Portfolio optimization via machine learning models for Italian and American Market.

Francesco Pivi, Elisa Castagnari, Matteo Fusconi
Master's Degree in Artificial Intelligence, University of Bologna
{ francesco.pivi, elisa.castagnari, matteo.fusconi4 }@studio.unibo.it

Abstract

In this study, we tackle two main challenges in quantitative finance: selecting an optimal stock portfolio and improving returns annually. To achieve this, we leverage various clustering techniques to identify promising shares for the upcoming month across different markets. We then optimize portfolio weights based on the Sharpe ratio. Finally, we tune different models to guide investment decisions for the following month. Our results reveal superior performance compared to the FTSEMIB index in the Italian market, while achieving outcomes comparable to the S&P 500. This difference may be attributed to the higher liquidity in the American one, renowned for its resilience during crashes and rapid recovery mechanisms.

1 Introduction

Portfolio optimization involves selecting the best asset distribution to achieve the following objectives: maximizing expected return and minimizing financial risks.
The common strategy for long-term investments involves selecting the best-evaluated stock, from each economic sector, to build a robust portfolio [1]. However, this approach has drawbacks. It exposes investors to guaranteed losses during total market crashes, as seen in historical events like 1929, 1987, 2008, and 2020.
Additionally, it fails to maximize returns during market bubbles and tends to perform similarly to the tracking index of the respective market, which is positive for the ones with demonstrated endogenous growth [2], like the U.S. But this does not hold true for Italy, which has lost about 20% of its value over the past 20 years.
Our approach involves buying a subset of available stocks at the beginning of each month and selling them at the end of it. Each of these is based on clustering methods.
The aim for selecting these assets is to have: a low volatility and a high buy rate. We proceed to fine-tune the portfolio weights based on the Sharpe ratio [3] over a one-year time frame, aiming to identify the optimal portfolio for the upcoming month.
To reduce the impact of market downturns, we focus on predicting the general trend of stock prices instead of its specific values. If our analysis shows a negative trend in the expected portfolio value for the next month compared to the current value, we will avoid making new investments to protect our capital. In order to tackle that problem, we developed various techniques (discussed specificly in section 3).
The algorithm has been tested on the American and Italian model separately.
Our approach appears to be sensitive to the clustering method and the temporal window considered. The choice of clustering method, such as DBSCAN or K-Means, significantly impacts the selection of stocks for the portfolio. Their sensitivity suggests that the underlying structure of the data, particularly the relationships between different stocks, plays a crucial role in the effectiveness of your strategy. Additionally, the temporal window considered for optimization, such as the one-year historical window, also affects the results.
This is because financial markets are dynamic, and strategies that perform well in one period may need adjustments to adapt to its changing dynamics.
We also experimented with an XGBoost technique, which yielded inferior results, and attempted to implement an LSTM seq2seq model. However, due to the limited dataset, we ultimately chose classical machine learning models.

2 Background

Before delving into portfolio optimization, it is important to grasp key concepts of its theory.
The daily return of a stock represents the percentage change in its consecutive daily prices. From this, we calculate both daily and annual volatility.

The former is the standard deviation of daily return values, while the latter is the daily volatility multiplied by the square root of 252, representing the number of trading days in the stock market. Understanding a stock's annual volatility is crucial for assessing the associated risk from an investor's perspective.
Once we have calculated the volatility and return of individual stocks using their historical prices, the next step is to determine the covariances and correlations within a portfolio. This information is encoded in matrices, which help illustrate the degree of association between pairs in the portfolio [4]. An effective portfolio seeks to balance risk and return, in particular, minimize risk and maximize returns.
To achieve risk minimization, it is important to select stocks with low correlations among themselves. This approach enhances portfolio diversity, contributing to the overall goal of optimizing the risk-return trade-off.
The expected return of a portfolio (denoted as E(R)) containing n different stocks $S_1, \ldots, S_n$ with associated weights $w_1, \ldots, w_n$ is given by:

$$E(R) = \sum_{i=1}^{n} w_i E(R_{S_i})$$

In order to calculate the variance of a portfolio, we utilize the variances of the individual stocks (σ_i) within the portfolio and consider the covariances between each pair ($Cov(i, j)$). The formula for computing the total variance, denoted as Σ, is expressed as follows [5]:

$$\Sigma = \sum_{i=1}^{n} w_i \sigma_i^2 + 2 \times \sum_{i,j} w_i w_j Cov(i, j)$$

