Optimal Asset Allocation using GARCH, Dynamic Conditional Correlation and Machine Learning

Business Project Submission

DIPAN BISWAS
MASTERS IN FINANCE
SPRING TERM 2024

Abstract

Standard deviations and correlations play a pivotal role in portfolio construction and hedging strategies. However, accurately estimating these correlations and volatilities presents challenges, particularly in fluctuating market conditions. This report delves into various methodologies to address this issue, with a specific focus on variants of GARCH and Dynamic Conditional Correlation (DCC) models, and the application of Deep Learning techniques using ResNets and Self-Attention mechanisms. These modern approaches offer flexibility in estimating time-varying conditional covariances. Drawing insights from both academic research and industry practices, the report assesses the effectiveness of different DCC models in constructing Markowitz Minimum Variance Portfolios. The results reveal that portfolios constructed using GARCH and DCC models outperform those relying on traditional covariance measures, highlighting the significance of incorporating dynamic conditional correlation estimates in portfolio management. Furthermore, the report introduces a novel method of forecasting standard deviations and asset correlations using Deep Learning models. This innovative approach can be utilised to pivot from conventional estimation of covariance as linear co-movement to non-linear paradigms. Despite the study's limitations, such as its limited asset pool, the implications of these findings remain relevant for portfolio and risk managers. They provide valuable insights for optimizing portfolio performance and managing risk in the changing landscape of financial markets.

Acknowledgement

I would like to express my sincere gratitude to Professor Marco Grotteria for his invaluable guidance, unwavering support, and profound insights throughout the duration of this research project. Professor Marco's expertise and encouragement have been instrumental in shaping the direction and success of this endeavour. His mentorship has not only enriched my understanding of the subject but has also inspired me to strive for excellence. I am deeply thankful for his contributions and commitment to my academic and professional growth.

Table of Contents

1. Introduction

Volatility and correlation forecasting stand as pillars in the realm of asset management, serving as crucial elements for portfolio optimization, risk assessment, and derivative pricing. In the intricate landscape of financial markets, where uncertainty reigns supreme, the ability to accurately predict future volatility and correlations of asset returns holds immense significance for a multitude of stakeholders, ranging from risk managers and portfolio managers to investors and academicians. The seminal recognition of the importance of volatility forecasting was underscored in 2003 when Professor R.F. Engle was awarded the Nobel Prize for his groundbreaking contributions in modelling volatility dynamics.

Standard deviation, commonly acknowledged as a gauge of volatility and portfolio risk, holds a crucial position in asset pricing theory. The accuracy of volatility estimates serves as a cornerstone in models like the Black-Scholes equation. Furthermore, regulatory mandates such as the Basel Accords mandate that financial institutions integrate volatility and correlation forecasts into their risk management strategies, firmly establishing its indispensable role throughout financial institutions. The broader applications of volatility and correlation forecasting span various domains, including quantile prediction and density forecasting, which are pertinent to risk management, market making, market timing, and portfolio optimization. Stakeholders utilize volatility and correlation forecasts to assess expected portfolio performance, precisely value derivatives, and execute well-informed trading decisions. For instance, volatility and correlation forecasting facilitates evaluation of the risk-return trade-off—a fundamental principle in contemporary trading and investing. A risk manager must ascertain the likelihood of portfolio decline in future, while an option trader seeks insight into expected volatility over the contract's lifespan. Additionally, understanding the volatility and correlation structures aids in hedging strategies. Portfolio managers may opt to divest assets before volatility escalates, and market makers may widen bid-ask spreads in anticipation of heightened future volatility.

This study investigates various modifications to the Generalized Autoregressive Conditional Heteroskedasticity (GARCH) model for volatilities and Dynamic Conditional Correlation (DCC) models, aiming to align with the economic realities of financial markets. The impact of these adjustments on the future performance of stylized portfolios, constructed using forecasted volatility and correlation estimates, is examined. The research finds that GARCH and DCC models exhibit superior performance over the long term due to their capacity to incorporate persistence and autocorrelation in both volatility and correlation. This advantage stems from better model fits achieved through parameter optimization, in contrast to the commonly used rolling historical standard deviation and Pearson correlation techniques, which necessitate arbitrary selection of window length. Additionally, the incorporation of various forms of mathematically imposed asymmetries proves beneficial in better modelling market phenomena such as fat tails, jumps, and regime changes. The effectiveness of these innovations in enhancing the risk performance of stylized portfolios is demonstrated.

Moreover, this study acknowledges that assuming correlation to be a linear relationship may be misleading. Linear models often fail to incorporate convexities and sharp bounces in correlation structures, particularly during tail events. Despite the emergence of modern Machine Learning techniques, research on utilizing Deep Learning for modelling covariance remains scarce. Therefore, the principal contribution of this research lies in developing a Residual Neural Network model with Self-Attention mechanism to capture the evolution of standard deviations and correlations. It is demonstrated that Deep Learning models can better capture correlation dynamics, particularly in automating decisions for swift reactions during tail events or persistence otherwise. By employing intelligent model construction, this study illustrates that forecasting accuracy can be enhanced beyond what is achievable by linear models, leading to better management of risk in portfolios.

2. Literature Review

The literature on volatility modelling and forecasting has undergone significant evolution since the seminal contributions of Markowitz (1952) and Sharpe (1964) in the development of the Capital Asset Pricing Model (CAPM). This framework elucidates how the anticipated return, both in its raw and risk-adjusted forms, across various assets is influenced by the mean and volatility dynamics of the overall market return, as well as the covariance between the market and individual assets. Engle's (1982) groundbreaking work on stochastic volatility ARCH (Autoregressive Conditional Heteroskedasticity) models marked a pivotal moment, albeit originally aimed at measuring inflation uncertainty. The methodologies devised for modelling volatility dynamics have found applications across a diverse array of disciplines, including economics, social sciences, natural sciences, and medicine. The interest in volatility modelling and forecasting surged following Engle's ARCH paper, which laid the foundation for modelling volatility as a time-varying function of current information. Subsequently, the GARCH family of models, with GARCH(1,1) being the most widely utilized, was introduced by Bollerslev (1986) and independently discussed by Taylor (1986).

The literature presents various adaptations and extensions of the GARCH model. For instance, studies have indicated that QGARCH performs admirably in predicting stock return volatility in several European countries (Franses 1996), while GJR-GARCH has been identified as superior in forecasting volatility in the Australian market (Brailsford, 1996). However, the selection of the most suitable GARCH specification often varies depending on the specific context, as evidenced by contradictory findings in different studies.

One major critique of GARCH revolves around their inability to generate accurate forecasts, particularly given that volatility is unobservable ex-post. Andersen (1998) argues that the apparent predictive failure of GARCH models is attributable to inherent noise in the return-generating process.

More than a decade after the popularity of the first ARCH model, researchers identified asymmetries in correlations among stock indices under dynamic conditions, accompanied by fatter tails in return distributions. This phenomenon suggests that traditional GARCH models alone may underestimate or overestimate return forecasts (Boyer, 1997). In response to these challenges, Engle and Kroner (1995) introduced the BEKK formulation, offering a general quadratic form for conditional covariance equations. However, parameter estimation complexity remained a challenge. Alexander (2000) proposed Factor GARCH models as an alternative for estimating large covariance matrices, although interpretation difficulties persisted. Engle (2002) proposed the Dynamic Conditional Correlation (DCC) model as a simple solution, which has subsequently become very popular. The DCC model enables the modelling of conditional correlation structures, providing insights into market synchronization and volatility clustering in financial series. The GARCH-DCC methodology has emerged as a prominent paradigm in covariance modelling due to its flexibility and ease of estimation. This approach involves modelling conditional variances using GARCH and conditional correlation matrices using the DCC model. Cappiello (2006) extended the DCC model to incorporate conditional asymmetries in volatilities and correlations. Haas (2004) integrated Hidden Markov States into GARCH models to capture asymmetries, while Jondeau (2006) presented a copula-based methodology for measuring conditional dependency in GARCH models of correlated assets.

These advancements underscore the ongoing efforts to enhance the accuracy and robustness of standard deviation and correlation forecasting techniques. By incorporating dynamic effects and asymmetries into models, researchers aim to provide more reliable estimates of volatility and correlations, ultimately facilitating better-informed decision-making in financial markets.

3. Data and Methodology

For this analysis, we develop monthly covariance projections for five equity indices: the S&P 500 of the USA, Nifty 50 of India, Hang Seng of Hong Kong, Stoxx 600 of Europe, and IBovespa of Brazil. These indices are selected to ensure a robust representation of both developed and emerging markets, with significant coverage across major industrial zones. Additionally, the selection is guided by the understanding that diversified portfolios, including those following strategies like 60/40 and roll down portfolios, typically incorporate exposures to both developed and emerging markets. The period of our study is from January 2008 to December 2023.

	BOVESPA	SP500	HSI	NIFTY50	STOXX
Annual Return	5%	8%	-3%	8%	2%
Standard Deviation	23%	16%	22%	21%	16%
Sharpe	0.19	0.46	-0.15	0.38	0.10
Sortino	0.06	0.13	-0.05	0.11	0.03
Max Drawdown	50%	48%	55%	47%	46%

Table 1: Performance measures of indices under consideration from 2008 to 2023

As our reference points for the standard deviation of the indices and their correlation, we utilize the unconditional standard deviation and Pearson correlation between January 2000 and December 2007. These values remain constant and are not updated. They serve as benchmarks for creating stylized portfolios from 2008 to 2023. By employing dynamic models of standard deviations and correlations, we will compare and elucidate advantages over these commonly used constant unconditional values.

For each month post June 2008, we create a 1-month forward projection of the covariance matrix of the 5 indices. We use GARCH(1,1) models on daily returns data of each index to independently model the volatility distributions. Then, we utilise the results as inputs into variants of Dynamic Conditional Correlation models (as described in Section 3.1) to generate 1-month forward correlation matrices.

Alternatively, we also use 6-month rolling standard deviation and 6-month rolling Pearson correlation as 1-month forecasts. These don't require parameter optimisation unlike GARCH and DCC models but assumes an arbitrary window length. They are similar to unconditional estimate but use shorter history and are updated every month. The covariance matrices calculated based on these modelled standard deviations and correlations are then employed in stylised portfolio construction, discussed later. A brief overview and mathematical formulation of the GARCH and DCC models are provided below.

3.1 Methodology

In this section, I will go over the brief theory and mathematical formulation of the conditional standard deviation and correlation models. Further details on these models are provided in the appendix.

Generalized Autoregressive Conditional Heteroskedascity (GARCH): GARCH models are built on the concept of conditional heteroskedasticity in financial time series, where variance changes over time based on past information. They capture how past volatility influences future volatility, reflecting the clustering and persistence of volatility seen in financial data. These models adjust forecasts in response to volatility spikes, with components addressing volatility shock persistence and the impact of past errors on future volatility. The standard GARCH(p, q) model can be broken down to the autoregressive component and the moving average component.

Mathematically, $\sigma_t^2 = \omega + \sum_{i=1}^{p} \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2$

where σ_t^2 represents the conditional variance of the financial time series at time t, ω denotes the intercept term capturing the long-term average volatility level, α_i and β_j are the parameters of the model that govern the persistence of volatility shocks, ε_{t-i}^2 represents the squared residual (error term) at time t-i, σ_{t-j}^2 represents the conditional variance at time t-j. The GARCH model's persistence, determined by the sum of α and β, indicates the longevity of variance deviations from its long-term average. High persistence, seen when α + β approaches 1, suggests that deviations in variance from its long-term average will endure for an extended period, while low persistence, approaching 0, implies spikes in volatility dissipates quickly.

Dynamic Conditional Correlation (DCC): DCC models extend the GARCH framework to capture time-varying correlations among financial assets. DCC models adapt to changes in correlation patterns, providing a dynamic framework for understanding asset interrelationships. Mathematically, DCC models comprise univariate GARCH models for individual asset returns and a dynamic conditional correlation estimator, allowing for the evolution of correlations over time.

