**ISE347 Final Project Report: Passive Portfolio Management**

**Intro**

Passive portfolio management offers a compelling alternative to traditional forecasting-based investment strategies, leveraging diversification to maintain market performance while minimizing complexity. In this study, we examined the weekly returns of the S&P100 index over the past five years, using this data to construct a smaller index fund with varying asset counts (q = 10, 25, 50, or 75). By employing an integer programming model in AMPL, we optimized portfolio rebalancing to create a reduced-asset index fund that closely tracks the returns of the full S&P100 without investing in all 100 assets.

The analysis revealed that increasing the number of selected assets enhances portfolio similarity to the S&P100. However, as q grows, the model begins to overfit the in-sample data, leading to diminished performance when applied to out-of-sample datasets. Through rigorous evaluation, we determined that selecting between 40 and 50 assets provides an effective balance between minimizing tracking error and avoiding overfitting, yielding a robust passive management strategy that maintains market representativeness while reducing portfolio complexity.

**Data Collection**

To construct a robust index fund with fewer assets while maintaining performance similar to the S&P100, we gathered historical weekly return data spanning five years (262 weeks) from Yahoo Finance. This dataset includes returns for each of the S&P100 assets, with notable companies such as Apple, Microsoft, Amazon, and NVIDIA. The timeframe ranges from May 2019 to May 2025, ensuring a sufficient sample size for evaluating portfolio performance under varying market conditions. The asset returns were directly sourced without additional cleaning, as missing values were not present in the dataset.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **t** | **AAPL** | **ABBV** | **ACN** | **ADBE** | **ADI** |
| 0 | 1 | 2 | 3 | 4 | 5 |
| 1 | -0.03779 | 0.025952 | 0.023121 | 0.00492 | -0.08834 |
| 2 | -0.05307 | 0.007551 | 0.002692 | -0.01815 | -0.0239 |
| 3 | -0.02179 | -0.04184 | -0.00414 | -0.01408 | -0.01829 |
| 4 | 0.086137 | 0.009386 | 0.027236 | 0.0268 | 0.047891 |
| 5 | 0.013621 | 0.016273 | 0.011207 | -0.01395 | 0.033072 |
| 6 | 0.031338 | 0.001144 | 0.003189 | 0.09133 | 0.092771 |
| 7 | -0.00433 | -0.07692 | -0.00426 | -0.01563 | -0.00704 |
| 8 | 0.031882 | 0.003713 | 0.035828 | 0.031665 | -8.86E-05 |

This shows a summary of the weekly returns for each asset

Additionally, the weight of each asset within the index fund was determined based on its market capitalization, retrieved from Yahoo Finance. The allocation of these weights followed the proportional valuation of each asset relative to the entire S&P100. This methodology ensured that the constructed index fund accurately represented key market trends while reducing the total number of assets invested. The analysis revealed that selecting between 40 and 50 assets strikes a balance between diversification and overfitting, preventing excessive tracking error for out-of-sample performance. This structured data collection approach allowed for the effective implementation of the optimization model in AMPL, ensuring meaningful portfolio construction and rebalancing constraints.

|  |  |  |
| --- | --- | --- |
| **assets** | **i** | **w** |
| MSFT | 2 | 10.98% |
| AAPL | 1 | 9.98% |
| NVDA | 7 | 9.58% |
| AMZN | 4 | 6.90% |
| GOOGL | 3 | 6.27% |
| META | 6 | 5.02% |
| TSLA | 5 | 3.23% |
| JPM | 8 | 2.37% |
| V | 11 | 2.27% |

This shows that the S&P100 has the highest weight in Microsoft, Apple, NVIDIA, Amazon, and Google as the top five assets.

**Modelling**

***Decision Variables***

* xi: Fractional allocation of asset i, with xi >= 0 and ∑in xi = 1
* yi: Binary variable indicating whether asset i is included in the fund.
* zt: Deviation variable capturing tracking error over T periods.

***Objective Function***

Minimizing the sum of tracking errors across all periods.

***Constraints***

Asset selection limit where at most q assets can be selected

If xi is nonzero, then yi must be 1.

Upper bound on rebalancing.

Lower bound on rebalancing.

This formulation captures the integer programming index fund optimization with constraints on rebalancing and tracking error minimization.

**Results and Analysis**

The results of this analysis demonstrate a clear relationship between the number of assets selected in the constructed index fund (q) and its expected return relative to the S&P100. By averaging returns over 10-week intervals across 262 weeks, the trend shows that as q increases, the expected return of the index fund more closely aligns with the performance of the S&P100.

For smaller values of q, deviations from the benchmark index are more pronounced, indicating that fewer assets struggle to replicate overall market movements. However, as q approaches higher values, particularly around 50, the expected return stabilizes and achieves a better match with the S&P100. This finding highlights the trade-off between diversification and overfitting, reinforcing that selecting an appropriate number of assets is crucial for maintaining tracking accuracy while avoiding excessive sensitivity to in-sample data.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **i** | **Benchmark** | **q10** | **q25** | **q50** | **q75** |
| 1 | 0.34% | 0.30% | 0.35% | 0.37% | 0.35% |
| 2 | 0.29% | 0.26% | 0.28% | 0.28% | 0.26% |
| 3 | 0.24% | 0.47% | 0.39% | 0.22% | 0.24% |
| 4 | 0.26% | 0.29% | 0.39% | 0.28% | 0.27% |
| 5 | 0.22% | 0.24% | 0.23% | 0.23% | 0.22% |
| 6 | 0.16% | 0.24% | 0.20% | 0.17% | 0.17% |
| 7 | 0.16% | 0.05% | 0.07% | 0.20% | 0.17% |
| 8 | 0.14% | 0.10% | 0.18% | 0.16% | 0.14% |
| 9 | 0.12% | 0.26% | 0.14% | 0.12% | 0.13% |
| 10 | 0.18% | 0.31% | 0.21% | 0.17% | 0.19% |

The construction and evaluation of the optimized index fund followed a structured approach, balancing in-sample calibration (86 weeks window) with out-of-sample validation (85 weeks window) and at least 90 weeks in the dataset available for a rolling window approach to ensure robustness. Initially, 33% of the available data was used for parameter estimation, stock selection, and weight allocation, while 33% served as out-of-sample testing to assess the fund's real-world performance. Additionally, our rolling window was large enough to generate 30 instances of in-samples where each sample moved forward by three weeks.