We have now created a two-dimensional space known as the risk-return space. The X-axis represents the variance, while the Y-axis is the return. By adjusting the portfolio weights, we can generate various combinations of returns and volatilities. We will choose the portfolio that maximizes the **Sharpe ratio** defined as :

$$SR = \frac{R_c - R_f}{\sigma_c}$$

where R_c, R_f and σ_c are the return of the current portfolio, the risk free portfolio and the standard deviation of the former.
This problem can be solved if we reformulate it as an optimization problem for the weights. We will solve a quadratic programming problem using Lagrange Multipliers [6]:

$$\min(\frac{1}{2}\vec{w}^T \Sigma \vec{w}) \text{ s.t. } \vec{R}^T \vec{w} \geq R_b \text{ and } \sum_i w_i = 1$$

where $\vec{R}$ is the vector of returns of each asset and R_b is the maximum return found for that given weights volatility. This formulation aligns with the Pareto frontier concept in economics.
In addition we give a brief description of the techniques used in order to construct the dataset for our research proposal, where we incorporated various financial indicators. The **Fama Fetch model** [7] tries to give an empirical explanation of the expected return of a given portfolio. The formula proposed in order to do so is:

$$E(R_i) - R_f = \beta_i (E(R_m) - R_F) + s_i E(SMB) + h_i E(HML)$$

Where we have that:

- R_f: the risk-free return.
- $E(R_m) - R_f$: the additional return expected from investing in a diversified portfolio.
- $E(SMB)$: expected return difference between portfolios with low and high market capitalization.
- $E(HML)$: expected return difference between portfolios with low and high book-to-market value ratios.
- a_i, b_i, s_i: parameters estimated through linear regression.

Lastly we employed the **ARIMA** model to inform our decision-making process on whether to make purchases. This modeling approach is proficient in transforming a non-stationary distribution into a stationary one, where the unconditional joint probability mass function does not change when shifted in time, primarily through differencing. That approach is particularly useful because the stock market itself does not exhibit stationarity, but returns do, because tend to be normally distributed around a zero mean also when time-shifted.
The model leverages historical data to forecast future values, operating under the assumption that past information is indicative of future trends. The key parameters include p (the order of the auto

regressive term noted as AR), d (the order of differencing for achieving stationarity), and q (the order of the moving average term (MA)).
The formula that describes the value in the series at a given time with $d = 1$ is:

$$y_t - y_{t-1} = y'_t = c + \sum_{i=1}^{p} \phi_i y'_{t-i} + \sum_{j=1}^{q} \theta_j \epsilon_{t-j} + \epsilon_t$$

where the first sum represent the forecasted series based on the past value of the it, whereas the second one on the past errors.

3 System description

We acquired data for the Italian and American markets through the Yahoo Finance API. Subsequently, we calculated various technical indicators on a daily basis, which were then aggregated on a monthly basis. Employing diverse clustering techniques, we identified a set of potentially high-performing stocks for the upcoming month based on the previous one.
Following this, we applied various portfolio optimization methods:

- **Baseline:** Our initial approach involves purchasing all promising assets identified through clustering at the beginning of each month and selling them at its end. Portfolio weights are optimized according to principles outlined in the background section.

- **Random Buyer:** For this strategy, at the start of each month, a coin is flipped. If it lands heads, assets are purchased for the next month; if tails, positions are held.

- **Persistency:** A variation of the previous strategy, where only the price change between the previous day and the one before is considered. If the change is positive, a position is bought for the next month; if negative, the position remains untouched.

