Adaptive Supervised Learning for Volatility Targeting Models

Eric Benhamou, David Saltiel, Serge Tabachnik, Corentin Bourdeix, François Chareyron, Beatrice Guez

Adaptive Supervised Learning for Volatility Targeting Models

Eric Benhamou[1,2], David Saltiel[2], Serge Tabachnik[3,4], Corentin Bourdeix[3], François Chareyron[3], and Beatrice Guez[2]

[1] Dauphine University, France
[2] Ai for Alpha, France
[3] Lombard Odier Asset Management, Switzerland
[4] Panthéon-Assas University, France

Abstract. In the context of risk-based portfolio construction and pro-active risk management, finding robust predictors of future realised volatility is paramount to achieving optimal performance. Volatility has been documented in economics literature to exhibit pronounced persistence with clusters of high or low volatility regimes and to mean-revert to a normal level, underpinning Nobel prize-winning work on Generalized Autoregressive Heteroskedastic (GARCH) models. From a Reinforcement Learning (RL) point of view, this process can be interpreted as a model-based RL approach where the goal of the models is twofold: first, to represent the volatility dynamics and forecast its term structure and second, to compute a resulting allocation to match a given target volatility: hence the name "volatility targeting method for risk-based portfolios". However, the resulting volatility model-based RL approaches are hard to distinguish as each model results in similar performance without a clear dominant one. We therefore present an innovative approach with an additional supervised learning step to predict the best model(s), based on historical performance ordering of RL models. Our contribution shows that adding a supervised learning overlay to decide which model(s) to use provides improvement over a naive benchmark consisting in averaging all RL models. A salient ingredient in this supervised learning task is to adaptively select features based on their significance, thanks to minimum importance filtering. This work extends our previous work on combining model-free and model-based RL. It mixes different types of learning procedures, namely model-based RL and supervised learning opening new doors to combine different machine learning approaches.

Keywords: Volatility targeting · Supervised learning · Best ordering · Model-based · Portfolio allocation · Walk-forward · Features selection.

1 Introduction

Volatility targeting has become popular in the financial industry and academic literature, with the specific objective to design a portfolio, made up of cash and risky assets with a constant predetermined level of portfolio volatility over time. This risk budgeting methodology overcomes the difficulty of inferring future expected performance and covariance structure of the risky assets as required by the traditional mean–variance framework created by Markowitz in 1952 [27]. The strategy, which was first introduced by Qian in 2005 [32], allocates capital between risky and non-risky assets, thanks to volatility forecasts of the risky assets. If the volatility of the risky assets is expected to be high, the weight of the risky asset is reduced and vice versa as presented in [11], [14], [19] and [8]. In addition, [15] has shown that left tail risk is significantly reduced thanks to a targeted portfolio's exposure inversely proportional to the forecasted portfolio volatility. However, this is contrasted by the work of [29] that states exactly the opposite, motivating for a machine learning overlay to determine which volatility targeting models to favor. In general, the economic value of time volatility, measured by significant performance and utility gains for investors who synchronize volatility in multi-asset portfolios, has been extensively reviewed by [16], [17], [22] and [38].

This has motivated the search for a stream of models to forecast volatility based on multiple methodologies that attempt to harness the informational content of this inherently unobservable latent process. These volatility models range from a simple empirical approach leveraging moving averages to more sophisticated ones based on ARCH and GARCH forecasts [9], [10], Heterogeneous Autoregressive model of Realized Volatility (HAR-RV), proposed by [13], multivariate high-frequency-based volatility (HEAVY) models [37]

and prospective forward-looking measures such as implied volatility [24].

From a modern and machine learning perspective, most of these volatility predictors can be interpreted as model-based reinforcement learning models. They aim to specify the complex structure of volatility and to make specific predictions on which portfolio allocations are sized in inverse proportion to predicted volatility. In certain cases, their parameters are determined to maximize specific performance measures such as the Sharpe ratio of the corresponding strategy. Although initially not formulated as a model-based RL approach, volatility targeting follows its traditional ingredients. The process attempts to identify an evolution of a complex system. Optimal allocation is determined thanks to a cumulative reward. The supervised learning task considered in this paper fulfills specific criteria: the portfolio allocation across the different volatility models. However, because of the complexity and non-stationary behavior of volatility, the appropriate choice of the volatility targeting model remains an open question. Furthermore, the intrinsic high correlation between these volatility models, both in terms of predictors and corresponding performances, present an interesting challenge for classification and supervised learning tasks.

