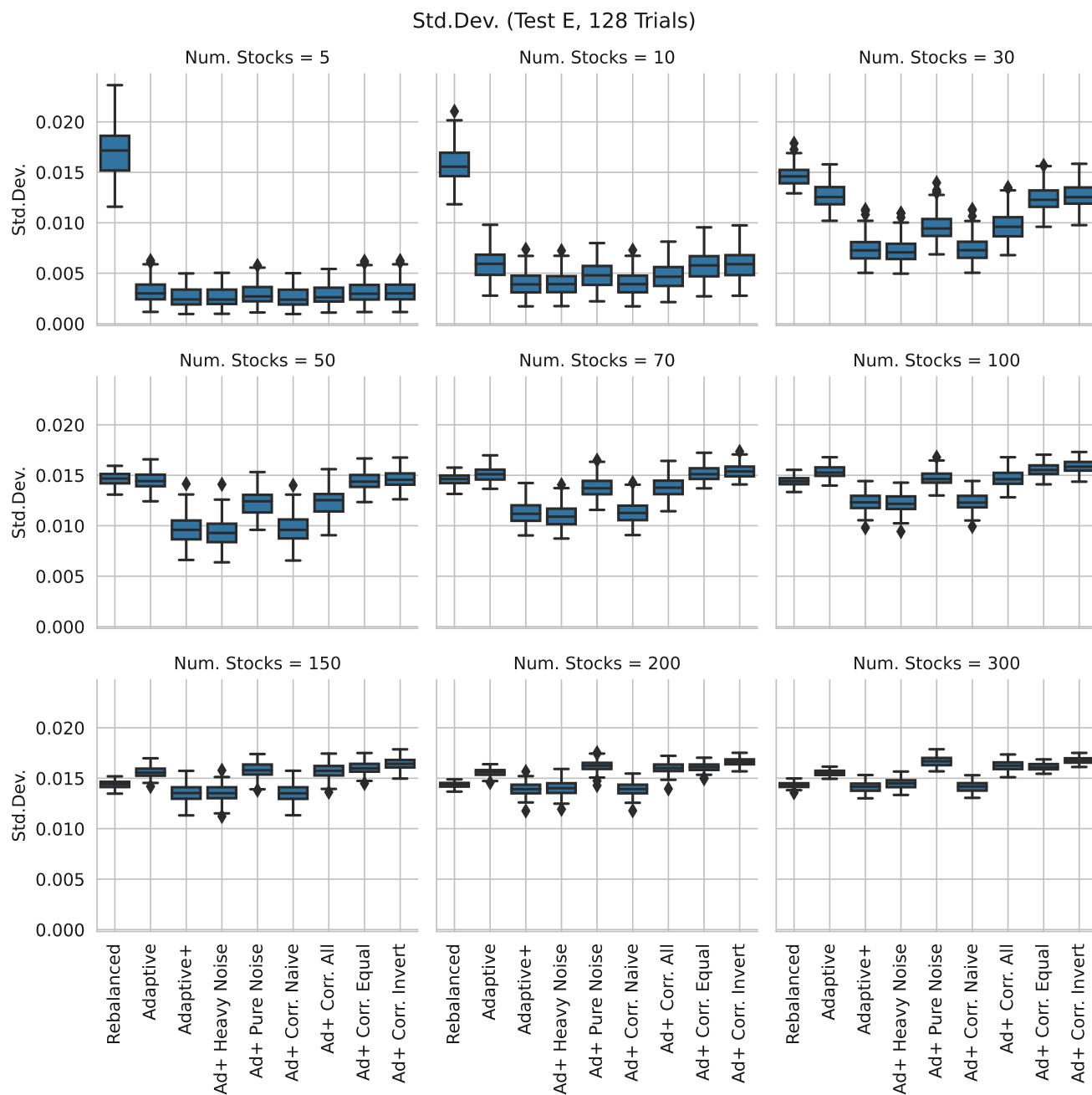


Figure 68: **Test E** – Compare **Std.Dev.** of the daily return of different portfolio methods and sizes.

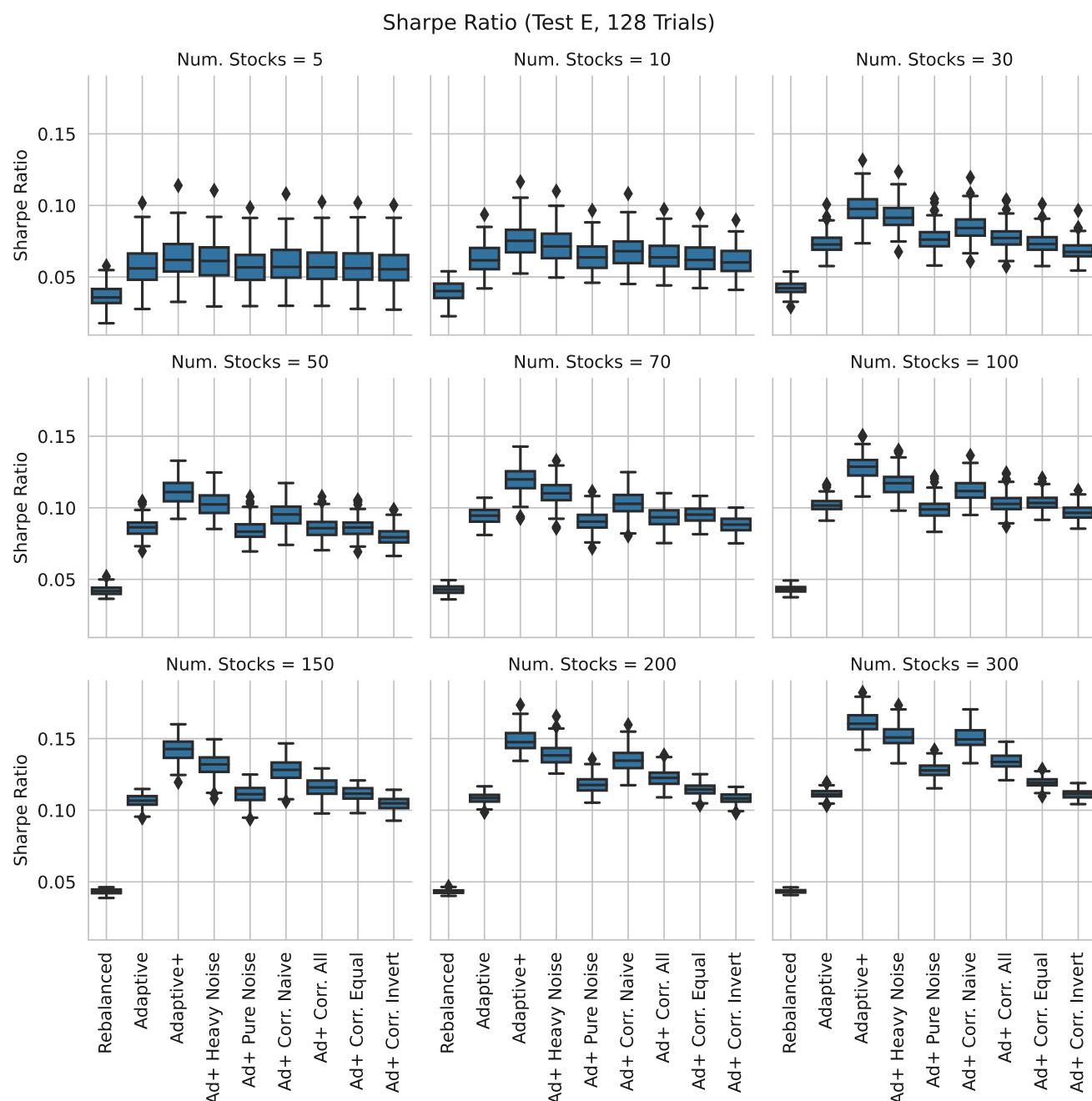


Figure 69: **Test E** – Compare **Sharpe Ratios** for daily returns of different portfolio methods and sizes.

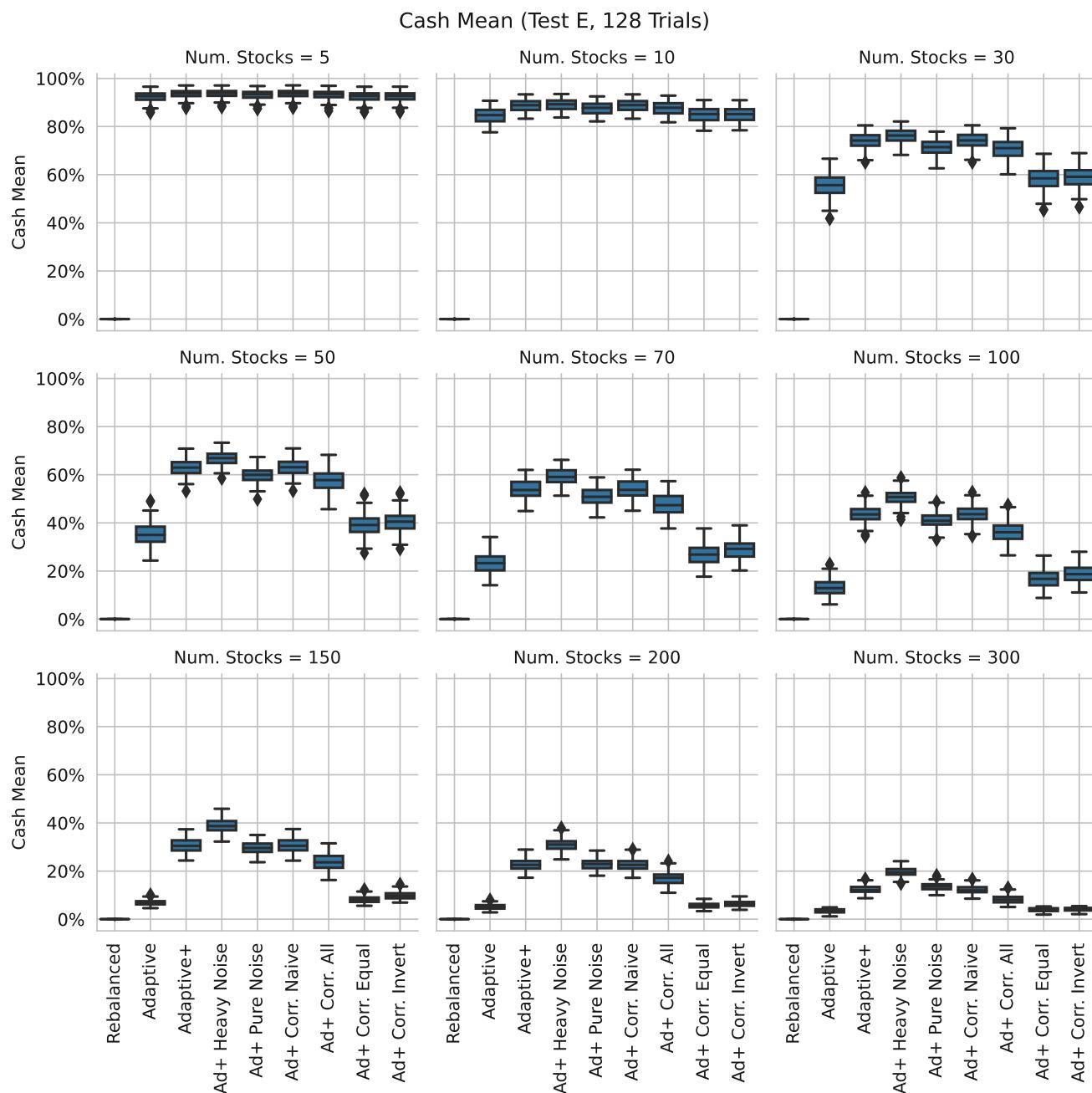


Figure 70: **Test E** – Compare **Cash Mean Daily Position** of different portfolio methods and sizes.

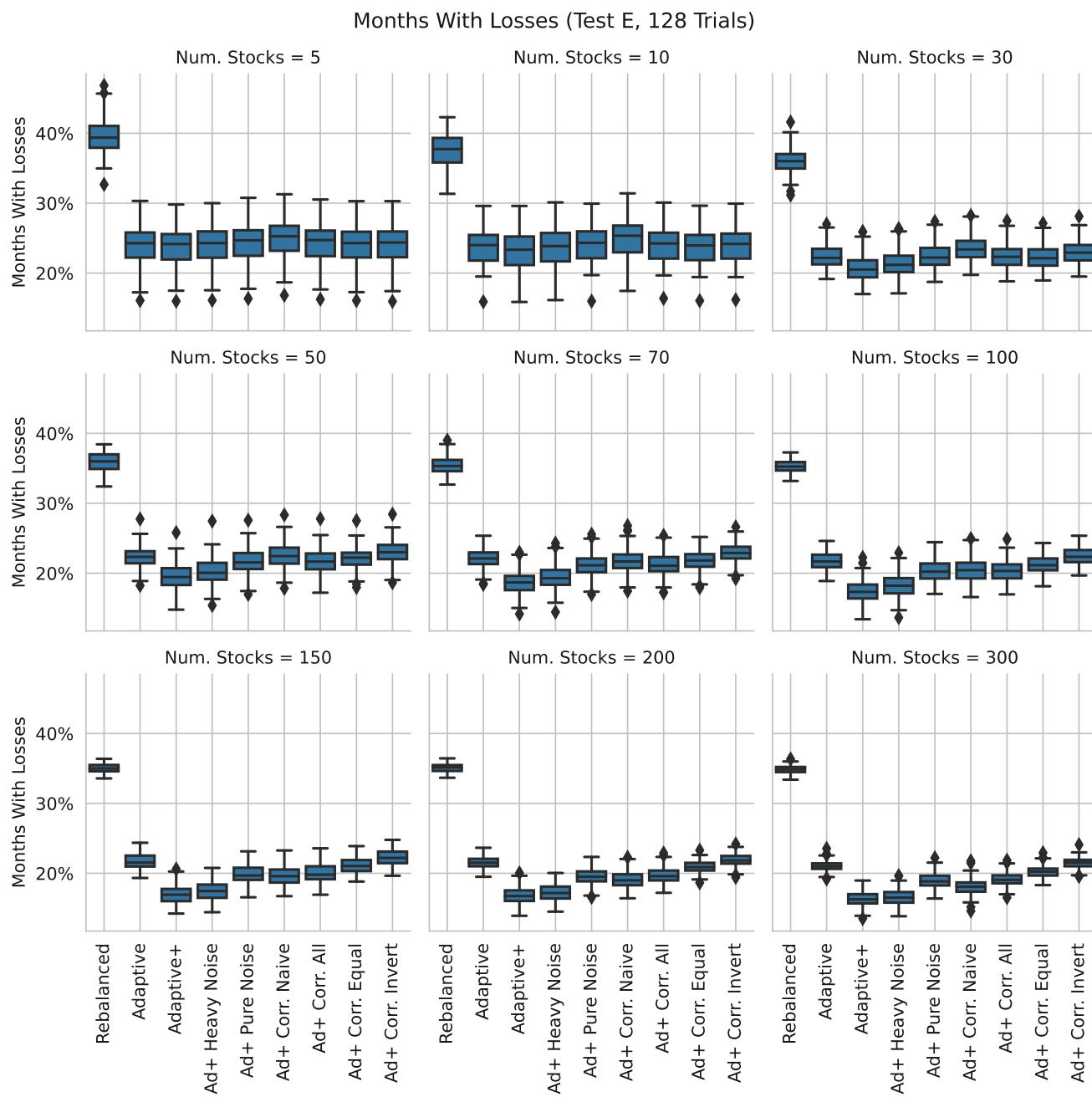


Figure 71: **Test E – Compare Months With Losses** of different portfolio methods and sizes.

