

# Forecasting volatility with the Shapley value based forecast combination

Samuel Wesselo (541765)

---



---

Supervisor:	Prof dr. P. Franses
Second assessor:	Dr. Wendun Wang
Date final version:	30th June 2023

---

## Abstract

This paper investigates the effectiveness of the Shapley value weighted forecast combination method to predict volatility. We compare the Shapley method to the  $1/N$  equal weight method and the OLS weighted method. The  $1/N$  method is generally regarded as a benchmark method for combining forecasts. Outperforming this method indicates the possibility of realizing a higher quality of forecasts, investors could increase the quality of their risk management, investment and option pricing. To test the accuracy of our combination methods we split our data in three parts. One part for parameter estimation, the second part involves allocation of weights to each forecast model and the final part is to compare the combined forecasts with each other. The performance of the methods is compared by looking at the  $R^2$  and the mean average error before performing a Diebold Mariano test. We use four different GARCH models in the combination process. First we perform four runs of a thousand simulations based on GARCH(3,3) models as data generating process to get a correct image of the accuracy of the three combination methods. Secondly, we use actual data from the Dow Jones Industrial Average to test whether the Shapley method works better for actual stock market data. We use data from the Federal Reserve Bank of St. Louis from July 2013 until June 2018. We find out the Shapley method significantly outperforms the other methods in most of the simulations, while the Shapley forecasts and the equal forecasts are much alike due to the fact the models do not differ much. For our stock market data we partly get the surprising result of the OLS method outperforming the other methods. The Shapley method however still outperforms the equal weight method for this data set.

The views stated in this thesis are those of the author and not necessarily those of the supervisor, second assessor, Erasmus School of Economics or Erasmus University Rotterdam.

**Contents**

- 1 Introduction** **2**
- 2 Literature** **3**
- 3 Data** **4**
- 4 Methodology** **7**
  - 4.1 Data division . . . . . 7
  - 4.2 GARCH models . . . . . 8
  - 4.3 Forecast combination . . . . . 9
    - 4.3.1 Shapley values . . . . . 9
    - 4.3.2 Equal . . . . . 10
    - 4.3.3 OLS . . . . . 10
  - 4.4 Forecast comparison . . . . . 10
- 5 Empirical results** **11**
  - 5.1 Data Generating Process . . . . . 11
  - 5.2 Dow Jones Industrial Average . . . . . 14
- 6 Conclusion** **19**
- References** **20**
- A Calculation of Shapley weights with four models** **21**
- B Programming code explanation** **22**

# 1 Introduction

Volatility forecasting plays a very important role in risk management, investment and option pricing. Accurate volatility predictions help investors determine when to buy, sell or hold assets. There are a great number of different ways to forecast volatility. For instance the standard GARCH models (Bollerslev, 1986), the Threshold GARCH models (Wu, 2010) and the real-time GARCH model (Smetanina, 2017). Bates and Granger (1969) is the first paper in which it is proven that a combination of forecasts could give a lower mean-squared error than either of the original forecasts. Since then, a lot of possible ways to combine different forecasts have been suggested. This paper builds upon Franses (2023), which introduces a new method to combine forecasts. This method bases the weights of the combined forecasts on their Shapley value, which is especially useful when the forecasts are strongly correlated. The main research question of the thesis is: Is the Shapley-values-based forecast combination a good way to forecast the volatility of the Dow Jones Industrial Average? To the best of our knowledge there has not been done a Shapley-value-based forecast combination on the Dow Jones Industrial Average (DJIA). The DJIA is one of the most widely recognized stock market indices in the world, making it a reliable indicator of market sentiment and trading behavior during significant economic events.

We build upon Franses (2023) by performing a simulation experiment and an empirical experiment to test whether this specific way of combining forecasts is useful. To test whether the Shapley method is a reliable method we compare this method with the commonly used  $1/N$  'equal' method and with the OLS method. We first perform a simulation experiment with a data generating process. We choose four models which are used in forecasting, the models we use are GARCH(1,1), GARCH(2,1), TGARCH(1,1) and TGARCH(2,1). We divide the data in three separate parts. The first part of the data is used to estimate the parameters from each model separately. In the second sample we make one-step ahead forecasts with a rolling window, after which we adjust the parameters based on the new data point. Based on the Shapley-values from this part of the data we give each of the forecasting models a weight as explained in section 4.3.1. In the third and last part of the sample we also make the one-step ahead forecasts. We do not change the weights given to each model. Based on the results of the forecasts in the final quarter we compare the Shapley values-based forecast combination with the  $1/N$  forecast combination and the OLS based forecast combination. We do this based on their  $R^2$ , MAE and with a Diebold Mariano test. Secondly, we look at the volatility of the DJIA Industrial Average. We look at data from July 2013 until June 2018. We use the same models as in the simulation experiment which can be used to forecast the DJIA. We divide the data in two different ways for robustness and repeat the steps taken in the simulation experiment.

In our paper we find that for the simulation experiment the Shapley value based forecast significantly outperforms the equal weight method and the OLS method. The difference in accuracy of the Shapley method and the equal weight method is relatively small. The equal weight method produces a significantly better forecast for a part of the simulations, but the Shapley method produces the better forecast in the majority of the simulations. Furthermore, due to the similarity of the GARCH models, the Shapley method weights do not differ very much from the equal weights. The DJIA data gives us some more surprising results. For this data set the OLS weighted method surprisingly produces the most accurate forecast when dividing

the data according to the first division method. The reason for this is a peak in variance in February 2018, where the Shapley method and the equal weight method produce excessively high forecasts. In the other data division method the Shapley method significantly produces the most accurate forecasts. For both methods the Shapley method does outperform the equal weight method significantly for this data. To conclude, we see that most of the time the Shapley method produces more reliable forecasts than the  $1/N$  equal weight method. Generally both of these methods are more accurate than the OLS method, however it also is possible that the OLS method coincidentally produces better forecasts. With this finding we contribute to the current literature by showing that a Shapley value based forecast is a good alternative for predicting volatility. The  $1/N$  combination method, often seen as a benchmark, is outperformed by the Shapley method in most of the cases of the simulation experiment and for actual stock market data.

In Section 2 we discuss some of the papers written on the subject of combining forecasts and using GARCH models for forecasting volatility. The data used in this paper is discussed in Section 3. Section 4 is about the methods used to answer our research question. The empirical results from our simulations and our DJIA data are presented in Section 5. In Section 6 we talk about the implications of our research paper, what we could have done differently and what possible future research might be interesting on this subject.

## 2 Literature

Since Bates and Granger (1969) propose combining forecasts there have been an incredibly great deal in optimising the combining of forecasts. Wang, Hyndman, Li and Kang (2022) present an extensive review on these forecast combinations. This paper mentions that the first idea of combining multiple forecasts was when Francis Galton attended an ox-weight-judging competition. He observed that the average of 787 estimates of the weight came very close to the actual weight, which gave him a realization that this might be extended to different kind of predictions. Wang et al. (2022) states that there are three sources of uncertainty which are reasons to not rely on one single "best model"; data uncertainty, parameter uncertainty and model uncertainty. When combining different models deviations or errors in each individual model will balance each other out, this way the forecast becomes more robust against misspecification biases. Timmermann (2006) states there are two levels of aggregation in the combination problem. The first step is to summarize information from the individual forecasters to make point forecasts. The second step aggregates the vectors of the point forecasts to the consensus measure. Because of the information aggregation generally it is expected that the bias is increased but the variance of the forecast error is reduced. Optimally the combination should trade off these components.

