FULL LEGAL NAME	LOCATION (COUNTRY)	EMAIL ADDRESS	MARK X FOR ANY NON-CONTRIBUTING MEMBER
Chiransh Gulati	US	gulatichiransh@gmail.com	
Harshil Sumra	India	harshilsumra1997@gmail.com	
Pritam Biswaranjan Datta	India	pritamdattaedu@gmail.com	

Statement of integrity: By typing the names of all group members in the text boxes below, you confirm that the assignment submitted is original work produced by the group (excluding any non-contributing members identified with an "X" above).	
Team member 1	Chiransh Gulati
Team member 2	Harshil Sumra
Team member 3	Pritam Biswaranjan Datta

Use the box below to explain any attempts to reach out to a non-contributing member. Type (N/A) if all members contributed. **Note:** You may be required to provide proof of your outreach to non-contributing members upon request.
N/A

Introduction

In today's world, all information/intelligence has ended up becoming a very strategic resource. How efficiently and effectively this resource is utilized has a very direct and observable impact in today's world. It is just visible in a more quantitative sense when observed in the case of applications involving mathematical models to financial data. This application of mathematical models utilizing this large variety of data raises several questions.

First, is our idealistic or constrained mathematical model generalizable? Or even if the idealistic solution offered fits the market well most of the time, what happens when the market changes? Or of all the wide variety of data or information streams available to us, which ones are significant to our dependent variable in focus? From a little more financial aspect, how to create a truly diversified portfolio?

In order to answer some of the above questions, many different techniques were created. For example, a model's validity can be easily tested by applying the model across a cross-section of market environments and back testing the model validity in all of them respectively. Factor selection for the model can be done using regularization techniques like Principal component analysis. Another such technique is "Denoising and Detoning" then portfolio covariance matrix in order to get more accurate weights. Clustering techniques help us segregating a collection of data points into smaller exhaustive subsets of data points where each subset has its own set of unique defining features and those defining features end up becoming the point of difference between any two different subsets.

1.1 Denoising and Detoning

Covariance is the center piece of financial mathematics. It is a necessary and defining characteristic of financial variables in the field of mathematical finance. in Markowitz's Portfolio Optimization theory, we aim to select a portfolio which maximizes our Sharpe's ratio given a set of constraints. Higher Sharpe's ratio basically means higher returns or minimum variance. Hence, we try and choose the stocks with good returns which together offer sufficient diversification (via techniques like clustering), that is, offset each other's non-systemic risk.

Denoising- is the technique which compares distribution of eigenvalues from the covariance matrix to the standard Marcenko-Pastur distribution, this distribution represents ideal distribution of noise. The eigenvalue-factor pair not inline to the ideal noise distribution represent the signals. Therefore, once we replace the variances of these noise corresponding factors with the average of all individual factor variances, it makes our covariance matrix more stable for mathematical implementation.

Detoning- is more about is more about removing certain common signal across all factors. For example, the contribution of market risk premium is common across all financial products in the market. Therefore, we can surely remove the variance coming from markets systematic risk. The removal of this systemic risk signal will allow us to focus more on the non-systemic risk signal, which can be helpful if all individual portfolio members are from different clusters, that is, the portfolio is diversified.

Benefits:

1. Enhanced Signal-to-Noise Ratio: By reducing the noise in the data, denoising techniques can improve the signal-to-noise ratio, making it easier to identify meaningful patterns and trends in the portfolio data.
2. Smoother Performance Analysis: Denoising can result in smoother and more accurate performance analysis. By removing short-term fluctuations or outliers caused by noise, it becomes easier to assess the true performance of the portfolio over time.
3. Improved Risk Assessment: Noise in portfolio data can distort risk assessments. Denoising helps in obtaining a clearer picture of risk by eliminating noise-induced distortions and providing a more accurate measure of portfolio volatility or downside potential.
4. Enhanced Decision Making: Denoising techniques can contribute to more informed decision-making processes. By reducing noise, portfolio managers can make decisions based on more reliable and relevant information, leading to potentially better investment choices and improved performance.
5. Enhanced Predictive Models: Denoising can benefit the development and accuracy of predictive models used in portfolio management. By eliminating noise, the models can focus on the underlying trends and relationships in the data, resulting in more robust predictions and forecasting capabilities.
6. Reduced False Signals: Noise in portfolio data can generate false signals or misleading indicators. Denoising helps filter out such false signals, providing a clearer view of genuine market trends and reducing the risk of making decisions based on erroneous or misleading information.
7. Enhanced Risk-Return Trade-off: Denoising techniques can assist in achieving a better risk-return trade-off. By reducing noise, portfolio managers can identify investment opportunities with potentially higher returns and lower risk, leading to more optimal portfolio allocations.
8. Non-Systemic analysis in a diversified portfolio- Detoning allows us to get a more individually focused signal pertaining to each individual factors.