- **Linear Regression:** In this strategy, instead of automatically purchasing promising stocks at the beginning of the month, a linear regression is fitted to the previous 14 days' data for each stock. The obtained slopes are multiplied by the weights assigned to each stock for the month. If the sum of the weighted slopes is positive, assets are purchased; otherwise, capital remains untouched. This approach helps filter out periods of intense market crashes.

- **ARIMA:** A variation of the linear regression strategy, where an ARIMA model is utilized to anticipate trends for each stock in the upcoming month. The model is trained using data from the previous 14 days to predict trends for the following week for each chosen asset. If the total weighted predicted trend is positive, assets are bought; otherwise, capital is maintained. It is important to note that this model is highly sensitive to hyperparameters.

Additionally, we compared the results achieved with those obtained without clustering but using the same optimization methods.
After the employment of these strategies, we calculate the daily return of our portfolio. Subsequently, we visualize and compare the obtained results.

4 Data

The dataset includes daily information for each stock from 2006 to the present, encompassing: the highest and lowest prices of the day, opening and closing prices, the number of stocks exchanged, and the adjusted close (the closing prices after adjustments for splits and dividend distributions).
Next, we extract various technical indicators and features for each stock. This includes:

- **Garman-Klass volatility:** considers both the opening and closing prices of a stock [8], leveraging the higher activity observed during the opening and closing of trading sessions for a more accurate volatility estimation. In this approach, we assume a continuous diffusion process, akin to geometric Brownian motion. However, it is important to note a drawback: the method may not handle opening price jumps and trend movements robustly.

$$\text{GKV} = \frac{(\ln(\text{High}) - \ln(\text{Low}))^2}{2} - (2\ln(2) - 1)\,(\ln(\text{Adj Close}) - \ln(\text{Open}))^2$$

- **Relative strength index (RSI):** provides a momentum indicator within technical analysis [9]. It gauges the velocity and extent of recent price changes in an asset, providing insights into whether the stock is currently in a bullish or bearish period. It ranges from 0 to 100,

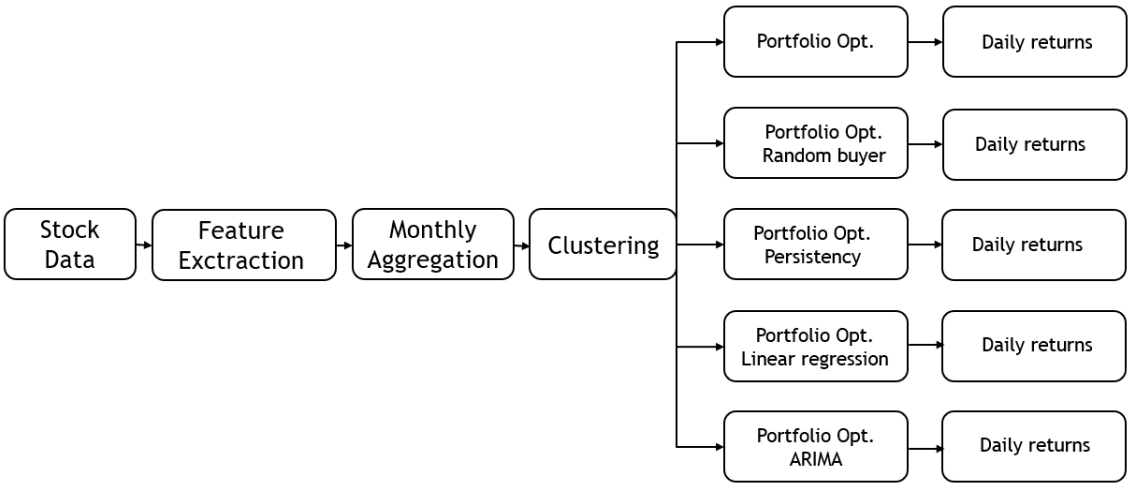

Figure 1: Pipeline developed.

enabling us to an easy understanding of its values. It is defined as:

$$RSI = 100 \times \frac{U}{U + D}$$

where U represents the average of upward closing price differences over a 14-day period, while D denotes the average absolute values of downward closing differences over the same days.