There are multiple ways to look for the optimal combination. Clemen (1989, p.559) writes: "The results have been virtually unanimous: combining multiple forecasts leads to increased forecast accuracy . . . in many cases one can make dramatic performance improvements by simply averaging the forecasts." Averaging the forecasts is the most obvious way, which is also a relatively easy method to compute. Stock and Watson (1998) look at 49 univariate forecasting methods and their optimal combinations. One of their subquestions is to find out whether combination forecasts outperform forecasts based on a single method across a range of time

series and if this is the case how heavily these combination forecasts should weight the currently-best performing forecasting methods. They find that pooled forecasts are found to outperform forecasts from any single method, especially the linear combination and median forecasts. The most reliable pooling method are the procedures that place weight on all forecasting models. This can be based on inverse MSE weighting, median or equal weighted. Median combination forecasts is a method where the median from a group of methods is used, this is better than linear combinations in the case of non Gaussian forecast errors. Stock and Watson (1998) also find there is not much effect from using a reduced or rolling sample for computing the weights. In the majority of the literature the weights of individual forecasts should add up to unity. Granger and Ramanathan (1984) test three alternative approaches and show that the best method is to add a constant term, instead of constraining the weights. They show through matrix computations that this method results in the smallest mean squared error and an unbiased combined forecast. Combining forecasts helps in forecasting when there is a structural break. Bates and Granger (1969) conclude that methods combining forecasts allowing for the weight to change lead to better forecasts. Rolling weighted least squares and time-varying parameter techniques are shown to be useful.

In this paper we predict the volatility of the American stock market, through the DJIA. Poon and Granger (2003) summarizes and compares 93 papers studying forecasting performance of various volatility models. They state that financial market volatility is clearly forecastable. Something else they conclude is that simple ARCH models do not result in accurate forecasts, in contrast to more comprehensive GARCH models. GARCH models are popular models for predicting volatility. Gokcan (2000) and Sharma, Aggarwal and Yadav (2021) both compare different (non-)linear GARCH models. The non-linear GARCH models were developed to examine leverage effect and volatility clustering. The leverage effect is the difference in effect between negative news and positive news. Usually stock markets react more extreme to negative news than to positive news. Both papers conclude that the GARCH(1,1) model outperforms the non-linear models. Charles and Darné (2014) look back to the volatility of the DIJA over the period 1928-2013. They determine the events that cause large shocks using models from the ARCH-family. They found large volatility shocks are principally due to the major financial crashes, US elections, wars, monetary policies, macroeconomic news and declarations on the economic situation, terrorist attacks, bankruptcy and regulation. This means that these kind of shocks should be taken into account in modeling volatility of returns.

### 3 Data

The data we use consists of the daily closing stock market position from the Dow Jones Industrial Average from the period 01-07-2013 until 29-06-2018. We get this data from the Federal Reserve Economic Data (FRED), the database from the Research division of the Federal Reserve Bank of St. Louis. We choose to exclude the COVID period from our data because this would not give accurate representations of the quality of forecasts. The reason for this is that COVID is such an unusual and excessive shock that we do not think this is representative. There are 1262 trading days of which we look at the simple returns. We multiply the returns by  $10^2$  for optimal functioning of the code in Python. In our research we use realized variance as a measure for the

actual volatility. We use the realized variance from the following formula:

$$RV_i = \frac{1}{T} \sum_{i=1}^T \left[ \ln \left( \frac{S_i}{S_{i-1}} \right) \right]^2 \quad (1)$$

Where  $S_i$  is the stock price on day  $i$ . We choose  $T$  to be 21, because on average there are twenty one trading days in a month. This means the value of the realized variance on a certain trading day is based on the returns for the past month.

Figure 1 shows the daily returns (multiplied by  $10^2$ ) from 01-07-2013 until 29-06-2018. As can be seen the returns vary much, with periods of extreme values in August and September 2015 and in February and March 2018. The first period falls in the 2015-2016 stock market selloff, which is a period in which stock prices declined globally. The volatility began in China due to uncertainty. On Monday, August 21 2015, the DJIA fell 3.17%, while in the next week the DJIA quickly recovered with record point gains. In February 2018 the DJIA fumbled 12% in two weeks, but after this the stocks recovered three-quarters of these losses within a couple of weeks. The reason for the big losses was the threat of inflation. The rise in volatility in these periods can also be seen in Figure 2, which shows the volatility of the DJIA in our five year period, this is calculated by taking the square root of the realized variance. The black line shows the mean volatility which is 6.133 (multiplied by  $10^{-3}$ ). As can be seen the value for volatility remains high a little bit longer than can be seen in the graph with the returns. The reason for this is that the realized volatility is based on the results of the past month.

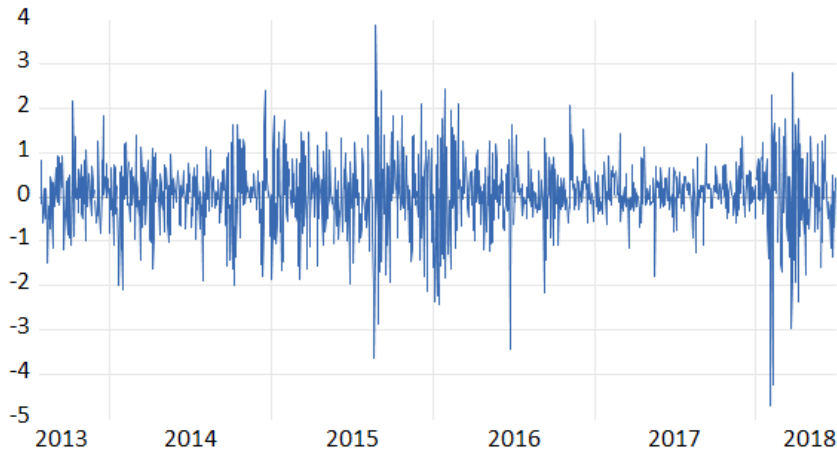


Figure 1: The daily returns of the DJIA multiplied by  $10^2$  for the period from 01-07-2013 until 29-06-2018

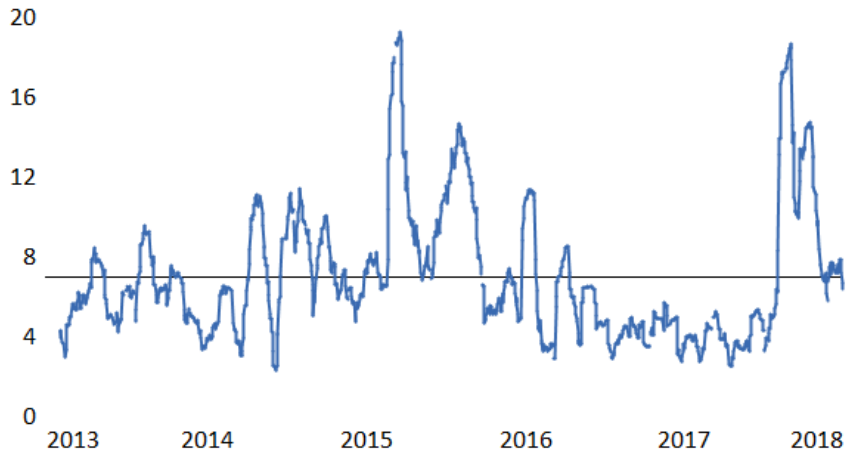


Figure 2: The realized volatility of the DJIA multiplied by  $10^3$  for the period from 01-07-2013 until 29-06-2018