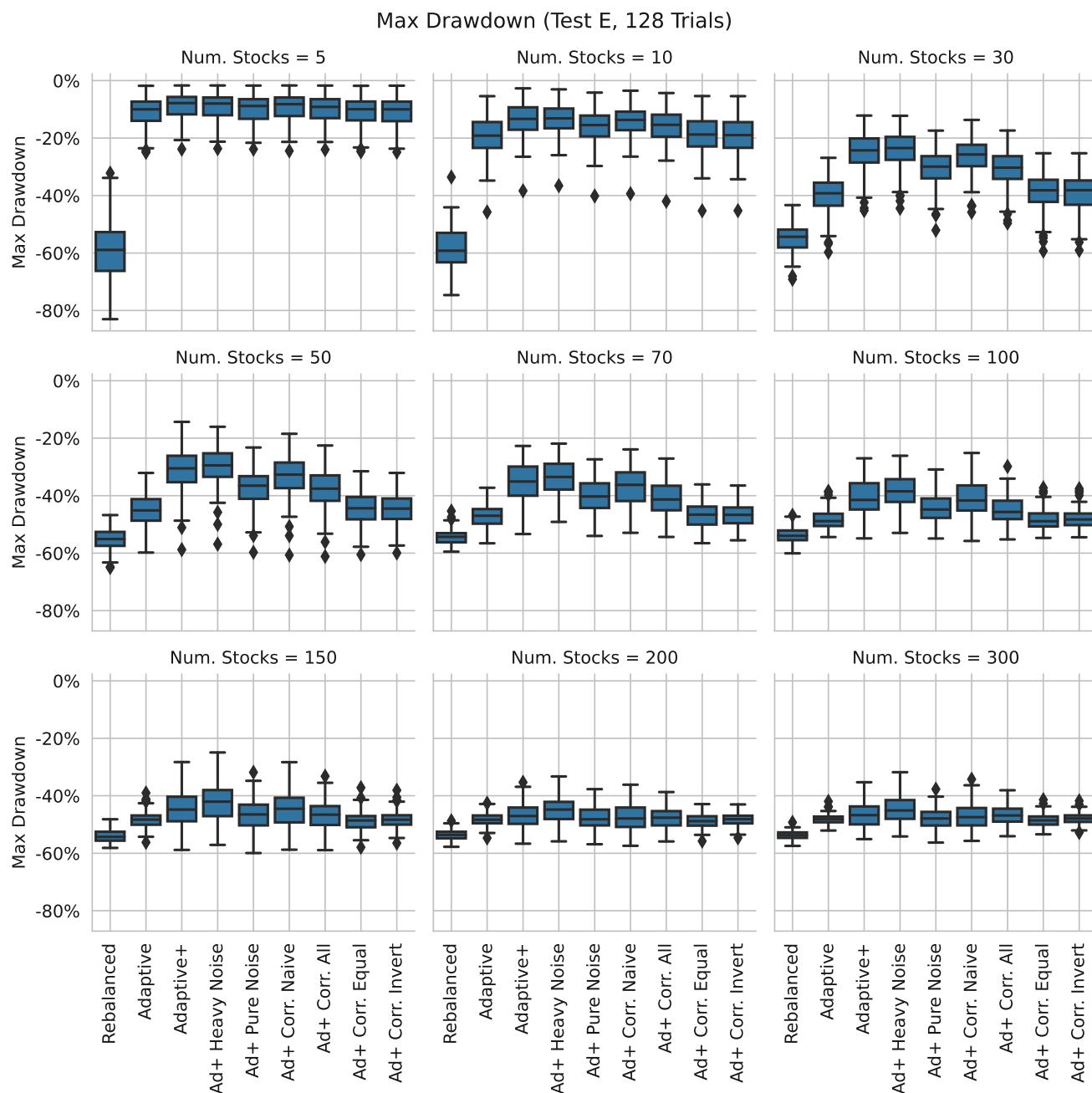


Figure 72: **Test E – Compare Max Drawdown** of different portfolio methods and sizes.

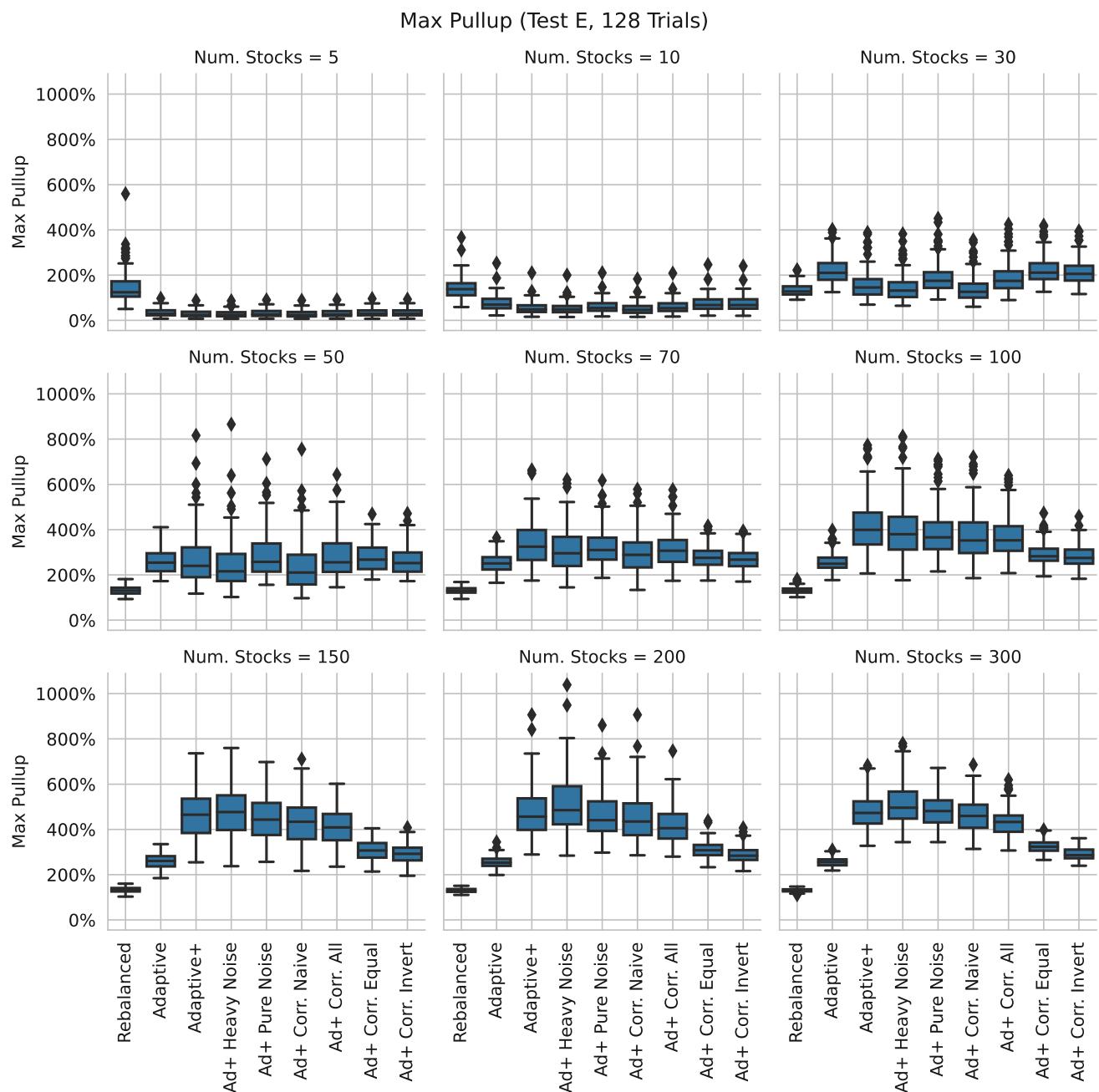


Figure 73: Test E – Compare Max Pullup of different portfolio methods and sizes.

15 Test F – Noisy Returns & Correlations (Robustness)

In the previous section we tested the diversification method from Section 8.8 with correlation matrices that were very noisy or completely malformed. But the original portfolio weights that were calculated using the Adaptive filtering process in Section 7.2, were still calculated from the actual future 2-3 year average stock-returns. It is possible that the diversification method was only able to cope with the heavy noisy and malformed correlations, because the original portfolio weights were calculated from the actual future stock-returns, and because the diversification method is only allowed to decrease the portfolio weights, the portfolios still performed well in the long run.

In this section we will therefore test the diversification method when both the correlation matrix and the future stock-returns are very noisy. The noise for the correlation matrix is the same as used in Test E from Section 14, and the noise for the future stock-returns is the same as in Test D from Section 13.

15.1 Test F – Portfolio Values

Figure 74 compares the portfolio values over time when each portfolio has 30 random stocks. The top-plot shows the Adaptive+ portfolios mostly performed better than the Adaptive portfolios. This is just a base-line comparison which shows that when the Adaptive+ portfolios used the actual future 10-day correlations, they performed better than the Adaptive portfolios which did not adjust the portfolio weights with regard to stock-correlations. But keep in mind that the future 2-3 year stock-returns are very noisy, and these are used to calculate the weights for both the Adaptive and Adaptive+ portfolios.

The remaining plots in Figure 74 are very similar to each other, where the effect of the various kinds of noisy and malformed correlations seem to have had a somewhat random effect on the portfolio values. Sometimes the Adaptive+ variants were better than the Adaptive portfolios, and sometimes they were worse. Note that the scales on the y-axes are different. The “Ad+ Corr. Equal” portfolios where all the correlations are equal to 0.1 seemed to have had the smallest effect on the portfolio values, while the “Ad+ Heavy Noise” and “Ad+ Corr. Naive” often had a large impact on the portfolio values, although it seems to be random whether the impact was positive or negative. This is probably just because the diversification method does not work so well for smaller portfolios, as it moves a lot of the portfolio into cash, which makes it under-perform the Adaptive portfolios that do not adjust for correlations.

Figure 75 compares the portfolio values when each portfolio has 100 random stocks. All the Adaptive+ variants performed better than the Adaptive portfolios. There were only very few of the trials for some of the malformed correlation matrices that made those portfolios perform worse, see e.g. the bottom-plot which uses the actual future 10-day stock-correlations that are all inverted. The Adaptive+ variants with very noisy or completely malformed correlations, usually performed significantly better than the Adaptive portfolios which did not adjust for stock-correlations. Note again that the scales on the y-axes are different in the sub-plots, which means that some of the noisy correlations were more harmful than others. After the 11-year investment period, the “Ad+ Corr. Naive” portfolios which used the last 10-day stock-correlations as a “naive” estimate of the future correlations, seemed to perform slightly better than the others. And the Adaptive+ variants with equal or inverted correlations performed the worst.

Figure 76 compares the portfolio values when each portfolio has 300 random stocks. This is very similar to Figure 75 that we just discussed for portfolios of 100 stocks. All the Adaptive+ variants performed significantly better than the Adaptive portfolios which did not adjust the portfolio weights for correlations. Once again it seems that the “Ad+ Corr. Naive” portfolios were slightly better than the others, and the Adaptive+ variants that used equal or inverted correlations were the worst, although these also managed to significantly improve the portfolio values compared to the Adaptive portfolios.

These plots have shown that for small portfolios with only 30 stocks, the diversification method from Section 8.8 has a seemingly random effect on the investment return over time. This is probably because portfolios with only 30 stocks are too small for the diversification method to function properly. Already for portfolios with 100 stocks, the diversification method significantly improves the investment return over time, and it works even better for portfolios with 300 stocks. It is truly remarkable that the diversification method works so well in the presence of very noisy and malformed correlations, as well as heavy noise in the estimates for the future stock-returns.

15.2 Test F – Performance Statistics

Figure 77 compares the arithmetic mean daily return for the different portfolio methods. For smaller portfolio sizes of only 5 or 10 stocks, all the Adaptive and Adaptive+ portfolios performed worse than the Rebalanced portfolios, and Figure 80 shows the reason is that most of those portfolios were held in cash. For portfolios with 30 stocks, the Adaptive portfolios were much better than the Rebalanced portfolios, and the Adaptive+ portfolios which used the actual future 10-day stock-correlations were better still, while the Adaptive+ variants with noisy or malformed correlations were a bit worse than the Adaptive portfolios, albeit still much better than the trivial Rebalanced portfolios. For portfolio sizes of 50 stocks or more, the Adaptive+ variants usually performed significantly better than the Adaptive portfolios. For portfolios with 100 stocks or more, all the noisy Adaptive+ variants performed roughly on par with the Adaptive+ portfolios that used the actual future 10-day correlations; with exception of the variants that used equal or inverted correlations, which performed slightly worse than the Adaptive+ portfolios, but still better than the Adaptive portfolios which did not adjust for correlations.