It is worth noting that the implementation of above techniques can vary, ranging from simple moving averages and exponential smoothing to more sophisticated filtering methods such as wavelet transforms or machine learning algorithms, but the driving logic remains the same, which is to achieve aforementioned benefits. Implementing these techniques in portfolio management, investors can benefit from improved data quality, enhanced decision-making processes, and potentially superior portfolio performance.

1.2 Clustring

In simple terms, “Clustering is basically the technique to gather similar things together in one basket while having several sets of baskets each having a set of similar things. The “things” in the above analogy can be financial firm/stock statistics or even image’s data information in case of facial recognition applications. Another financial application would be to segregate loan applicants into different clusters to do demographic analysis to identify the credit worthy and credit demographic characteristics. The first thing we need to perform clustering is distance metric which helps us move towards an optimal cluster from a suboptimal one. There are several different distance metrics like Euclidean distance (most common), Manhattan distance and Cosine distance (angular distance between two-unit vectors, it helps us test in case of natural language processing applications). Then we decide on the definition of cluster center, based on which both the inter and intra cluster distance is measured. Common seen implementation is of median (robust to outliers but doesn't incorporate information from all data points) and mean (gets impacted by outliers). There is another criterion of type of distribution, in which case the factors with similar distribution characteristics are clubbed together. Choice of number of clusters can be done using the elbow method or by simulating the process for different number of clusters and deciding on the least number of clusters which have maximum segregation.

There are two most common clustering algorithms:

K-Means algorithm: In this we start with "K" random cluster points. Based on these centers, we put all available data points in different clusters. Then based on the definition of cluster center chosen to define new cluster centers. Then the process starts a fresh with these new cluster centers till we reach the point where two consecutive cluster centers are identical.

Hierarchical clustering is of two types: Agglomerative and Divisive clustering.

- Agglomerative clustering – all data points are considered individual clusters. For each data point we calculate its distance from all other data points then we merge the two closest data point pair to form a new cluster. We keep repeating this process till we reach the desired number of clusters. Hence, this approach is also called "bottom to top approach".
- Divisive clustering- the whole of the data set is considered one single cluster and then we keep on recursively dividing the data into smaller clusters till all the data points end up becoming individual clusters. Hence, this approach is also called "top-down approach".

We use silhouette score metric as a metric of dissimilarity as a good clustering algorithm has small intra cluster distance and large inter cluster distance, symbolized by higher silhouette score (closer to 1). Clustering can help us in identifying stocks/firms with similar properties which could help us do industry wise analysis or even help in the portfolio managers endeavor to diversify the portfolio.

Benefits:

1. Portfolio Segmentation: Clustering allows for the segmentation of portfolios based on similar characteristics, such as asset type, sector, market capitalization, or risk profile. This segmentation enables portfolio managers to better understand the composition and diversification of their portfolios.
2. Risk Management: Clustering can help identify groups of assets with similar risk profiles. By analyzing the risk characteristics within each cluster, portfolio managers can make informed decisions about risk management, such as adjusting allocations, implementing hedging strategies, or rebalancing portfolios to achieve desired risk exposures.