This is precisely the goal of this work. We aim to determine the best allocation for volatility targeting models in order to achieve the maximum *ex post* Sharpe ratio.

1.1 Related Works

Our work can be related to an ever growing body of machine learning literature applied to financial markets. Indeed, with ever increasing competition and data processing speed in the financial markets, robust forecasting methods have become a vital subject for asset managers. The premise of machine learning algorithms - to offer a way to find and model non-linearity behaviour in financial time series - has attracted ample attention and efforts that can be traced back to the late 2000's when machine learning research started to pick up in the financial industry. Rather than listing a large related body of work, we will refer to various publications that have reviewed the existing literature in chronological order.

In 2009, [3] surveyed more than 100 related published articles using neural and neuro-fuzzy techniques derived and applied to forecasting stock markets, or discussing classifications of financial market data and forecasting methods. In 2010, [23] conducted a survey on the application of artificial neural networks in forecasting financial market prices, including exchange rates, stock prices, and financial crisis prediction as well as option pricing. In addition, the stream of machine learning publications was not only based on neural network but also on generic and evolutionary algorithms as reviewed in [1].

More recently, [39] reviewed the application of cutting-edge NLP techniques for financial forecasting, using text from financial news or Twitters. [33] covered the wider topic of using machine learning techniques, including deep learning, to financial portfolio allocation and optimization systems. [30] focused on support vector machine and artificial neural networks to forecast prices and regimes based on fundamental and technical analysis. Later, [36] discussed the challenges and research opportunities of machine learning techniques as applied to finance, including issues for algorithmic trading, backtesting and live testing on single stocks and more general predictions of financial markets. In addition, [35] reviewed not only deep learning methods but also other machine learning methods to forecast financial time series. As the recent focus has been mostly on deep learning approaches, it is not surprising that most of the reviewed articles are related to this field. Finally in [25], the authors compare machine learning to conventional quantitative research methodologies in finance and discuss the idiosyncrasies of finance and the challenges that financial markets pose to machine learning methodologies. They also examine the opportunities (and applications) that machine learning offers for financial research.

While there are existing studies related to machine learning applied portfolio optimization including volatility targeting strategies (e.g. [20], [28] and [40]), the specific problem of volatility model allocation using machine learning in the volatility targeting context is fairly novel. This stems from the two approaches being traditionally very different in spirit: volatility targeting assumes rules while machine learning prides itself on having no *a priori* or preconceived rules.

Reformulating volatility targeting methods as a model-based RL approach as in [5] opens new doors. In this work, [5] show that using a model free Deep Reinforcement Learning (DRL) approach can help to decide which volatility model(s) to choose. This is underpinned by the fact that DRL appears to be a promising tool to tackle regime changes [4], [6] or [7].

However, DRL faces at least two issues. First, DRL can be problematic in financial markets as

only one scenario, namely the historical one, can be replayed as opposed to games where a simulator can generate many multiples of experiences (such as chess or the game of Go). Although some new synthetic market data-generating processes exist as nonparametric, AI-based Monte Carlo methods and neural network–based approaches including variational autoencoders (VAEs) and generative adversarial networks (GANs), none of these techniques are completely satisfactory in replicating realistic market scenarios even if they can be useful to mitigate overfitting on financial datasets [31]. Second, DRL does not work well when using many features and is still hard to interpret as opposed to a supervised learning method based on decision tree algorithms that can rank and classify specific real features thereby rendering them more tangible and interpretable. In our past specific volatility targeting allocation exercise, choosing the right model(s) in the context of a DRL approach has proven to be a serious challenge [5].

In this study, we are interested in testing a new supervised method that leverages macro data as well as model performance data to select the best volatility targeting model(s). In addition, this new methodology enables a certain understanding of the role of the selected features and the detection of regime shifts thanks to the selected features' changes.

1.2 Contributions

Our contribution is precisely motivated by the shortcomings presented in the aforementioned remarks. We therefore add a supervised learning overlay to decide which model to use. The goal of this supervised learning task is to order all our models and choose the best model or weight them according to their predicted relative performance. As many data sources are used, we add, in our supervised learning task, a feature filtering step to determine the critical features.