The mean daily return for the noisy Adaptive+ variants was around 1.0011, which corresponds to an annualized return around 31.6%, calculated as $1.0011^{250} - 1 \approx 31.6\%$ when assuming there are 250 trading-days in a year. Compare this to a mean daily return around 1.0009 for the Adaptive portfolios, which corresponds to an annualized return around 25.2%, so adjusting for stock-correlations resulted in excess annualized returns of more than 6% on average. This is an exceptionally good performance, especially considering that we were using the future 2-3 year average stock-returns with heavy noise, and we were also using various kinds of heavy noisy and malformed stock-correlations.

Figure 78 compares the standard deviation for the different portfolio methods. For smaller portfolios with only 5 or 10 stocks, the standard deviation is much lower for all the Adaptive and Adaptive+ variants, but Figure 80 shows the reason is merely that those portfolios were mostly held in cash. For portfolios with 50 stocks or more, all the Adaptive and Adaptive+ variants had roughly the same standard deviation as the trivial Rebalanced portfolios, and Figure 80 shows that already for portfolios with 70 stocks, the different portfolios were nearly always fully invested in stocks. A few of the

Adaptive+ variants always had lower standard deviation than even the Rebalanced portfolios, including the portfolios that were using the “naive” forecasting of the stock-correlations.

Figure 79 compares the Sharpe Ratios for the different portfolios. Regardless of the portfolio size, the following portfolio types had the highest and therefore best Sharpe Ratios: The Adaptive+ portfolios which used the actual future 10-day stock-correlations, the portfolios that added heavy noise to these correlations, and the “naive” portfolios that used the previous 10-day stock-correlations. For larger portfolio sizes of 150, 200 and especially 300 stocks, the Adaptive+ portfolios with the “naive” correlation forecasts usually performed significantly better than all the others. Even the Adaptive+ portfolios with equal or inverted correlations performed on par with (or perhaps slightly better than) the Adaptive portfolios which did not adjust the portfolio weights with regard to stock-correlations.

Figure 81 compares the percentages of months with losses for the different portfolio methods. For portfolio sizes of 30 stocks or more, the performance patterns are roughly the same, namely that the Rebalanced portfolios had losses in roughly 35% of months, while the Adaptive portfolios had losses in roughly 32% of months so they were slightly better, and the Adaptive+ portfolios using the actual future 10-day stock-correlations had losses in roughly 30% of months, and slightly less for the larger portfolios with 150, 200 or 300 stocks. The Adaptive+ variants with noisy or malformed correlations performed roughly on par with this, with the “naive” forecasts for the stock-correlations having almost exactly the same performance as the Adaptive+ portfolios using the actual future 10-day correlations. The worst Adaptive+ variants were the ones using equal or inverted correlations, but they still just had roughly the same performance as the Adaptive portfolios which did not adjust for correlations at all.

Figure 82 compares the Max Drawdowns of the different portfolios. For smaller portfolios the Adaptive and Adaptive+ variants performed much better than the Rebalanced portfolios, but that was simply because they held a lot of their portfolios in cash. For portfolios with 50 stocks or more, the Adaptive and Adaptive+ variants had Max Drawdowns that were roughly on par with each other, and they were mostly better than for the Rebalanced portfolios.

Figure 83 compares the Max Pullups of the different portfolios, to see how well they recovered from stock-market crashes. For small portfolios with only 5 or 10 stocks, all the Adaptive and Adaptive+ variants performed much worse than the Rebalanced portfolios, but that was probably just because they held a lot of their portfolios in cash. Already for portfolios with 30 stocks, the Adaptive and Adaptive+ variants were usually better than the Rebalanced portfolios. For the larger portfolio sizes with 150, 200 and especially 300 stocks, the Adaptive portfolios were nearly always better than the Rebalanced portfolios, and the Adaptive+ variants were nearly always better than the Adaptive portfolios, perhaps with exception of the Adaptive+ variants using equal or inverted correlations. The other Adaptive+ variants had roughly the same performance with Max Pullups around 200%.

15.3 Summary

This section used heavy noise in the estimates for the future stock-returns, as well as very noisy and malformed stock-correlations. For smaller portfolios the diversification method under-performed because it moved a lot of the portfolios into cash. But for larger portfolios it worked extremely well, which is very impressive considering how noisy the estimated stock-returns and correlations were!

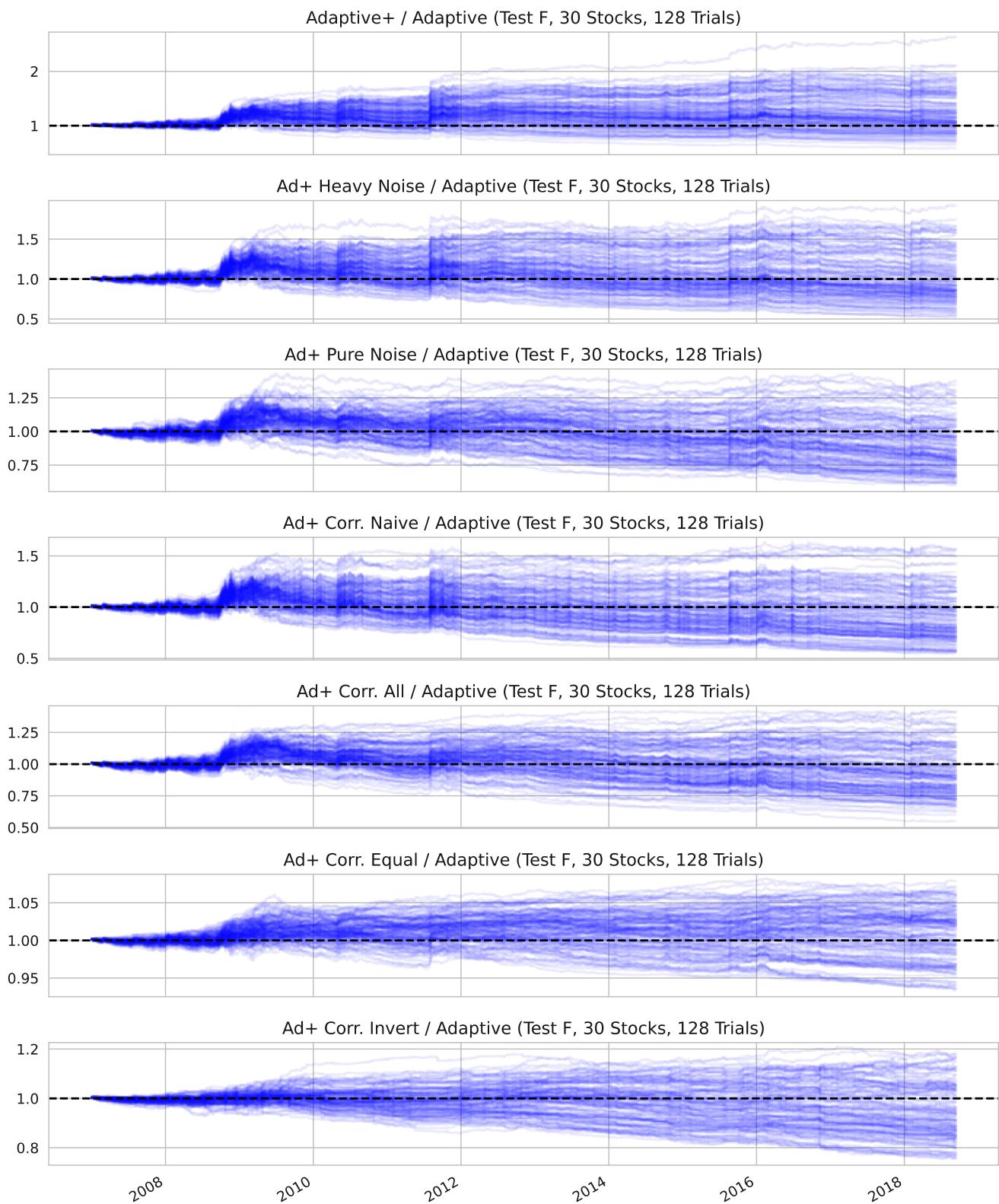


Figure 74: **Test F** – Compare values of 128 random portfolios with **30 stocks** each.

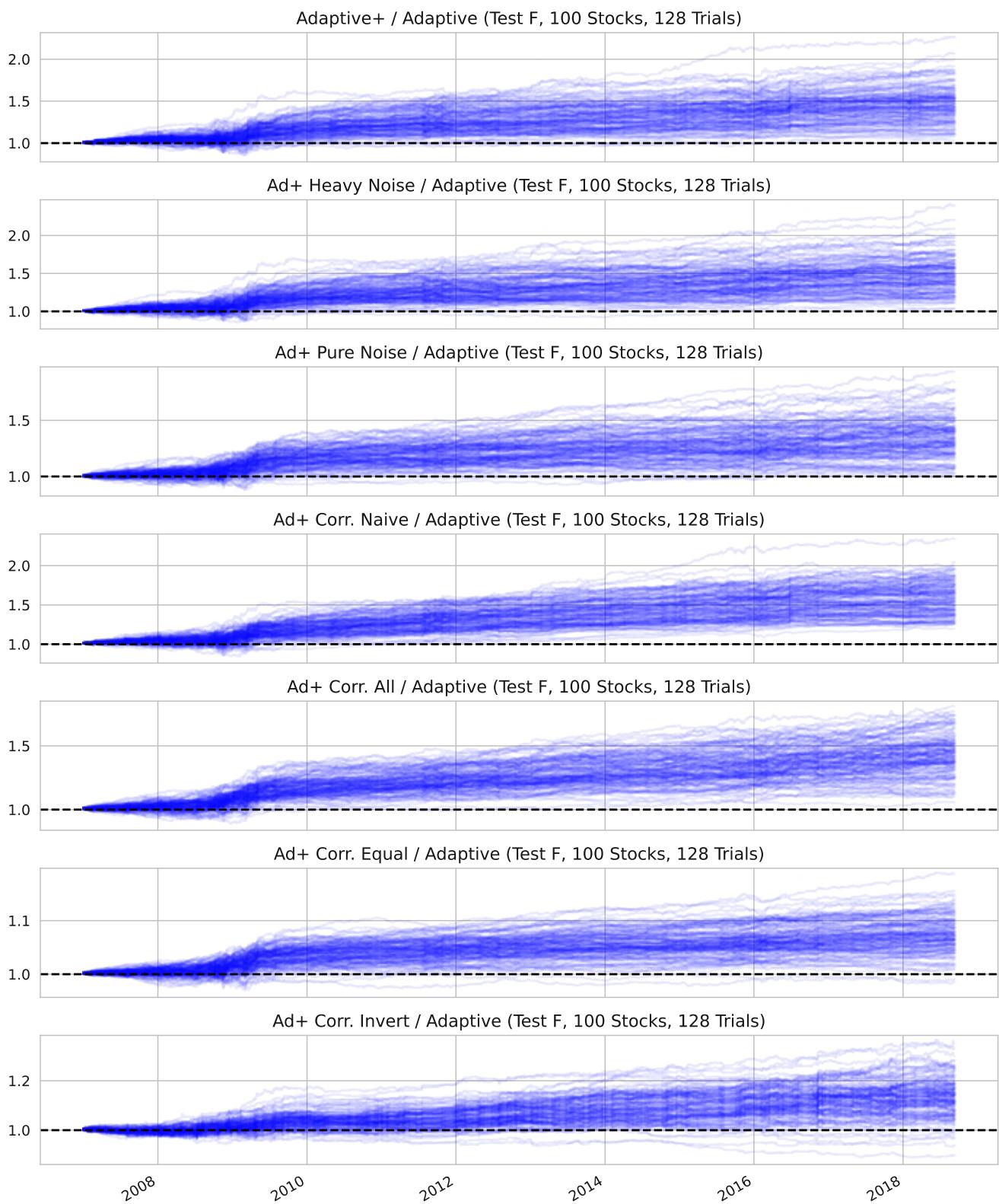


Figure 75: **Test F** – Compare values of 128 random portfolios with **100 stocks** each.

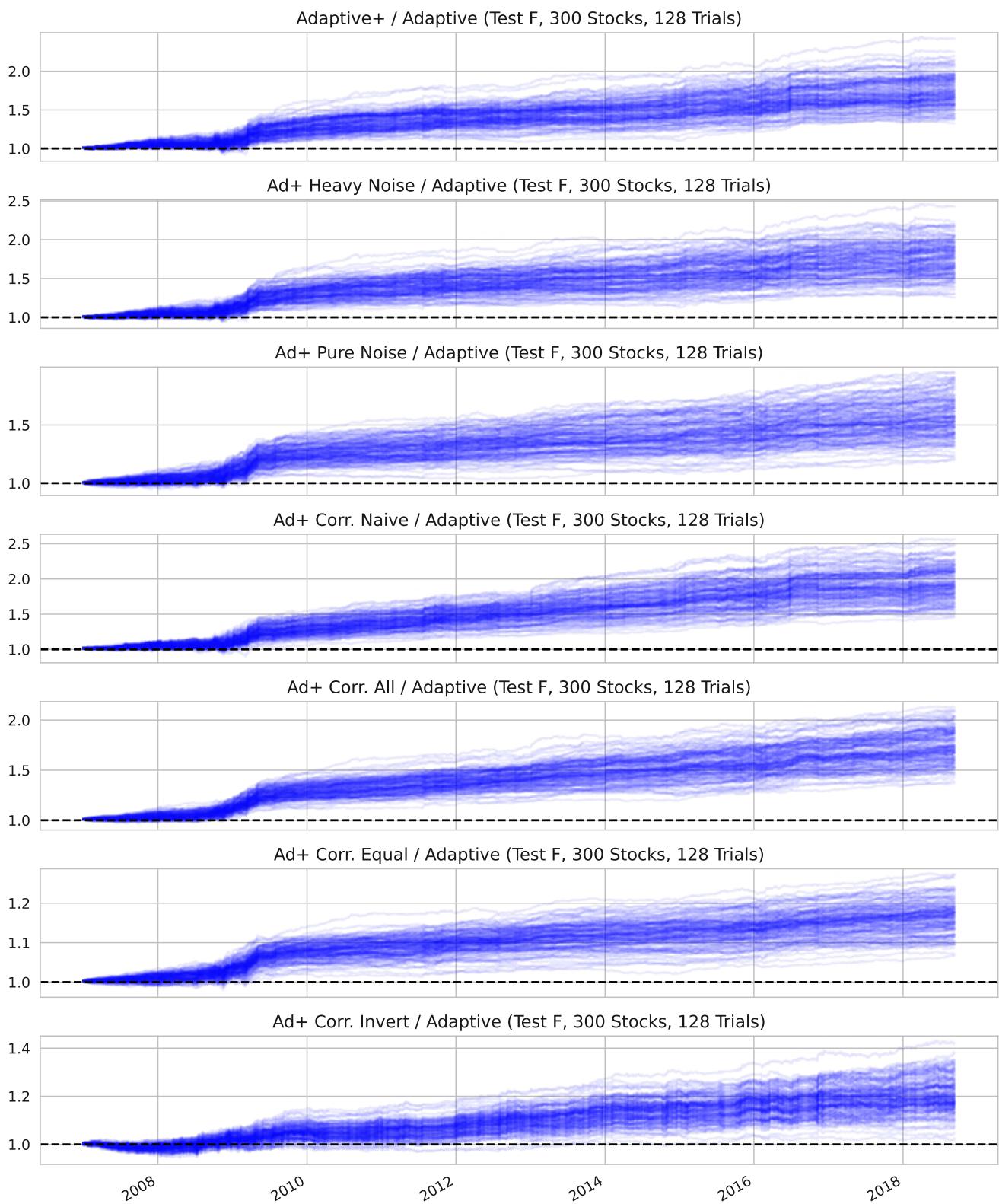


Figure 76: **Test F** – Compare values of 128 random portfolios with **300 stocks** each.