3. Asset Allocation Strategies: Clustering can assist in identifying asset allocation strategies by grouping assets that exhibit similar historical performance or correlation patterns. Portfolio managers can then allocate assets across different clusters to create diversified portfolios with varying risk-return profiles.
4. Performance Evaluation: Clustering can aid in performance evaluation by comparing the performance of assets within each cluster. This analysis allows portfolio managers to assess the performance of individual clusters or subgroups and identify areas of strength or weakness.
5. Investment Opportunities: Clustering can uncover investment opportunities by identifying clusters of assets that have exhibited similar price movements or other relevant factors. Portfolio managers can focus on these clusters to identify potential investments that align with their investment strategies and objectives.
6. Portfolio Optimization: Clustering can be incorporated into portfolio optimization processes by grouping assets with similar risk and return characteristics. By considering cluster-specific constraints and objectives, portfolio managers can optimize portfolio allocations within each cluster to achieve desired risk-return outcomes.
7. Market Analysis: Clustering can provide insights into market dynamics and trends. By clustering assets based on market behaviour, portfolio managers can gain a deeper understanding of market segments, investor sentiment, and potential shifts in market conditions.

It is important to note that the choice of clustering algorithm and the selection of relevant features or variables for clustering depend on the specific portfolio management goals and the characteristics of the data being analyzed or even the choice of the portfolio manager.

1.3 Backtesting

This is simply the process of model validation based historical data. It has a wide range of applications in the field of finance. It helps us see the goodness of the designed model/algorithm given the market moved as it did in the specified historical time frame. This has also been known to be used to validate model results using different cross-sections of data in order to test whether the model is generalizable.
Benefits:

1. Strategy Evaluation: Backtesting allows portfolio managers to evaluate the effectiveness of different trading or investment strategies. By applying the strategy to historical data, they can assess its performance, including returns, risk metrics, drawdowns, and other relevant indicators.
2. Risk Assessment: Backtesting enables portfolio managers to quantify the risk associated with a particular strategy. By analyzing historical performance, they can estimate key risk measures such as volatility, maximum loss, or Value-at-Risk (VaR), providing insights into the potential downside risks and risk-adjusted returns of the strategy.
3. Fine-tuning Strategies: Backtesting helps in fine-tuning and optimizing trading or investment strategies. By analyzing the performance of different variations or parameters within a strategy, portfolio managers can identify the most optimal settings or adjustments to enhance returns, reduce risk, or improve other performance metrics.

4. Scenario Analysis: Backtesting allows for scenario analysis by applying the strategy to historical data during different market conditions or economic environments. This analysis provides insights into how the strategy would have performed during different market cycles, helping portfolio managers understand its robustness and adaptability.
5. Benchmark Comparison: Backtesting facilitates the comparison of a strategy's performance against relevant benchmarks or alternative investment options. By measuring and comparing returns, risk metrics, and other performance indicators, portfolio managers can assess the strategy's outperformance or underperformance relative to the benchmarks.
6. Decision Support: Backtesting provides a historical track record of the strategy's performance, which can serve as a valuable decision-support tool. Portfolio managers can use the backtesting results to gain confidence in the strategy, make informed decisions on whether to deploy the strategy in real-time trading, or allocate resources accordingly.
7. Learning and Improvement: Backtesting allows portfolio managers to learn from past performance and improve their strategies over time. By analyzing backtesting results, they can identify strengths, weaknesses, and areas for improvement, refining their approaches to enhance performance and adapt to changing market conditions.

It is important to note that backtesting has limitations, and past performance does not guarantee future results. Assumptions, data quality, and modeling choices can impact the accuracy and reliability of backtesting results. Therefore, it is crucial to conduct rigorous analysis, consider additional risk factors, and validate results with real-time performance when implementing strategies in live trading.

By leveraging backtesting effectively, portfolio managers can gain valuable insights into strategy performance, risk assessment, and decision-making, leading to more informed and potentially improved portfolio management outcomes.