Table 1 shows descriptive statistics of the daily simple returns, the realized volatility and the realized variance. The daily simple returns are calculated with the formula  $R_t = \frac{S_t - S_{t-1}}{S_{t-1}}$ . We see some interesting results in this table. The mean return is positive, which means the stock price of the DJIA has been rising on average. The skewness of the returns is negative, which means there are more extreme negative values. This is what is generally expected by stock markets, the market generally reacts stronger to bad news than to good news. The realized volatility has excess kurtosis, this means the price changes are too peaked and have too thick tails compared to a normal distribution. The results of the Jarque-Bera normality test point to the fact that the returns, the realized volatility and the realized variance are not following a standard distribution. There are 1262 data points, these are divided in three different samples. For robustness we divide the samples in two different ways. Firstly, we divide the data in a half, a quarter and another quarter and secondly we divide the data in three equal parts. Figure 3 shows the realized variances with lines for the dates where the data is split. The red lines show the points for the first way and the black lines show the points for when the data is split in three equal parts.

Table 1: Summary statistics of the daily returns, the realized volatility and the realized variance

	Daily returns ( $\times 10^2$ )	Realized volatility ( $\times 10^3$ )	Realized variance ( $\times 10^5$ )
Mean	0.038	7.096	6.133
St. dev	0.780	3.315	6.414
Skewness	-0.578	1.383	2.487
Kurtosis	6.667	4.910	9.941
Jarque-Bera	775.643 (0.000)	583.248 (0.000)	3765.073 (0.000)

Note: the notation ( $\times 10^i$ ) means that the data has been multiplied by this factor. The realized volatility is calculated by taking the square root of the realized variance. The null hypothesis of the Jarque-Bera test is normality.

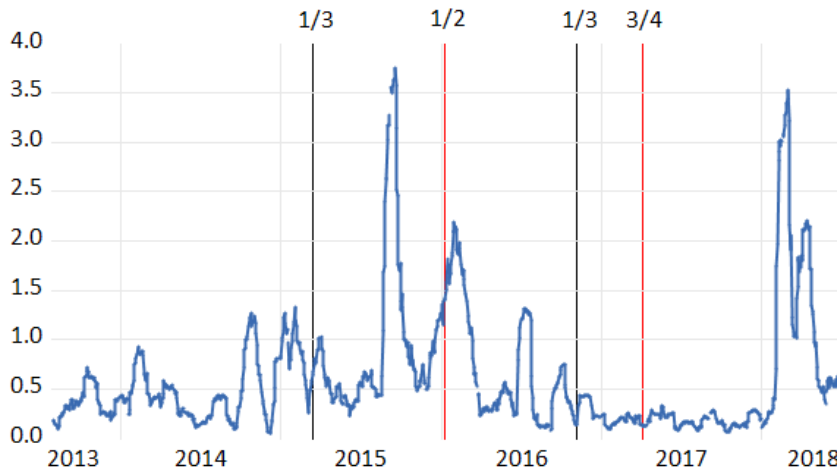


Figure 3: The realized variance of the DJIA multiplied by  $10^4$  for the period from 01-07-2013 until 29-06-2018 with lines showing the sampling of the data

## 4 Methodology

In the replication part and in the extension part we use four different models for forecasting. In this section we first explain how we divide our data and how we use each part. Secondly, we discuss the GARCH models we use for the forecast combination and for the data generating process. The next subsection is about how the weights are determined in each of our three methods. Lastly, we explain the way the combined forecasts are compared in their accuracy.

### 4.1 Data division

For both our simulation runs and for our actual data we divide our data in three parts. As explained in Section 3, we will use two different ways of dividing our empirical stock market data for more robustness in our results. Method *I* is done by first using the first half of the data, then a quarter and as third part the last quarter. The second method is *II*, where we divide



the data in three equal parts. The data from the DGP has 1000 data points for each simulation and the data from the DJIA has 1262 data points. Because we need data from the past month to calculate realized variance we have 1239 data points left.

The first part is used for parameter estimation for each of the four models. This means for our DGP the first 500 values are used and for the DJIA the first 620 or 413 values are used for parameter estimation, this is done by fitting the models to the data. Based on these parameters a forecast is made for the second half of the data by using a rolling window of data. The rolling window size is 500, 620 or 413 and we roll one observation ahead. This means each forecast is based respectively on the past 500, 620 or 413 observations.

The second part of the data consists of a quarter or one third of the data and is used for estimating the weights of each model. The forecasts for this part of the data are compared with the actual values. The weight determination is done differently for each of the three methods. Each of these ways are is explained in subsection 4.3.

The last quarter or last one-third of the data is used for comparing the different forecasts from each combination method, the way the forecasts are compared are explained in subsection 4.4.

## 4.2 GARCH models

The Generalized Autoregressive Conditional Heteroskedasticity (GARCH) model assumes that the conditional variance is a function of the errors at earlier times. The GARCH(1,1) model takes into account the lagged error from time  $t - 1$  and the lagged value of the conditional variance and is defined as follows:

$$\sigma_t^2 = \omega + \alpha\epsilon_{t-1}^2 + \beta\sigma_{t-1}^2 \quad (2)$$

Where  $\sigma_t^2$  is the conditional variance at time  $t$ , the  $\omega$  is a constant and the  $\alpha$  is the parameter for the lagged squared error term  $\epsilon_t$ .  $\beta$  is the parameter for the lagged conditional variance  $\sigma_{t-1}^2$ . To guarantee that  $\sigma_t^2 \geq 0$  for all  $t$ :  $\omega > 0$ ,  $\alpha > 0$  and  $\beta \geq 0$ .

In many GARCH models the return function looks as follows:

$$r_t = \mu + \epsilon_t \quad (3)$$

In this function  $r_t$  is the return on day  $t$  and  $\mu$  is the expected return based on the information set until  $t - 1$ .  $\epsilon_t$  can be written as  $\epsilon_t = z_t\sigma_t$ , with  $z_t \sim iid N(0, 1)$ .

Another model we use to combine different forecasting models is the GARCH(2,1) model, this model accounts for a second lagged error term from time  $t - 2$ . This model has the following equation:

$$\sigma_t^2 = \omega + \alpha_1\epsilon_{t-1}^2 + \alpha_2\epsilon_{t-2}^2 + \beta\sigma_{t-1}^2 \quad (4)$$

The threshold GARCH (TGARCH) model is an extension that accounts for asymmetric reactions to positive and negative shocks. This is done by adding a parameter  $\gamma$  for the lagged error term. The TGARCH(1,1) model has the following equation:

$$\sigma_t^2 = \omega + \alpha \epsilon_{t-1}^2 + \gamma \epsilon_{t-1} + \beta \sigma_{t-1}^2 \quad (5)$$

The  $\gamma$  accounts for different reactions when the shocks are negative.