- **Bollinger bands:** consists of envelopes at a standard deviation level above and below a simple moving average of the price. These bands dynamically adjust to fluctuations in the underlying price, responding to changes in volatility. They are calculated over a period of 20 days. When the price chart exits the upper band and then re-enters, a sell signal is generated, indicating a rapid price increase followed by a subsequent slowdown or adjustment. Conversely, when the price chart exits the lower band and then re-enters, a buy signal is triggered. This signifies a sharp decline in price, potentially coming to a halt and, perhaps, reversing the trend.

- **Average true range (ATR):** considers not only the volatility of that day but also takes into account the one of the previous 14 ones usually. Its formula is:

$$ATR_t = \frac{ATR_{t-1} \times (n-1) + TR_t}{n}$$

where TR it is called true range and can be obtained by $\max(high, close_{prev}) - \min(low, close_{prev})$, where n is the number of considered days. The first ATR of the series is defined as $\frac{\sum_i^n TR_i}{n}$.

- **Moving Average Convergence Divergence (MACD):** It represents the disparity between two moving averages: one calculated over a 12-day period and the other over 26 days. If it is positive it means we are on a bullish period, while if it is negative in a bearish one [10].

- **Dollar volume:** it is simply the product of the number of stocks exchanged that day with their relative adjusted close.

- **Fama Fetch Factors:** include the market risk premium (RMRF), the size effect (SMB - Small Minus Big), and the value effect (HML - High Minus Low), providing a more comprehensive framework for understanding and explaining stock returns in financial research. They are better discussed in the background section.

5 Experimental setup and results

We experimented various clustering methods to aggregate stocks, specifically exploring: DBSCAN, K-Means, and Agglomerative. The most effective results were achieved using the first two, while the last one yielded subpar outcomes in silhouette analysis. We also experimented our optimization approaches skipping the clustering step. All techniques were evaluated with all the optimization approaches.

In our analysis, DBSCAN was employed to determine the optimal number of clusters each month. This was accomplished through a grid search over the maximum distance (eps) between two data points and the minimum number of samples in a neighborhood for a single one in order to to be considered a core point. The best number of clusters was identified by sorting the clusters based on silhouette score and the percentage of unclustered data.

Regarding K-Means, we observed an elbow point in the inertia plot consistently around 3 for the Italian market and 4 for the American one, each month during our training period from 2006 to 2016. As a result, we chose a fixed number of clusters for that algorithm based on these observations.

In the portfolio optimization phase, we experimented various time windows for optimizing the weights for each month. Ultimately, we found out that using a one-year historical window provided the optimal balance between computational efficiency and the quality of results.

Relatively to the filters applied by the regression and ARIMA models, we opted to look back only 14 and 30 days for the Italian and American market respectively. Additionally, for latter, we chose a forecasting period of five days for the adjusted close.

This translates to a strategy of initiating purchases at the start of each month, conditioned on a positive forecasted trend for the first week of the upcoming month.

However, extensive tuning wasn't feasible due to limited hardware capabilities. The results obtained over the training (2008-2018) and validation dataset (2016-2024) are reported in the following tables.

To assess and compare our results, we chose the Sharpe ratio for each strategy. This metric provides a useful measure of both portfolio volatility and return. Additionally, we included the Return on Investment (ROI) to identify which one of these techniques yielded the highest overall performance.

		ROI	σ	SR
NO-Clust	Base	24.9%	0.011	0.015
	Persistency	39.8%	0.008	0.024
	Regression	51.6%	**0.006**	0.035
	Arima	**79.5%**	0.008	**0.041**
	Mean	48.9%	0.008	0.029
	Random bs	-7%	0.006	-0.004
K-Means	Base	-23.7%	0.013	-0.004
	Persistency	23.3%	**0.009**	0.016
	Regression	**54.5%**	0.010	0.027
	Arima	42.4%	0.010	0.022
	Mean	24.1%	0.010	0.015
	Random bs	53.4%	0.008	**0.031**
DBSCAN	Base	104%	0.016	0.016
	Persistency	**313%**	**0.012**	**0.061**
	Regression	149%	0.013	0.042
	Arima	79.4%	0.014	0.027
	Mean	161%	0.014	0.040
	Random bs	2.7%	0.008	0.004
FTSE MIB		-59.6%	0.018	-0.015

Table 1: Results for the Italian market over the training set.