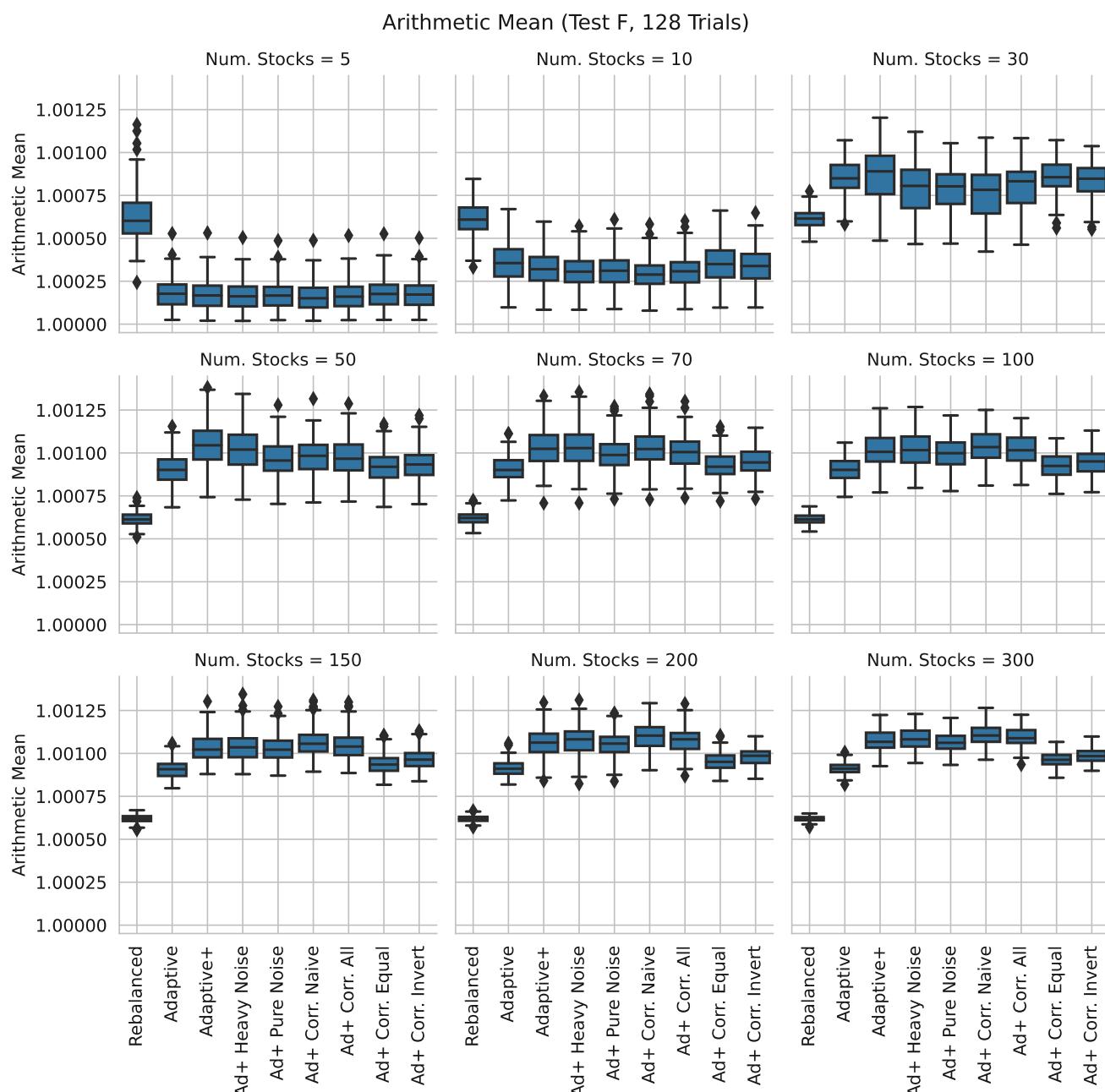


Figure 77: **Test F – Compare Arithmetic Mean** daily return of different portfolio methods and sizes.

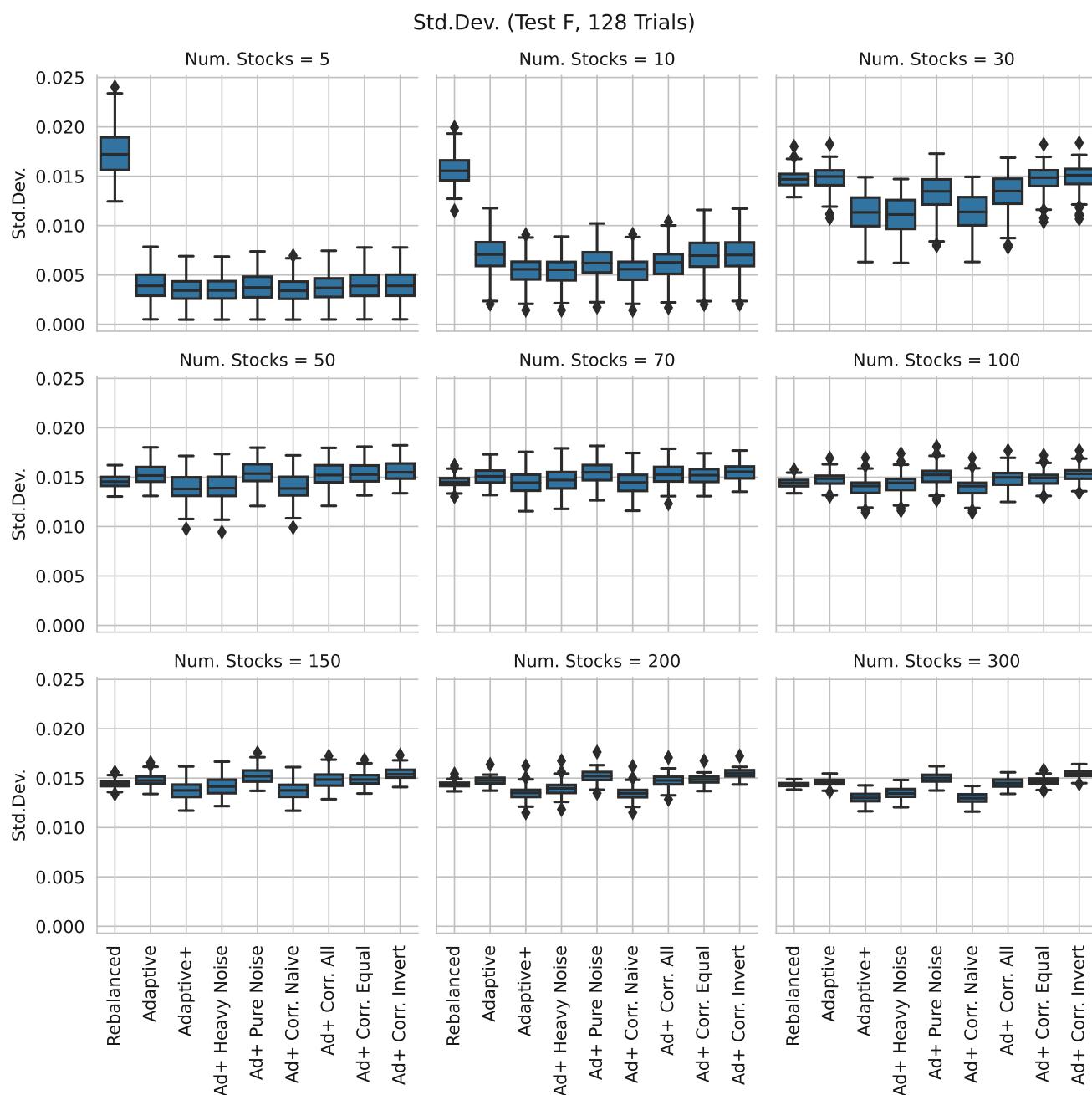


Figure 78: **Test F** – Compare **Std.Dev.** of the daily return of different portfolio methods and sizes.

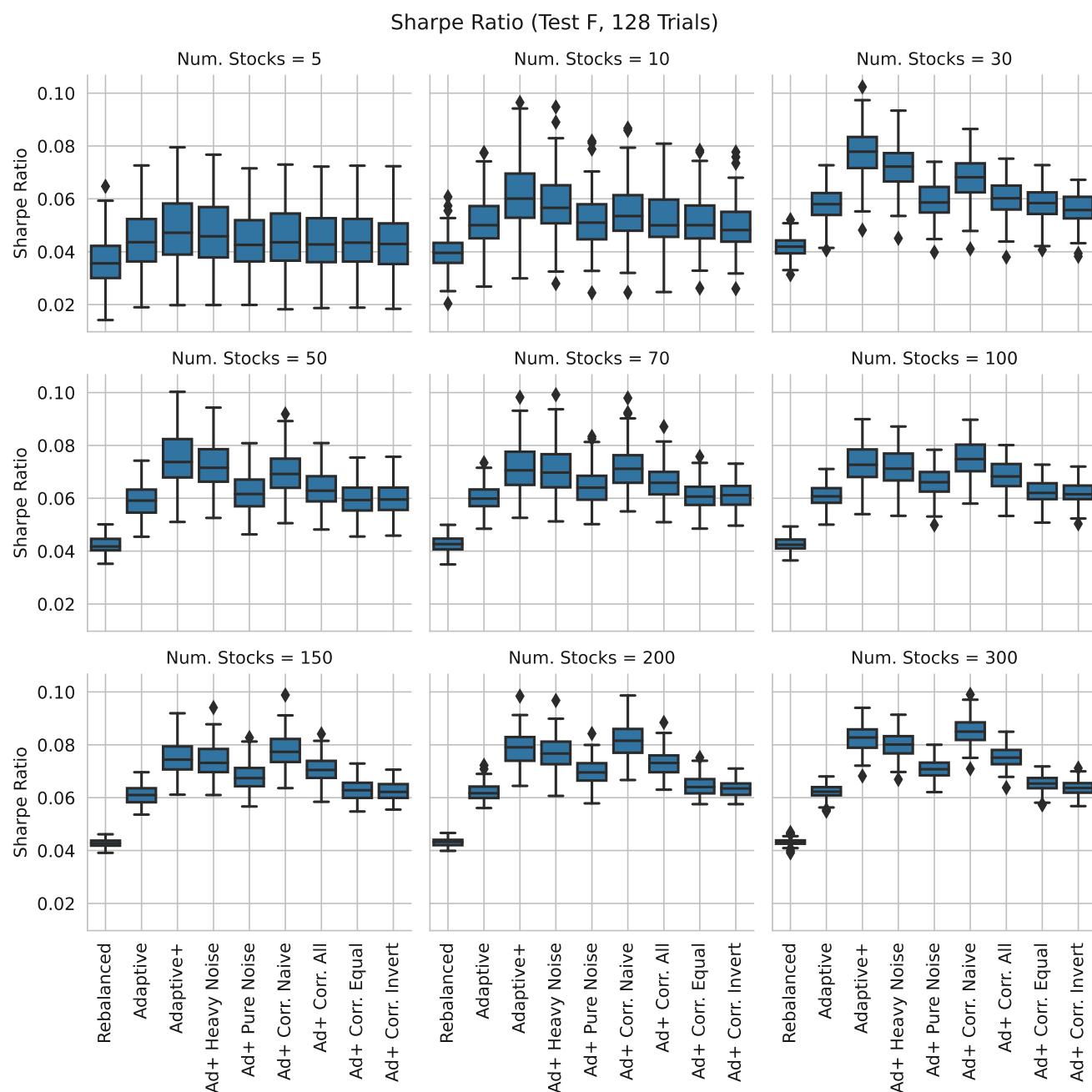


Figure 79: **Test F** – Compare **Sharpe Ratios** for daily returns of different portfolio methods and sizes.