For the GARCH model represented by $h_{i,t} = E_{t-1}(r_{i,t}^2)$, $\quad r_{i,t} = \sqrt{h_{i,t}}\varepsilon_{i,t}$

We represent the covariance matrix by $H_t = D_t R_t D_t$ where $D_t = \text{diag}(\sqrt{\sigma_{1,t}^2}, \sqrt{\sigma_{2,t}^2}, \ldots, \sqrt{\sigma_{n,t}^2})$

To estimate R_t, we first find Q_t as

$$Q_t = S(1-\alpha-\beta) + \alpha(\varepsilon_{t-1}\varepsilon_{t-1}') + \beta Q_{t-1}$$

where S is unconditional correlation matrix of the epsilons or historical average correlation considered to be the long-term mean. To ensure correlation fall between 0 and 1, we find R_t as

$R_t = (Q_t^*)^{-1} Q_t (Q_t^*)^{-1}$ where Q_t^* is a diagonal matrix of Q_t. (see appendix 7.2 for further details)

Asymmetric Dynamic Conditional Correlation (ADCC): ADCC is based on the theory that second moment of equity returns often displays asymmetricity, rising more after negative shocks in returns. To address this, we extend Engle's DCC GARCH model to accommodate conditional asymmetries in volatilities and correlations. This model incorporates series-specific parameters and adjusts univariate volatility parameterizations. Economically, two theories explain asymmetric volatility: the leverage effect and volatility feedback. The leverage effect suggests that after a stock's value decline, the firm's debt-to-equity ratio increases, leading to higher non-leveraged portion volatility. Alternatively, volatility feedback suggests the anticipation of higher future volatility prompts investors to sell, causing markets to decline, leading to larger volatility increases following negative shocks. We capture this in ADCC using the following equation (see appendix 7.3 for further details):

$$Q_t = (\bar{P} - a^2\bar{P} - b^2\bar{P} - g^2\bar{N}) + a^2\varepsilon_{t-1}\varepsilon_{t-1}' + g^2 n_{t-1}n_{t-1}' + b^2 Q_{t-1}$$

where $a^2 + b^2 + \delta g^2 < 1$, $n_t = I[\varepsilon_t < 0]$ and $\delta = \text{maximum eigenvalue } [\bar{P}^{-1/2}\bar{N}\bar{P}^{-1/2}]$

Markov Switching GARCH – DCC (MSGARCH-DCC): Markov Switching GARCH (MSGARCH) models are adept at capturing the complexities of financial data. They incorporate non-linear specifications and regime-dependent behaviour, where different states influence a time series' evolution. By utilizing a hidden Markov chain, MSGARCH models adapt to different regimes, reflecting varying volatility levels. For example, during market stress, volatility increases, indicating higher uncertainty, while stability leads to decreased volatility. These models offer valuable insights into financial market dynamics, aiding in more accurate volatility forecasts and risk management strategies. In this study, we assume states of each index is independent and use these univariate state-dependant GARCH models as inputs into the DCC model. (see appendix 7.4 for details)

The following is a mathematical representation of a MSGARCH(1,1) model.

$$\sigma_t^2(s_{1:t};\varphi_{t-1}) = \omega_{s_t} + \alpha_{s_t} a_{t-1}^2 + \beta_{s_t} \mathbb{E}\left[\sigma_{t-1}^2(s_{1:t-1};\varphi_{t-2})\middle|s_t;\varphi_{t-1}\right]$$ where s_t represents the states

Copula GARCH (cGARCH): Copulas are mathematical tools used to model the dependency structure between random variables. Unlike traditional methods that focus on the multivariate distributions, copulas allow us to separate the marginal distributions from the dependency structure, providing a flexible framework to capture complex dependencies. At its core, a copula is a function that describes the joint distribution of random variables based on their independent marginal distributions. Examples include the Gaussian Copula, the Student-t Copula for heavier tails, and Archimedean Copulas such as Clayton, Gumbel, and Frank Copulas, each characterized by unique parameters determining dependency strength and type.

$$C(u_1,u_2;\rho) = \Phi_\rho\left(\Phi^{-1}(u_1),\Phi^{-1}(u_2)\right)$$ where C is the copula function. (see appendix 7.5 for details)

For this study, we use Student-t copula modelled on marginal distributions derived from independent GARCH models of indices. Previous research has demonstrated the ability of Student-t copula to capture tail dependencies between assets better than the Gaussian copula and hence, justified its applicability in analysing asset volatilities.

The parameters of the GARCH and DCC models are optimized to fit the data using SLSQP optimization (refer to the appendix 7.6). We obtain 1-month volatility and correlation forecast for each index and index-pair respectively at the beginning of each month with these models and analyse them.

3.2 Deep Learning – Data Preparation and Model Construction

We create the Deep Learning model with the help of a Residual Network (ResNet) architecture. We enhance the regime detection ability of the ResNet with Self-Attention mechanism.

ResNets address the vanishing gradient problem encountered in traditional deep neural networks by introducing skip connections between layers. These connections facilitate direct information flow and preserve gradient signals, enabling the training of deeper networks with improved convergence. By integrating residual blocks with skip connections, ResNets effectively capture both shallow and deep features from the data and thus, any non-linearity or deep dependency. (see appendix 7.8)

The Attention Mechanism plays a crucial role in modern deep learning, especially for tasks involving sequential or structured data. It allows models to selectively focus on different parts of the input, akin to human cognitive processes. Using a scoring function, it calculates attention weights for each element in the sequence, which are then normalized to represent importance. By dynamically adjusting these weights, the model can effectively process and learn from relevant information. Self-attention, a variant, enables input elements to attend to other input elements, aiding in modelling long-range dependencies. (see appendix 7.9)

Integrating self-attention into ResNets offers a promising path for financial feature extraction. This approach leverages attention to identify distinct financial regimes. By applying attention weights to the output units of residual blocks after activation functions, feature selection becomes more nuanced. Placing attention within the block enhances the model's ability to capture essential signals, enabling deeper learning and adaptability to evolving market conditions. Unlike static models like Markov-switching models, the Attention-ResNet framework acknowledges the dynamic nature of financial markets. Regime probabilities are not fixed but dynamically inferred from input variables, allowing the model to capture subtle market shifts accurately. This adaptive approach offers a responsive and accurate representation of financial dynamics (refer to the appendix 7.10 for further details).

Figure 1 represents schema of the Neural Network architecture used. One ANN model is fitted to predict volatilities of the indices and one to predict correlations of index pairs.

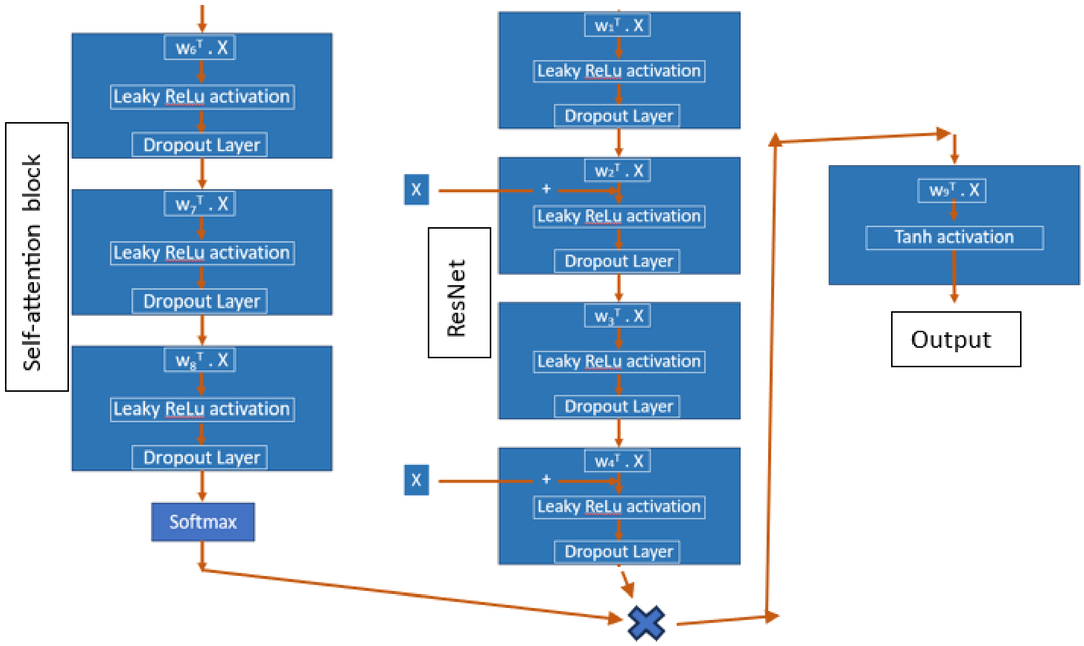


Fig 1: Schema of Self-attention ResNet architecture used to model standard deviations and correlations

The neural network architecture comprises layers with 32 neurons, except for the final layer, which contains 4 neurons. Activation functions are selected based on the output requirements, with Tanh used for correlation prediction and ReLU for standard deviation estimation, ensuring non-negative outputs. To mitigate overfitting, a dropout rate of 0.2 is implemented, randomly setting 20% of the weights to zero, and cross-validation sets comprising 20% of the training data are used during each epoch. The Adam optimizer is employed for gradient descent, offering fast convergence initially and gradually slowing down as the weights approach optimal values, thanks to its adaptive weight updation mechanism. Moreover, the learning rate decays exponentially during training, prioritizing gradient updates at the start and gradually reducing their impact to fine-tune the model. Data preprocessing involves creating various features for both individual indices and index pairs, including daily returns, squared returns, rolling means, and correlations. For the standard deviation model, the target variable is the next 60 days' actual standard deviation, while for the correlation model, it's the next 60 days' actual correlation coefficient. The dataset spanning from 2008 to 2023 is divided into training data from 2008 to 2018 and testing data from 2019 to 2023. This division ensures that the model's performance is assessed on an independent testing period, allowing for the evaluation of its generalizability. Finally, the predicted standard deviations and correlations are leveraged to construct stylized portfolios for each month in the testing period.

3.2 Stylized Portfolio Construction

The stylised portfolio that we are going to create with the indices is the Markowitz Minimum Variance Portfolio (MVP), an extension of Modern Portfolio Theory, for crafting portfolios aimed at minimizing risk exposure (refer to the appendix 7.11 for further details). Since the overarching goal of forecasting standard deviations and correlations is to accurately estimate portfolio risk, comparing the realized standard deviation of MVPs for different models will enable us to compare the strength of each model. We would deem a portfolio better if it has lower risk, irrespective of the returns. We create MVPs for modelled standard deviation and correlations at the beginning of each month from June 2008, hold them for one month, record the returns and then rebalance them with fresh forecasts for next month.

4. Results and Discussion

4.1 Standard Deviation models

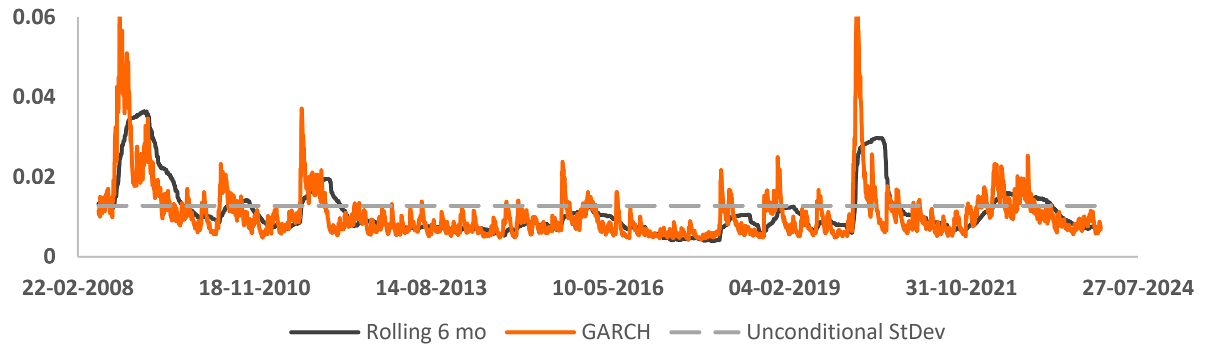


Fig 2: Daily forecast of S&P 500 Standard Deviation using rolling 6-month standard deviation, unconditional standard deviation estimate and GARCH model

Figure 2 above illustrates the daily standard deviation estimate of S&P500 derived rolling 6-month and GARCH models. The horizontal line represents the unconditional standard deviation experienced between 2000 and 2007. Analysing both the 6-month rolling and GARCH models reveals that the standard deviation exhibits characteristics of both autoregressive and mean-reverting processes. Assuming unconditional standard deviation is the long-term average, it becomes apparent that short-term standard deviation more or less oscillates around this mean. Additionally, the presence of alternating periods of high and low volatility supports the notion of persistence or autoregressive behaviour.

A notable disparity between the Pearson and GARCH models lies in their responsiveness to volatility spikes. The GARCH model demonstrates swift reactions to such spikes, whereas the rolling 6-month model exhibits a slower incorporation of increased volatility. This discrepancy implies that the GARCH model holds an advantage in scenarios where frequent portfolio rebalancing is necessary or when the holding period is short, as the rolling 6-month model may not accurately assess risk within such contexts.

Furthermore, it becomes evident that the unconditional standard deviation often proves to be an inadequate estimator of risk. Portfolios frequently exhibit behaviour divergent from what the unconditional standard deviation would predict, highlighting the stochastic nature of standard deviation and the necessity for meticulous modelling techniques.

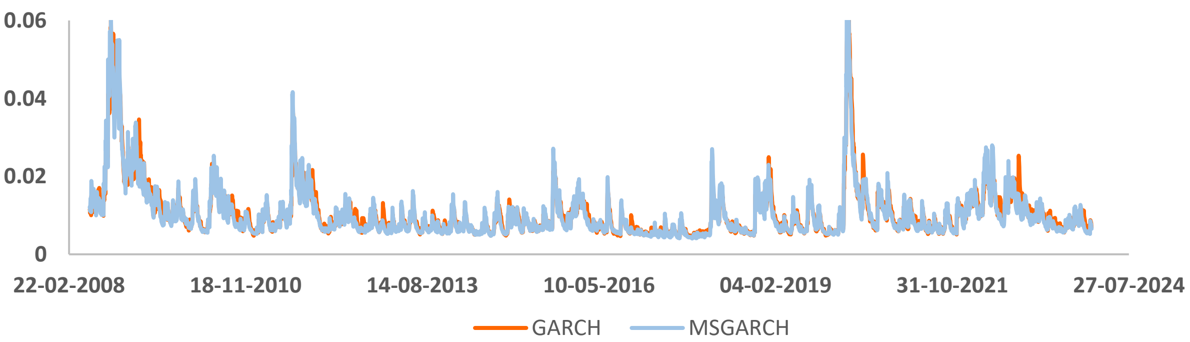


Fig 3: Daily forecast of S&P 500 Standard Deviation using GARCH and MS-GARCH models

Figure 3 juxtaposes the daily standard deviation estimates of the S&P 500 obtained from GARCH and MSGARCH models. Although they share similarities, subtle distinctions are noticeable. There are instances where the volatility levels and directions derived from GARCH and MSGARCH models diverge. This discrepancy arises because while GARCH models a single state, MSGARCH calculates its estimates based on the expected values in two states and the transition probability between them. Consequently, the MSGARCH model incorporates regimes, assigns weights to them, and then determines the standard deviation estimate within those regimes. The probability-weighted average of these regimes yields the estimated standard deviation, occasionally leading to deviations from the GARCH model, which operates within a single regime. Importantly, these regimes are purely statistical and lack economic causality, making it challenging to attribute economic significance to them. The effectiveness of these regimes will be assessed based on the performance of the stylized portfolios created using these models.