We also use the TGARCH(2,1) model which has the following equation:

$$\sigma_t^2 = \omega + \alpha_1 \epsilon_{t-1}^2 + \alpha_2 \epsilon_{t-2}^2 + \gamma \epsilon_{t-1} + \beta \sigma_{t-1}^2 \quad (6)$$

The data we use for the simulation part is simulated by a data generating process (DGP). The DGP we use is a GARCH(3,3) model. This means the data is generated from the following equation:

$$\sigma_t^2 = \omega + \alpha_1 \epsilon_{t-1}^2 + \alpha_2 \epsilon_{t-2}^2 + \alpha_3 \epsilon_{t-3}^2 + \beta_1 \sigma_{t-1}^2 + \beta_2 \sigma_{t-2}^2 + \beta_3 \sigma_{t-3}^2 \quad (7)$$

To be sure of the robustness of our simulation we simulate with different parameters four times. Each time we make sure the values of the  $\alpha$  parameters add up to 0.1 and the  $\beta$  parameters add up to 0.8. The parameters we use are in Table 2. We used varying parameters on purpose, such that we investigate whether the Shapley method performs better than the equal weight method with different data generating processes.

Table 2: The parameters used for the four simulation runs

	$\omega$	$\alpha_1$	$\alpha_2$	$\alpha_3$	$\beta_1$	$\beta_2$	$\beta_3$	$\mu$
run 1	0.1	0.03	0.03	0.04	0.4	0.3	0.1	2
run 2	0.1	0.01	0.07	0.02	0.1	0.5	0.2	2
run 3	0.1	0.02	0.02	0.06	0.2	0.2	0.4	2
run 4	0.1	0.06	0.03	0.01	0.5	0.1	0.2	2

Note: this table shows the values of the parameters used for the DGP in Equation 7.

### 4.3 Forecast combination

#### 4.3.1 Shapley values

When combining different forecasts each of the forecasts gets a weight, with all the weights adding up to one. In case of  $K$  forecasts the Mincer Zarnowitz regression is used:

$$y_t = \alpha + \beta_1 f_{1,t} + \beta_2 f_{2,t} \dots + \beta_K f_{K,t} + \epsilon_t \quad (8)$$

In this regression  $y_t$  is the actual value the models try to predict. The Shapley-values then are computed as follows:

$$SH_j = \sum_{\substack{S \subset K \\ j \in S}} \frac{(s-j)!(k-s)!}{k!} [R^2(S) - R^2(Sj)] \quad (9)$$

In this equation  $R^2(S)$  is the  $R^2$  of the model with all forecasts in a set  $S \subset K$ . The  $R^2$  are calculated by regressing the forecasts on the actual value as in equation 8. The Shapley weights

are found as follows:

$$s_j = \frac{SH_j}{R_{12..K}^2} \quad (10)$$

The Shapley values weighted forecast is then computed as follows:

$$f_{SH,t} = s_1 f_{1,t} + s_2 f_{s,t} + \dots + s_3 f_{S,t} \quad (11)$$

### 4.3.2 Equal

The name of the equal weight method already explains the size of the weights. All weights are one divided by the amount of models combined. The combined forecast of the equal weight method is composed as follows with  $S$  is the amount of combined models:

$$f_{EQUAL,t} = \frac{1}{S} f_{1,t} + \frac{1}{S} f_{s,t} + \dots + \frac{1}{S} f_{S,t} \quad (12)$$

### 4.3.3 OLS

The OLS method first performs a linear regression on the forecasts as in the following equation:

$$y_t = \alpha + \beta_1 f_{1,t} + \beta_2 f_{2,t} + \dots + \beta_S f_{S,t} + \epsilon_t \quad (13)$$

The coefficients of this regression are then used as weights for the forecast combination, which results in a combined forecast as follows:

$$f_{OLS,t} = \beta_1 f_{1,t} + \beta_2 f_{2,t} + \dots + \beta_S f_{S,t} \quad (14)$$

## 4.4 Forecast comparison

The forecast performances of the combination methods are evaluated in absolute and relative terms.

We compare the different forecasts based on  $R^2$ . This is a measure that represents the proportion of the variance for a dependent variable that is explained by an independent variable in a regression model. It is defined as follows:

$$R^2 = 1 - \frac{SumSquaredRegression}{TotalSumofSquares} = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y}_i)^2} \quad (15)$$

Where  $\hat{y}_i$  is the forecast and  $\bar{y}_i$  is the average.

We also use the mean absolute error (MAE) to compare the forecast accuracy of the combination methods. The MAE is calculated by the following:

$$MAE = \frac{1}{n} \sum_{i=1}^N |\lambda_i - \hat{\lambda}_i| \quad (16)$$

In this equation  $\lambda_i$  stands for the actual realized variance and  $\hat{\lambda}_i$  is the forecasted variance. The MAE is an accuracy measure commonly used for financial data. This is the case because it is easy interpretable and is relatively robust to outliers compared to squared error functions.

We also compare the forecasts based on the one sided Diebold-Mariano test (Diebold & Mariano, 2002). We call the  $k$ -step forecasts of model  $i$  and model  $j$  respectively  $\hat{\lambda}_{i,t_k|t}$  and  $\hat{\lambda}_{j,t_k|t}$ , with corresponding errors  $\epsilon_{i,t_k|t}$  and  $\epsilon_{j,t_k|t}$ . The loss differential is the difference between those errors:  $d_{t+k} = \epsilon_{i,t_k|t} - \epsilon_{j,t_k|t}$ . The null hypothesis is that the two forecasting models have the same accuracy:  $H_0 : E(d_t) = 0 \forall t$  with alternative hypothesis  $H_1 : E(d_t) > 0$

The Diebold Mariano (DM) statistic is defined as follows:

$$DM = \frac{\bar{d}}{\sqrt{V(d_{t+k}) \setminus T}} \sim^{asy} N(0, 1) \quad (17)$$

$\bar{d}$  is the mean of the loss differentials,  $V(d_{t+k})$  is the variance of the loss differential, which is defined as follows:

$$V(d_{t+k}) = \frac{1}{T-1} \sum_{t=1}^T (d_{t+k} - \bar{d})^2 \quad (18)$$

We take a significance level of 0.05. The one sided DM test is a one-tailed test, which means the critical value is 1.645 (z-value)

## 5 Empirical results

In this section we discuss the results from the combined forecasts. First we compare the Shapley weighted forecast with the forecasts based on equal weights and OLS weights for four runs of 1000 simulations with a data generating process. Based on these results we conclude whether the Shapley combined forecast is not only theoretically better than the equally weighted forecast, but also empirically. Secondly, we compare the three combination methods for our actual data from the DJIA. To compute our results we made use of Python (2023) and the figures are made with EViews 11.

### 5.1 Data Generating Process

Table 3 shows the results of the  $R^2$  of the four runs from 1000 simulations with the data generating process. In the table the mean  $R^2$  is mentioned and the amount of times a method came out on top. The OLS method has the highest value of  $R^2$  the most amount of times in every run, so the OLS method explains the largest part of the variance most often. The mean  $R^2$  however, is higher for the Shapley method in three of the four runs. When we look at the average of all runs we conclude the OLS method has the highest  $R^2$  the majority of the time, but the Shapley method is more constantly reliable when it comes to the  $R^2$  with a higher mean  $R^2$ . Comparing the Shapley method with the equal weight method we conclude the Shapley method has a higher mean  $R^2$  and more often has the highest  $R^2$  of a simulation, this is the case for all four runs.