The motivation is twofold. Firstly, using too many features tends to add noise to our signal as some features are either non-predictive or redundant with other features. Secondly, we would like to change the selected features when there are regime changes. We therefore add, in our supervised learning step, a filtering selection. We test the salience of this features selection as well as a hyperparameter selection process thanks to a walk-forward approach as described later in the paper.

2 Models Presentation

2.1 Volatility Targeting Models

In our study, we use nine different models to represent and forecast volatility. These models are identical to the ones presented in [5]. They begin with simple concepts such as moving averages, exponential moving averages (RM2006) or a two-step approach as presented in [26] to account for a short memory process. Then, they incorporate more sophisticated and statistically-justified models like the famed GARCH model, as in [12], the Glosten-Jagannathan-Runkle GARCH model as in [18], microstructure-based models like the HEAVY model (which relies on high-frequency data as in [37]), or the HAR model as in [13]. Last but not least, we add two additional prospective implicit models that are based on forward-looking variables such as the VIX. One of them uses Principal Component Analysis to decompose a set of implied volatility indices into the main eigenvectors. The resulting implied volatility proxies are then rescaled to match a realized volatility metric thereby voiding the famous volatility risk premium and Jensen's inequality-related bias. These nine models are quite diverse overall and work alternatively well in various market environments, making the choice of the best model or any type of ranking difficult to tackle.

As presented in [5], volatility targeting is achieved using the volatility forecasts given by the models. If we denote by σ_{target} the target volatility of the strategy and if the model i predicts a future's volatility $\sigma_{t-1}^{i,pred}$, based on information available at time $t-1$, the allocation in the future's model i at time t is given by the ratio between the target volatility and the predicted volatility $b_t^i = \sigma_{target}/\sigma_{t-1}^{i,pred}$.

Hence, we can compute the daily amounts invested in each of the future volatility models and create a corresponding time series of returns $r_t^i = b_t^i \times r_t^{future}$, consisting of investing according to the allocation computed by the volatility targeting model i. This provides n time series of returns r_t^i. The striking point, as illustrated by Figure 1 is the very high correlation between the different volatility targeting models' returns, thereby motivating the search for a robust method to determine which model(s) to include in the portfolio construction process. As volatility regimes change, the best model varies over time, making the exercise of choosing the best model non-trivial.

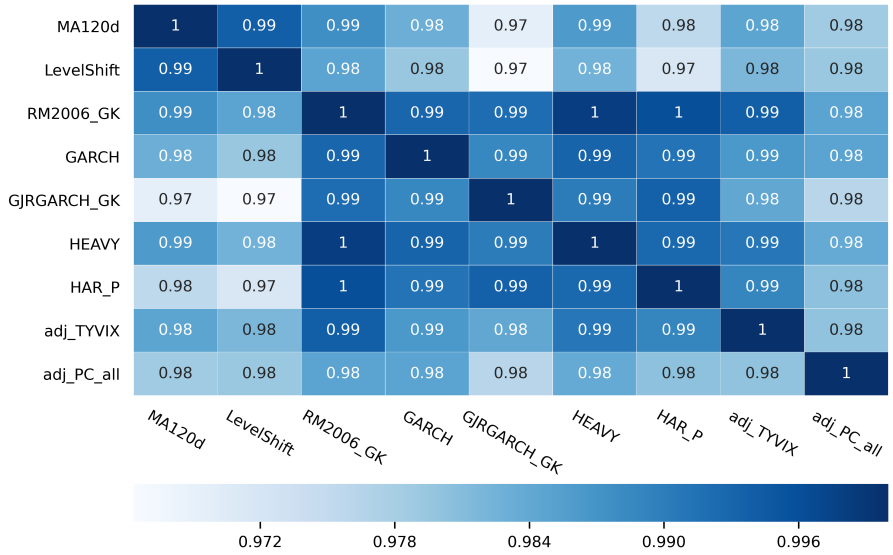

Fig. 1: Correlation between volatility targeting models for the U.S. 10-year Treasury Note futures contract

2.2 Supervised Learning Overlay

In order to identify the best volatility model(s) with a supervised learning task, we use Gradient Boosting Decision Trees (GBDT) to find the model(s) that provide(s) the best next timestep return. More specifically, we selected LightGBM (LGBM) as our framework [21]. The label(s) are for each day the best model(s). The inputs are various financial and non-financial data that are described at length later in this paper. The specific choices of the GBDT model, in particular LGBM, are motivated by multiple reasons:

- GBDT are the most suitable machine learning methods for small data set classification problems. In particular, they are known to perform better than their state-of-the-art cousins, Deep Learning methods, for small data sets. As a matter of fact, GBDT methods are preferred by Kagglers and have won multiple challenges.
- GBDT methods are less sensitive to data rescaling, compared to logistic regression or penalized methods.
- They can cope with imbalanced data sets.
- They allow for very fast training when using the leaf-wise tree growth (compared to level-wise tree growth).
- Last but not least, among GBDT methods, LGBM is one of the most numerically efficient method, with computing time divided by 2 to 3 compared to XGBoost.

2.3 Features

Our supervised learning task aims to determine the label (the best volatility targeting models) for a given set of inputs. These inputs are derived from several features :

- **PnLs series** : the 9 PnL times series of the different models.
- **Volatility forecasts**: the forecast of the nine volatility models.
- **Market indicators**: the values of the S&P 500, Nasdaq, Dow Jones, Euro Stoxx, FTSE, Nikkei, MSCI World, Emerging Markets and ACWI indices, that represent the most important equity index markets.
- **Commodities indicators**: the prices of several commodity indicators spanning the following sectors: energy markets, industrial and precious metals, agriculture, grains, livestock and soft commodities sectors.
- **FX carry indices**: the 4 carry indices between the dollar and the Euro, the Swiss franc, the Pound sterling and the Japanese Yen.
- **Macro indices** : 5 features corresponding to different proprietary market risk appetites.
- **Bond indicators** : levels of the U.S., U.K., Japan and German 10-year government bonds and the slope computed as the difference between the 10 and 2 years yields.
- **Economic surprises** : economic surprise indices based on the G20 countries, Europe, the U.S., Asia, emerging markets.

– **Implied volatility indicators** : several implied volatility indicators based on FX carry indices, Gold, Crude oil, government bonds and equities indices.
– **Credit spreads** : the TED spreads of the U.S. and the E.U. as well as other international spread indicators.
– **Put/Call ratio** : the put/call ratio for the S&P 500 option market.

Including the volatility model PnL time series, the market indicators, the commodity indicators and the FX carry indices, there is a total of 32 features in a first set. In addition, we compute other statistics such as averages, standard deviations, Sharpe ratios and other related technical indicators, summing up to a total of 416 input features in a second set. Volatility forecasts and macro indices represent another 13 features. We also calculate other statistical figures including short-term moving averages and distance from the average to get an additional 78 inputs. The last 78 macro features are not transformed. In total, we end up with 572 input features in our supervised learning model. We use LightGBM gain as feature importance to select a certain percentage of the most important features, and retrain LightGBM with them to filter the number of features every time the algorithm is trained.

2.4 Walk-Forward Methodology

The whole procedure is summarized in Figure 2. We use n models to represent the dynamics of the market volatility. We then add other features that provide orthogonal information to the models, such as economic health indices and technical analysis. *In fine*, the most important features are selected. As mentioned previously, LightGBM enables us to rank our various volatility targeting models according to the forecasted return probability of the next timestep. We then construct several strategies using these predictions. In order to test the robustness of our aggregated meta-model out-of-sample, we use the well-known methodology called walk-forward analysis.

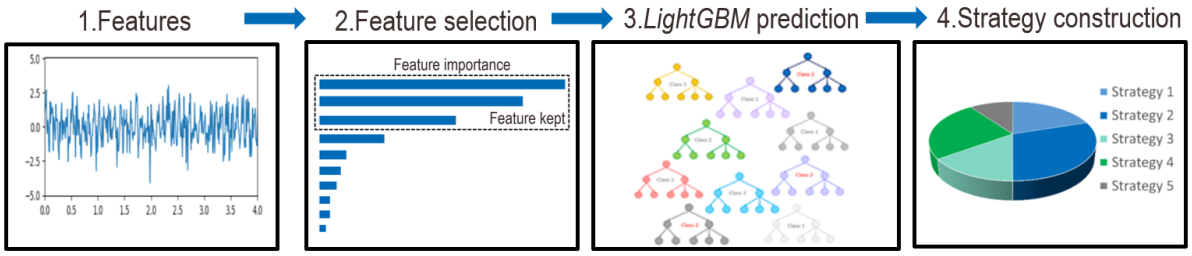

Fig. 2: Overall architecture of the supervised learning task applied to the selection and allocation of volatility targeting strategies