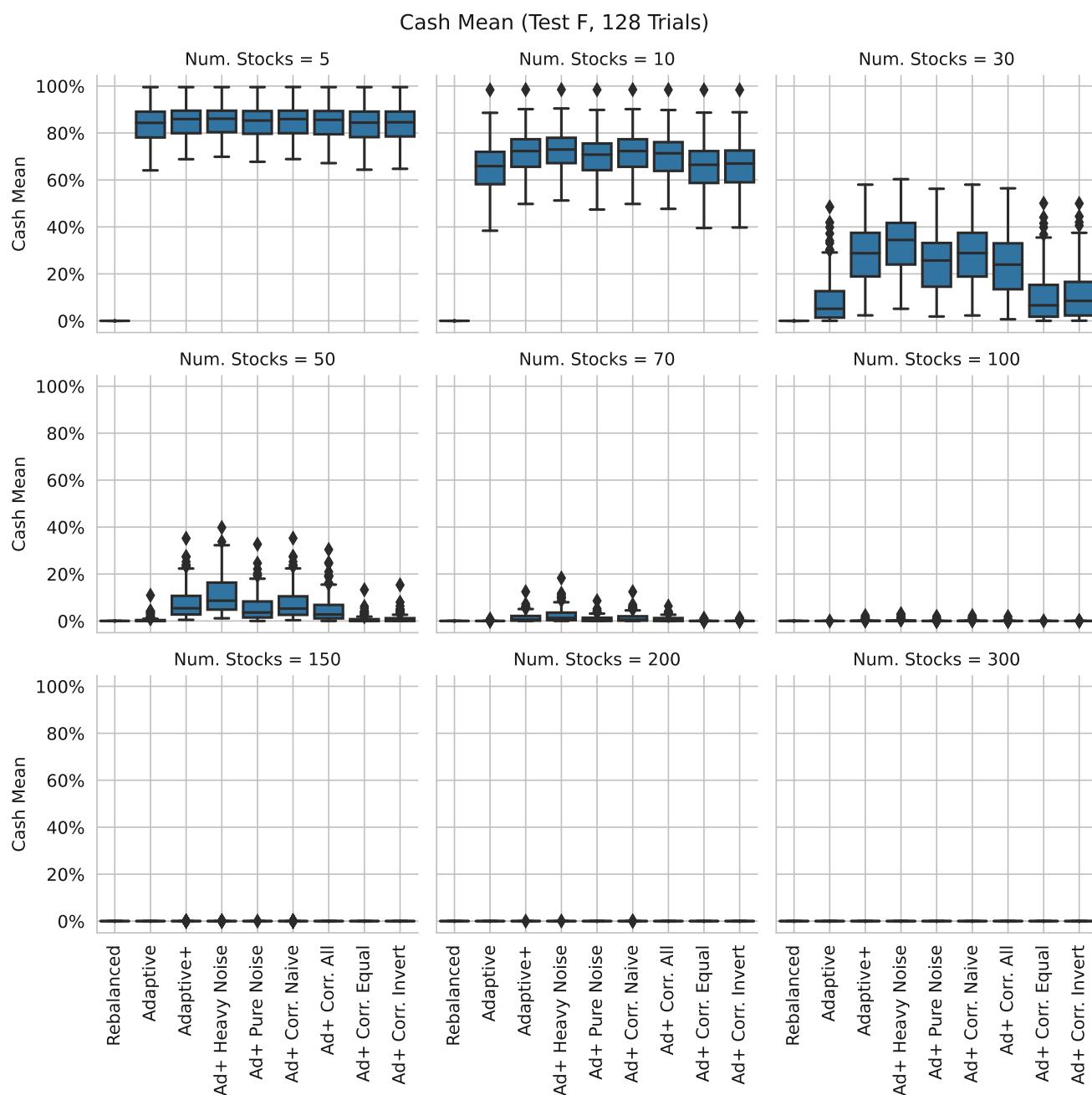


Figure 80: Test F – Compare Cash Mean Daily Position of different portfolio methods and sizes.

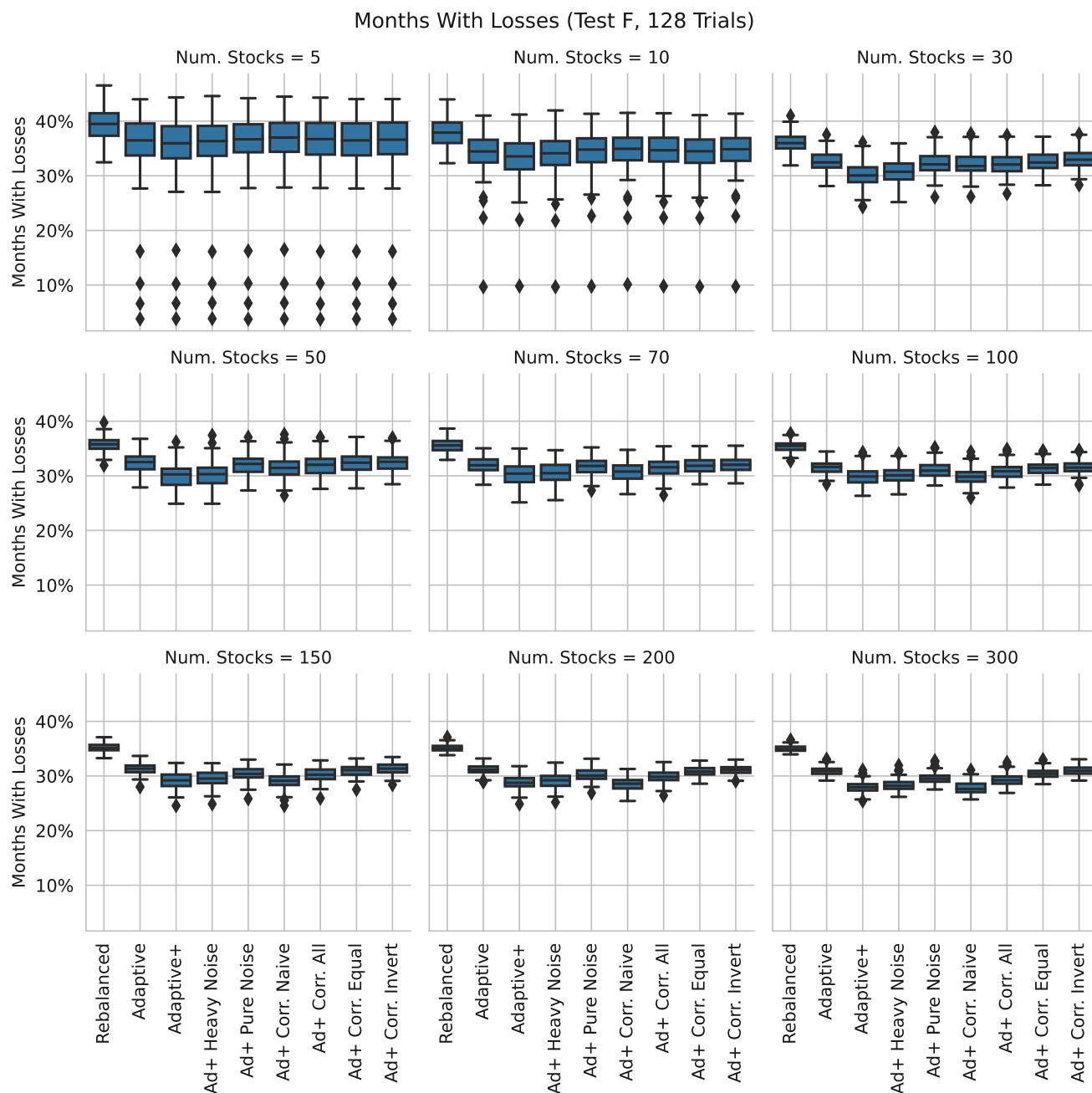


Figure 81: **Test F – Compare Months With Losses** of different portfolio methods and sizes.

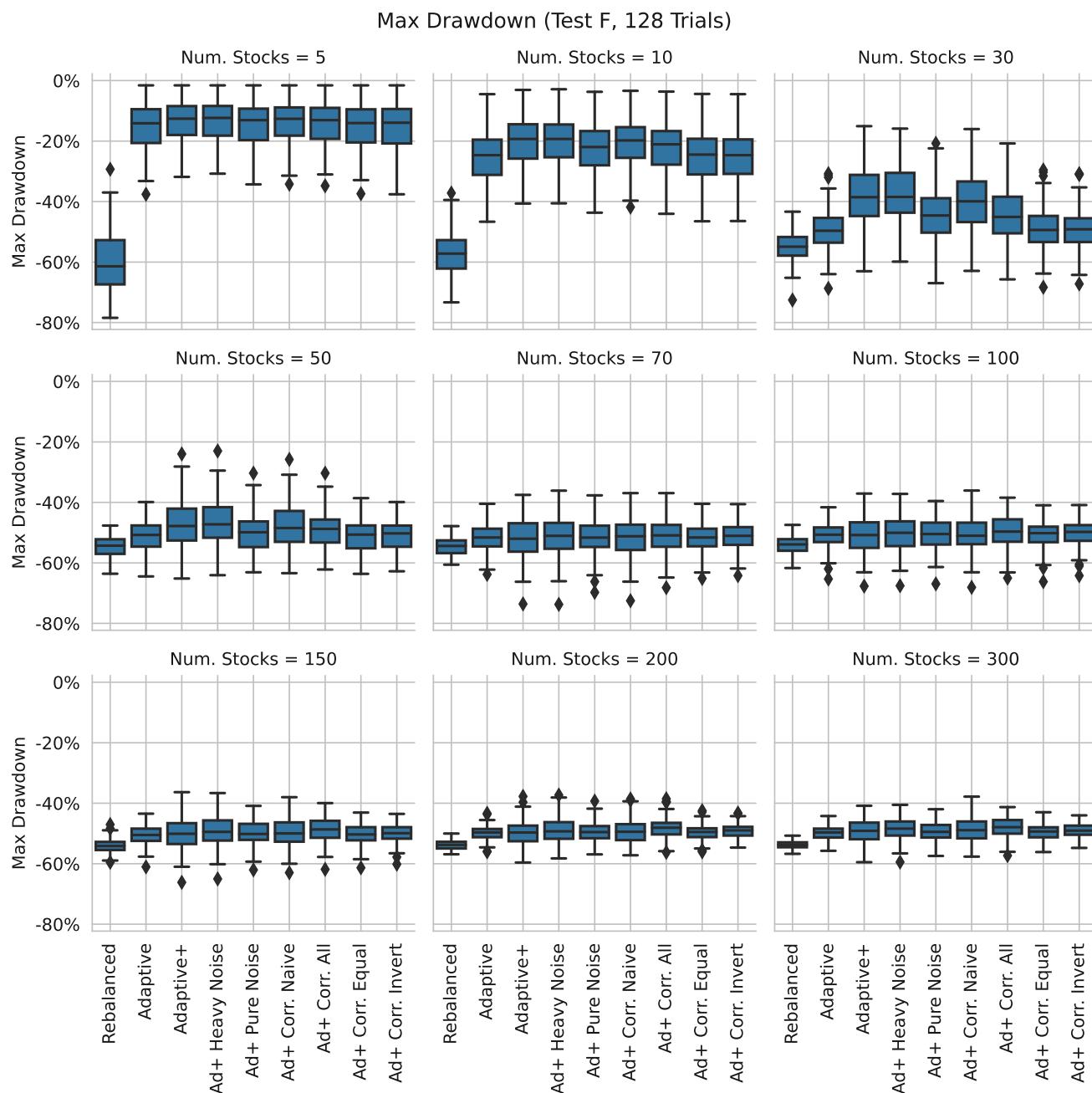


Figure 82: Test F – Compare Max Drawdown of different portfolio methods and sizes.

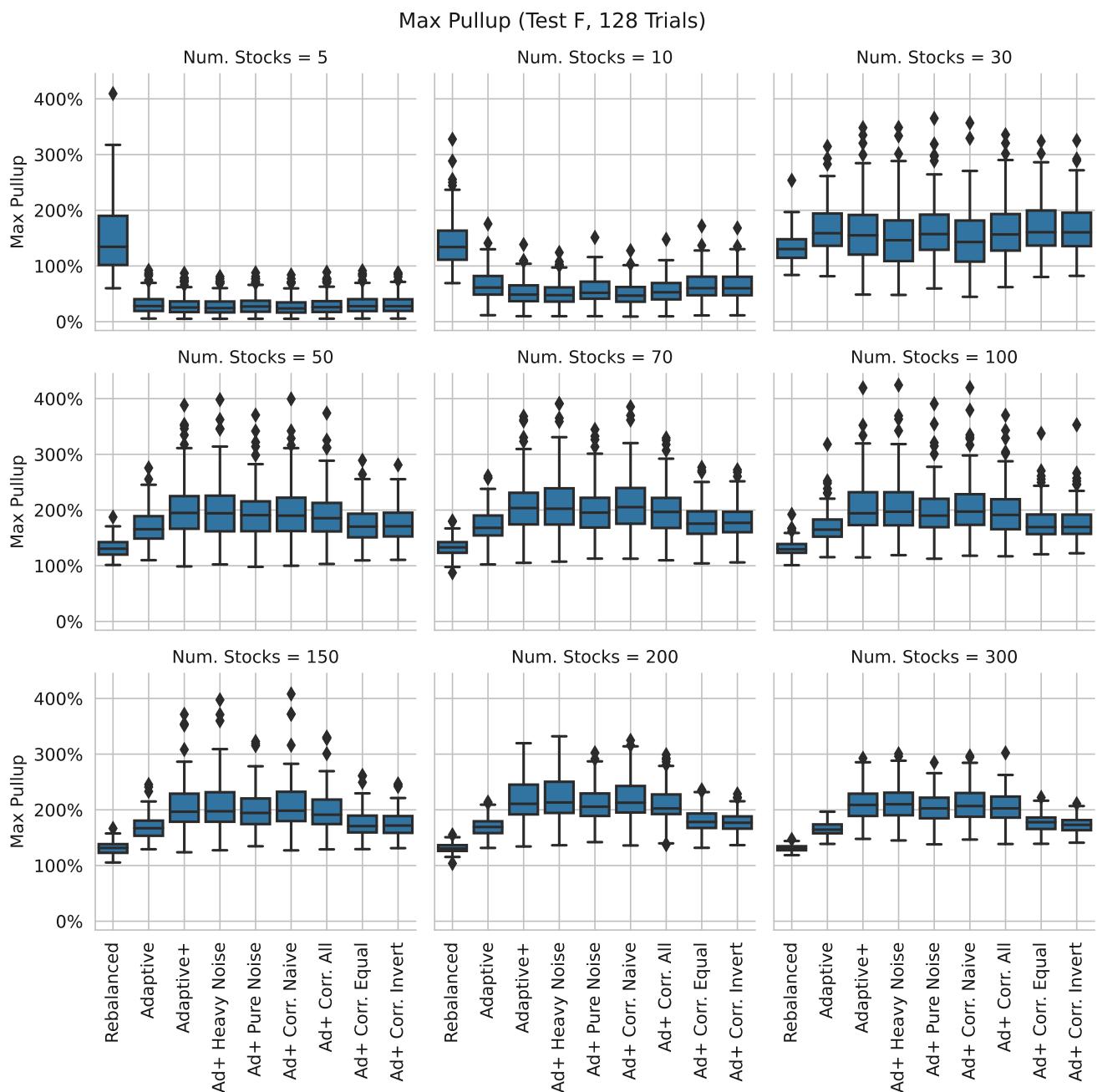


Figure 83: Test F – Compare Max Pullup of different portfolio methods and sizes.