Table 3: The  $R^2$  of the combination models received in the simulations

$R^2$	Run 1		Run 2		Run 3		Run 4		Average	
	Mean	Wins	Mean	Wins	Mean	Wins	Mean	Wins	Mean	Wins
Shapley	<b>0.630</b>	298	0.521	256	<b>0.434</b>	332	<b>0.796</b>	246	<b>0.595</b>	283
Equal	0.604	108	0.486	97	0.412	118	0.775	138	0.569	115
OLS	0.598	<b>594</b>	<b>0.539</b>	<b>647</b>	0.402	<b>550</b>	0.768	<b>616</b>	0.577	<b>602</b>

Note: the notation ( $\times 10^i$ ) means that the data has been multiplied by this factor. The results from these accuracy measures are from the last quarter of the data. The numbers in bold show which of the combination methods comes out as the best in a category.

The mean MAE and the amount of times a method has the lowest MAE are in table 4. The results from this table are unambiguous. The Shapley method has the lowest MAE in every run, while also coming out on top in more than 62% of the simulations every run. It is clear that the OLS method is outperformed by both the Shapley method and the equal weight method. On average the MAE of the OLS method is around eight times higher than the MAE of the other two methods. The difference in accuracy between the Shapley and equal weight method is not very big. The difference between the mean MAE of the four runs is 0.000164. Despite this small mean difference the Shapley method does gets the lowest MAE in on average 64.5% of the simulations, while the equal weight method has the lowest MAE in 32.5% of the simulations.

Table 4: The MAE of the combination models received in the simulations

MAE ( $10^2$ )	Run 1		Run 2		Run 3		Run 4		Average	
	Mean	Wins	Mean	Wins	Mean	Wins	Mean	Wins	Mean	Wins
Shapley	<b>0.845</b>	<b>648</b>	<b>0.905</b>	<b>627</b>	<b>0.939</b>	<b>647</b>	<b>0.801</b>	<b>656</b>	<b>0.873</b>	<b>645</b>
Equal	0.862	312	0.918	350	0.957	321	0.817	315	0.889	325
OLS	6.451	40	8.944	23	7.302	32	5.581	29	7.070	31

Note: the notation ( $\times 10^i$ ) means that the data has been multiplied by this factor. The results from these accuracy measures are from the last quarter of the data. The numbers in bold show which of the combination methods comes out as the best in a category.

As mentioned the difference in accuracy between the Shapley method and the equal weight method does not seem very big based on the MAE. To test whether the Shapley method is performing significantly better than the equal weight method we perform one-sided Diebold Mariano tests. The results of these tests are in Table 5. We see that for each run the Shapley method performs significantly better than the equal weight method more often than the other way around. Except for in run 2 the Shapley forecast performs better in the majority of the 1000 simulations, in run 2 it was around half of the time with 498 times out of 1000. When we look at the average results from the Diebold Mariano test we conclude that in 53.5% of the times the Shapley method has a significantly higher accuracy, while the equal weight method only has a higher accuracy in 18.0% of the time.

Table 5: The results of the one-sided Diebold Mariano tests of the simulations

DM	Shapley	Equal
Run 1	559	163
Run 2	498	188
Run 3	534	172
Run 4	548	198
Average	535	180

The average Shapley weights allocated to each model are in table 6. The Shapley method gave weights between 0.156 and 0.360 to each of the four models. The biggest difference in weights is in run 2, when the GARCH(2,1) model performs much better than the GARCH(1,1) model. The reason for this, bigger than usual, difference in weights is that the  $\alpha_2$  of the DGP has the high value of 0.7, this means the GARCH(2,1) model fits better than the GARCH(1,1) model. Looking at the average Shapley weights they can be seen as fairly equal. This is as expected because the four models are a lot like each other. These fairly equal weights and similarities between the models are the reasons for the small difference in performance accuracy. When we look at the Diebold Mariano results we do see that this small difference in weight does result in significantly better forecasts most of the times.

Table 6: The Shapley weights received by each individual GARCH model in the simulations

Mean Shapley Weight	GARCH(1,1)	GARCH(2,1)	TGARCH(1,1)	TGARCH(2,1)
Run 1	0.190	0.345	0.231	0.235
Run 2	0.156	0.360	0.213	0.271
Run 3	0.189	0.355	0.223	0.233
Run 4	0.230	0.310	0.233	0.227
Average	0.192	0.343	0.225	0.241

Note: the weights in this table are based on the  $R^2$  in the third quarter of the data.

The weights allocated by the OLS method are in Table 7, something that directly stands out is the presence of negative weights. The weights of the OLS method are, contradictory to the Shapley weights, very different from the equal weights. As mentioned in Franses (2023) a problem with OLS might occur in case the forecasts are strongly correlated. A result of this correlation is by example negative weights. It is also visible that every GARCH model sometimes has a positive mean weight, while in another run with different parameters the model has a negative mean weight.

Table 7: The OLS weights received by each individual GARCH model in the simulations

Mean OLS Weight	GARCH(1,1)	GARCH(2,1)	TGARCH(1,1)	TGARCH(2,1)
Run 1	-0.099	2.664	6.963	-8.737
Run 2	-0.167	0.307	1.436	-0.875
Run 3	-0.086	-0.302	-1.021	2.178
Run 4	0.001	0.037	1.869	-1.149
Average	-0.878	0.677	2.312	-2.146

Note: the weights in this table are based on an OLS regression in the third quarter of the data.

The MAE of all individual GARCH models for every run are in Table 5.1. We see that the GARCH(2,1) model outperforms the other models in every run. This was to be expected based on the relatively high weights this model received from the Shapley method as can be seen in Table 6. When we look further into the MAE of the GARCH(2,1) model we see that the GARCH(2,1) model even has lower values for MAE than all the combination methods. This means that combining models does not definitely lead to a higher accuracy for this DGP.

MAE ( $\times 10^2$ )	GARCH(1,1)	GARCH(2,1)	TGARCH(1,1)	TGARCH(2,1)
Run 1	1.110	<b>0.834</b>	0.921	0.937
Run 2	1.177	<b>0.904</b>	0.980	0.998
Run 3	1.192	<b>0.924</b>	1.021	1.012
Run 4	1.048	<b>0.798</b>	0.908	0.890
Average	1.132	<b>0.865</b>	0.958	0.959

Note: the notation ( $\times 10^2$ ) means that the data has been multiplied by this factor. The numbers in bold show which of the models comes out as the best in a certain category. These accuracy measures are based on the actual values and the forecast of the last quarter of the data

## 5.2 Dow Jones Industrial Average

Table 8 shows the  $R^2$  and the MAE from the Shapley method, the equal weight method and the OLS method for our data from the DJIA. For our division method  $I$  we see that the OLS method results in the highest  $R^2$ , which means relatively the biggest part of the variance is explained by the OLS based forecast. We also see that the MAE is the lowest for the forecast based on OLS weights. This is a surprising result, as the OLS method came out as the worst method in our simulations. The second division method gives results that might be expected based on the simulations, the Shapley method and the equal weight method perform better based on both  $R^2$  and MAE. The equal weight method has a bit higher  $R^2$ , while the Shapley method has a lower MAE. For both accuracy measures there is not much of a difference between the Shapley method and the equal weight method. However, the Shapley method outperforms the equal weight method both times for MAE. The difference in  $R^2$  is below 0.01 each time and the difference in MAE is around 0.03 for both division methods.