The GARCH estimates for all five indices are utilized in DCC, ADCC, and cGARCH models to estimate correlation. Conversely, the MSGARCH estimates are input into a separate DCC model to derive correlation estimates for the MSGARCH-DCC model.

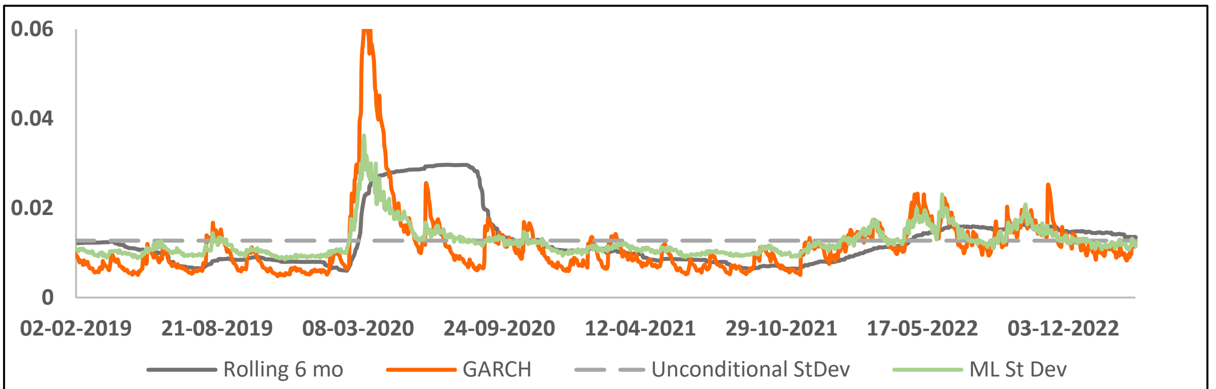


Fig 4: Daily forecast of S&P 500 Standard Deviation using unconditional estimate, rolling 6 month, GARCH and Machine Learning models

Figure 4 above juxtaposes standard deviation estimates derived from rolling 6-month, GARCH, and Machine Learning (ML) models. It's important to note that this chart pertains to data from 2019, as our ResNet model's testing period spans from 2019 to 2023. A notable observation is that the ML model exhibits fewer peaks and troughs compared to the GARCH model, displaying more persistent and smoother features. Although reactive, the spikes in the ML model are not as pronounced as those in the GARCH model, suggesting that the ML model reacts less severely to changes in standard deviation while predicting future values. Additionally, although spikes in both the GARCH and ML models occur simultaneously, they dissipate in a distinct manner, indicating differing treatment of autocorrelation and mean reversion.

In contrast to the GARCH vs. MSGARCH comparison, which revolves around regime disagreements, the disagreement between ML and GARCH lies in the level of standard deviation. The ML model consistently predicts either higher or lower standard deviation for extended periods without converging with the DCC model. The implications of these disparities will become evident as we analyse the performance of stylized portfolios.

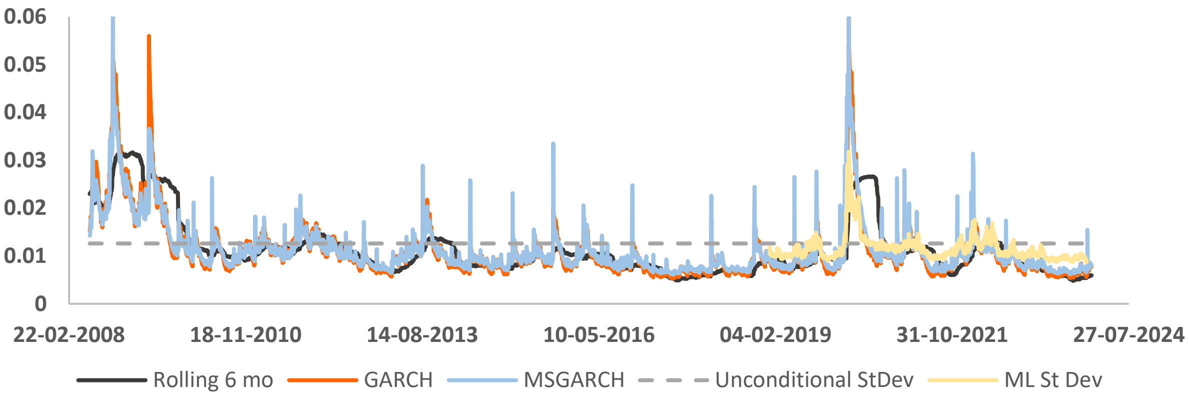


Fig 5: Daily forecast of Nifty 50 Standard Deviation using unconditional estimate, rolling 6 month, GARCH, MSGARCH and Machine Learning models

Figure 5 illustrates the standard deviation estimates of the Nifty 50 obtained from different models. Similar observations can be made as with the S&P 500. Notably, the influence of regimes is evident in the estimates derived from the MSGARCH model. This particular model indicates that regime transitions occur in the Nifty 50 with greater frequency compared to the S&P 500. Once more, the implications of this modelling approach will become apparent when we examine the performance of our stylized portfolios. We can create similar time series for the other 3 indices.

4.2 Correlation models

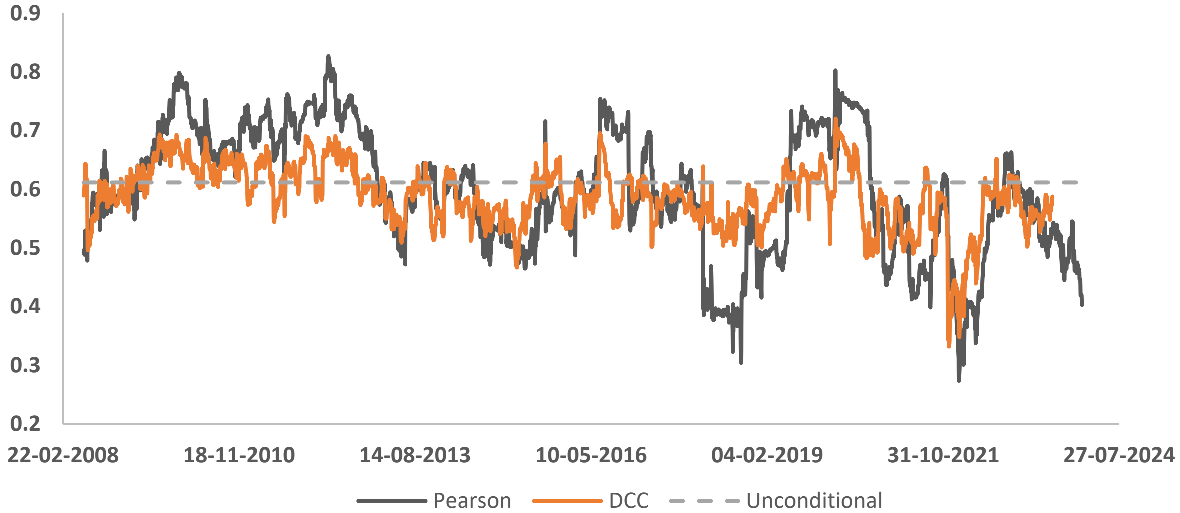


Fig 6: Daily forecast of S&P 500 and STOXX Correlation using rolling 6-month Pearson, unconditional Pearson estimate and DCC model

Figure 6 illustrates the daily correlation estimates between the S&P 500 and STOXX 600, derived from rolling 6-month Pearson and DCC models. Both the Pearson and DCC estimates exhibit cyclical patterns, oscillating around the unconditional correlation value. This behaviour, akin to standard deviation, underscores the autoregressive, mean-reverting and overall stochastic nature of correlation.

The Pearson correlation estimate appears to have sharp jumps, likely due to its sole reliance on the past 6 months of data without consideration for long-term mean or evolution. Moreover, the arbitrary choice of the rolling window may lack relevance to the actual correlation evolution, devoid of any autoregressive component or memory of past behaviour.

In contrast, the DCC model demonstrates smoothness and greater stability for the index pair. Given that both the S&P 500 and STOXX are indices of developed economies with strong daily unconditional correlation, it is improbable for very similar investors to suddenly change their behaviour drastically across these two indices. Therefore, the DCC correlation model is expected to yield more stable portfolios with lower turnover. However, since Pearson model is fully dependant on short-run dynamics, it may be more suitable for portfolios with very short holding periods. The ramifications of these disparities on the performance of stylized portfolios will be explored in a subsequent section.

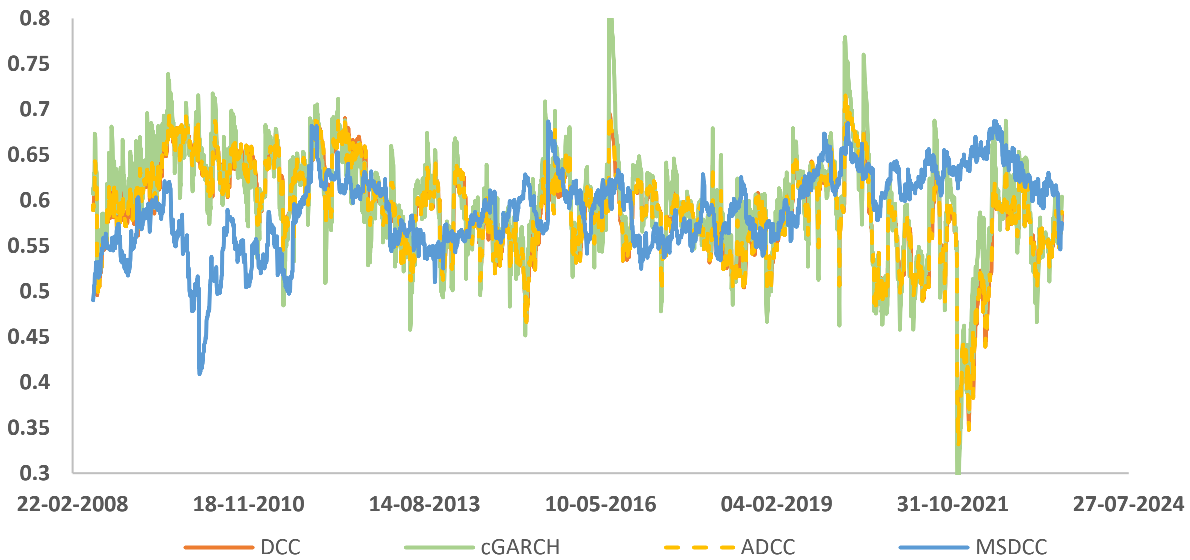


Fig 7: Daily forecast of S&P 500 and STOXX Correlation using DCC, ADCC, MSGARCH and cGARCH models

Figure 7 compares the daily correlation estimates between the S&P 500 and STOXX derived from various models - DCC, ADCC, cGARCH, and MSGARCH-DCC. Notably, the DCC and ADCC models exhibit overlap, indicating that the parameter capturing asymmetry in the ADCC model lacks statistical significance. This suggests that the model fails to discern meaningful differences in correlation behaviour during periods of negative shocks versus positive shocks. This implies that financial regimes might possess nuances beyond simply distinguishing between negative and positive returns regimes.

The MSGARCH-DCC model, on the other hand, establishes regimes and estimates correlation evolution by fitting probability-weighted standard deviations on a DCC model. While this approach can introduce contrarian and directionally opposite movements compared to DCC models, the drawback is that the regimes identified by MSGARCH-DCC are machine-learned and thus lack economic interpretability. However, despite this limitation, the chart reveals significant shifts in behaviour within similar mathematical frameworks of the DCC models.

The effectiveness of these regimes in enhancing the predictability of correlation and yielding portfolios with reduced risk will be assessed through stylized portfolio analysis. This evaluation aims to validate the hypothesis that financial time series are not smooth functions and are interspersed with jumps.

Finally, the cGARCH model demonstrates a similar evolution to DCC and ADCC but with more pronounced spikes. As previously mentioned, the utilization of the Student-t copula in cGARCH renders it more sensitive to tail movements than the Gaussian copula. Consequently, cGARCH may serve as a better model for assessing tail scenarios, Value at Risk, and conditional VaR measures.