16 Test G – Parameter Tuning (Omniscient)

In the previous tests, the Adaptive and Adaptive+ portfolios had very large cash holdings for smaller portfolio sizes, which caused them to under-perform in some respects. The reason is that the Adaptive filter from Section 7.2 only allows stocks into the portfolio if they have sufficiently high estimated future returns, so if there are only few stocks available for investment, and each portfolio weight is only allowed to be a small fraction of the entire portfolio, then a lot of the portfolio will be held in cash.

16.1 Parameters

The Adaptive filter is just a mathematical formula with a few parameters that determine how the filter behaves for different estimates of the future stock-returns, so the formula makes a portfolio weight larger when the future stock-returns are estimated to be higher, and vice versa.

We can change the parameters for the Adaptive filter and thereby change how it behaves for different estimates of the future stock-returns. There are three such parameters for the Adaptive filter in Eq. (28) and Eq. (29). In all the previous tests, the parameters were set to these values:

$$\mu_{min} = 10\%, \quad \mu_{max} = 30\%, \quad Weight_{max} = 5\% \quad (92)$$

These parameters were chosen according to what I personally thought were reasonable: The minimum future stock-return should be 10% per year, for which the portfolio weight would be zero. The portfolio weight then increases linearly until the future stock-return is estimated to be 30%, for which the portfolio weight is set to 5%, which is the maximum portfolio weight that is allowed. Using these parameters, the Adaptive filter worked extremely well for larger portfolio sizes of 100-300 stocks, but it caused the smaller portfolios to hold too much cash.

16.2 Optimization

We now want to find parameters for the Adaptive filter that are better suited for portfolios of only 30 stocks. The question is how we should find such parameters? In this case we only have 3 parameters that need tuning, but for a more sophisticated filtering process, you could have many more parameters. And the number of possible parameter combinations increases exponentially with each new parameter, which makes it impossible to try all parameter combinations. So we need a clever way of tuning the parameters automatically using a computer.

This is essentially just an optimization problem, where the search-space consists of all valid parameter combinations, and we then want to find the parameters that perform best on some metric, such as the arithmetic mean daily return. But we need to use a special kind of optimization method, which does not require the mathematical gradient of the problem that is being optimized, because the performance metric is calculated from simulation results, when the portfolio method is used on e.g. 128 random portfolios for a given set of parameters with the Adaptive filter.

If there is only a single objective that needs optimization, such as the mean daily return of the portfolio, then we could use one of many so-called Evolutionary Algorithms, which usually work quite well.

16.3 Multi-Objective Optimization

If we want to find the parameters for the Adaptive filter that perform best on two or more performance metrics simultaneously, then we need to use a so-called Multi-Objective optimizer. In this test we will use the well-known multi-objective optimization method called NSGA-2 [Deb 2000], which does not require the mathematical gradient of the problem being optimized, so it is very flexible and allows you to optimize any reasonable combination of performance metrics that you might formulate.

It is important to note that the two performance metrics (or objectives) must be conflicting, otherwise there is no reason to use a multi-objective optimizer such as the NSGA-2, because a single-objective optimizer would probably work much better. For example, if we want to maximize both the arithmetic and geometric mean daily returns by changing the parameters of the Adaptive filter, then we don't need to use a multi-objective optimizer, because those two performance metrics are nearly identical.

In this test we want to optimize the parameters of the Adaptive filter for these two performance metrics:

1. The excess mean daily return of the Adaptive portfolios over the Rebalanced portfolios. This must be maximized. It is averaged over 128 trials of portfolios with 30 random stocks each.
2. The Max Drawdown of the Adaptive portfolios. This must also be maximized so e.g. -20% is better than -50%. This is averaged over the same 128 trials of random portfolios.

Note there is a superficial similarity to the mean-variance portfolio optimization described in Section 2, which maximizes the mean return of a portfolio while simultaneously minimizing its variance, which is also a form of multi-objective optimization. Although we are also doing multi-objective optimization here, it is actually very different from mean-variance portfolio optimization, which is used to find the actual portfolio weights, while we are using the multi-objective optimization to find good parameters of the Adaptive filtering process, which in turn is used to calculate the portfolio weights.

16.4 Computation Flowchart

Figure 84 shows a simplified computation flowchart for the multi-objective optimization of the parameters for the Adaptive portfolio method. The computer code is publicly available, see Section 19.

The overall idea is that the NSGA-2 optimizer first makes some random guesses for the parameters of the Adaptive portfolio method, and then it tries to iteratively improve those parameters, by combining the parameters that performed well on the two performance metrics described above. Each time a new set of parameters must be tested, the algorithm generates 128 portfolios with 30 random stocks in each. It then calculates the performance metrics and reports those back to the NSGA-2 optimizer, which repeats the process until it finds the optimal parameters for the Adaptive portfolio method.

We are using the actual future 2-3 year average stock-returns when tuning the parameters in this test. The parameter tuning was also run with heavy noise in the future stock-returns, but it resulted in the parameters for the Adaptive portfolios being much more conservative and only allowing stocks into the portfolio if their estimated future returns were very high. This is something you could experiment with.

This took about 100 minutes to run on a quad-core computer with a 2.6 GHz CPU (boost 3.5 GHz).

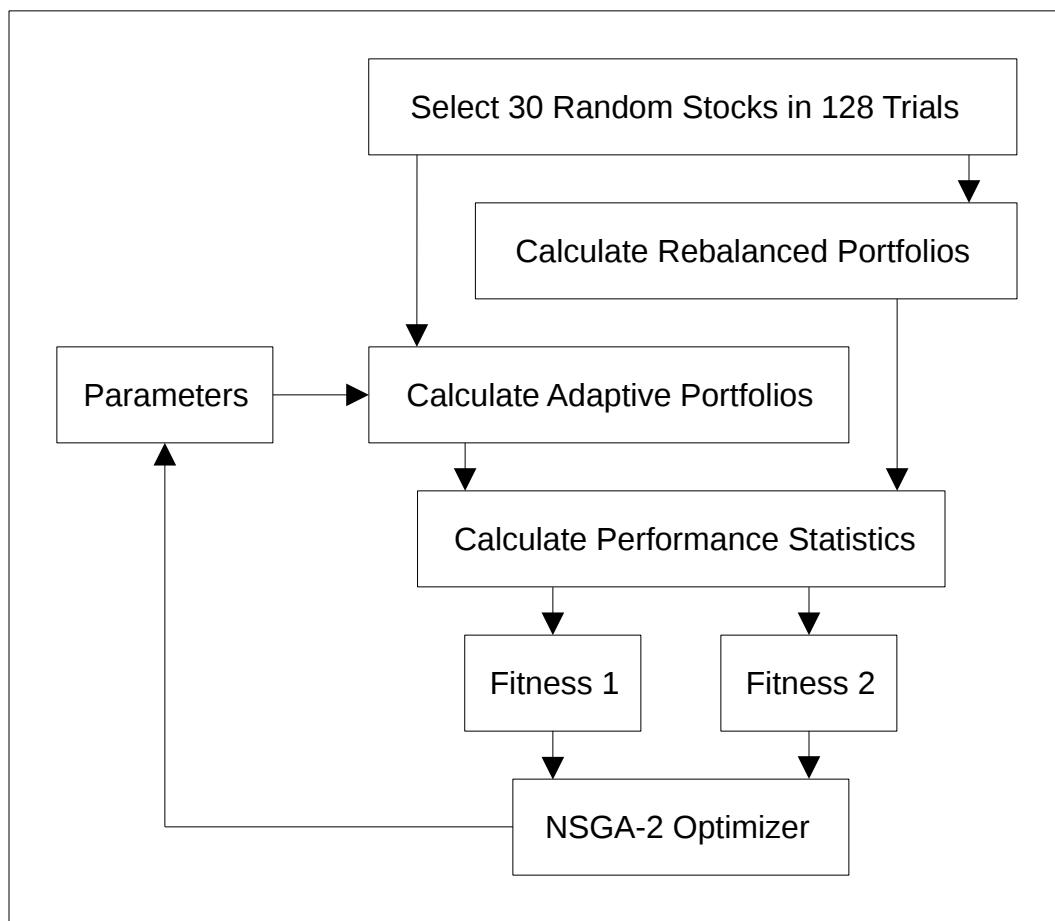


Figure 84: Simplified computation flowchart for Multi-Objective Optimization of the parameters for the Adaptive portfolio method.

16.5 Pareto Front

Figure 85 shows the so-called Pareto Front for parameters of the Adaptive portfolio method. Each blue dot shows the performance of one parameter set that were found by the NSGA-2 optimizer. These parameters were the ones that performed best on both performance metrics simultaneously. So these parameters are the optimal compromises between the two performance metrics. The three red dots show the Adaptive parameters that will be tested in the following sections.

This plot is very similar to the so-called Efficient Frontier in mean-variance portfolio optimization, but that is again a superficial comparison because the portfolio methods work very differently. Here we have merely tuned the parameters of the Adaptive portfolio method to perform well on these two metrics, and because the two metrics are in conflict, they lie on a so-called Pareto or Efficient Frontier.

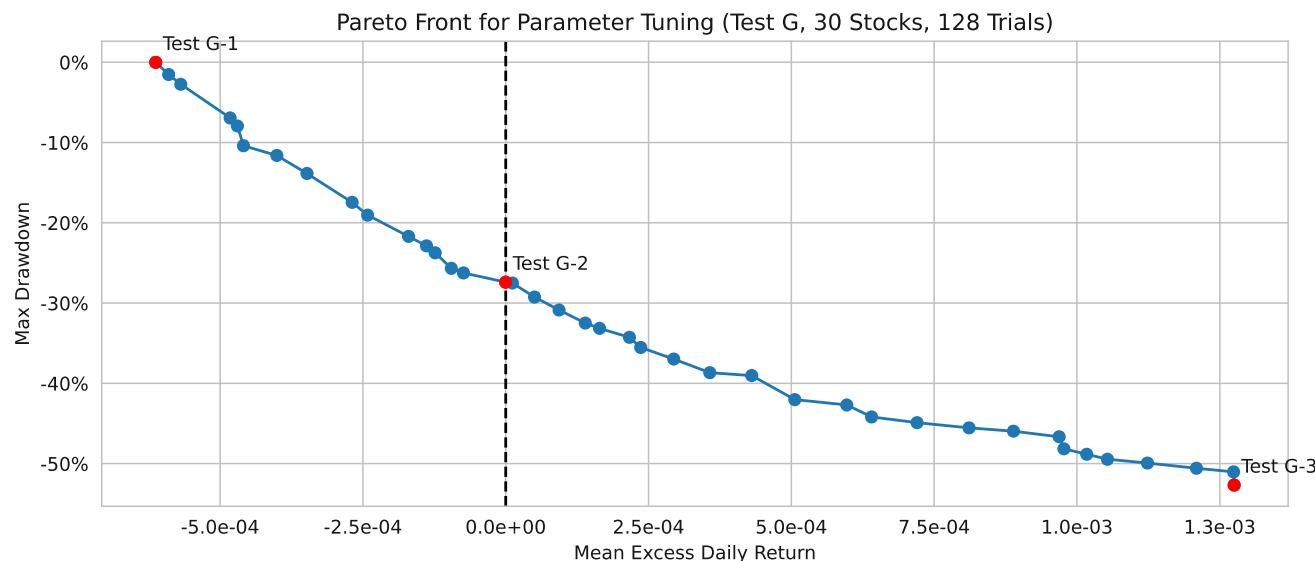


Figure 85: The Pareto Front for parameters of the Adaptive portfolio method.

16.6 Test G-1 – Best Max Drawdown

We now test the following parameters which gave the best Max Drawdown for the Adaptive portfolios:

$$\mu_{min} = 98.9\%, \quad \mu_{max} = 192.2\%, \quad Weight_{max} = 0.27\% \quad (93)$$

Note that the Threshold, Adaptive and Adaptive+ portfolios all use the same parameters in Eq. (93), and the Rebalanced+ portfolios also use the same $Weight_{max}$.

Figure 86 compares the different portfolio methods on 128 trials where the portfolios have 30 random stocks each. You should first focus on the Rebalanced and Adaptive portfolios, which are the portfolios that were used in the parameter tuning. In this case we see that the arithmetic mean daily return was zero for the Adaptive portfolios, because the Adaptive portfolios had 100% in cash, so the Max Drawdown was 0%. This is one extreme of the Pareto-Optimal Frontier: If we want the best Max Drawdown then we should always hold the entire portfolio in cash.