To check whether there is a significant difference between the forecast accuracy of the three methods we perform one sided Diebold Mariano tests. When performing Diebold Mariano tests with method  $I$  between the OLS methods and the other methods we get respective p-values of

0.009 and 0.010 when comparing with the Shapley and the equal weight method. This means the OLS method significantly outperforms the other two methods for this sample choice. The outcome of this DM test is not surprising looking at the values of the accuracy measures in Table 8. For the data division with three equal parts the p-values of the Shapley method and the equal weight method outperforming the OLS method are respectively 0.058 and 0.144 This means that with our significance level of 0.05 the OLS method is not significantly outperformed by the other methods.

We also test whether the Shapley method is significantly better than the equal weight method. The p-value from method *I* is 0.021 and from method *II* 0.043. This means we do reject the null hypothesis of equal accuracy between these tests and the Shapley method works significantly better for the DJIA in this period.

Table 8: Forecast comparison DJIA

	$R_I^2$	MAE <sub><i>I</i></sub>	$R_{II}^2$	MAE <sub><i>II</i></sub>
Shapley	0.398	0.361	0.423	<b>0.291</b>
Equal	0.388	0.364	<b>0.427</b>	0.294
OLS	<b>0.668</b>	<b>0.321</b>	0.410	0.361

Note: the results from these accuracy measures are from the last quarter of the data.

Figure 4 shows the actual Realized Variance and the forecasts for all three methods of method *I* for the last quarter of the data. A few things stand out. The first thing that stands out is that it seems like the line for the Shapley method is missing. What is actually happening is that the differences between the Shapley forecast and the equal forecast are so small the lines are almost exactly 'on top of each other'. We also saw this in Table 8, where the values of the accuracy measures were very close to each other. Secondly, we see a high peak in the period in the end of February 2018. This is something we already discovered in the data section. However, we also learn something new from this figure. We see that the Shapley forecast and the Equal forecast are much higher than the actual variance, while the OLS forecast is much closer to the actual value. It is probable that this period with much irregularities is the biggest cause of the OLS forecast being significantly better than the others. This is confirmed by the fact that the MAE of the Shapley and equal weight method is actually lower. It seems that for the rest of the period the three forecasts are about as good as each other, this is something we investigate further in the last part of this result Section.



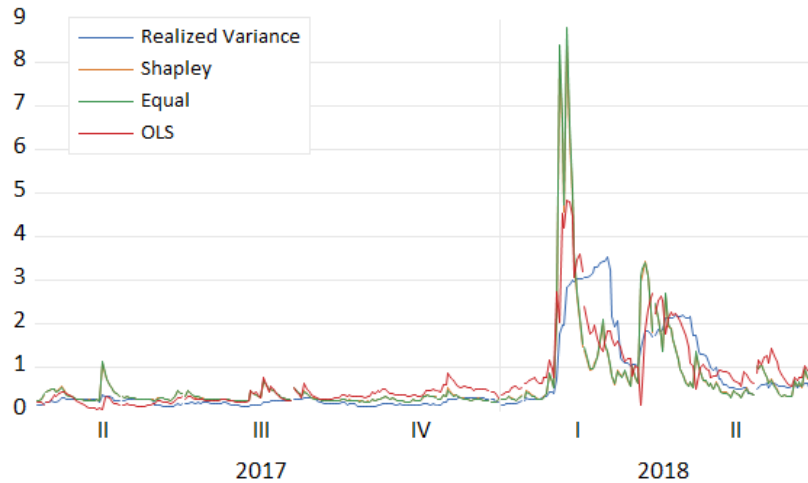


Figure 4: The actual Realized Variance and the three weighted forecasts with method *I*

Note: the results are from the last quarter of the data.

Figure 5 shows the last one-third of the Realized variance and the method *II* forecasts. The biggest difference with Figure 6 is the OLS forecast, which looks to be much worse. In the first quarter of 2018 there are even some negative values for the predicted variance, while this is not even possible in reality. Further the same things apply for this figure, with the Shapley method and the equal weight method showing almost no differences. Both of these forecasting methods get a much higher peak around February 2018.

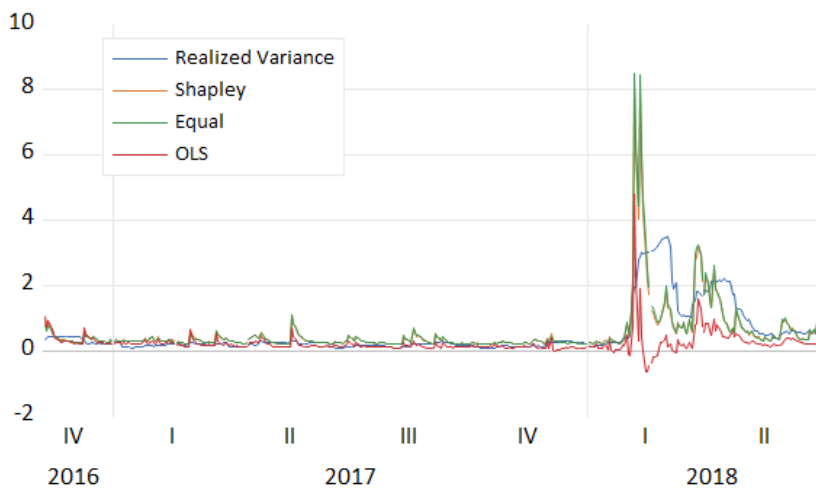


Figure 5: The actual Realized Variance and the three weighted forecasts with method *II*

Note: the results are from the last one-third of the data.

The reason the forecasts from the Shapley method and the equal weight method are so close to each other is that the weights do not differ very much from another as can be seen in Table 9.

The weights allocated to each GARCH model are between 0.147 and 0.372. These weights are based on the  $R^2$  from the individual models and combinations of all models. The first two rows of Table 9 show the  $R^2$  of the individual models in the second part of the data. We see that the models with a higher  $R^2$  get allocated a higher weight, which is what might be expected beforehand. The reason for this relatively equal weight allocation is that the models we use are very much alike. All four models are GARCH models with at most two additional variables compared to each other. The estimates from the parameters from the first half of the data are in Table 10. Here it is confirmed how much alike the models we use are. The value of omega is between 0.052 and 0.060 for all models and the estimates of  $\beta_1$  are all between 0.609 and 0.706. The most extreme finding is that with the value of  $\alpha_2$  being zero the TGARCH(1,1) and TGARCH(2,1) models are almost exactly the same. The difference between the values of the parameters only come after three decimals. When the models are much alike the accuracy of the models are also not far apart from each other. This can also be seen in the first two rows of Table 9 where we see that the  $R^2$  from the GARCH models do not differ very much.

In the fifth and sixth row of Table 9 we see the weights allocated by the OLS method. As we explained earlier this section the negative weights are due to correlation between the models. Two very extreme cases can be seen here, with the TGARCH models being almost exactly the same. To still profit from combining the forecasts the two models get allocated the extreme weights of -525.889 for the TGARCH(1,1) model and 525.562 for the TGARCH(2,1) model for method *I*. We see that in the case of our first division method it does work to have these negative weights, while for method *II* it does deliver worse results.