		ROI	σ	SR
NO-Clust	Base	92.4%	0.012	0.033
	Persistency	60.4%	0.009	0.031
	Regression	49.2%	0.009	0.025
	Arima	**100%**	**0.009**	**0.043**
	Mean	75.6%	0.010	0.033
	Random bs	50%	0.036	0.059
K-Means	Base	136%	0.013	0.039
	Persistency	**146%**	**0.009**	**0.056**
	Regression	133%	0.013	0.039
	Arima	**146%**	0.011	0.047
	Mean	140%	0.011	0.04
	Random bs	51%	0.032	0.007
DBSCAN	Base	4.7%	0.015	0.009
	Persistency	57.9%	0.011	0.025
	Regression	12.2%	0.013	0.011
	Arima	**156%**	**0.010**	**0.044**
	Mean	57.8%	0.013	0.022
	Random bs	2.7%	0.008	0.004
FTSE MIB		17.9%	0.013	0.014

Table 2: Results for the Italian market over the validation set.

		ROI	σ	SR
NO-Clust	Base	64.7%	0.015	0.024
	Persistency	**98.0%**	**0.009**	**0.043**
	Regression	80.1%	0.010	0.034
	Arima	52.7%	0.011	0.025
	Mean	73.9%	0.011	0.031
	Random bs	39.6%	0.007	0.026
K-Means	Base	39.4%	0.015	0.019
	Persistency	**99.7%**	**0.009**	**0.044**
	Regression	75.9%	0.011	0.031
	Arima	26.1%	0.012	0.016
	Mean	60.3%	0.012	0.027
	Random bs	16.6%	0.008	0.013
DBSCAN	Base	53.2%	0.016	0.021
	Persistency	101.2%	**0.011**	0.038
	Regression	**127.6%**	0.012	**0.040**
	Arima	43.2%	0.013	0.021
	Mean	81.3%	0.013	0.030
	Random bs	30.0%	0.008	0.023
S&P 500		42.4%	0.014	0.020

Table 3: Results for the American market over the training set.

		ROI	σ	SR
NO-Clust	Base	**151%**	0.012	0.044
	Persistency	108%	**0.007**	**0.053**
	Regression	67.8%	0.009	0.034
	Arima	111%	0.009	0.047
	Mean	110%	0.009	0.044
	Random bs	67.1%	0.006	0.045
K-Means	Base	**130%**	0.013	**0.038**
	Persistency	74.1%	0.011	0.031
	Regression	88.4%	**0.010**	0.035
	Arima	89.7%	**0.010**	**0.038**
	Mean	95.6%	0.011	0.035
	Random bs	50.8%	0.006	0.035
DBSCAN	Base	32.8%	0.015	0.017
	Persistency	42.0%	0.011	0.021
	Regression	**116%**	0.011	0.041
	Arima	**116%**	**0.010**	**0.042**
	Mean	76.7%	0.012	0.030
	Random bs	17.9%	0.008	0.015
S&P 500		141%	0.012	0.043

Table 4: Results for the American market over the validation set.

6 Discussion

Our goal is to choose the strategy with the least variability to minimize risk while also aiming for the highest return on investment, as discussed in the background section. Interestingly, we point out that the two clustering techniques behaved in a different way (DBSCAN filtered more tickers than K-Means), but in the end both obtained positive results in all the tested environments.

6.1 Italian market

Observing the training set, we note a significant market downturn due to the 2008 stock market crash. The market struggles to recover the nearly 60 % points lost in the early years. Our developed pipelines outperform the FTSEMIB index while also exhibiting lower volatility.
Specifically, we observe that DBSCAN consistently constructs a better stock portfolio monthly with respect to other technique. The introduction of the ARIMA filter, helps to mitigate market collapse periods, albeit causing losses during some upward trends.
Regarding K-Means, we can observe that the clustering performed is able to achieve a better collection of stocks compared to the overall market trend. However, the clustering appears to be less effective than the one developed by DBSCAN. The application of filters also seems to yield positive results in this case.