In machine learning, a standard approach is to do k-fold cross-validation. This approach happens to break the chronology of the data and, in certain ill-conceived cases, may potentially use test data in the train set which embodies a blatant case of in-sample optimization. [34] has conducted a comprehensive empirical study of eight common validation schemes and demonstrated that commonly used cross-validation schemes often yield estimates with the largest bias and variance, while forward-validation or walk-forward schemes yield better estimates of the out-of-sample error, even more so in the context of financial time series. We therefore settled on a forward-sliding test set which uses non-overlapping previous past data as the corresponding training sets. In other words, we favor adding incrementally new data in the training set, at each new step, to ensure stability and robustness in the validation process. This method is sometimes referred to as an "anchored walk-forward" validation methodology as we have anchored the training data. Finally, as the test set is always ensuing the training set, the walk-forward analysis creates fewer steps compared with a cross-validation approach which could lead to another source of overfitting.

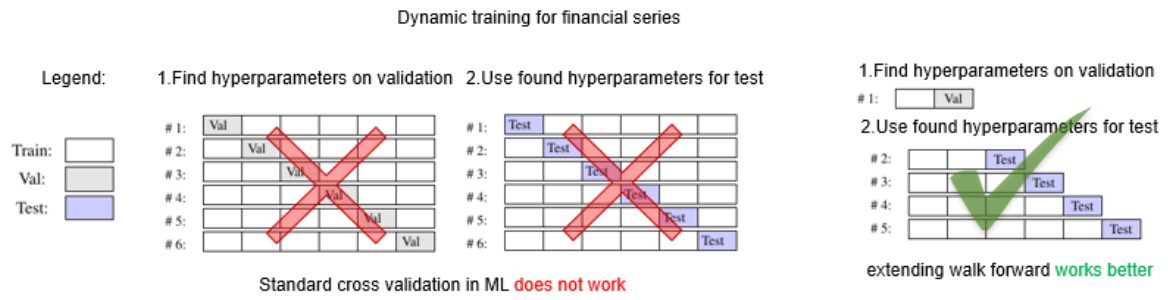

Fig. 3: Overall training, validation and test processes for a walk-forward methodology

In practice, and for our given data sets, we train our models from 2003 to the end of 2010 (giving us at least 7 years of training period), and then use a repetitive validation period of one year from 2011 up to 2016 to determine the best hyperparameters thanks to Optuna. This is a recent open-source Python library developed in 2019 which mixes sampling and pruning algorithms with Bayesian optimization [2].

Once these hyperparameters are determined, we train our model from 2003 to 2016 and use a repetitive test period of one year from 2017 onwards. In other words, we test the model out-of-sample on the 2017-2020 period. The entire process is summarized in Figure 3.

This method enables the calibration of the hyperparameters on a validation set. These LightGBM hyperparameters are:

- *max_depth*, *boost_round*,
- *min_sum_hessian_in_leaf*,
- *min_gain_to_split*,
- *feature_fraction*,
- *bagging_fraction*,
- *lambda_l2*,
- *min_importance*.

3 Financial Data Experiments

3.1 Market Data

We test our model on two mainstream futures contracts: the continuously rolled CBOT's future on the U.S. 10-year Treasury Note denoted by *TY1* in the rest of the paper, and the continuously rolled CME's E-mini future on the S&P 500 index denoted by *ES1*. As mentioned before, daily data range from 2003 to 2020.

3.2 Hyperparameters and Features Selection

Table 1 shows the details of the selected hyperparameters for the two futures contracts *TY1* and *ES1* during the validation period using the Sharpe ratio as the reward function. For *TY1*, the model has a maximum depth of one and selects approximately one hundred features. For *ES1*, the best model is very different. We find that it corresponds to a more complex system with a maximum depth of two, but fewer features are selected, which is somehow intuitive as maximum depth and number of features play similar roles. These results are summarized in Table 1.