Table 9: Allocated weights of each method

	GARCH(1,1)	GARCH(2,1)	TGARCH(1,1)	TGARCH(2,1)
$R^2_I$	0.551	0.571	0.396	0.396
$R^2_{II}$	0.471	0.431	0.337	0.337
Shapley weight <sub><i>I</i></sub>	0.336	0.370	0.147	0.147
Shapley weight <sub><i>II</i></sub>	0.372	0.281	0.173	0.173
OLS weight <sub><i>I</i></sub>	-10.002	11.495	-525.889	525.562
OLS weight <sub><i>II</i></sub>	1.637	-0.877	268.868	-268.838

Note: this table shows the  $R^2$  of the GARCH models from the second part of the data. The weights are also based on the performance in this part of the data.

Table 10: Estimated parameters from first half of the data

	GARCH(1,1)	GARCH(2,1)	TGARCH(1,1)	TGARCH(2,1)
$\omega$	0.060	0.056	0.052	0.052
$\alpha_1$	0.278	0.214	0.134	0.134
$\alpha_2$		0.030		0.000
$\beta_1$	0.609	0.669	0.706	0.706
$\gamma$			0.166	0.166
$\mu$	0.083	0.068	0.052	0.052

Note: the parameters are based on the estimation in the first half of the data.

Table 11 shows the outcomes of the accuracy measures from each individual GARCH model. A few things stand out from these outcomes. Firstly, we see that the TGARCH models have (almost) the exact same outcomes. This was something that might be expected when looking at the parameters in Table 10. The GARCH models have higher  $R^2$  and MAE than the TGARCH models. This might be expected from the values of  $R^2$  in Table 9, where the GARCH models also had higher values for  $R^2$ . The GARCH(2,1) model outperforms all other models and even has better values for all performance measures than the combined forecasts based on Shapley values and on equal weights. The GARCH(1,1) model also has comparable values to the Shapley and equal weight method. The reason is that the performance of the combined forecasts gets dragged down by the TGARCH models.

We perform a DM test to check whether the GARCH(2,1) and the GARCH(1,1) model forecasts are significantly better than the Shapley values based forecast. For the GARCH(2,1) model we find p-values of 0.018 and 0.036 respectively for method *I* and *II*, which means this forecast is significantly better than the Shapley value based forecast. The p-values for the test with the GARCH(1,1) model respectively are 0.055 and 0.267, this is not significant as we work with a significance level of 5%. The fact that the GARCH(2,1) model performs better than the Shapley method, and thus the equal weight method, points to the fact that combining forecasts does not always result in a higher accuracy. The reason for the deterioration of the forecast when combining is the low accuracy from the TGARCH models.

Table 11: Forecast comparison of the GARCH models

	$R_I^2$	MAE <sub><i>I</i></sub>	$R_{II}^2$	MAE <sub><i>II</i></sub>
GARCH(1,1)	0.396	0.359	0.354	0.294
GARCH(2,1)	<b>0.427</b>	<b>0.343</b>	<b>0.474</b>	<b>0.287</b>
TGARCH(1,1)	0.367	0.395	0.422	0.312
TGARCH(2,1)	0.367	0.395	0.422	0.312

Note: these are the values of  $R^2$  and MAE from the last part of the data

As discussed and can be seen in Figure 4 and Figure 5 there is a specific period which has a large influence on the accuracy measures of the forecasts. This period is the first part of February 2018. When using data division method *I*, the week from the fifth of February until the ninth of February has an MAE of 4.246 from the Shapley method, this is an extremely big difference in comparison with the MAE of 0.361 in Table 8. We remove the 105 data points from the beginning of February 2018 until the end of our data from our forecasts and compare the smaller sample of forecasts. The results from this comparison are in Figure 5.2. Because we removed a very hard to predict volatile period the MAE becomes smaller for each combination method. The Shapley, equal weight and OLS method have a respective MAE of 0.124, 0.120 and 0.181. In contrast to when using the comparison sample from method *I* the OLS method has the highest MAE and the Shapley method has the lowest MAE. We also see this in Figure 5.2 where the OLS method often gives worse predictions than the other methods. When performing a DM test between the Shapley method and the OLS method we get a p-value of 0.000. Which means the Shapley method performs significantly better. We also test the null hypothesis of equal forecast accuracy between the Shapley and equal weight method against an alternative hypothesis of a

better equal weight method forecast. This results in a p-value of 0.334, which means there is no significant difference in performance between these methods. From this difference in results after removing the extremely volatile period we can conclude that these unstable periods in stock markets have a big influence on the relative performance of different forecasting methods.

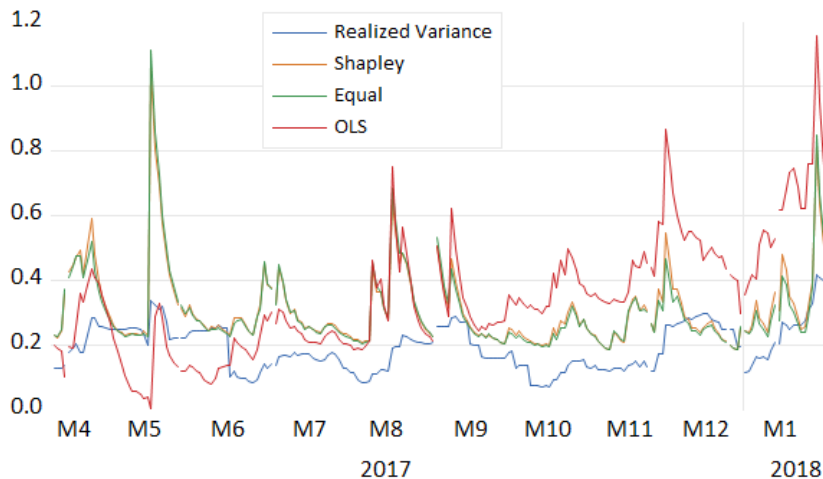


Figure 6:

## 6 Conclusion

In this paper we investigate whether the Shapley-values-based forecast combination is a good way to forecast the volatility of the DJIA. We partly adopt the methodology from Franses (2023) and extend this paper by performing a thousand simulations four times and testing the different combination methods on actual stock market data. From our simulation experiment we find the Shapley method outperforms both the equal weight method and the OLS method. The accuracy of the Shapley and equal weight method does not differ much, because the Shapley method weights are comparable with the equal weights. Still, the Shapley method has better average values for  $R^2$ , MAE and also is significantly better according to the Diebold Mariano test.

For our DJIA data we find the surprising result that the OLS method significantly outperforms the Shapley and equal weight method when dividing our data according to method  $I$ . With data division with three equal parts the OLS method performs worse than the other two methods as expected. When comparing the Shapley method and equal weight method among each other we do find that the Shapley method produces significantly more accurate predictions for both ways of data division. When looking further into the accuracy of the models where the combined forecasts were composed of we see that the TGARCH(1,1) and TGARCH(2,1) model are almost identical for this data. More importantly, we see that the GARCH(2,1) makes significantly better forecasts than the Shapley and equal weight method. For this data combining forecasts does not lead to better results. Both the simulation results and the DJIA results lead us to conclude that the benchmark  $1/N$  method indeed can be improved by using the Shapley

method.