In the validation set, we observe the market growing until 2020, when we have a crash due to the COVID crisis, and then a subsequent recovery. In this context, with hyperparameters optimized on the training set, we notice that all techniques, except for the DBSCAN base, outperform the stock market index, while also exhibiting lower volatility underline the quality and robustness of our pipeline.
In this scenario, models that appear to achieve the best results are those clustering stocks using K-Means, contrary to what was observed in the training set.
The discrepancy between clustering techniques in the training and validation sets suggests a potential shift in the underlying data distribution or characteristics not captured by our data.

However, the implementation of the filters introduced enables us to bypass the market crash

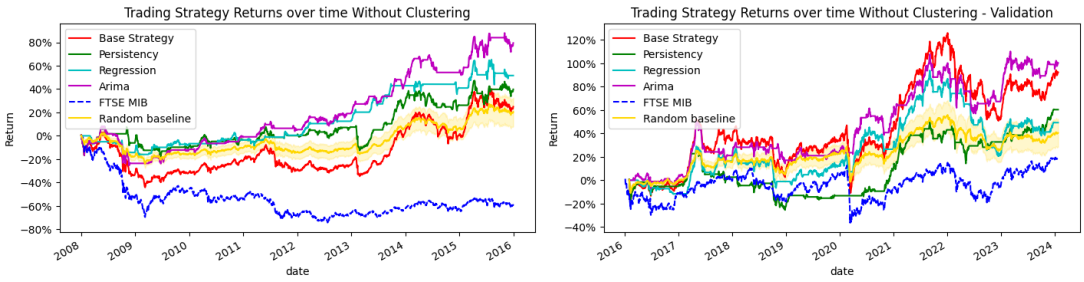

Figure 2: Result over training (left) and validation (right) without any clustering scheme for Italian market.

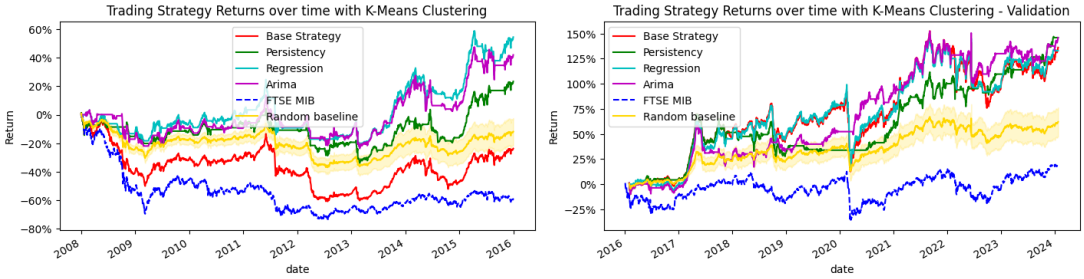

Figure 3: Result over training (left) and validation (right) with K-Means for Italian market.

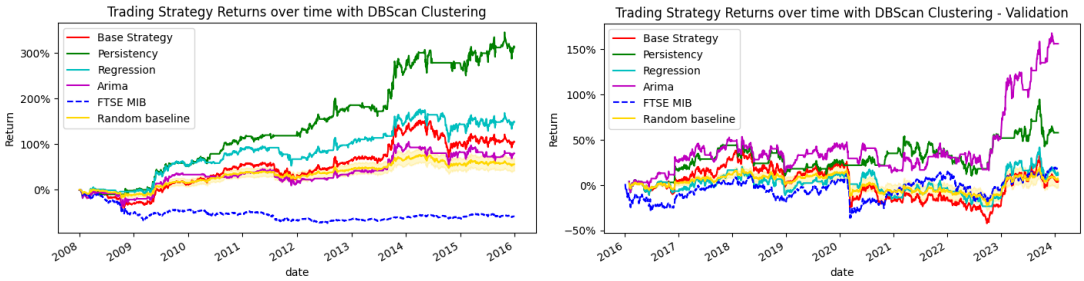

Figure 4: Result over training (left) and validation (right) with DBSCAN for Italian market.

in 2020 and avoid losing the results achieved up to that point.