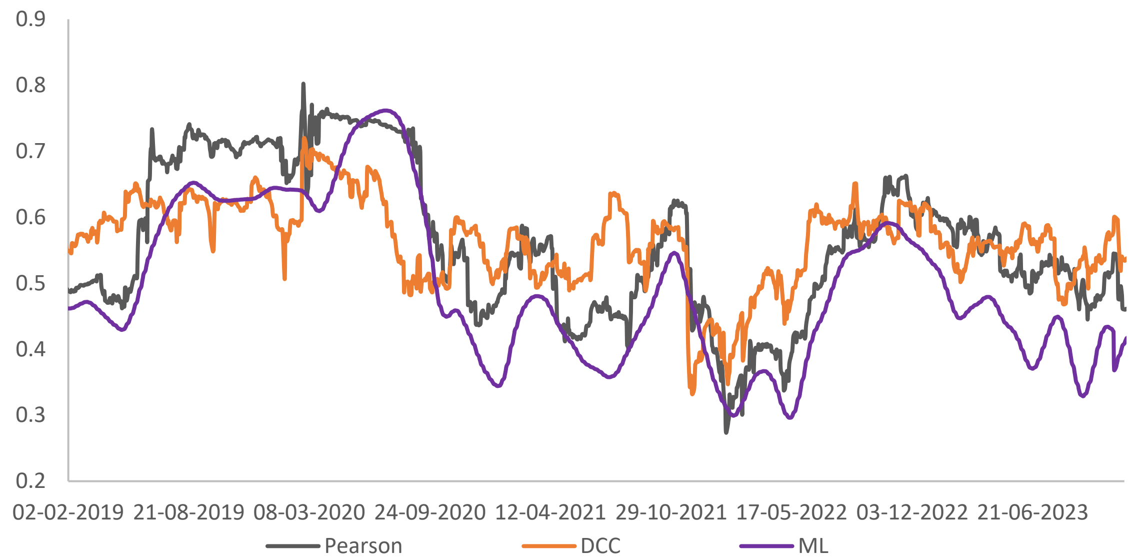


Fig 8: Daily forecast of S&P 500 and STOXX Correlation using rolling 6 month Pearson, DCC and Machine Learning models

Figure 8 juxtaposes correlation estimates obtained from rolling 6-month Pearson, DCC, and Machine Learning models. It's crucial to acknowledge that this chart corresponds to data from 2019, as the testing period for our ResNet model spans from 2019 to 2023. Notably, the correlation estimates from the ResNet model appear smoother compared to both Pearson and DCC estimates. Throughout most of this timeframe, the Machine Learning estimate also tends to be lower than the estimates from linear models. It's widely recognized that the activation functions utilized in Neural Networks effectively capture non-linearities, discontinuities, and jumps in data. Consequently, the Machine Learning model suggests that non-linearities in financial return time series exert a diminishing effect on net correlation, implying that the idiosyncratic movement of each index exceeds what linear models suggest. It would be intriguing to investigate whether this behaviour persists on longer timeframes, such as monthly and annual, but the insufficient historical data prevents us from fitting data-intensive Machine Learning models on higher timeframes. Similar to the Machine Learning model's approach to standard deviation, the self-attention framework implemented in the ResNet model dynamically models several regimes and weights of attention, dynamically adjusting them in each time period forecast. Once more, the implications of these disparities with the DCC models become apparent as we analyse the performance of stylized portfolios.

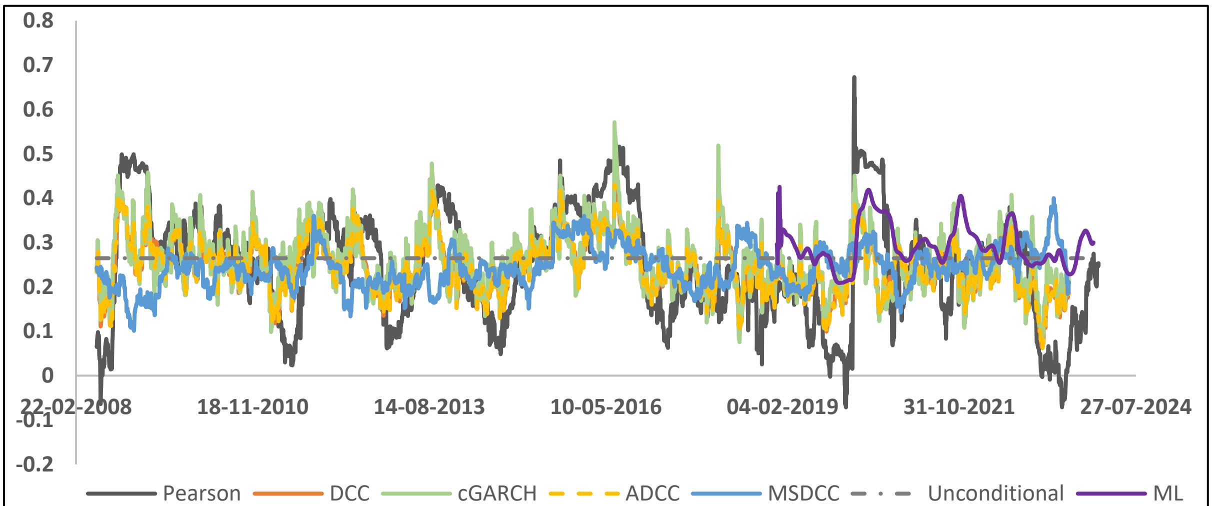


Fig 9: Daily forecast of S&P 500 and Nifty 50 Correlation using various models

Figure 9 presented above showcases correlation estimates between the S&P 500 and Nifty 50 derived from various models. Similar observations can be drawn compared to the analysis of the S&P 500 and STOXX 600. Notably, only the Pearson correlation estimate approaches zero and occasionally dips into negative territory. However, this does not align with the growing trend of globalization, which increasingly intertwines economies and financial markets. It seems highly unlikely for two equity indices to exhibit very low or negative correlations for an extended period. Consequently, the DCC and ML models provide correlation estimates that better reflect the underlying relationship between the indices, closer to actual market dynamics. Furthermore, if such low or negative correlations were indeed accurate, portfolios would experience significant diversification effects and consequently exhibit low net volatilities. The validity of this hypothesis will be tested through the performance analysis of stylized portfolios in the next section.

4.3 Minimum Variance Portfolios (MVP)

After examining the standard deviation and correlation forecasts from various models, we now shift our focus to analysing how these forecasts impact a stylised portfolio comprising the 5 indices. In our analysis, we refrain from attempting to forecast or optimize returns. Instead, our portfolios prioritize risk management and remain indifferent to returns. Consequently, we opt to construct minimum variance portfolios that impose no constraints on returns but aim to minimize risk, measured as the standard deviation of the portfolios observed between 2008 and 2023. A covariance model that accurately reflects the financial realities of the market should be capable of generating the best forecast of future covariance. Such a model can then be utilized to derive weights for the 5 indices, resulting in portfolios characterized by minimal risk.

We have a closed form solution for weights of Markowitz Minimum Variance portfolios:

$$w = \frac{\Sigma^{-1}\mathbf{1}}{\mathbf{1}'\Sigma^{-1}\mathbf{1}}$$

where Σ is the forecasted covariance matrix of indices and **1** is an identity matrix.

The expected returns generated by the portfolios is therefore $\mu_\pi = \mu' w$ and the portfolio risk is given by $\sigma_\pi^2 = w'\Sigma w$

As we can see, the formula for weights of MVP is completely independent of the returns of the assets and depends only of the covariance matrix of the index returns.

The covariance matrix for each period is given by

$$\begin{bmatrix} Var(x_1) & \cdots\cdots & Cov(x_n,x_1) \\ \vdots & \ddots & \vdots \\ Cov(x_n,x_1) & \cdots\cdots & Var(x_n) \end{bmatrix}$$

where $x_1,\ldots,x_n$ are assets.

The variance terms for our MVP is the 1-month forward squared standard deviations that we obtain from our standard deviation models. The Covariance (x_i, x_j) = Correlation (x_i, x_j) * $Var(x_i)$ * $Var(x_j)$, where the correlation values are derived from the corresponding correlation models.

As mentioned before, at the beginning of every month from June 2008 to December 2023, we generate our 1-month forward covariance matrix forecast, generate the MVP weights from the formula above and create the MVP with those weights. At the end of the month, we record the returns generated by our portfolio. We repeat this exercise every month.

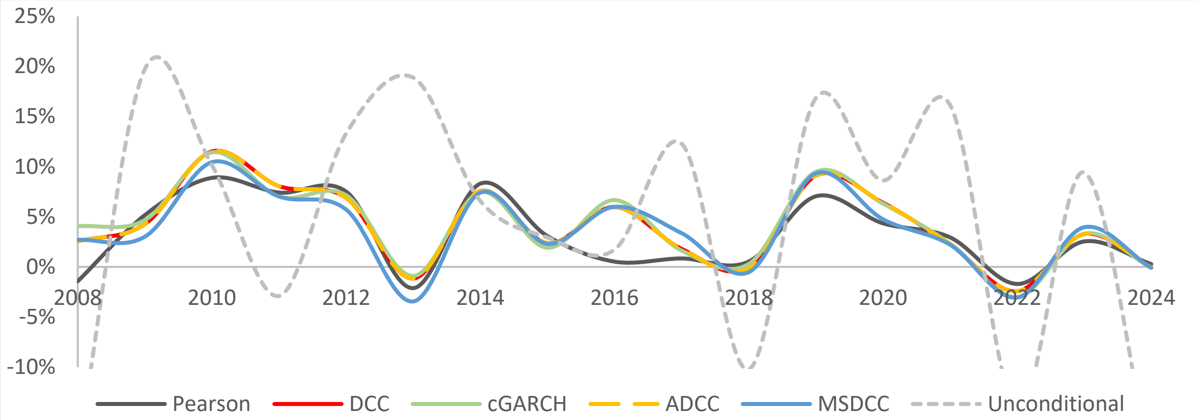


Fig 10: Annual Returns of the Minimum Volatility Portfolio of 5 indices rebalanced monthly

Figure 10 illustrates the annual returns of each portfolio over the years. It is evident that all the return profiles exhibit cyclical patterns. Through naïve observation, we can say that the unconditional estimates of standard deviation and correlation creates very volatile portfolios and does not fulfil our objective of minimizing volatility. On the other hand, while all the conditional models display similar evolutionary trends, notable deviations occur in the Pearson MVP portfolios during certain years (2008, 2016, and 2019). This highlights that there is a fundamental difference between the Pearson model and the DCC model, as we hypothesised in the earlier sections. It suggests that if we were to forecast or simulate asset prices or returns, the outcomes may significantly differ between Pearson-based and DCC-based models. This distinction is crucial as asset managers often simulate correlated returns to construct efficient frontiers and determine the allocation of standard portfolios such as 60/40, Aggressive, or Conservative. Additionally, for risk managers at trading desks, the allocation of capital based on Pearson models versus DCC models may yield entirely different return profiles. This underscores the importance of model selection in both portfolio construction and analysis.

2008 - 2023	**Unconditional**	**Pearson**	**DCC**	**ADCC**	**cGARCH**	**MSDCC**
Compunded Return (annualized)	4.2%	3.5%	4.5%	4.5%	4.6%	4.0%
Standard Deviation	16.69%	13.41%	11.95%	11.95%	11.57%	11.09%
Sharpe Ratio	0.10	0.08	0.17	0.17	0.18	0.14

Table 2: Annual returns, standard deviation and Sharpe ratio of MVPs from various models

Table 2 provided above presents the collective performance metrics of the modelled portfolios. It's evident that portfolios based on unconditional correlation exhibit by far the highest actual standard deviation among MVPs. This proves that unconditional estimates are too simplistic and misleading to use when designing portfolios as they don't incorporate the dynamic nature of risk measures.

The Pearson MVP has the next highest standard deviation. It performs better than unconditional MVP because it uses dynamic estimates. However, since it does not incorporate long-run dynamics, it falls short of the DCC models. In contrast, portfolios derived from DCC, ADCC, MSDCC, and cGARCH models all demonstrate standard deviations lower than that of the Pearson portfolio. While these differences may appear minor, it's crucial to recognize that these portfolios are aimed at minimizing volatility. When our objective shifts, particularly towards portfolios positioned around different regions of the efficient frontier with higher expected returns, these differences can become significant. Notably, the volatilities observed in cGARCH and MSDCC portfolios appear to be even lower than those in DCC portfolios. cGARCH achieves this by effectively capturing tail dependencies through the utilization of leptokurtic distributions (Student-t). MSDCC models statistical regimes that may have a similar effect in incorporating leptokurtosis into the underlying model by creating regimes that cater specifically to nature of financial markets during tail events or shocks.

Furthermore, it's evident that Pearson portfolios exhibit both the lower returns and the higher standard deviation than DCC models. This confirms our earlier hypothesis that the low correlations predicted by Pearson correlations were misleading and led to incorrect allocations in equity portfolios. While DCC, ADCC, and cGARCH portfolios demonstrate higher return performance compared to MSDCC, it's essential to note that return expectations were not utilized in portfolio creation and should therefore not be considered in our performance analysis.

Moreover, it's apparent that MSDCC portfolios yield the lowest standard deviations, followed by cGARCH and DCC/ADCC. Additionally, the risk-adjusted ratios are notably superior to those derived from vanilla Pearson correlations. Thus, we have demonstrated that correlations, like returns and volatilities, exhibit inherent persistence that can be effectively modelled by incorporating a long-term mean and lagged values. The models are enhanced by integrating the realities of financial markets, such as asymmetric behaviours in various market regimes and tail events.

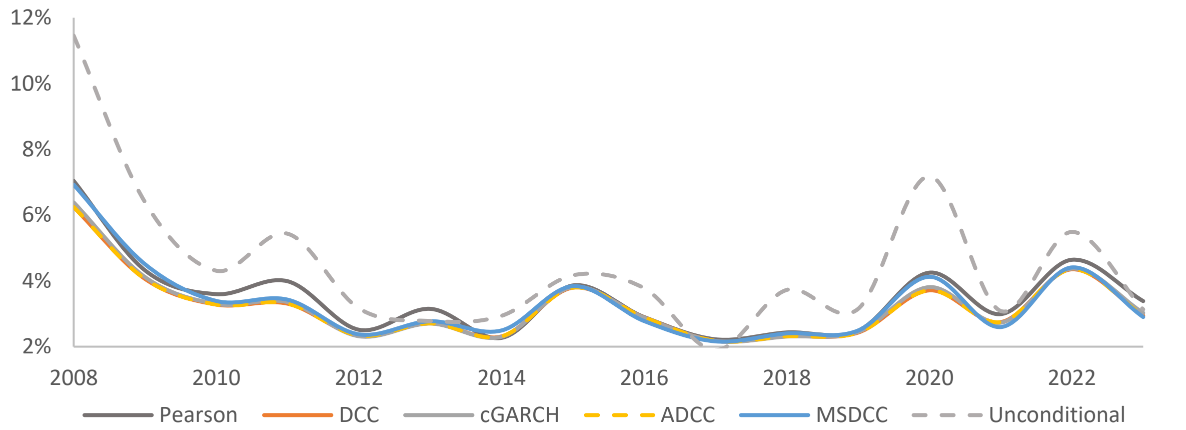


Fig 11: Standard deviation of the Minimum Volatility Portfolio of 5 indices rebalanced monthly

Finally, we come to the analysis of the Machine Learning model. The Table 3 is the performance metrics of the portfolios between 2019 and 2023. We constrict the window of observation for all models because the testing period of the ResNet model being 2019 to 2023, its only fair that it be compared against other models in the same period.