2. Applying improvements on GPW2 (combination of the MVO, Black-Litterman and Kelly)

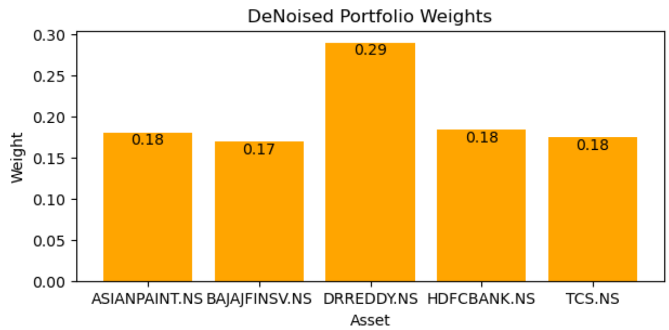
In GWP2 we had compared portfolio constructed across 3 models. The best risk adjusted return was achieved for the combination of the MVO, Black-Litterman and Kelly denoted as ‘Combine 3’. Therefore this was the baseline taken forward for the GWP3 as required. We start with the portfolio weight for the 5 stocks in the ratio of 23%, 6%, 48%, 5% and 18%

	Combination	Return	Std. Dev.	Sharpe Ratio	AsianPaint Wt	BajajFin Wt	DrReddy Wt	HDFCBank Wt	TCS Wt
0	Combine 3	0.6840	0.0268	1.5610	0.23	0.06	0.48	0.05	0.18
1	BL + Kelly	0.5900	0.0274	1.3116	0.20	0.04	0.51	0.06	0.18
2	BL + LC	0.5706	0.0283	1.2261	0.27	0.12	0.38	0.02	0.21
3	Kelly + LC	0.5768	0.0283	1.2379	0.30	0.06	0.43	0.00	0.17

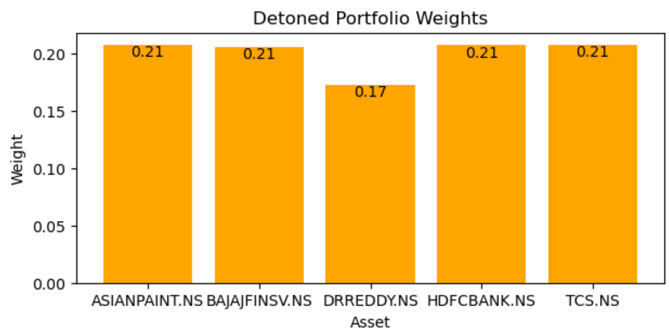
Approach to denoising, clustering and backtesting.

We have used code suggested (mentioned in references) to implement the algorithms.

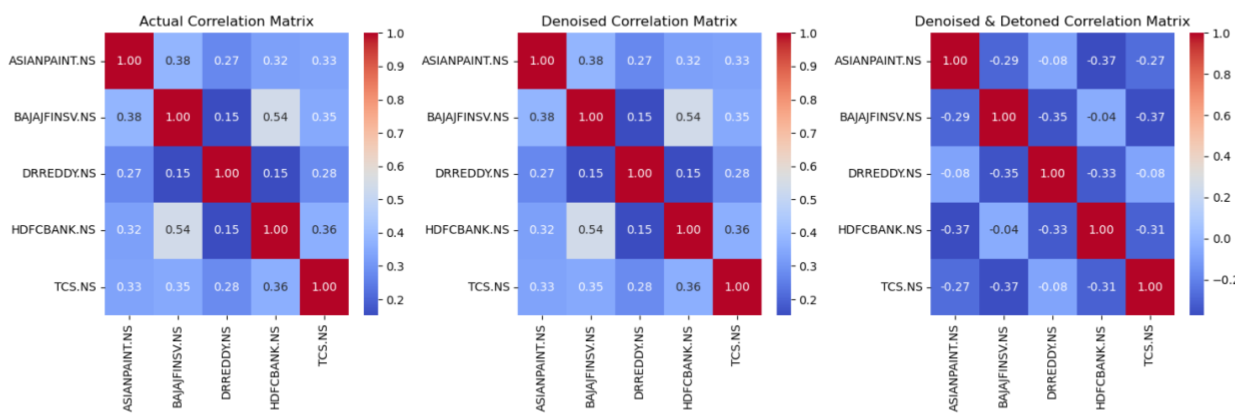
The denoised correlation matrix was obtained from the daily returns data of the 5 stocks. For expected returns we used the annualized returns of the 5 stocks. We needed to get the weights for the denoised data for which we used the MVO technique.