Hyperparameters	*TY1*	*ES1*
max_depth	1	2
boost_round	650	800
min_importance	0.7	0.275
min_sum_hessian_in_leaf	0.125	1.04
min_gain_to_split	0.432	0.028
feature_fraction	0.686	0.834
bagging_fraction	0.49	0.553
lambda_l2	0.184	0.647

Table 1: Hyperparameters selected for *TY1* and *ES1* during the validation period (2011-2016)

For *TY1*, the selected *min_importance* hyperparameter shows that around one hundred features are responsible for 70% of all explanatory information that is used to predict which models have better returns at the next timestep. Then, each year, the model selects the most important features that account for 70% of the total gain. It uses 650 trees with a maximal depth of 1. For *ES1*, it is the same logic but with different hyperparameters. Tables 2 and 3 show the 10 most important features se-

lected on the test set from the 572 inputs. Notably, features directly related to volatility models are very often selected. The model focuses on volatility changes and abnormal behaviors using the distance from the average and the percentage change over 10 days. These 10 features are responsible for more than 39% of the total information. They all relate to the volatility of *TY1*.

For both *TY1* and *ES1*, models are retrained each year. Table 2 provides the most important features for *TY1*. Table 4 confirms the stability of the supervised methodology as about 85% of selected features remain identical between two consecutive years. For *ES1*, the conclusion is similar, but we can notice the large presence of volatility features as the model selects the same 24 features each year on average (Table 4).

Feature name	2017	2018	2019	2020	Average
GJRGARCH_GK_Vol_distance_to_250_MA	**6.10**%	**6.37**%	**6.87**%	**7.67**%	**6.75**%
HEAVY_Vol_distance_to_250_MA	5.91%	5.64%	6.16%	6.01%	5.93%
HAR_P_Vol_distance_to_250_MA	4.58%	5.06%	5.34%	5.50%	5.12%
GARCH_Vol_distance_to_250_MA	4.90%	4.82%	5.19%	4.89%	4.95%
adj_PC_all_std_60	4.65%	4.41%	3.94%	4.43%	4.36%
LevelShift_Vol_pct_change_10	2.99%	2.87%	2.79%	2.93%	2.89%
HEAVY_Vol_pct_change_10	3.03%	2.71%	2.59%	2.48%	2.70%
adj_PC_all_std_125	2.89%	2.63%	2.47%	1.71%	2.43%
GJRGARCH_GK_std_125	1.80%	2.28%	2.90%	2.24%	2.31%
GJRGARCH_GK_Vol	1.88%	2.28%	2.47%	1.96%	2.15%

Table 2: Normalized feature importance of the 10 most important features for *TY1*

Feature name	2017	2018	2019	2020	Average
RAEGARCH_vol_distance_to_250_MA	**10.86**%	**11.76**%	**11.61**%	**12.05**%	**11.57**%
GARCH_vol_distance_to_250_MA	7.49%	7.93%	7.81%	6.71%	7.48%
adj_VIX9D_vol_distance_to_250_MA	5.56%	8.03%	8.32%	7.57%	7.37%
LevelShift_std_60	5.56%	6.74%	7.91%	8.01%	7.05%
adj_VIX9D_vol	7.31%	6.79%	6.51%	6.97%	6.90%
MacroSignal_diff_1	6.90%	6.01%	5.05%	4.67%	5.66%
HEAVY_vol_distance_to_250_MA	4.32%	4.69%	6.19%	5.61%	5.21%
RM2006_GK_vol_pct_change_1	4.12%	4.20%	4.36%	4.31%	4.25%
adj_VIX9D_vol_pct_change_5	4.26%	3.95%	4.22%	4.12%	4.14%
adj_PC_equity_DM_vol_distance_to_250_MA	3.82%	3.64%	4.12%	4.20%	3.95%

Table 3: Normalized feature importance of the 10 most important features for *ES1*

Year	Features rate on *TY1*	Features number on *TY1*	Features rate on *ES1*	Features number on *ES1*
2017	/	98	/	26
2018	84%	95	92%	24
2019	87%	93	100%	23
2020	82%	94	88%	24

Table 4: Number and rates of features kept for *TY1* and *ES1*

3.3 Comparison with Benchmark

The model output is a probability for each volatility model to have the highest return at the next timestep. We therefore examine several strategies from these probabilities :

- **Weighted**: the probability of each model is used directly as an allocation,
- **3 best**: only the three best models are taken into account and allocation is computed as the three re-normalized probabilities,
- **Follow the best**: 100% investment in the model with the highest probability.