2019-2023	ML	Unconditional	Pearson	DCC	cGARCH	ADCC	MSDCC
Compunded Return (annualized)	13.0%	10.70%	7.0%	8.8%	9.0%	8.8%	8.7%
Standard Deviation	11.85%	16.66%	14.26%	14.18%	13.8%	14.18%	13.66%
Sharpe Ratio	0.89	0.49	0.31	0.44	0.47	0.44	0.46

Table 3: Annual returns, standard deviation and Sharpe ratio of MVPs including Machine Learning (ML) model

A significant improvement in performance is evident when comparing DCC models to Machine Learning approaches. Notably, the returns of the minimum variance portfolio show increase. However, since our optimization did not prioritize returns, we will disregard this aspect in our analysis.

Of particular significance is the remarkable reduction in standard deviation, from 13.66% for MSDCC to 11.85% in the ResNet model. This improvement consequently boosts the Sharpe ratio from 0.46 to 0.89, representing a remarkable increase of nearly 100%. These findings yield several noteworthy conclusions. Firstly, correlations may not consistently adhere to linear patterns, as traditionally hypothesized by Pearson and DCC models. The ResNet model appears adept at capturing and leveraging the intricate relationships between assets, thereby generating more optimal forecasts. With the incorporation of self-attention module, the model is able to dynamically construct regimes and take implicit decisions about which regime to focus on, given inputs about market conditions.

Additionally, the positioning of the minimum variance portfolio towards the left end of the efficient frontier suggests that ResNet portfolios, from both returns and risk perspectives, tend to align closer to the true efficient frontier compared to portfolios generated by Pearson or DCC models. This has tremendous asset management implication. Firstly, for managers, it is evident that there would be a significant gap between the efficient frontier of Pearson or DCC models as compared to that generated by Machine Learning. Secondly, for risk management and hedging, using Machine Learning models may enable them to create superior capital allocation decisions that both increase returns and reduce risk – creating a “model arbitrage”. We acknowledge that this is a preliminary work on the utility of Deep Learning in forecasting correlation. Additional research in model construction and fitting can significantly improve the performance in a way that fits real-world financial dynamics even better.

5. Conclusion

The analysis presented in this report highlights the effectiveness of utilizing GARCH, Dynamic Conditional Correlation (DCC) and Deep Learning (ResNet) models for forecasting standard deviation and correlation, thereby enhancing the performance of Minimum Variance Portfolios in minimising risk. The results highlight the drawback of using simple unconditional estimates as they fail to consider the stochastic nature of risk. Further, portfolios constructed using DCC outperform those based on Pearson correlation when considering risk measures. The essence of this analysis lies in the incorporation of time-varying conditional volatility and dynamic conditional correlation, which enables more accurate estimation of the covariance matrix, consequently enhancing portfolio diversification. It was also observed that standard deviation and correlation exhibit their own evolutionary processes, displaying both autoregressive behaviour with lagged values and mean-reversion around the long-run unconditional mean. Furthermore, the study emphasizes the importance of mathematically integrating financial market realities such as regime dependence and tail reactions, which enhance forecasting accuracy and contribute to superior portfolio construction.

Moreover, this study introduces an innovative approach leveraging Deep Learning models for forecasting correlations and standard deviations, ushering in a new paradigm for constructing markedly more efficient portfolios while integrating improved expectation values of covariance matrices. This innovation stems from Deep Learning models' ability to discern intricate non-linear relationships within time series data, a feat beyond the reach of models like Pearson and DCC, which can only extract linear relationships. Additionally, these models dynamically incorporate factors such as regime dependence, tail reactions, and other higher-order phenomena within financial markets, all without explicit specification in the mathematical structure. These hidden aspects, often overlooked in conventional analyses, are automatically accounted for by Deep Learning models, enriching their predictive power and providing a more comprehensive understanding of market dynamics.

However, it's important to acknowledge the limitations of this study, notably its focus on a limited asset pool and the exclusion of factors like transaction costs, portfolio churns, and liquidity constraints. Future research could expand the analysis by encompassing a wider array of assets and setting constraints on transaction costs and portfolio turnover.

In summary, this study underscores the effectiveness of employing GARCH, DCC and Machine Learning models with Minimum Variance Portfolio strategies in crafting well-diversified portfolios. The implications of these findings are pertinent for investors and risk managers aiming to enhance portfolio performance, mitigate risk and improving effectiveness of hedging. By incorporating time-varying conditional volatility and correlation into the portfolio construction and capital allocation process, finance practitioners can enhance trade execution, better manage risk exposure and generate more realistic expectations of portfolio performance.

6. References

1. Markowitz, H. (1952). Portfolio selection. The Journal of Finance, 7(1), 77–91
2. Sharpe, William F. "Capital asset prices: A theory of market equilibrium under conditions of risk." *The journal of finance* 19.3 (1964): 425-442.
3. Engle, Robert F. "Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation." *Econometrica: Journal of the econometric society* (1982): 987-1007.
4. Engle, Robert F., and Tim Bollerslev. "Modelling the persistence of conditional variances." *Econometric reviews* 5.1 (1986): 1-50.
5. Taylor, Stephen J. "Forecasting the volatility of currency exchange rates." *International Journal of Forecasting* 3.1 (1987): 159-170.
6. Engle, Robert. "Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroskedasticity models." *Journal of Business & Economic Statistics* 20.3 (2002): 339-350.
7. Andersen, Torben G., et al. "Volatility and correlation forecasting." *Handbook of economic forecasting* 1 (2006): 777-878.
8. Bollerslev, Tim. "On the correlation structure for the generalized autoregressive conditional heteroskedastic process." *Journal of time series analysis* 9.2 (1988): 121-131.
9. Engle, Robert F., and Kenneth F. Kroner. "Multivariate simultaneous generalized ARCH." *Econometric theory* 11.1 (1995): 122-150.
10. Alexander, Carol. "Principal component models for generating large GARCH covariance matrices." *Economic Notes* 31.2 (2002): 337-359.
11. Cappiello, Lorenzo, Robert F. Engle, and Kevin Sheppard. "Asymmetric dynamics in the correlations of global equity and bond returns." *Journal of Financial econometrics* 4.4 (2006): 537-572.
12. Cappiello, Lorenzo, Robert F. Engle, and Kevin Sheppard. "Asymmetric dynamics in the correlations of global equity and bond returns." *Journal of Financial econometrics* 4.4 (2006): 537-572.
13. Jondeau, Eric, and Michael Rockinger. "The copula-garch model of conditional dependencies: An international stock market application." *Journal of international money and finance* 25.5 (2006): 827-853.
14. Black, Fischer. "Studies of stock market volatility changes." *Proceedings of the American Statistical Association, Business & Economic Statistics Section, 1976* (1976).
15. Markowitz, Harry M., and Harry M. Markowitz. *Portfolio selection: efficient diversification of investments*. J. Wiley, 1967.
16. Necula, Ciprian. "A copula-garch model." *Economic research-Ekonomska istraživanja* 23.2 (2010): 1-10.

7. Appendix

7.1 Generalized Autoregressive Conditional Heteroskedasticity (GARCH)

Financial assets are characterized by their inherent volatility, making the prediction and management of risk a paramount concern for investors. In this pursuit, GARCH models have emerged as indispensable tools for modelling and forecasting volatility dynamics in financial time series data.

The foundation of GARCH models lies in the recognition of conditional heteroskedasticity in financial time series, where the variance of the series changes over time based on past information. The key insight behind GARCH models is that past volatility affects future volatility, capturing the clustering and persistence of volatility observed in financial data. The autoregressive nature of GARCH models acknowledges that volatility shocks tend to persist over time, reflecting the phenomenon of volatility clustering commonly observed in financial markets. To grasp the intuition behind GARCH models, consider a simple analogy of weather forecasting. Just as meteorologists use past weather data to predict future weather patterns, GARCH models utilize past volatility data to forecast future volatility levels in financial markets. Like weather systems, financial markets exhibit periods of calmness and turbulence, and GARCH models aim to capture these fluctuations. When volatility spikes occur, GARCH models adjust their forecasts, recognizing the impact of past volatility shocks on future volatility levels.

At the core of GARCH models are a set of mathematical equations that govern the dynamics of volatility over time. The standard GARCH (p, q) model comprises two main components: the autoregressive conditional heteroskedasticity (ARCH) component and the moving average (MA) component. The ARCH component captures the persistence of volatility shocks, while the MA component accounts for the impact of past forecast errors on future volatility. The formulation of a GARCH (p, q) model can be expressed as follows:

$$\sigma_t^2 = \omega + \sum_{i=1}^{p} \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2$$

where:

- σ_t^2 represents the conditional variance of the financial time series at time t.
- ω denotes the intercept term capturing the long-term average volatility level.
- α_i and β_j are the parameters of the model that govern the persistence of volatility shocks.
- ε_{t-i}^2 represents the squared residual (error term) at time t-i.
- σ_{t-j}^2 represents the conditional variance at time t-j.

The parameters α_i and β_j determine the extent to which past squared residuals and past conditional variances influence the current conditional variance, respectively. The number of lagged values to be considered is usually determined by some form of goodness of fit metric like AIC and BIC. By estimating these parameters using historical data, GARCH models provide insights into the dynamics of volatility and enable the forecasting of future volatility levels.

The GARCH model's persistence, determined by the sum of α and β, indicates the longevity of variance deviations from its long-term average. High persistence, seen when α + β approaches 1, suggests that deviations in variance from its long-term average will endure for an extended period, while low persistence, approaching 0, implies spikes in volatility dissipates quickly. This persistence level significantly influences risk management and asset allocation strategies. High persistence demands cautious risk management, anticipating enduring volatility, while low persistence allows for relaxed measures as volatility tends to dissipate. Accurate estimation of this parameter, varying across assets and time frames, is crucial, achievable through Maximum Likelihood Estimation.

7.2 Dynamic Conditional Correlation (DCC)

Dynamic Conditional Correlation (DCC) models represent a sophisticated extension of the Generalized Autoregressive Conditional Heteroskedasticity (GARCH) framework, offering a nuanced approach to capturing time-varying correlations among financial assets. At their core, DCC models aim to capture the dynamic nature of correlations among financial assets, recognizing that correlations evolve over time in response to changing market conditions and information arrivals. The theoretical foundation of DCC models builds upon the insight that correlations are not static but rather fluctuate over different market regimes. By incorporating this dynamic aspect, DCC models provide a more realistic depiction of the interconnectedness among assets in financial markets.

To grasp the intuition behind DCC models, consider the analogy of a synchronized dance performance. In a dance troupe, the movements of individual dancers are coordinated and synchronized, with the degree of synchronization evolving over time. Similarly, in financial markets, the movements of assets are intertwined, with correlations reflecting the degree of synchronization among asset returns. DCC models adaptively capture changes in correlation patterns, akin to the fluid coordination observed in a dance performance. Thus, DCC models offer a dynamic framework for understanding the interrelationships among assets in evolving market environments.

The mathematical formulation of DCC models builds upon the GARCH framework, extending it to capture time-varying correlations. The DCC model comprises two main components: the univariate GARCH models for individual asset returns and the dynamic conditional correlation estimator. The conditional correlation matrix in a DCC model is updated over time, allowing for changes in correlations between assets.

We represent the underlying GARCH model as the conditional expectation of the square of returns.

$h_{i,t} = E_{t-1}(r_{i,t}^2), \qquad r_{i,t} = \sqrt{h_{i,t}}\varepsilon_{i,t}$ for asset i at time instance t

Then the covariance matrix at time is represented as:

$H_t = D_t R_t D_t \qquad D_t = \mathrm{diag}(\sqrt{\sigma_{1,t}^2}, \sqrt{\sigma_{2,t}^2}, \dots, \sqrt{\sigma_{n,t}^2})$

D_t is a diagonal matrix of conditional volatilities and R_t is the conditional correlation matrix.

$$\varepsilon_t = D_t^{-1} r_t$$

We can see that both D_t and R_t changes with t as opposed to a constant correlation model where R would be invariant with t. To calculate R_t, we first estimate the matrix Q_t as

$$Q_t = S(1-\alpha-\beta) + \alpha(\varepsilon_{t-1}\varepsilon_{t-1}') + \beta Q_{t-1}$$

where *S* is unconditional correlation matrix of the epsilons or historical average correlation considered to be the long-term mean.

$$S = \frac{1}{T}\sum_t (r_t r_t')$$

To ensure that correlations fall between 0 and 1,

$R_t = (Q_t^*)^{-1} Q_t (Q_t^*)^{-1}$ where Q_t* is a diagonal matrix of Q_t

DCC models find wide-ranging applications in financial econometrics, including portfolio management, risk assessment, and volatility forecasting. In portfolio management, DCC models facilitate the construction of diversified portfolios by incorporating dynamic correlations among assets, thereby optimizing portfolio performance. Risk assessment measures, such as Value-at-Risk (VaR) and Conditional Value-at-Risk (CVaR), benefit from DCC models' ability to capture time-varying correlations, providing more accurate estimates of downside risk. Moreover, DCC models play a crucial role in volatility forecasting, enabling market participants to anticipate changes in correlation patterns and adjust their trading strategies accordingly.