We proceed to use the denoised data to detone where the market component is removed from the data. The effect of market trends are generally eliminated from the correlation and idiosyncraric stock behaviour is more pronounced.



A comparative glance across the 3 correlation matrix shows that there is minimal to very low signal-to-noise ratio in the data because the actual and denoised matrix is identical. Eliminating the market component depicts negatively correlation amongst the stocks. The 2 financial sector stocks HDFC and BAJAJ are however are comparatively tightly correlated.



An interesting observation is that the pharma stock DrReddy seems to be the exception amongst the pack. The weight assigned to it across the models is almost always in the opposite direction to the other 4 stocks, which provides a very anticipatory entry to the next topic of clustering !

Clustering:

We user the correlation and covariance matrix to arrive at the cluster formation, inter cluster allocation and stock allocation.

The clusters formed are:

{0: ['DRREDDY.NS'],
 1: ['ASIANPAINT.NS', 'BAJAJFINSV.NS', 'HDFCBANK.NS', 'TCS.NS']}

As anticipated by earlier model observations, the pharma stock is in cluster all by itself while the pther 4 stocks are in the other cluster.

The inter cluster allocation is ~30% and 70% respectively.:

Inter Cluster Allocation:

0	0.298356
1	0.701644

The 30% weightage on DrReddy could be a cause for concern which can be addressed by increasing the number of stocks from 5 to perhaps 7 or 10 and achieve a higher intra cluster diversification.

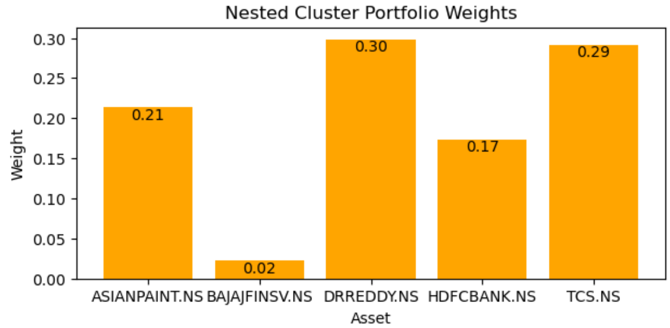
Another interesting metric is the silhouette score for the stocks:

Silhouette:

ASIANPAINT.NS	0.056186
BAJAJFINSV.NS	0.186184
DRREDDY.NS	0.000000
HDFCBANK.NS	0.177169
TCS.NS	0.053897

Where we observe the significant overlap between the 2 financial sector stocks of HDFC anb BAJAJFINSV.

The portfolio allocation is as below:



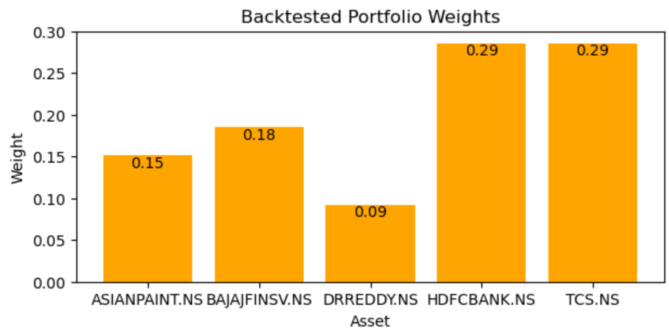
Backtesting:

For the exercise we start with an initial weight distribution and the ctual returns data. A simulation of the portfolio return is calculate for M=1000 times. The probabilistic Sharpe ratio is calculated for the 1000 entries. We provided the Nifty50 return as the sharpe ratio (with rf = 0.03) benchmark.

The Deflated Sharpe Ratio observed was:

Deflated Sharpe Ratio: 0.9647529787184895

This indicates high probability of the return from this allocation could be less than the risk-free rate.