These strategies are compared to the **benchmark** which is an equal-weighted strategy.

Figure 4 shows the allocation of the weighted strategy on *TY1*. The model focuses on few volatility models such as *adjusted TYVIX*, *adjusted Principal Component* and *moving average 120-days*. But the probabilities are not stable which indicates that this strategy exhibits a potentially high turnover.

For *ES1*, the model is less stable. This is confirmed by the *weighted* strategy on Figure 5 as allocations switch frequently between several models.

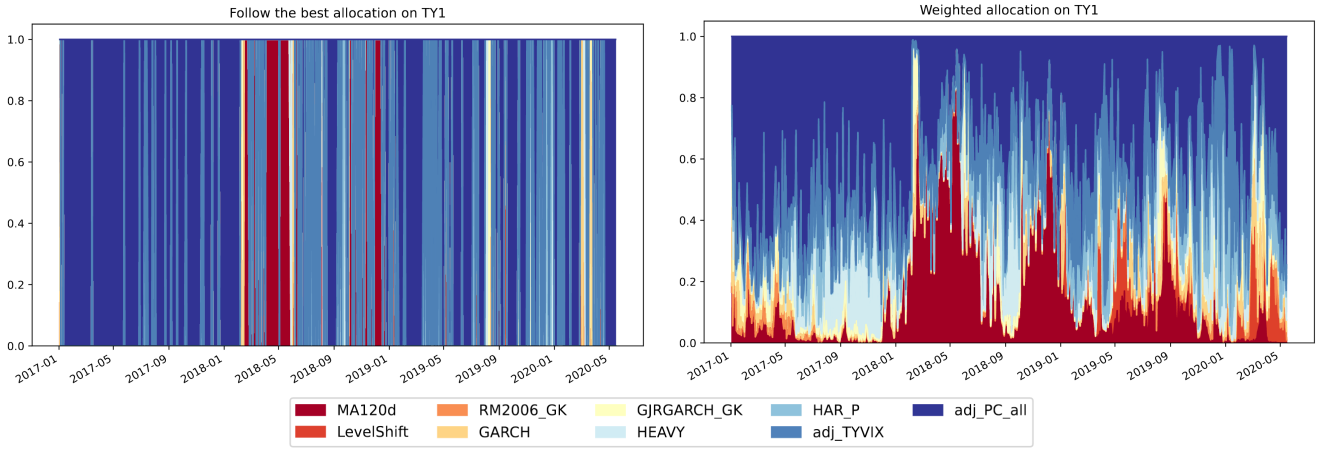

Fig. 4: Allocation of the *weighted* and *follow the best* strategies for *TY1*

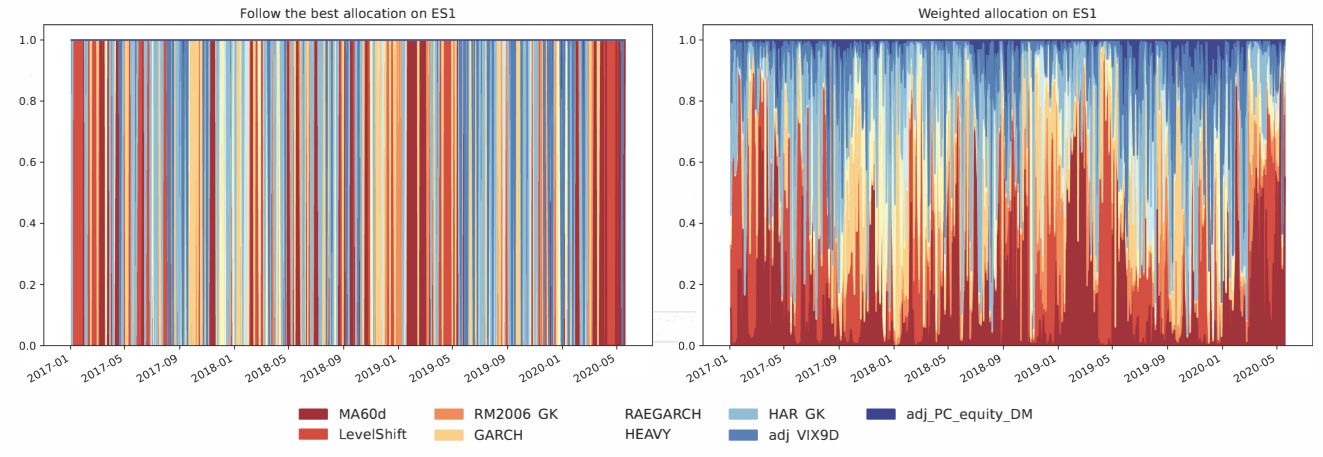

Fig. 5: Allocation of the *weighted* and *follow the best* strategies for *ES1*

For *TY1*, the *follow the best* strategy yields better results than the others as it exhibits a better performance with a lower volatility, resulting in a higher Sharpe ratio. Different performances are displayed in Figure 6. In addition, the *follow the best* strategy has the smallest drawdown which indicates that the model invests in the high-volatility model when markets plummet. However, the turnover is seven times higher than the *benchmark* which implies much larger transaction costs (TC) and reduces the final performance. Note that we have used a heuristic linear 2 bps per trade model for transaction costs applied to both the fixed income and equity proxies. From Figure 4, a large majority of the volatility models do not have a high probability of being the best at the next timestep, hence the allocation between *weighted* and *3 best* strategies are extremely alike and give similar performances. This information is provided in Table 5.