The field of combining forecasts is a major field not yet fully explored. We believe there are many methods of forecast combination still to be discovered and tested. In the future the Shapley method could be compared to newly discovered combination methods. For comparison between the Shapley method and the equal weight method there are also a few things that could be done differently in the future. The first thing that could be done in another way is that it would be advisable to use models that differ more. In our research the TGARCH models were almost identical, this does not help in the reliability of our outcomes. Also on a bigger scale four GARCH models are quite similar, some more variation might result in different results. Looking back on our research this is something we could have done differently. Secondly, more models could be included to test whether the Shapley method also works better with more forecasts to combine. With different and with more models way the Shapley weights might be less equal to the equal weights. The research could also be done with different data sets, such as the New York Stock Exchange, the NASDAQ or the Japan Stock Exchange. Lastly, one could specifically look for difference in accuracy in periods with much volatility or in periods with little volatility. As seen in our results from the DJIA data the biggest difference between the accuracy of the methods is in the peak in February 2018. Some methods could work better in highly volatile markets, while other methods perform better in a quiet market.

## References

- Bates, J. M. & Granger, C. W. (1969). The combination of forecasts. *Journal of the Operational Research Society*, 20(4), 451–468.
- Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31(3), 307–327.
- Charles, A. & Darné, O. (2014). Large shocks in the volatility of the dow jones industrial average index: 1928–2013. *Journal of Banking & Finance*, 43, 188–199.
- Clemen, R. T. (1989). Combining forecasts: A review and annotated bibliography. *International Journal of Forecasting*, 5(4), 559–583.
- Diebold, F. X. & Mariano, R. S. (2002). Comparing predictive accuracy. *Journal of Business & Economic Statistics*, 20(1), 134–144.
- Franses, P. H. (2023). Shapley-values-based forecast combination.
- Gokcan, S. (2000). Forecasting volatility of emerging stock markets: linear versus non-linear garch models. *Journal of Forecasting*, 19(6), 499–504.
- Granger, C. W. & Ramanathan, R. (1984). Improved methods of combining forecasts. *Journal of Forecasting*, 3(2), 197–204.
- Poon, S.-H. & Granger, C. W. J. (2003). Forecasting volatility in financial markets: A review. *Journal of Economic Literature*, 41(2), 478–539.
- Sharma, S., Aggarwal, V. & Yadav, M. P. (2021). Comparison of linear and non-linear garch models for forecasting volatility of select emerging countries. *Journal of Advances in Management Research*, 18(4), 526–547.
- Smetanina, E. (2017). Real-time garch. *Journal of Financial Econometrics*, 15(4), 561–601.

- Stock, J. H. & Watson, M. W. (1998). A comparison of linear and nonlinear univariate models for forecasting macroeconomic time series. *National Bureau of Economic Research Cambridge, Mass., USA*.
- Timmermann, A. (2006). Forecast combinations. *Handbook of Economic Forecasting, 1*, 135–196.
- Wang, X., Hyndman, R. J., Li, F. & Kang, Y. (2022). Forecast combinations: an over 50-year review. *International Journal of Forecasting*.
- Wu, J. (2010). Threshold garch model: Theory and application. *The University of Western Ontario*, 1–42.

## A Calculation of Shapley weights with four models

In our paper we combine four models with the Shapley value based method. In this Appendix we show how the calculation of each of the Shapley weights in our paper. The regression is as follows:

$$y_t = \alpha + \beta_1 f_{1,t} + \beta_2 f_{2,t} \dots + \beta_K f_{K,t} + \epsilon_t \quad (19)$$

To compute Shapley values we need all the  $R^2$  from each individual and each combined forecast. The Shapley values are computed as follows:

$$\begin{aligned} SH_1 = & \frac{1}{4}R_1^2 + \frac{1}{12}(R_{12}^2 - R_2^2) + \frac{1}{12}(R_{13}^2 - R_3^2) + \frac{1}{12}(R_{14}^2 - R_4^2) + \frac{1}{12}(R_{123}^2 - R_{23}^2) + \frac{1}{12}(R_{124}^2 - R_{24}^2) \\ & + \frac{1}{12}(R_{134}^2 - R_{34}^2) + \frac{1}{4}(R_{1234}^2 - R_{234}^2) \end{aligned}$$

$$\begin{aligned} SH_2 = & \frac{1}{4}R_2^2 + \frac{1}{2}(R_{12}^2 - R_1^2) + \frac{1}{12}(R_{23}^2 - R_3^2) + \frac{1}{12}(R_{24}^2 - R_4^2) + \frac{1}{12}(R_{123}^2 - R_{13}^2) + \frac{1}{12}(R_{124}^2 - R_{14}^2) \\ & + \frac{1}{12}(R_{234}^2 - R_{34}^2) + \frac{1}{4}(R_{1234}^2 - R_{134}^2) \end{aligned}$$

$$\begin{aligned} SH_3 = & \frac{1}{4}R_3^2 + \frac{1}{12}(R_{13}^2 - R_1^2) + \frac{1}{12}(R_{23}^2 - R_2^2) + \frac{1}{12}(R_{34}^2 - R_4^2) + \frac{1}{12}(R_{123}^2 - R_{12}^2) + \frac{1}{12}(R_{134}^2 - R_{14}^2) \\ & + \frac{1}{12}(R_{234}^2 - R_{24}^2) + \frac{1}{4}(R_{1234}^2 - R_{124}^2) \end{aligned}$$

$$\begin{aligned} SH_4 = & \frac{1}{4}R_4^2 + \frac{1}{12}(R_{14}^2 - R_1^2) + \frac{1}{12}(R_{24}^2 - R_2^2) + \frac{1}{12}(R_{34}^2 - R_3^2) + \frac{1}{12}(R_{124}^2 - R_{12}^2) + \frac{1}{12}(R_{134}^2 - R_{13}^2) \\ & + \frac{1}{12}(R_{234}^2 - R_{23}^2) + \frac{1}{4}(R_{1234}^2 - R_{123}^2) \end{aligned}$$

The weights allocated to each forecast are calculated with the following formula:

$$s_i = \frac{SH_i}{r_{1234}^2} \quad (20)$$

## B Programming code explanation

The code is attached in the zip file. To make the code fully understandable we explain a few things from the code in this Appendix. We run the code in two separate ways for the research paper. First we do a thousand simulations. For this way we choose the main file to have “data = makegdp()”, this way the actual data is a GARCH(3,3) model newly generated for each run. For the DJIA data it should say “data = pd.read\_excel('finaldatacorrect2.xlsx')” and the actual data should be in an excel file with this name in the same directory as the Python file. The return should be under ‘Returns’ and the realized variance should be under ‘Variance’. In the RSQUARES class and in the GARCH11 class there are variables called a, b and c. These respectively represent the point of a quarter, the half and the third quarter of the data. For the simulation these should be 250, 500 and 750. For the DJIA data it should be 310, 620 and 930 or 413, 826 and 413. Also f in the GARCH11 class should be 500 for the simulations and 620 or 413 for the DJIA data. and ”Length In the Comparison class the variables named ‘a’ should have a value of 250 for simulations and 309 or 413 for the actual data. One final remark is that the value of ‘n’ in the main class determines how many times the loop should run. This should be a thousand for the simulation and just one for the stock market data.