7.3 Asymmetric DCC

Conditional estimates of equity's second moment often reveal the phenomenon of asymmetric volatility, where it tends to escalate more following a negative shock compared to a positive one of the same magnitude. To address this, we introduce a model capable of accommodating conditional asymmetries not only in volatilities but also in correlations. This model represents an extension of Engle's Dynamic Conditional Correlation (DCC) GARCH model. This framework expands the original model in two key dimensions: firstly, by allowing for series-specific news impact and smoothing parameters, and secondly, by permitting conditional asymmetries in correlations. Furthermore, modifications are made to the univariate volatility parameterizations from standard GARCH(1,1) models to accommodate conditional asymmetries.

Economically, two theories offer explanations for asymmetric volatility: the leverage effect and time-varying risk premia (volatility feedback). The leverage effect, as outlined by Black (1976), posits that following an unexpected decline in a stock's value, the firm's debt-to-equity ratio increases. Consequently, the volatility of the entire firm, assumed to remain constant, must manifest as increased volatility in the non-leveraged portion (equity). An alternative perspective, initially proposed by French (1987), suggests that news indicating future higher volatility prompts risk-averse investors to sell positions until the expected return compensates for the heightened risk. Consequently, markets decline in anticipation of volatility increases, leading to a larger rise in volatility following a negative shock (volatility feedback). While these explanations are primarily applicable to systematic risks, they are not mutually exclusive.

Despite the proposed explanations for asymmetries in return volatility, a theoretical framework justifying the evidence of asymmetric responses to joint adverse movements in correlations remains scarce. One plausible explanation lies in time-varying risk premia. Specifically, a negative systematic shock exerts downward pressure on returns across stock pairs, consequently increasing the variances of these securities in a CAPM-type environment. Assuming betas remain unchanged, covariances will increase. If idiosyncratic variances do not proportionally change, correlations will also increase. Consequently, correlations may exhibit higher levels following a negative shock than after a positive innovation of equivalent magnitude.

Mathematically we modify the DCC equation from

$$Q_t = (1 - a - b)\bar{P} + a\varepsilon_{t-1}\varepsilon'_{t-1} + bQ_{t-1}, \qquad P_t = Q_t^{*-1} Q_t Q_t^{*-1}$$

to $$Q_t = \left(\bar{P} - a^2\bar{P} - b^2\bar{P} - g^2\bar{N}\right) + a^2\varepsilon_{t-1}\varepsilon'_{t-1} + g^2 n_{t-1}n'_{t-1} + b^2 Q_{t-1}$$

where $a^2 + b^2 + \delta g^2 < 1$ and $n_t = I[\varepsilon_t < 0]$

$$\delta = \text{maximum eigenvalue } [\bar{P}^{-1/2}\bar{N}\bar{P}^{-1/2}]$$

The mathematical formulation of ADCC models extends the traditional DCC framework to accommodate asymmetries in correlation dynamics. While the basic structure of ADCC models remains similar to that of DCC models, additional parameters are introduced to capture the asymmetries. Specifically, separate parameters are estimated to govern the dynamics of correlations during positive and negative market regimes. Thus, ADCC models offer a nuanced approach to understanding the complex relationship between asset returns during different market conditions.

7.4 Markov Switching GARCH – DCC (MSGARCH-DCC)

Markov Switching GARCH (MSGARCH) models is as powerful tool capable to capture the intricate dynamics of financial data by incorporating non-linear specifications and regime-dependent behaviour. At the heart of a Markov Switching model lies the notion of different states of the world exerting influence on the evolution of a time series. In the context of MSGARCH models, dynamic properties are contingent upon the present regime, with regimes representing realizations of a hidden Markov chain characterized by a finite state space. This non-linear specification allows MSGARCH models to capture the inherent complexities and regime-dependent behaviours observed in financial time series data. Moreover, the dependence inherent in the Markov chain engenders conditional heteroskedasticity, reflecting the varying degrees of volatility across different regimes.

To grasp the intuition behind MSGARCH models, we can envision a financial market characterized by two distinct states or regimes. During periods of market stress, the regime may transition to a state where volatility is elevated, reflecting increased uncertainty and risk aversion among market participants. Conversely, during periods of market stability, the regime may transition to a state characterized by subdued volatility, signalling confidence and optimism in the market. By capturing these regime-dependent behaviours, MSGARCH models offer insights into the underlying dynamics of financial markets, facilitating more accurate volatility forecasts and risk management strategies.

The following is a mathematical representation of a MSGARCH(1,1) model.

$$\sigma_t^2(s_{1:t};\varphi_{t-1}) = \omega_{s_t} + \alpha_{s_t} a_{t-1}^2 + \beta_{s_t} \mathbb{E}\left[\sigma_{t-1}^2(s_{1:t-1};\varphi_{t-2})\middle|s_t;\varphi_{t-1}\right]$$

where $a_{t-1}^2 = [r_{t-1} - b_{t-1}(s_{1:t-1};\varphi_{t-2})]^2 = [r_{t-1} - \mu_{s_{t-1}} - \phi_{s_{t-1}} r_{t-2}]^2$

The latent state variables st ∈ {1, 2} for a two-state process is governed by constant transition probabilities:

$$p_{ij} = \mathbb{P}(s_t = j | s_{t-1} = i; \varphi_{t-1})$$

Φ_{t-1} denotes the past observed returns up to time t-1 and a_t is the innovation component at time t.

The transition probabilities have constraints $0 < p_{ij} < 1$ and $\Sigma^2_{j=1} p_{ij} = 1$ for i ∈ {1, 2}. The latent state process is assumed to be irreducible and aperiodic Markov chain with stationary probability measures.

The last term of the conditional variance measure can be evaluated as:

$$\mathbb{E}\left[\sigma_{t-1}^2(s_{1:t-1};\varphi_{t-2})\middle|s_t;\varphi_{t-1}\right] = \mathbb{E}\left[\sigma_{t-1}^2(s_{t-2:t-1};\varphi_{t-2})\middle|s_t;\varphi_{t-1}\right]$$

$$= \sum_{j=1}^{2}\sum_{i=1}^{2} \sigma_{t-1}^2(s_{t-2}=i, s_{t-1}=j;\varphi_{t-2})\mathbb{P}(s_{t-2}=i, s_{t-1}=j|s_t;\varphi_{t-1})$$

We assume states of each asset is independent and use these univariate state-dependant GARCH models as inputs into the DCC model.

7.5 Copula GARCH

Copula-GARCH presents yet another promising avenue to measure conditional dependency in a GARCH model, offering insights into multivariate distributions when marginal distributions are known.

Copulas are mathematical tools used to model the dependency structure between random variables. Unlike traditional methods that focus solely on the marginal distributions of individual variables, copulas allow us to separate the marginal distributions from the dependency structure, providing a flexible framework to capture complex dependencies. At its core, a copula is a function that describes the joint distribution of random variables based on their marginal distributions. It essentially serves as a bridge between the marginal distributions and the joint distribution, enabling us to model the relationship between variables independently of their individual distributions.

The key advantage of copulas is their ability to capture various forms of dependency that may not be adequately described by traditional correlation measures. For example, copulas can capture non-linear dependencies, tail dependencies and asymmetries in the joint distribution.

There are many types of copulas, each with its own characteristics and properties. Some common examples include:

1. Gaussian Copula: This is one of the simplest copulas and is often used when the variables have Gaussian distributions. It assumes that relationship between variables is linear and symmetric.
2. Student-t Copula: Unlike the Gaussian copula, the Student-t copula allows for heavier tails in the joint distribution, making it more suitable for capturing extreme events and tail dependencies.
3. Archimedean Copulas: These are a family of copulas characterized by certain mathematical properties. Examples include the Clayton copula, Gumbel copula, and Frank copula, each of which has its own shape parameter that determines the strength and type of dependency.

Consider two random variables X_1 and X_2 with marginal cdfs $F_i(x_i) = Pr[X_i <= x_i]$, i = 1,2. The joint cdf is denoted $H(x_1,x_2) = Pr[X_1 <= x_1, X_2 <= x_2]$. All cdfs $F_i(.)$ and $H(.,.)$ range in the interval [0,1].

From Sklar's theorem for Copulas, the copula function C defined over margins $F_i(x_i)$, so that
$H(x_1 , x_2) = C(F_1(x_1) , F_2(x_2))$.

For example, the Gaussian copula is defined by the cdf $C(u_1,u_2;\rho) = \Phi_\rho\left(\Phi^{-1}(u_1), \Phi^{-1}(u_2)\right)$

and the density by $c(u_1,u_2;\rho) = \dfrac{1}{\sqrt{1-\rho^2}}\exp\left(-\dfrac{1}{2}\psi'\left(R^{-1}-I_2\right)\psi\right)$

We can derive similar relations for Student-t copula. Academic research has shown evidence that the Student-t distribution fits well the univariate behaviour of financial time series.

Let us assume we have 2 GARCH(1,1) processes:

$$\begin{cases} x_{1t} = \mu_1 + \varepsilon_{1t} \\ \varepsilon_{1t} = \sigma_{1t} z_{1t} \\ z_{1t} \quad iid \quad t(\nu_1) \\ \sigma_{1t}^2 = \omega_1 + \alpha_1 \varepsilon_{1t-1}^2 + \beta_1 \sigma_{1t-1}^2 \end{cases} , \qquad \begin{cases} x_{2t} = \mu_2 + \varepsilon_{2t} \\ \varepsilon_{2t} = \sigma_{2t} z_{2t} \\ z_{2t} \quad iid \quad t(\nu_2) \\ \sigma_{2t}^2 = \omega_2 + \alpha_2 \varepsilon_{2t-1}^2 + \beta_2 \sigma_{2t-1}^2 \end{cases}$$

The dependency structure between z_{1t} and z_{2t} is modelled by the copula function, the parameters of which are themselves time variant.

$$\theta_t = T\left(a_1 + a_2\theta_{t-1} + a_3 z_{1t-1} z_{2t-1}\right)$$

T(.) is a transformation to ensure parameter Θ_t lies within the existence interval of the copula function. The parameters a_1, a_2 and a_3 are estimated using Maximum Likelihood methods. Θ_t is then used to find the correlation at time t using the knowledge of the copula distribution.

7.6 Sequential Least Squares Programming (SLSQP)

Sequential Least Squares Programming (SLSQP) stands as a powerful numerical optimization algorithm utilized to solve constrained non-linear optimization problems. Developed by Dieter Kraft in the 1980s, SLSQP has found widespread application in various fields, including finance, engineering, and machine learning. In the context of financial modelling, SLSQP serves as a valuable tool for estimating parameters in complex models such as the Dynamic Conditional Correlation (DCC) model.

At its core, SLSQP seeks to minimize a given objective function subject to a set of constraints, both equality and inequality. The algorithm employs a sequential approach, iteratively optimizing the objective function while ensuring that the constraints are satisfied at each iteration. Central to SLSQP is the use of BFGS method to approximate gradients. SLSQP efficiently explores the parameter space, converging towards those estimates that minimize the loss function, while adhering to model constraints. By iteratively refining the parameter estimates, SLSQP converges towards the optimal solution, providing robust and reliable parameter estimates even in complex optimization landscapes.

Consider a non-linear optimization problem of the form:

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{subject to} \quad & h(x) \geq 0 \\ & g(x) = 0. \end{aligned}$$

The Lagrangian of this problem is:

$$\mathcal{L}(x, \lambda, \sigma) = f(x) + \lambda h(x) + \sigma g(x)$$

$$\nabla \mathcal{L}(x, \lambda, \sigma) = 0$$

The optimization searches for the solution by iterating the following equation, where ∇^2_{xx} denotes the Hessian matrix:

$$\begin{bmatrix} x_{k+1} \\ \lambda_{k+1} \\ \sigma_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ \lambda_k \\ \sigma_k \end{bmatrix} - \underbrace{\begin{bmatrix} \nabla^2_{xx}\mathcal{L} & \nabla h & \nabla g \\ \nabla h^T & 0 & 0 \\ \nabla g^T & 0 & 0 \end{bmatrix}^{-1}}_{\nabla^2 \mathcal{L}} \underbrace{\begin{bmatrix} \nabla f + \lambda_k \nabla h + \sigma_k \nabla g \\ h \\ g \end{bmatrix}}_{\nabla \mathcal{L}}.$$

The DCC models relies on the estimation of various parameters to capture the dynamic nature of correlations. SLSQP optimization offers an effective approach to estimating these parameters, enabling practitioners to calibrate the DCC model to historical data and extract meaningful insights into correlation dynamics. SLSQP optimization is employed to maximize the likelihood function of the DCC model, which quantifies the probability of observing the historical data given the model parameters. The objective function to be minimized in this case is the negative log-likelihood function, which penalizes deviations between the observed data and the model's predictions. By minimizing the negative log-likelihood function, SLSQP effectively maximizes the likelihood of the data given the model, yielding parameter estimates that best fit the observed data.

7.7 Artificial Neural Networks

Artificial Neural Networks (ANNs) have emerged as a transformative force in the field of artificial intelligence, reshaping the landscape of technology and innovation across various domains. Rooted in the intricate workings of the human brain, ANNs are computational models designed to mimic the learning processes of biological neural networks.

At the heart of Artificial Neural Networks lie neurons, the fundamental building blocks that process and transmit information. These neurons are organized into layers, comprising an input layer, one or more hidden layers, and an output layer. Each neuron receives input signals, applies a transformation through an activation function, and produces an output signal that propagates through the network. The strength of connections between neurons, represented by weights, and the flexibility introduced by biases, collectively enable ANNs to learn complex patterns and make predictions.