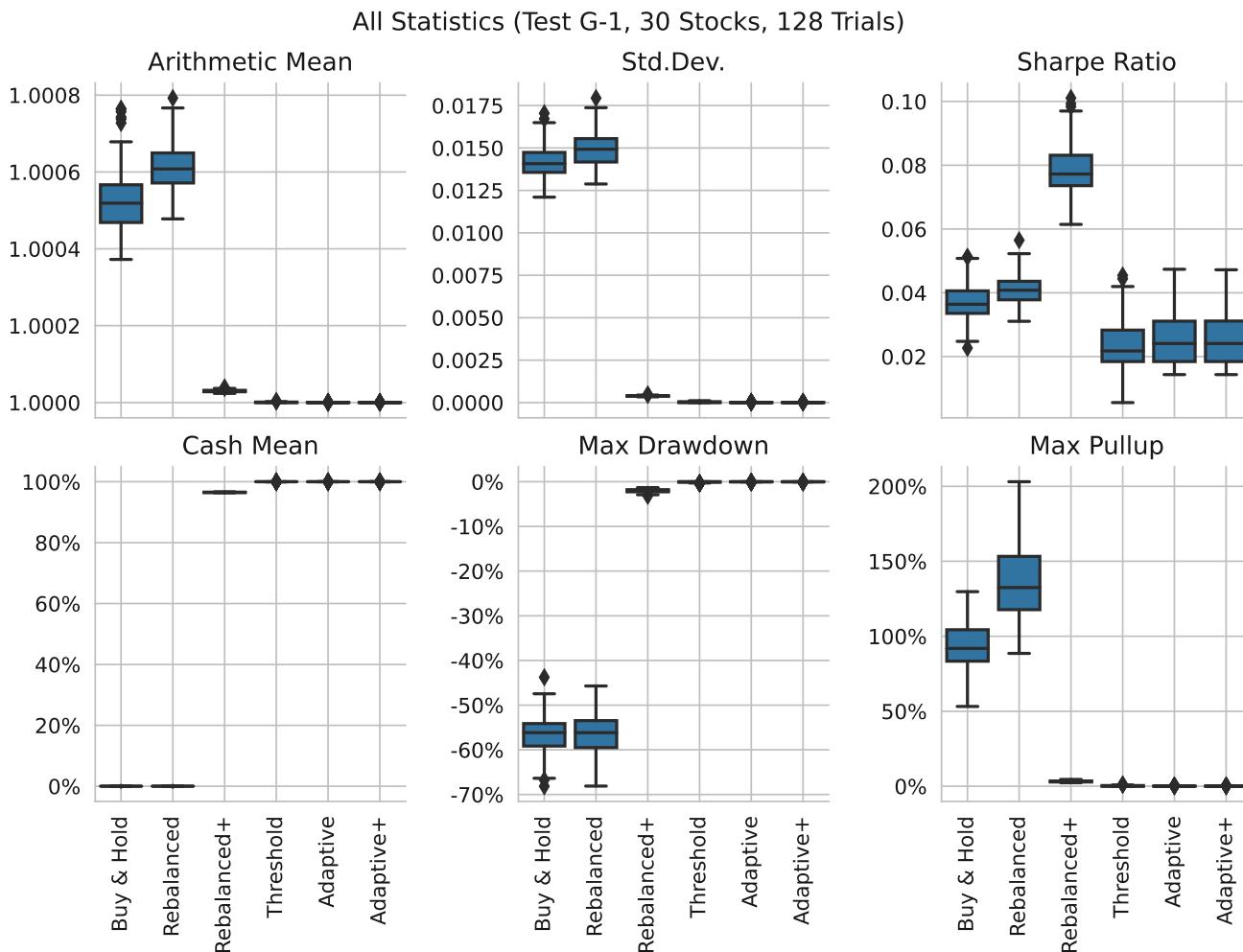


Figure 86: **Test G-1** – Compare performance statistics for the different portfolio methods, where some of them use the tuned parameters from Eq. (93).

16.7 Test G-2 – Compromise

We now test the following parameters, which resulted in the Adaptive portfolios having nearly the same mean daily return as the Rebalanced portfolios:

$$\mu_{min} = 9.9\%, \quad \mu_{max} = 105.2\%, \quad Weight_{max} = 12.1\% \quad (94)$$

So the future stock-returns must be at least 9.9% before the stock is allowed into the portfolio, and the weight increases linearly until the future stock-return is 105.2% where the portfolio weight is 12.1%.

Figure 87 compares the different portfolio methods when some of them are using the parameters from Eq. (94). The arithmetic mean daily return was about the same for the Rebalanced and Adaptive portfolios, but the Max Drawdown was much better for the Adaptive portfolios. So the multi-objective optimizer has found parameters that were a better compromise for those two performance metrics.

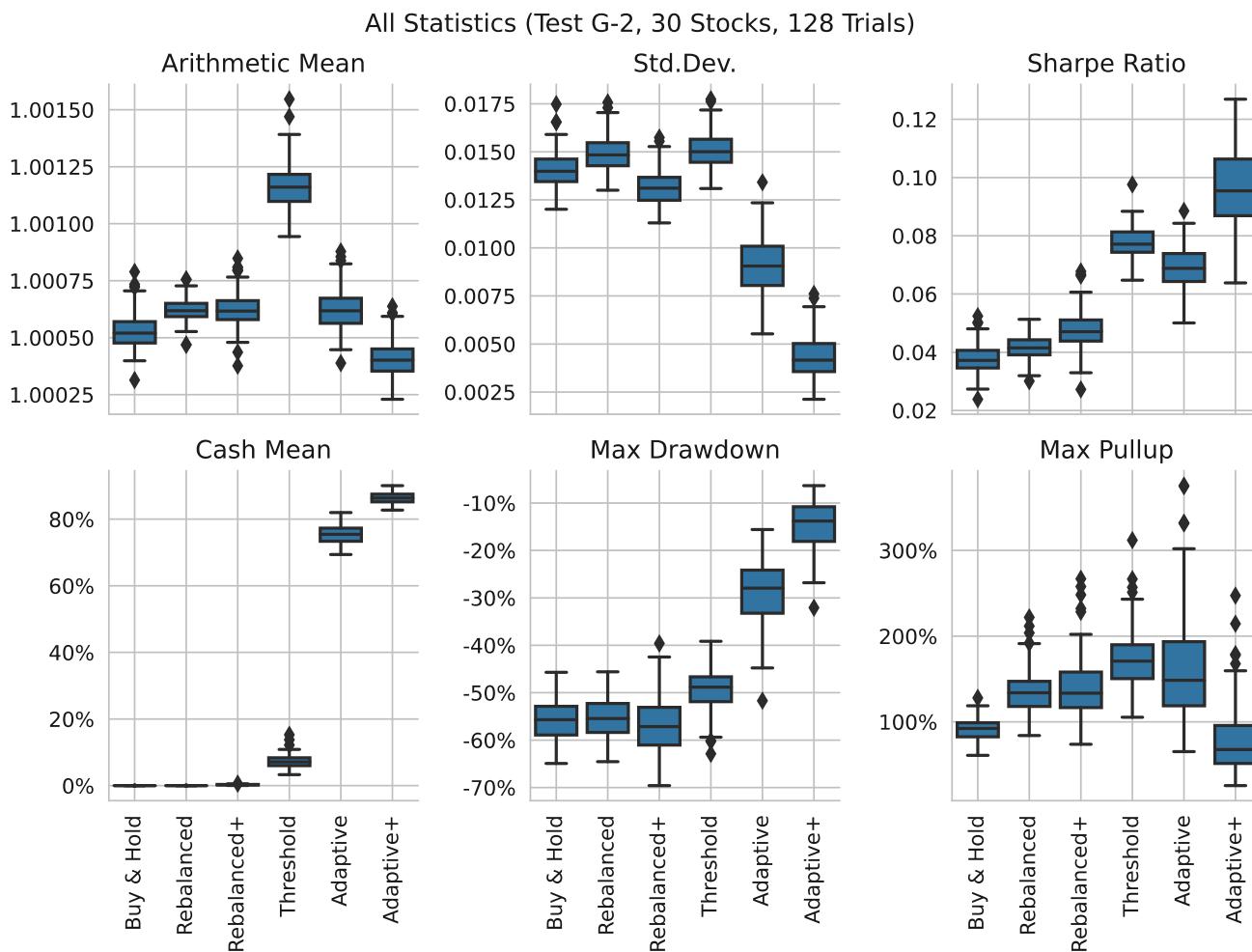


Figure 87: **Test G-2 – Compare performance statistics for the different portfolio methods, where some of them use the tuned parameters from Eq. (94).**

16.8 Test G-3 – Best Mean Excess Return

We now test the following parameters, which had the highest excess mean daily return for the Adaptive portfolios compared to the Rebalanced portfolios:

$$\mu_{min} = 18.8\%, \quad \mu_{max} = 54.2\%, \quad Weight_{max} = 97.0\% \quad (95)$$

So the future stock-returns must be at least 18.8% before the stock is allowed into the portfolio, and the weight increases linearly until the future stock-return is 54.2% where the portfolio weight is 97.0%. This makes it possible for the portfolio to be nearly fully invested in a single stock, but usually it will probably be much less because the portfolio weights are also normalized so they sum to one.

Figure 88 compares the different portfolio methods when using the parameters from Eq. (95). Now the Adaptive portfolios performed much better than the Rebalanced portfolios, and the Threshold and Adaptive+ portfolios also benefited greatly from the tuned parameters. They improved not only on the mean daily return for which they were tuned, but also on most of the other performance metrics.

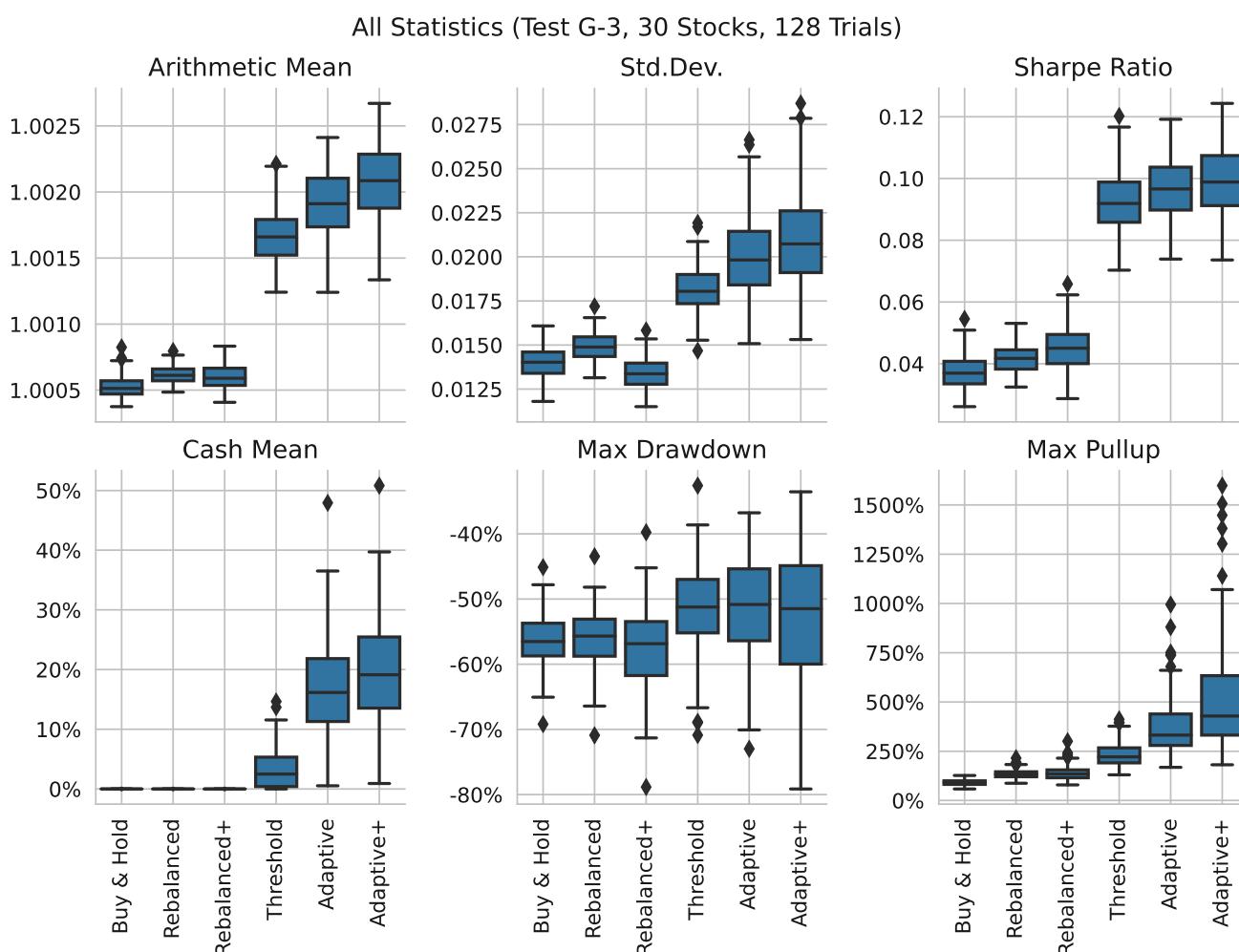


Figure 88: **Test G-3** – Compare performance statistics for the different portfolio methods, where some of them use the tuned parameters from Eq. (95).

17 Future Research

Suggestions for future research have been made throughout this paper and the more important ones are summarized here along with a few more. You are encouraged to do this research and write a paper. Some of these suggestions are fairly easy and would just require minor modifications to the computer code provided in Section 19, while other suggestions might require significant effort and ingenuity.

The suggestions for future research are:

- Further analysis of the diversification method, e.g.: Why is it so robust on noisy and malformed correlation matrices? How does it change the portfolio weights? Are there any weaknesses?
- Find a better or perhaps more general convergence proof for the diversification method. This would be a very big contribution. What are the requirements for the Full Exposure function that makes the algorithm converge, how fast does it converge, and is the solution always unique?
- Try other definitions of the Full Exposure that is used in the diversification algorithm. Can the performance be improved in some way? Or is the Full Exposure already optimal? Why?
- Try other variants of the filtering process. Perhaps an exponentially increasing function is better, because it would give exponentially more weight to assets with higher expected returns? Can other financial data for a company or stock be used successfully in the filtering process?
- Use low-risk bonds or a bond-ETF (Exchange Traded Funds) instead of cash in the portfolios. Does that improve or worsen the portfolio's performance on some metrics?
- The data-set with U.S. stocks has gone through several selection processes where a lot of stocks have been eliminated, including stocks that became worthless. This means there is so-called “survivorship bias” in all the tests in this paper. Try using the new portfolio method with a larger data-set that contains more stocks that became worthless to see how it performs.
- Try other variants of the correlation matrix, e.g.: Which correlation measure works best? How is the correlation matrix best forecast, e.g. what is the optimal number of days X that makes the diversification method work best, when using the previous X days of stock-correlations as a “naive” forecast for the future correlations? Is it useful to combine correlations from historic periods, such as the big stock-market crashes in the years 2009 and 2020?
- A big open problem is of course the forecasting of future stock-returns. We have seen in this paper that if we can make reasonable but also very noisy predictions of the future 2-3 year average stock-returns, then that is sufficient to make good portfolio allocations with high returns. The big question is how we can predict the future 2-3 year stock-returns? A good starting point is my previous paper on Long-Term Stock Forecasting [Pedersen 2020].
- On a theoretical level, what would happen to asset-prices if all market-participants used the same filtering and diversification methods with the same parameters and assumptions?

Godspeed!

18 Conclusion

In this paper we first showed that so-called “mean-variance” portfolio optimization is inherently broken because variance is a horrible risk-measure for investing, and because the portfolio’s mean and variance are being optimized simultaneously, so estimation errors can cause that method to concentrate its portfolio in losing assets that are highly correlated. We also dispelled some other common academic misbeliefs about the randomness and predictability of future stock-returns and correlations.