6.2 American market

It is important to note that this stock index behaves differently compared to its Italian counterpart. Looking at the training data, we can see that the crash in 2008 was significant, but the market was able to recover from this loss in a few years. This can be attributed to the higher capitalization and it is supported by the Zipf's law hypothesis where a high capital market works as a centralizer.

During this years, we observed that the DBSCAN clustering performs similarly to K-Means. It is worth noting that the regression filters and Arima are able to filter out some of the 2008 crash.

In the validation set, we observed that our results show a similar increase to the American market, although they cannot be achieved with any technique we developed and are consistently surpassed by the S&P 500.
This could be due to the structure of the index that includes the top 503 stocks in the American market, and the capital allocated to them depends on their relative capitalization within it. This helps capture financial bubbles caused by the growth of a sector and filters out losses, by moving less profitable companies out of the ranking and replacing them with others with higher recent evaluations. In this scenario, while our ability to filter out global market losses is evident, our clustering method proves to be inefficient in instances where sectors,

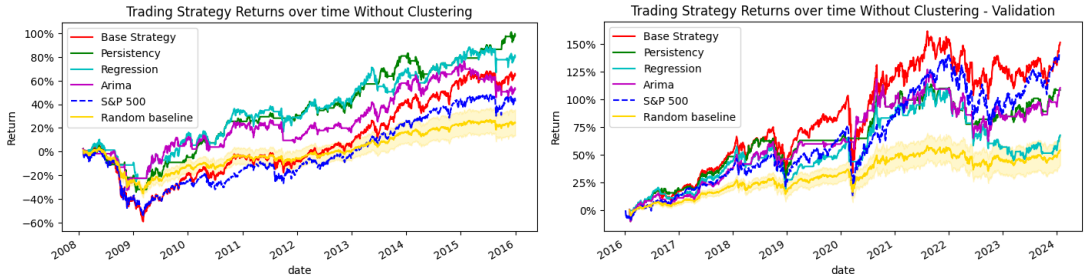

Figure 5: Result over training (left) and validation (right) without any clustering scheme for American market.

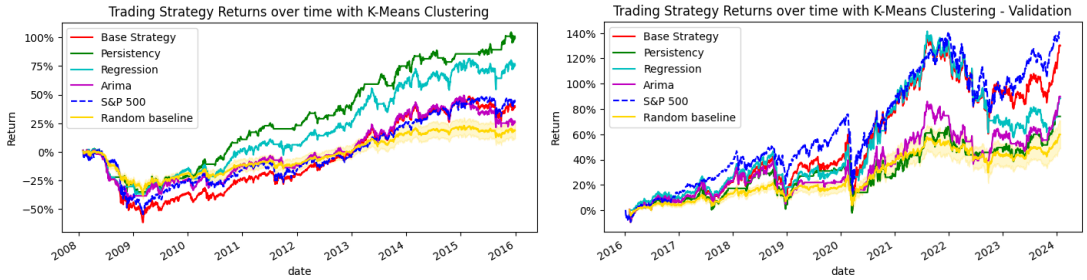

Figure 6: Result over training (left) and validation (right) with K-Means scheme for American market.

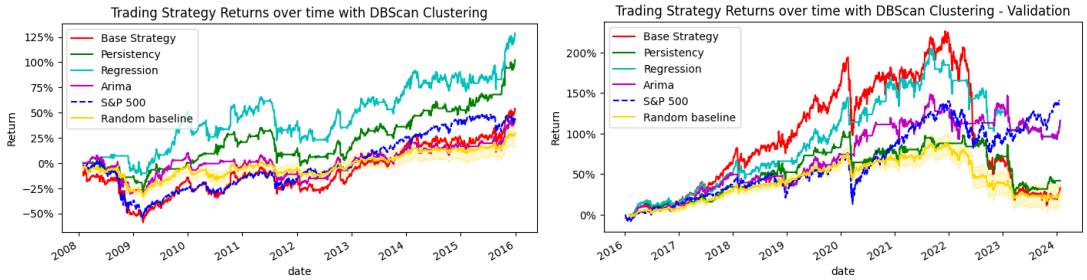

Figure 7: Result over training (left) and validation (right) with DBSCAN scheme for American market.

previously given less consideration, undergo swift growth—a contrast to the index, which adeptly captures such dynamic changes.