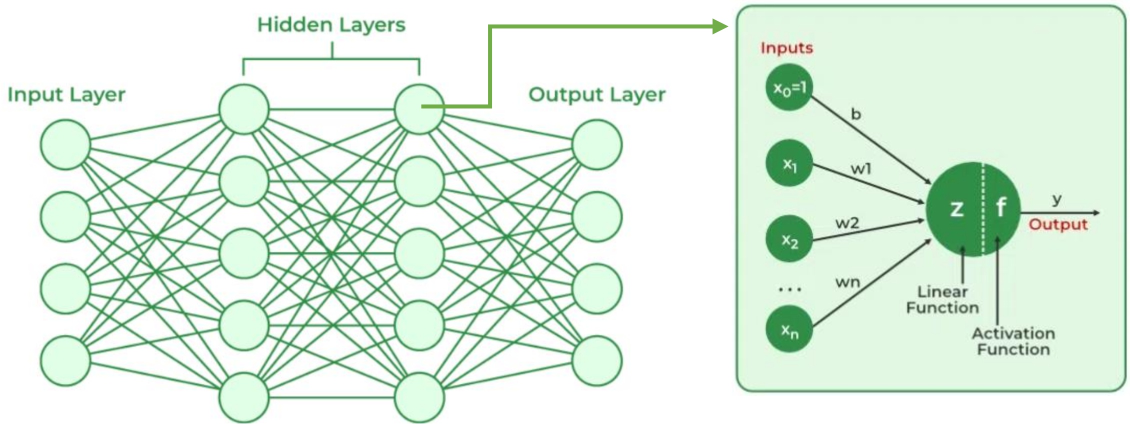


Fig 12: Neural Network Architecture

Central to the functioning of ANNs is the concept of activation functions, which introduce non-linearity into the network, allowing it to model intricate relationships within data. Common activation functions such as the sigmoid, tanh, ReLU (Rectified Linear Unit), and softmax functions, serve as crucial elements in shaping the behaviour and performance of neural networks.

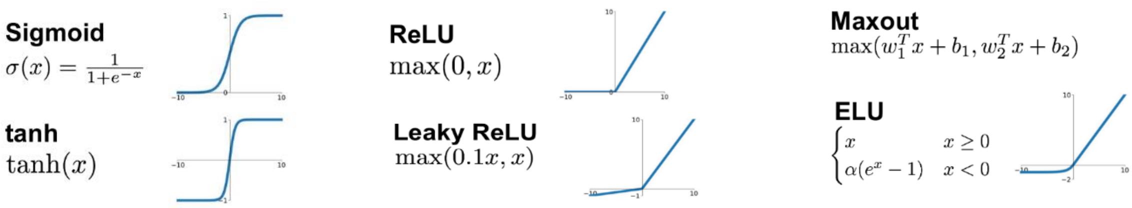


Fig 13: Activation Functions

The training process of ANNs is facilitated by the feedforward process and subsequent backpropagation algorithm, a cornerstone of modern machine learning. In the feedforward process, input data is propagated through the neural network, layer by layer, until it reaches the output layer. Each neuron computes a weighted sum of its inputs, applies an activation function, and passes the result to the neurons in the subsequent layer. Backpropagation involves iteratively adjusting the weights and biases of the network based on the error between predicted outputs and actual targets. Through the principles of gradient descent optimization, ANNs learn to minimize the loss function, gradually refining their predictions and improving performance over time.

Gradient descent is the fundamental optimization algorithm used to minimize the loss function in deep learning models. At its core, gradient descent operates by iteratively adjusting the parameters of the model in the direction of the steepest descent of the loss landscape. The basic form of gradient descent involves calculating the gradient of the loss function with respect to each parameter and updating the parameters by subtracting the gradient multiplied by the learning rate. This process continues until convergence is reached or a predefined number of iterations is completed. Variants of gradient descent, such as Stochastic Gradient Descent (SGD), Mini-batch Gradient Descent, and Batch Gradient Descent, offer different approaches to updating parameters and handling training data. SGD updates parameters using one randomly training sample, making it computationally efficient for large datasets. Mini-batch Gradient Descent strikes a balance between SGD and Batch Gradient Descent by updating parameters using small batches of data. Batch Gradient Descent updates parameters using the entire training dataset (called epoch) in each iteration, ensuring stability but often at the cost of computational efficiency. Despite their differences, all variants of gradient descent aim to iteratively optimize the model parameters to minimize the loss function and improve the model's performance.

Momentum (RMSProp), AdaGrad, and Adam are advanced optimization algorithms commonly used to enhance the performance of gradient descent in training neural networks. RMSProp introduces a velocity term that accelerates the parameter updates by accumulating gradients over time, enabling smoother and faster convergence by dampening oscillations in the optimization process. AdaGrad adapts the learning rate for each parameter based on the historical gradients, scaling down frequently updated parameters and scaling up infrequently updated ones, which can be beneficial for sparse data and non-stationary objectives. Adam, short for Adaptive Moment Estimation, combines the ideas of momentum and Adagrad by maintaining exponentially decaying averages of past gradients and squared gradients, adjusting the learning rates accordingly. This adaptive learning rate scheme makes Adam robust to various types of optimization problems and often leads to faster convergence and better generalization performance in practice.

7.8 Residual Networks

Traditional deep neural networks suffer from the vanishing gradient problem, where the gradients become increasingly small as they propagate through numerous layers during training. This phenomenon impedes the optimization process, hindering the convergence of the network and limiting its ability to capture complex patterns in the data. Residual Networks (ResNets) alleviate this issue by introducing skip connections that enable the direct flow of information from one layer to another, bypassing intermediate layers. By preserving the gradient signal throughout the network, ResNets facilitate the training of deeper architectures with improved convergence properties.

The intuition behind ResNets stems from the concept of residual learning, where instead of directly fitting the desired underlying mapping, the network learns to model the residual between the input and the desired output. This residual learning approach allows the network to focus on learning the residual transformations, which are typically easier to optimize, particularly in the case of very deep networks. By enabling the direct propagation of gradients through shortcut connections, ResNets facilitate the flow of information across layers, enabling the network to effectively learn both shallow and deep features from the data.

The construction of a ResNet involves the integration of residual blocks, each comprising multiple layers with skip connections. The key innovation lies in the presence of a shortcut connection that directly adds the input to the output of the block. This additive operation allows the network to learn residual mappings, capturing the difference between the input and output feature maps. Additionally, identity mappings are used to ensure that the dimensions of the input and output feature maps remain consistent, facilitating the seamless integration of shortcut connections across different layers of the network.

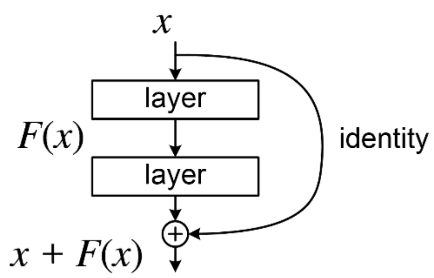


Fig 14: ResNet construction

7.9 Attention Mechanism

Attention Mechanism is a pivotal component in modern deep learning architectures, particularly in tasks involving sequential or structured data such as natural language processing. Its core function is to allow the model to focus selectively on different parts of the input, effectively learning to weigh the importance of each element. This mechanism mimics the human cognitive process of selectively focusing on relevant information while ignoring irrelevant details.

The intuition behind attention mechanism can be likened to the spotlight metaphor. Just as a spotlight illuminates specific regions of a stage while leaving others in darkness, attention mechanism enables the model to allocate more resources to certain parts of the input sequence, enhancing its ability to capture relevant information. By dynamically adjusting the attention weights for each element in the sequence, the model can adaptively attend to the most informative parts, leading to more effective processing and representation learning.

These attention weights are typically calculated using a scoring function that measures the compatibility between the current context and each element in the sequence. The scores are then normalized using a softmax function to obtain attention weights that sum to one, representing the importance of each element. Finally, the model computes a weighted sum of the input elements using the attention weights, generating a context vector that encapsulates the relevant information for the current step.

Unlike traditional attention mechanisms that focus on contextual information from external sources, self-attention enables each element in the sequence to attend to all other elements, including itself. By dynamically computing attention weights based on pairwise relationships between elements, self-attention facilitates the modelling of long-range dependencies without relying on recurrence or convolution. This mechanism has gained prominence in transformer-based models, revolutionizing natural language processing tasks by enabling efficient processing of sequential data with minimal computational overhead.

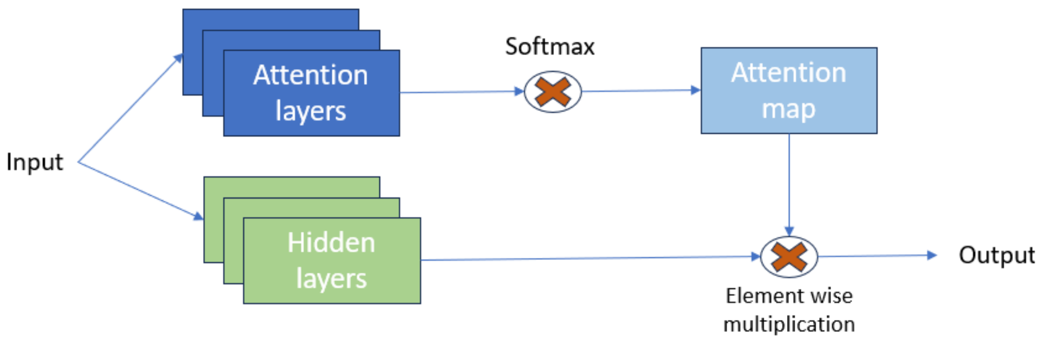


Fig 15: Self-attention mechanism

7.10 Attention ResNets

The integration of the self-attention mechanism within residual blocks presents an intriguing avenue for advancing financial feature acquisition. By harnessing the power of attention, this approach not only enables the identification of relevant features but also facilitates the discovery of distinct regimes within financial data.

Furthermore, the application of attention mask weights via elementwise product onto the output units of the residual block, positioned after the activation function, underscores a nuanced approach to feature selection. This strategic placement ensures that the attention mechanism influences the processing of data within the block, potentially enhancing the model's ability to capture essential financial signals. This construction not only deepens the network's capacity to learn complex patterns but also imbues it with the adaptability to discern and respond to evolving financial regimes.

The incorporation of the self-attention mechanism within residual blocks marks a departure from traditional approaches such as Markov-switching models, where transition probabilities remain time-invariant. Unlike these static models, the Attention-ResNet framework acknowledges the dynamic nature of financial markets, recognizing that regimes can evolve rapidly over time.

Moreover, within the Attention-ResNet paradigm, regime probabilities are not simply predefined or fixed. Instead, they are dynamically inferred through a complex non-linear combination of input variables. This adaptive nature allows the model to capture subtle shifts and nuances in market behaviour, offering a more responsive and accurate representation of financial dynamics.

The mathematical formulation is as follows:

Self-Attention Module: $z^{a,(n)}(X) = W^{a,(n)} \cdot f^{(n-1)}(X) + b^{a,(n)}$

$$f^{a,(n)}(X) = \sigma(z^{a,(n)}(X))$$

$$z^{a,(n+1)}(X) = W^{a,(n+1)} \cdot f^{a,(n)}(X) + b^{a,(n+1)}$$

$$f^{a,(n+1)}(X) = \Phi(z^{a,(n+1)}(X))$$

where σ is the activation function for the hidden layers, Φ is the softmax activation in the output layer, and $W^{a,(n+1)}$ are the weight parameters applied to inputs of each layer.

$$\Phi(z^{s,(N)}(X^s)) = \left[\frac{exp(z_1^{a,(N)}(X^a))}{\sum_c exp(z_c^{a,(N)}(X^a))}, \ldots, \frac{exp(z_c^{a,(N)}(X^a))}{\sum_c exp(z_c^{a,(N)}(X^a))}\right]^\top$$

Residual Block: $z^{(n)}(X) = W^{(n)} \cdot f^{(n-1)}(X) + b^{(n)}$

$$f^{(n)}(X) = \sigma(z^{(n)}(X))$$

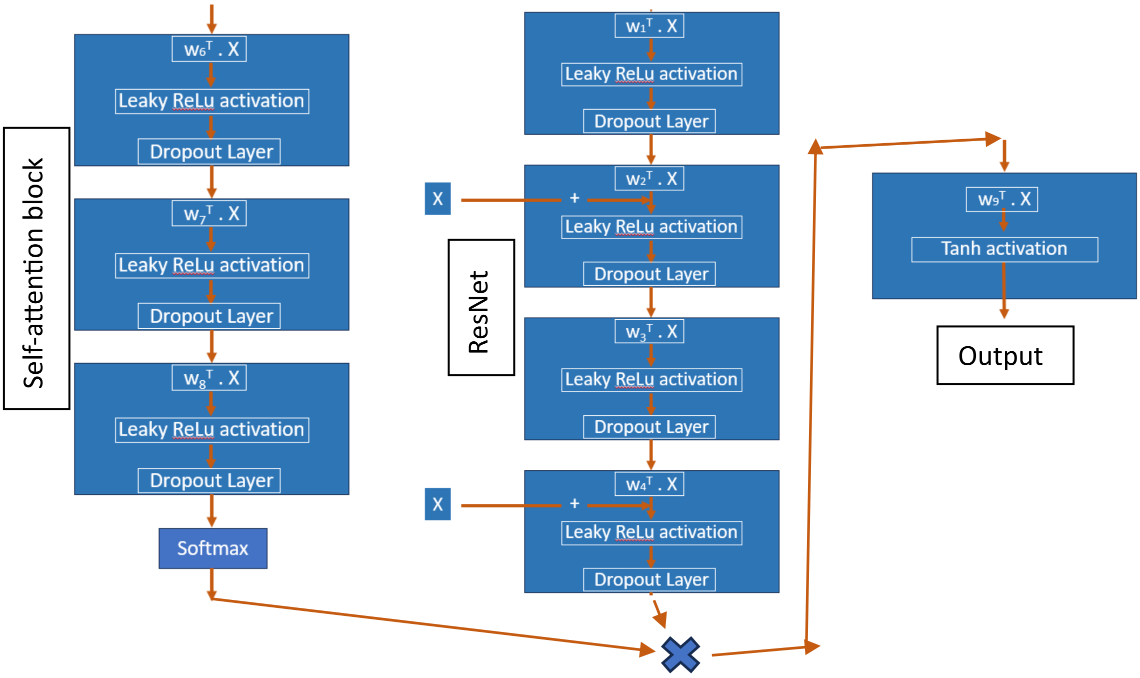
$$z^{(n+1)}(X) = W^{(n+1)} \cdot f^{(n)}(X) + b^{(n+1)}$$

$$f^{(n+1)}(X) = \sigma(z^{(n+1)}(X) + f^{(0)}(X))$$

Combination of Self-Attention module with Residual Block:

$$f^{(n+1)}(X) \otimes f^{a,(n+1)}(X)$$

Two Deep Neural Network models were designed to forecast standard deviations and correlations. The schema of the networks (shown below) has the same structure.