3. **Improvement matrices added in code file**

4. **Improvements training and testing on different sample**

We tested the model on the days of data between GWP2 and GWP3

```
test_data.head(3)
```

Date	ASIANPAINT.NS	BAJAJFINSV.NS	DRREDDY.NS	HDFCBANK.NS	TCS.NS
2023-07-06	3399.399902	1619.900024	5191.203613	1675.000000	3314.281494
2023-07-07	3343.699951	1614.900024	5142.286133	1660.400024	3320.615234
2023-07-10	3343.149902	1597.800049	5114.950195	1656.449951	3263.463623

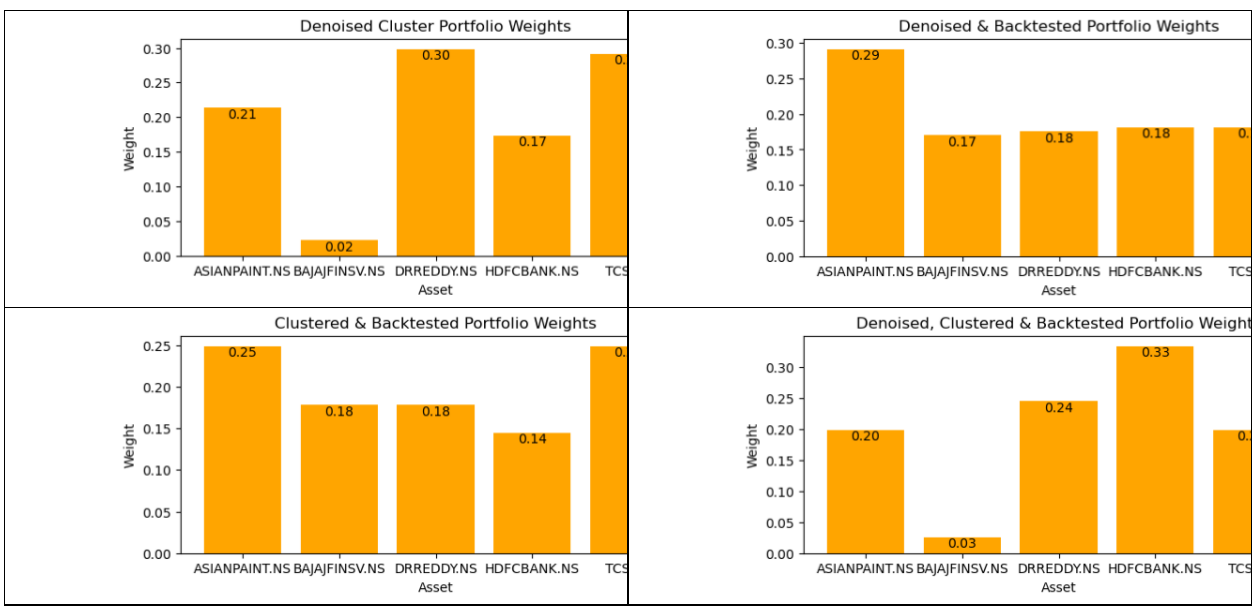
```
test_data.tail(3)
```

Date	ASIANPAINT.NS	BAJAJFINSV.NS	DRREDDY.NS	HDFCBANK.NS	TCS.NS
2023-07-21	3517.699951	1632.849976	5288.200195	1675.750000	3368.300049
2023-07-24	3543.699951	1657.500000	5391.700195	1678.400024	3394.750000
2023-07-25	3400.399902	1648.849976	5425.399902	1696.599976	3399.149902

We carried out 4 scenarios to combine the improvements.

1) Denoised + Clustering
2) Denoised + Backtested
3) Clustering + Backtested
4) Denoised + Clustering + Backtested

The approach was to use the output from one model as input to the next model to see the combined effect in terms of the correlation matrix and weights.



- Denoised + Clustered: Is completely influenced by the clustering model.
- Denoised + Backtested: Is largely dominated by the Denoising model

- Clustered + Backtested: Equally influenced
- Denoised + Clustered + Backtested: Influenced by Denoising more than Clustering or Backtesting.