6.3 Comparison

There is a noticeable structural difference between the two markets under consideration. The first one exhibits lower dimensionality on the international stage and struggles to recover from collapses, unlike its American counterpart. Moreover, the benchmark indices differ. The FTSEMIB comprises 40 industries listed on the Borsa di Milano market, selected based on factors such as capitalization, sector, and exchange frequency. Such an index demonstrates lower variability compared to the American market, where no sectorial assumption is done, and capital allocation is solely based on their capitalization. In this latter scenario, our algorithm seems to achieve better results on the Italian market. It is important to note that this work does not take into account maintenance costs, buying and selling expenses for each asset, as well as dividend distribution and its taxation.

7 Conclusion

Our study aimed to optimize stock portfolios using machine learning models for both the Italian and the American market. We developed a pipeline that consisted in aggregating financial indicators to select the most promising stocks via a clustering algorithm. Then, we proposed different filters, whose aim was to avoid the market's losses. Our results

showed superior performance compared to the FTSEMIB index in the Italian market and outcomes comparable to the S&P 500 in the American one. However, it is crucial to acknowledge the structural differences between the Italian and American markets. The former one often exhibits higher volatility, which can create opportunities for our strategy to exploit inefficiencies. On the other hand, the U.S. market, being more liquid and efficient, presents a more challenging environment where achieving alpha is generally considered more difficult.
Despite challenges, our approach demonstrates potential for enhancing portfolio management and investment decision-making. Further research could explore refining algorithms and incorporating additional factors for more accurate predictions.
We have not identified a universally superior clustering or filtering technique, as it greatly depends on the market trend and type. However, we have determined the most effective approach for each situation.

8 Further improvements

Considering our lack of technical expertise in the field, the dataset we have constructed might not encompass all the required information to achieve an optimal clustering.

Introducing deep learning models for predictive analysis of the following month could potentially yield better results than those we have obtained.

One constraint of this project is the treatment of clustering and its subsequent optimization as separate processes. Consolidating these components into a unified decision-focused learning model has the potential to improve performance.

9 Links to external resources

GitHub Repository: Portfolio Optimization via Machine Learning Models

References

[1] Arthur O'sullivan, Steven M Sheffrin, and Kathy Swan. Economics: Principles in action. 2003.

[2] Robert M Solow. A contribution to the theory of economic growth. *The quarterly journal of economics*, 70(1):65–94, 1956.

[3] Thomas Schneeweis, Garry B Crowder, and Hossein B Kazemi. *The new science of asset allocation: risk management in a multi-asset world*. John Wiley & Sons, 2010.

[4] Mark Rubinstein. Markowitz's" portfolio selection": A fifty-year retrospective. *The Journal of finance*, 57(3):1041–1045, 2002.

[5] Markus K. Brunnermeier. Lecture slides: Financial economics.

[6] Harry M Markowitz et al. The optimization of a quadratic function subject to linear constraints. *Naval research logistics Quarterly*, 3(1-2):111–133, 1956.

[7] Eugene F Fama and Kenneth R French. Multifactor explanations of asset pricing anomalies. *The journal of finance*, 51(1):55–84, 1996.

[8] Isaac Meilijson. The garman-klass volatility estimator revisited. *arXiv preprint arXiv:0807.3492*, 2008.

[9] ran-Moroan Adrian. The relative strength index revisited. *African Journal of Business Management*, 5(14):5855–5862, 2011.

[10] Nguyen Hoang Hung. Various moving average convergence divergence trading strategies: A comparison. *Investment management and financial innovations*, 2016.