Each layer in the network has 32 neurons, except the last layer after the attention mask is applied, which has 4 neurons. The inputs are fed into both the self-attention block and the ResNet block. The final activation is Tanh for correlation as Tanh restricts the output range between 0 and 1. For the network for standard deviations, the activation is ReLu as standard deviations cannot be negative.

To prevent overfitting, the following measures are taken:

1. Each neuron has a random dropout rate of 0.2, which means 20% of the weights are randomly set to 0. This prevents the model to be overfitted to a particular parameter.
2. Cross-validation sets of 20% of the training set is used during each epoch. This prevents the model to overfit on any given batch during compilation.

The optimizer used for gradient descent is Adam. This optimizer converges fast at the beginning but slows down towards the end when gradients reduce as the weights near optimum. This results in smooth learning. Furthermore, the learning rate also decays at an exponential schedule. This means that the model puts more weight on the gradients during initial updates but the weights gradually reduce with training iterations.

Having created the network architecture, we move to data preparation. The model being a supervised learning algorithm needs useful input information along with a target variable.

We create the following variables for standard deviation model (for each of the 5 indices separately):

1. For each t, daily returns, squared daily returns and abs(daily returns) from t-1 to t-19
2. For each t, 20 days, 50 days and 120 days rolling mean of daily returns
3. For each t, 20 days, 50 days and 120 days rolling mean of daily squared returns
4. For each t, 20 days, 50 days and 120 days rolling standard deviation of returns

The target variable for the standard deviation model is the next 60 days actual standard deviation.

We create the following variables for standard deviation model (for each 5 index pairs separately):

1. For each t and asset 1, daily returns, squared returns and abs(daily returns) from t-1 to t-19
2. For each t and asset 2, daily returns, squared returns and abs(daily returns) from t-1 to t-19
3. For each t, product of daily return of asset 1 and 2 from t-1 to t-19
4. For each t, abs(product of daily return of asset 1 and 2) from t-1 to t-19
5. For each t and asset 2, daily returns, squared returns and abs(daily returns) from t-1 to t-19
6. For each t and asset 1, 20 days, 50 days and 120 days rolling mean of daily returns
7. For each t and asset 2, 20 days, 50 days and 120 days rolling mean of daily returns
8. For each t and asset 1, 20 days, 50 days and 120 days rolling mean of daily squared returns
9. For each t and asset 2, 20 days, 50 days and 120 days rolling mean of daily squared returns
10. For each t, 20 days, 50 days and 120 days rolling Pearson correlation of daily squared returns

The target variable for the correlation model is the next 60 days actual Pearson correlation.

The historical data from 2008 to 2023 is split into training set from 2008 – 2018 and testing set from 2019 to 2023. This is to ensure the learnt models are transferable between two completely independent periods in the financial markets. We report the performance of the model for only the testing period.

We use the predicted standard deviation and correlations to construct the minimum variance portfolios for each month in the testing period and compare them against the DCC portfolios.

7.11 Markowitz Minimum Variance Portfolio

Modern portfolio theory is a mathematical framework for assembling a portfolio of assets such that the expected return is maximized for a given level of risk. Pioneered by Harry Markowitz in 1952, it is a mathematical formalization of the effect of diversification in investing, the idea that owning different kinds of financial assets is less risky than owning only one type. Its key insight is that an asset's risk and return should not be assessed by itself, but by how it contributes to a portfolio's overall risk and return.

The Minimum Variance Portfolio (MVP) methodology is an extension of the Modern Portfolio Theory for crafting portfolios aimed at minimizing risk exposure. Built on the assumption that investors exhibit risk aversion, the MVP framework prioritizes risk mitigation when designing portfolios. A critical step in this process involves constructing a covariance matrix representing the pairwise correlations and variances of assets within the portfolio. By leveraging this matrix, investors can determine optimal asset weights that minimize portfolio risk, framed as a quadratic programming problem subject to linear constraints.

Building upon preceding discussions, we utilize the conditional volatility outputs from GARCH models and conditional correlation outputs from the DCC models to estimate the covariance matrix of asset returns, a cornerstone of the MVP approach. Incorporating time-varying conditional volatility and correlation enables us to compute dynamic covariance matrices that adapt to evolving market risks. This adaptive feature is crucial for effective portfolio optimization, recognizing that correlations fluctuate over time, especially during periods of market stress. Employing a time-varying covariance matrix approach allows investors to adjust portfolio weights in response to changing market conditions, thereby fostering more resilient and stable portfolios.

While these models offer valuable insights into time-varying correlations, they are not without limitations. The MVP approach heavily relies on historical data to estimate time-varying covariance matrices, assuming that past behaviour can predict future asset price movements. However, this assumption may falter during significant shifts in market conditions, potentially compromising risk management effectiveness. Additionally, the MVP approach assumes stationary statistical properties of asset returns, which may not hold true in reality. Fluctuating mean and variance can impact the accuracy of the MVP model, particularly during market turbulence when volatility surges. Thus, investors should exercise caution and complement MVP with other risk-management tools like Value at Risk.

Mathematically, the weight vector that satisfies the Minimum Variance Portfolio constraints is as follows:

$$w = \frac{\Sigma^{-1}\mathbf{1}}{\mathbf{1}'\Sigma^{-1}\mathbf{1}}$$

where $\mu_{\pi} = \mu' w$ and $\sigma_{\pi}^{2} = w'\Sigma w$

whete Σ is the covariance matrix of assets under consideration and **1** is an identity matrix.

As we can see, the MVP is completely independent of the returns of the assets and depends only of the variance and correlation of the returns.

7.12 Indices

For this analysis, we develop monthly covariance projections for five equity indices: the S&P 500 of the USA, Nifty 50 of India, Hang Seng Index of Hong Kong, Stoxx 600 of Europe, and IBovespa of Brazil. These indices are selected to ensure a robust representation of both developed and emerging markets, with significant coverage across major industrial zones. Additionally, the selection is guided by the understanding that diversified portfolios, including those following strategies like 60/40 and roll down portfolios, typically incorporate exposures to both developed and emerging markets. This analysis aims to highlight the extent of diversification achieved through these selections.

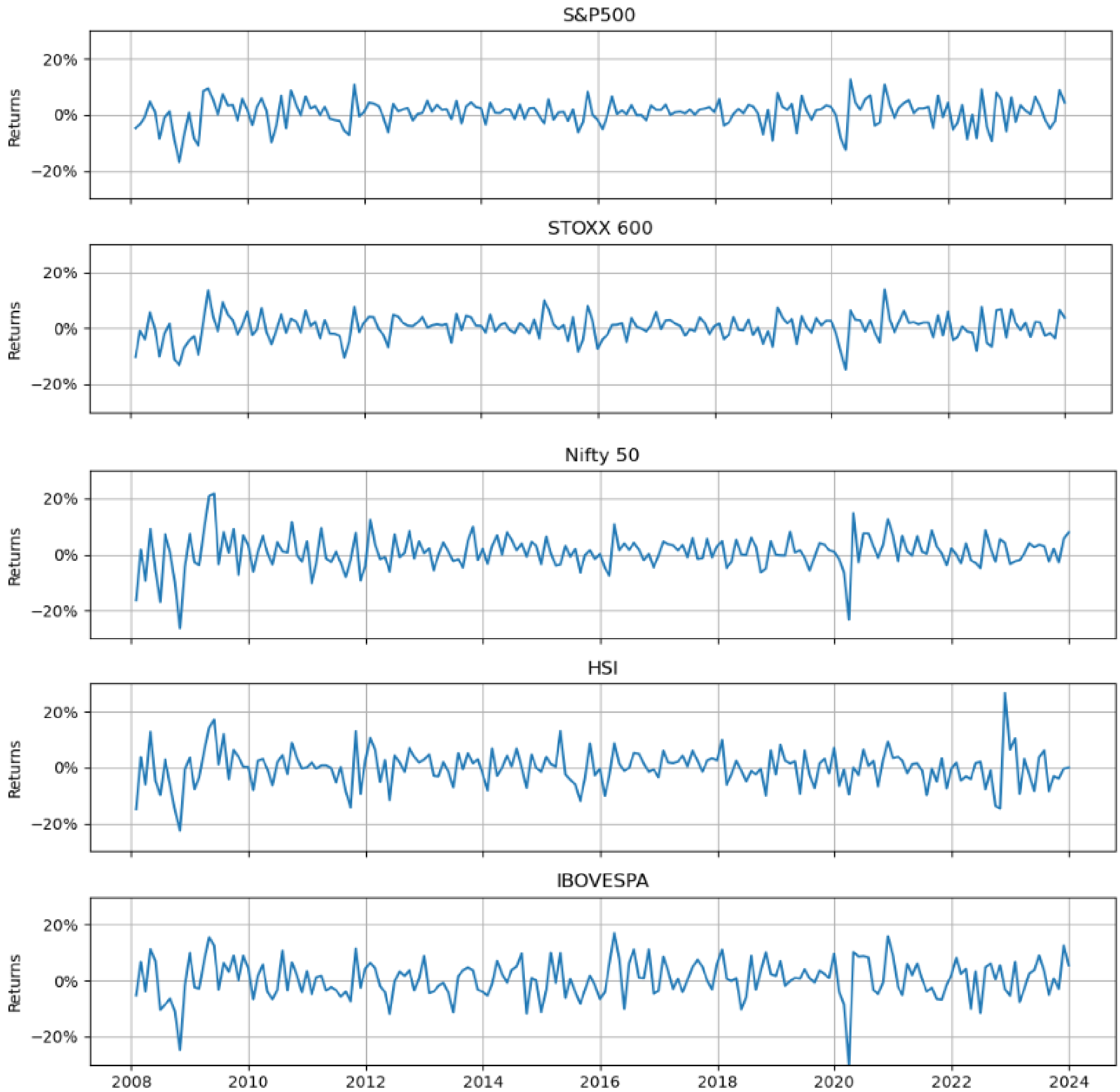
S&P 500 (USA): The S&P 500, often considered the benchmark index for the US stock market, consists of 500 of the largest publicly traded companies in the United States. It represents a diverse range of industries and sectors, including technology, healthcare, finance, and consumer goods. The index is weighted by market capitalization, meaning that larger companies have a greater influence on its movements. The S&P 500 is widely used by investors and analysts as a measure of the overall health and performance of the US equity market.

Nifty 50 (India): The Nifty 50 is the National Stock Exchange of India's benchmark stock market index. It comprises the top 50 largest and most actively traded Indian companies listed on the exchange, representing various sectors of the Indian economy. Similar to the S&P 500, the Nifty 50 is market capitalization-weighted, with larger companies having a greater impact on its movements. It serves as a barometer for Indian equity market and is closely monitored by investors, traders, and policymakers.

Hang Seng Index (Hong Kong): The Hang Seng Index (HSI) is the primary stock market index of the Hong Kong Stock Exchange. It tracks the performance of the 50 largest and most liquid companies listed on the exchange, covering a broad spectrum of industries such as finance, real estate, telecommunications, and technology. The HSI is also market capitalization-weighted.

Stoxx 600 (Europe): The Stoxx 600 is a stock index representing the performance of 600 large, mid, and small-cap companies across 17 European countries. It covers a wide range of industries, including healthcare, banking, automotive, and technology. Unlike some other indices, the Stoxx 600 is not limited to companies from a specific country or region within Europe, providing a more comprehensive view of the European equity market.

IBovespa (Brazil): The IBovespa is the benchmark stock index of the São Paulo Stock Exchange (B3) in Brazil. It comprises the most actively traded stocks on the exchange, representing various sectors of the Brazilian economy, including commodities, banking, energy, and manufacturing. The index is market capitalization-weighted. The IBovespa is widely followed as an indicator of the performance of the Brazilian and broader Latin American equity market.



	BOVESPA	SP500	HSI	NIFTY50	STOXX
Annual Return	5%	8%	-3%	8%	2%
Standard Deviation	23%	16%	22%	21%	16%
Sharpe	0.19	0.46	-0.15	0.38	0.10
Sortino	0.06	0.13	-0.05	0.11	0.03
Max Drawdown	50%	48%	55%	47%	46%

Table 1: Performance measures of the 5 indices from 2008 to 2013.