The testing methodology implemented a rolling-time window approach, where the out-of-sample data was divided into equally sized intervals. The fund was held over each interval, and the in-sample window was shifted forward to refine model parameters dynamically. Rebalancing constraints were introduced to limit excessive portfolio adjustments, ensuring practical feasibility. As the table illustrates, the expected returns of the constructed funds remained statistically comparable to the benchmark, except for higher values of 𝑞, where overfitting caused deviations in out-of-sample performance. By iteratively adjusting the fund composition, this approach provided valuable insights into the trade-offs between portfolio size, tracking accuracy, and stability in real market conditions.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **q** | **Normal Test Stat** | **p-value** | **Paired t-test stat** | **p-value2** |
| 10 | 0.97 | 0.50 | 0.58 | 0.57 |
| 25 | 0.95 | 0.13 | 0.23 | 0.82 |
| 50 | 0.95 | 0.13 | -2.04 | 0.05 |
| 75 | 0.96 | 0.32 | -2.39 | 0.02 |

The statistical analysis confirms that the expected returns across different values of 𝑞 are normally distributed, justifying the use of a paired t-test to compare the mean differences between the constructed index fund and the S&P100 benchmark. With a sample size of 30 observations per 𝑞, the paired t-test results indicate that lower values of 𝑞 (10, 25, and 50) do not show statistically significant deviations from the benchmark. However, for 𝑞 =75, the p-value falls below the significance threshold, leading to a rejection of the null hypothesis. This finding suggests that while increasing q enhances the similarity of expected returns to the S&P100, excessively high values introduce overfitting, which compromises out-of-sample performance. As a result, selecting an optimal range, such as 𝑞 ∈ [40, 50], ensures robust portfolio construction without sacrificing generalizability.

The analysis highlights a strong exponential relationship between the number of selected assets (q) and the objective value. As q increases, the objective value decreases significantly, indicating improved tracking accuracy between the constructed index fund and the S&P100. However, this trend also reveals a crucial trade-off: while higher values of q minimize tracking error, they also introduce overfitting, where the model optimally aligns with in-sample data but loses generalizability for out-of-sample performance. This overfitting effect suggests that beyond a certain threshold, particularly for very high q values, the model sacrifices robustness, leading to diminished practical effectiveness in real-world market conditions. Thus, selecting an optimal q range, such as [40, 50], ensures a balance between tracking precision and resilience against overfitting effects.

**Discussion**

The findings of this study underscore the effectiveness of passive portfolio management strategies that rely on diversification rather than predictive modeling. By selecting a subset of assets from the S&P100, the optimization framework successfully created index funds that minimize tracking error while maintaining strong correlation with the benchmark. The results demonstrate that increasing the number of selected assets improves similarity to the S&P100, but beyond a certain threshold, the model begins to overfit in-sample data, reducing its generalizability for out-of-sample returns. This emphasizes the importance of strategic asset selection, where an optimal range of 𝑞 ensures a balance between diversification and model robustness.

Furthermore, the statistical analysis confirms that lower values of 𝑞 maintain statistical similarity with the benchmark, while higher values (such as 𝑞=75) introduce overfitting complications. The paired t-test results validate the effectiveness of the selected methodology, reinforcing the need for controlled asset expansion when constructing reduced-asset index funds. This study provides valuable insights into the trade-offs between portfolio size and tracking accuracy, highlighting how integer programming-based optimization can guide passive investment strategies to achieve market performance with fewer assets.

**Appendix**

param n;

param w{i in 1..n};

param q;

param T;

param r{t in 1..T, i in 1..n};

var x{i in 1..n} >= 0;

var z{t in 1..T};

var y{i in 1..n} binary;

minimize objFunc: sum{t in 1..T} z[t];

subject to minAssets: sum{i in 1..n} y[i] <= q;

subject to allocateWeights: sum{i in 1..n} x[i] = 1;

subject to assetWeight {i in 1..n}: x[i] <= y[i];

subject to maxReturn {i in 1..n}: sum{t in 1..T} z[t] >= sum{t in 1..T} (w[i] - x[i]) \* r[t, i];

subject to minReturn {i in 1..n}: sum{t in 1..T} z[t] >= sum{t in 1..T} -(w[i] - x[i]) \* r[t, i];

* AMPL model file with rebalancing constraints

option solver CPLEX;

model p.mod;

data p\_1.dat;

solve;

display objFunc, q;

*# Export values to a txt*

printf "Outputs 30 runs\n" > out10.txt;

printf "objFunc:%f\n", objFunc >> out10.txt;

printf "\nx values:\n" >> out10.txt;

for {i in 1..99} {

    printf "%f\n", x[i] >> out10.txt;

}

reset;

model p.mod;

data p\_2.dat;

solve;

display objFunc, q;

printf "\nobjFunc:%f", objFunc >> out10.txt;

printf "\nx values:\n" >> out10.txt;

for {i in 1..99} {

    printf "%f\n", x[i] >> out10.txt;

}

* AMPL run file where 30 data files with different windows of our asset data used as the in-sample data