5. Q&A

1. What are the differences in performance according to these metrics for the different combination of improvements?
2. What reasons might explain these differences in performances (with specific reference to the financial products, the historical data, the math of the model dynamics, etc.),
3. In which cases, if any, does the incremental gain in performance according to the chosen metrics justify the additional complexity and effort of the improvements?

Comparative Performance:

We compared portfolio returns, sharpe ratio, std dev, max drawdown and sortino ratio for the individual and combination of models.

	Model	AsianPaint Wt.	BajajFinSv Wt.	DrReddy Wt.	HDFCBank Wt.	TCS Wt.	Port. Returns%	Sharpe Ratio	Std Dev	Max Drawdown	Sortino Ratio
0	GWP2[Optimal+BL+Kelly]	0.2300	0.0600	0.4800	0.0500	0.1800	2.8747	0.0081	0.2742	-0.0106	1.1873
1	Denoised	0.1805	0.1699	0.2902	0.1842	0.1752	2.3723	0.0068	0.2679	-0.0097	0.5627
2	Detoned	0.2073	0.2057	0.1727	0.2068	0.2075	2.0217	0.0068	0.2287	-0.0098	0.5400
3	Clustered	0.2138	0.0226	0.2984	0.1737	0.2915	2.4442	0.0079	0.2382	-0.0127	0.4737
4	Backtested	0.1513	0.1849	0.0924	0.2857	0.2857	1.9242	0.0074	0.2010	-0.0116	0.4481
5	Denoised & Clustered	0.2138	0.0226	0.2984	0.1737	0.2915	2.4442	0.0079	0.2382	-0.0127	0.4737
6	Denoised & Backtested	0.2913	0.1705	0.1759	0.1812	0.1812	1.8745	0.0071	0.2054	-0.0094	0.4349
7	Clustered & Backtested	0.2490	0.1785	0.1785	0.1450	0.2490	2.0325	0.0074	0.2125	-0.0106	0.4654
8	Denoised Clustered & Backtested	0.1984	0.0259	0.2443	0.3330	0.1984	2.1598	0.0071	0.2347	-0.0109	0.5028

When compared to the ‘best’ portfolio in terms of superior risk-reward ratio, the GWP2 combined model of MVO+BL+Kelly still stands out compared to any of the models developed by denoising, clustering or backtesting.

Models based on clustering is next based, followed by denoising and then backtesting.

Constraining the exercise to 5 stocks and testing the models over a small period of 14 days may not have been sufficient to train and then test the model. Yet clustering and denoising is pretty close to the best portfolio. Backtesting in its nature is conservative in approach to remove effects of “sins”. Backtesting models are less volatile and that is the only metric in its favour.

This exercise does not provide evidence that the complexity of building the models justify the effort for a risk-taking investor. However, this exercise is worthy to be carried out for risk-averse investor.

6. Conclusion:

The exercise was conducted using only one type of asset which were stocks. Other constraints like the number of assets and the testing time period also may have affected the observed outcomes.

The models are complex and application of those tend to make the portfolio constructed more balanced for risk-nuetral and risk-averse institutions and investors.

Using correlation and covariance matrix to develop the model is a good approach for quants and financial engineers. In terms of interpretability the change in allocation is a good way to show the outcome. However, explaining the why is tad bit difficult and will require more visualization that we could use in this assignment.

There is a appreciable observable difference in results using these models, although in terms of performance it did not improve to the model explored earlier in GWP2. Notwithstanding the constraints, these models do have merit and it is worth the effort for financial engineers and serious investors.

References:

1. Denoising & Clustering Code: https://github.com/emoen/Machine-Learning-for-Asset-Managers
2. Implementation of code snippets and exercises from Machine Learning for Asset Managers (Elements in Quantitative Finance) written by Prof. Marcos López de Prado.
3. https://github.com/rubenbriones/Probabilistic-Sharpe-Ratio/tree/master
4. Probabilistic Sharpe Ratio example in Python (by Marcos López de Prado)
5.