We then presented our new so-called “filter-diversify” portfolio method which has two separate phases: The filtering process only allows assets into the portfolio if they have sufficiently high estimated returns. The portfolio weights created from the filtering process are then fed into the diversification process, which uses a new algorithm to minimize the correlations between assets in the portfolio.

The new diversification algorithm can be dubbed “Hvass Diversification” for easy reference, and it has several benefits: It is fairly simple. It supports both long and short portfolios. It is very fast as it only takes a few milli-seconds to compute for a portfolio of 1000 assets. It has quadratic time-complexity so it can be used with even larger portfolios. It is guaranteed to converge to the optimal solution in a small number of iterations. And it is extremely robust to estimation errors in the correlation matrix.

Both the filtering and diversification algorithms have been extensively tested using real-world data for nearly 1000 U.S. stocks. It is common in academic research that testing of portfolio optimization is conflated with testing of stock-prediction models, so it is impossible to tell which part was responsible for the poor performance. In this paper we instead made so-called “omniscient” tests which used the actual future 2-3 year average stock-returns in the filtering process, and then we used the actual future 10-day stock-correlations in the diversification algorithm. This showed that our new portfolio method worked extremely well when it was given correct predictions about the future stock-returns and correlations. We then tested our new portfolio method for robustness by adding heavy noise to the future stock-returns and correlations. We even tested the diversification method with correlations that were completely malformed, and the diversification method still performed very well.

The reason that our new diversification algorithm performed so well in the presence of heavy noise or completely malformed correlation matrices, is probably that the diversification algorithm only allows the portfolio weights to decrease, so the worst that can happen is that it moves too much of the portfolio into cash. And even with very noisy correlations, the diversification algorithm was still able to improve on some performance metrics, probably because stock-correlations vary greatly over time, so even though the correlation matrix is very noisy, sometimes it is approximately correct; much like the proverbial broken clock that is still correct twice a day. But more research would be needed to understand exactly why the diversification method works so well under heavy noise. It is generally recommended that you extensively test the new portfolio method before using it in your own investing.

The diversification method does not have any user-adjustable parameters, but the filtering process does. For most of the tests we saw that our new portfolio method placed too much of the portfolio into cash when the portfolio size was small. This was because of the particular parameters used in the filtering process. We then showed how to use a so-called “Multi-Objective” optimizer to tune the parameters, so our new portfolio method could also be used effectively for smaller portfolios with only 30 stocks.

19 Data & Computer Code

This section briefly describes the computer code and lists all the stock-tickers used in the data-sets. The data-processing is described in Section 9.1.

19.1 Computer Code

The following computer code is freely available and is written in the Python programming language:

- The [Python Notebook](#) contains the computer code used to run all the tests and generate all the plots and statistics in this paper. It will automatically download all the financial data when run.
- [FinanceOps](#) is the GitHub repository that contains all of my newest research in finance, including the Python Notebook for this paper. It is generally recommended that you download the entire repository if you want to run a Python Notebook, because some of them may require data-files that are included with the repository. You can also run the Python Notebooks entirely in the cloud using the free Google Colab service. Instructions are found in the link above.
- [diversify.py](#) is the Python module / file that contains the computer code for the diversification algorithm from Section 8.8. This is the research-version which contains more features than the implementation found in the InvestOps package below.
- [InvestOps](#) is a Python package that you can easily import in your own Python program to use the diversification algorithm from this paper. Instructions are found in the link above.

19.2 Stock-Tickers

The following are all 949 stock-tickers from USA whose daily stock-prices were used in these tests.

A, AAL, AAP, AAPL, AAWW, ABC, ABG, ABMD, ABT, ACC, ACHC, ACM, ACN, ADBE, ADI, ADM, ADP, ADS, ADSK, ADTN, ADXS, AEE, AEO, AEP, AES, AET, AFG, AFL, AGCO, AGN, AGNC, AGO, AHL, AIG, AIZ, AJG, AKAM, AKS, ALB, ALE, ALGN, ALGT, ALK, ALKS, ALL, ALNY, ALR, ALV, ALXN, AMAT, AMD, AME, AMED, AMG, AMGN, AMKR, AMP, AMT, AMTD, AMZN, AN, ANDV, ANF, ANSS, ANTM, AON, AOS, APA, APC, APD, APH, ARE, ARG, ARNA, ARO, ARRS, ARW, ASB, ASH, ATI, ATO, ATR, ATVI, ATW, AVB, AVGO, AVNT, AVP, AVT, AVY, AWI, AWK, AXON, AXP, AXS, AYI, AZO, AZPN, BA, BAC, BAX, BB, BBBY, BBWI, BBY, BC, BCO, BCR, BDC, BDN, BDX, BEBE, BEN, BG, BGS, BHC, BHI, BID, BIG, BIIB, BIO, BJRI, BK, BKD, BKE, BKH, BKNG, BKS, BLK, BLL, BMRN, BMS, BMY, BOH, BPL, BPOP, BR, BRCD, BRK-A, BRO, BRS, BSX, BWA, BX, BXP, BYD, BZH, C, CA, CACI, CAG, CAH, CAKE, CAR, CASY, CAT, CAVM, CB, CBB, CBI, CBRE, CBRL, CBSH, CBT, CCE, CCI, CCK, CCL, CCOI, CDE, CDNS, CDR, CE, CELG, CEQP, CERN, CF, CFR, CFX, CGRN, CHD, CHE, CHH, CHRW, CI, CIEN, CINF, CL, CLC, CLF, CLGX, CLH, CLI, CLR, CLX, CMA, CMC, CMCSA, CME, CMG, CMI, CMP, CMPR, CMS, CNC, CNK, CNO, CNP, CNVR, CNX, COF, COG, COL, COLM, COO, COP, COST, CP, CPB, CPE, CPN, CPT, CR, CREE, CRI, CRL, CRM, CROX, CRR, CRS, CRUS, CRZO, CSC, CSCO, CSL, CSX, CTAS, CTSH, CTXS, CUBE, CUZ, CVA, CVLT, CVS, CVX, CW, CXO, CXW, CY, D, DAL, DAR, DBD, DBI, DCI, DD, DDS, DE, DECK, DEI, DFODQ, DFS, DFT, DG, DGX, DHI, DHR, DIS, DISCA, DISH, DK, DKS, DLB, DLR, DLTR, DLX, DOV, DPZ, DRE, DRI, DRQ, DTE, DUK, DVA, DVN, DXCM, DY, EA, EAT, EBAY, ECA, ECL, ED, EEFT, EEP, EFX, EGP, EIX, EL, EMC, EME, EMN, EMR, ENDP, ENR, ENS, EOG, EPAC, EPD,

EPR, EQIX, EQR, EQT, ES, ESL, ESRX, ESS, ETFC, ETN, ETR, EV, EVR, EW, EWBC, EXAS, EXC, EXEL, EXP, EXPD, EXPE, EXR, F, FAST, FCN, FCX, FDS, FDX, FE, FFIV, FHI, FHN, FICO, FIS, FISV, FITB, FL, FLEX, FLIR, FLO, FLR, FLS, FMC, FNB, FNSR, FOSL, FR, FRT, FRX, FSLR, FTI, FTNT, FUL, FULT, G, GBX, GCO, GD, GE, GEO, GHC, GILD, GIS, GL, GLW, GNTX, GNW, GPC, GPN, GPS, GRA, GRMN, GS, GT, GWR, GWW, GXP, H, HAIN, HAL, HAS, HBAN, HBI, HD, HE, HES, HFC, HIBB, HIW, HL, HLF, HMC, HOG, HOLX, HON, HP, HPQ, HR, HRB, HRC, HRL, HSIC, HST, HSY, HUM, HUN, HWC, HWM, HXL, IAC, IBKC, IBKR, IBM, ICE, IDA, IDCC, IDXX, IEX, IFF, IGT, ILMN, INCY, INFN, INGR, INT, INTC, INTU, IO, IONS, IP, IPGP, IPI, IPXL, IRBT, IRET, IRM, ISIL, ISRG, IT, ITRI, ITT, ITW, IVR, IVZ, J, JACK, JAKK, JBHT, JBL, JBLU, JCI, JCOM, JEF, JKHY, JLL, JNJ, JNPR, JOY, JPM, JWN, K, KATE, KBH, KBR, KDP, KEX, KEY, KIM, KLAC, KMB, KMT, KMX, KO, KR, KRC, KSS, KSU, L, LAMR, LAZ, LBRYA, LDOS, LEA, LECO, LEG, LEN, LH, LHX, LII, LKQ, LL, LLL, LLY, LMT, LNC, LNG, LNT, LOGI, LOGM, LOPE, LOW, LPX, LRCX, LSTR, LULU, LUMN, LUV, LVLT, LVS, LXP, LYV, M, MA, MAA, MAC, MAN, MANH, MAR, MAS, MASI, MAT, MCD, MCHP, MCK, MCO, MD, MDLZ, MDP, MDR, MDRX, MDSO, MDT, MDU, MELI, MET, MGLN, MGM, MHK, MIC, MIDD, MJN, MKC, MKL, MKSI, MKTX, MLHR, MLM, MMC, MMM, MMS, MNKD, MNRO, MNST, MO, MOH, MOS, MPW, MPWR, MRK, MRO, MRVL, MS, MSCC, MSCI, MSFT, MSI, MSM, MTB, MTD, MTG, MTN, MTOR, MTZ, MU, MUR, MWW, MXIM, MYGN, MYL, NATI, NAV, NBIX, NBL, NCR, NDAQ, NDSN, NEE, NEM, NEU, NFG, NFLX, NFX, NI, NKE, NKTR, NLOK, NLY, NNN, NOC, NOK, NOV, NRG, NS, NSC, NTAP, NTGR, NTRS, NUAN, NUE, NUVA, NVAX, NVDA, NVR, NWE, NWL, NXST, NYT, O, OA, OC, OCN, ODFL, ODP, OFC, OGE, OHI, OI, OII, OKE, OLED, OLN, OMC, OMI, ON, OPI, ORCL, ORI, ORLY, OSK, OVV, OXY, PAA, PACW, PAYX, PBCT, PBI, PCAR, PCG, PCH, PDCO, PDLI, PEAK, PEG, PENN, PEP, PETM, PFE, PFG, PG, PGR, PH, PHH, PHM, PII, PKG, PKI, PLCE, PLD, PM, PNC, PNM, PNR, PNRA, PNW, PODD, POM, POOL, PPC, PPG, PPL, PRGO, PRU, PRXL, PSA, PTC, PTEN, PVH, PWR, PX, PXD, PZZA, QCOM, QRTEA, R, RAD, RAI, RAX, RBC, RCII, RCL, RDC, RDN, RE, REGN, REN, RES, RF, RGA, RGLD, RHI, RHT, RIG, RJF, RL, RMD, RNR, ROK, ROP, ROST, RPM, RRC, RS, RSG, RTN, RTX, RYN, SAFM, SAM, SANM, SBAC, SBGI, SBH, SBUX, SCG, SCHW, SCI, SEE, SEIC, SF, SFLY, SGEN, SGMS, SGY, SHO, SHW, SIG, SIRI, SITC, SIVB, SJM, SKT, SKX, SLAB, SLB, SLG, SLM, SM, SMG, SMT, SNA, SNBR, SNDK, SNH, SNI, SNPS, SO, SOHU, SON, SONC, SPB, SPG, SPGI, SPLS, SPR, SPWR, SPXC, SPY, SRCL, SRE, SSYS, STE, STI, STJ, STLD, STRA, STT, STX, STZ, SUI, SVU, SWK, SWKS, SWN, SWX, SWY, SXT, SYK, SYNA, SYY, T, TAP, TCBI, TCO, TDC, TDG, TDW, TDY, TECD, TECH, TEL, TEN, TER, TEVA, TFC, TFX, TGI, TGNA, TGT, THC, THO, THS, TIF, TIVO, TJX, TKR, TLRD, TM, TMO, TMUS, TNL, TOL, TPR, TPX, TRN, TROW, TRUE, TRV, TSCO, TSM, TSN, TSS, TT, TTEK, TTWO, TUP, TWX, TXN, TXRH, TXT, TYL, UAL, UDR, UFS, UGI, UHS, UIS, ULTA, ULTI, UMPQ, UNFI, UNH, UNM, UNP, UPS, URBN, URI, USB, USG, UTHR, V, VAR, VFC, VIA, VIAC, VLO, VMC, VMI, VMW, VNO, VR, VRSK, VRSN, VRTX, VSAT, VSH, VTR, VZ, WAB, WAT, WBA, WBC, WBMD, WCC, WCG, WDC, WEC, WELL, WEN, WERN, WEX, WFC, WFM, WFT, WGL, WHR, WLK, WM, WMB, WMT, WOR, WPC, WR, WRB, WRE, WRI, WSM, WSO, WST, WTRG, WU, WWD, WWW, WY, WYNN, X, XCO, XEC, XEL, XLNX, XOM, XPO, XRAY, XRX, Y, YHOO, YUM, ZBH, ZBRA, ZION

20 Bibliography

- [Deb 2000] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan, “*A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II*”, Parallel Problem Solving from Nature VI, 2000.
- [Luenberger 1998] D.G. Luenberger, “*Investment Science*”, 1998.
- [Markowitz 1952] H. Markowitz, “*Portfolio Selection*”, Journal of Finance, 1952.
- [Markowitz 1959] H. Markowitz, “*Portfolio Selection: Efficient Diversification of Investments*”, 1959.
- [Pedersen 2014] M.E.H. Pedersen, “*Portfolio Optimization & Monte Carlo Simulation*”, 2014. [\[PDF\]](#)
- [Pedersen 2020] M.E.H. Pedersen, “*Long-Term Stock Forecasting*”, 2020. [\[PDF\]](#)
- [Pedersen 2021] M.E.H. Pedersen, “*Does Volatility Harvesting Really Work?*”, 2021. [\[PDF\]](#)
- [Thorp 1975] E.O. Thorp, “*Portfolio Choice and the Kelly Criterion*”, Stochastic Optimization Models in Finance, 1975.

My previous papers and books can all be downloaded through [SSRN](#) and [GitHub](#).