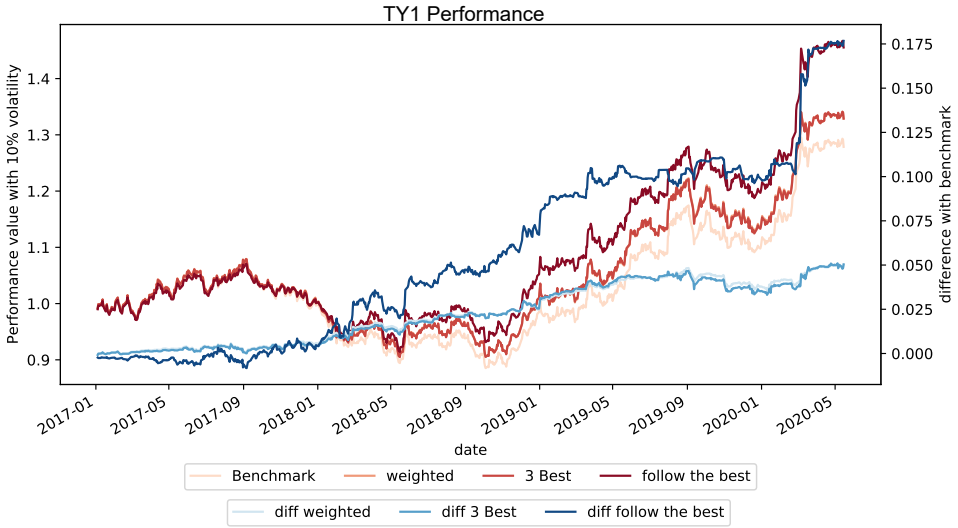

Fig. 6: *TY1* strategy performance on the test set

	Benchmark	*Weighted*	*3 Best*	*FtB without TC*	*FtB with TC*
Ann. ret	0.068	0.073	0.072	**0.098**	0.083
Ann. vol	9.0%	8.3%	**8.2%**	8.3%	8.3%
Sharpe ratio	0.759	0.883	0.883	**1.176**	1.001
Sortino	1.177	1.368	1.368	**1.875**	1.601
Drawdown	16.2%	13.5%	13.4%	**12.2%**	13.1%
Ann. Turnover	**8.90**	31.47	33.96	66.16	66.16

Table 5: *TY1* strategy performance for the test period (2017-2020)

Conversely, Figure 7 shows that the *follow the best* strategy is riskier than the corresponding *benchmark* for *ES1*. The *weighted* and *3 best* strategies have lower performance than the *benchmark* but are better than *follow the best*. The volatility of the output strategies have a high volatility compared to the *benchmark*, which is due to the frequent changes between volatility models. It is therefore not surprising that these strategies do not exhibit strong returns - the *benchmark* strategy having the best Sharpe ratio. In addition, the supervised model does not seem to work as intended as the *follow the best* strategy has the worst drawdown. We can therefore conclude that the model does not predict well which volatility model will have the best return at the next timestep for *ES1*. Further information is provided in Table 6.

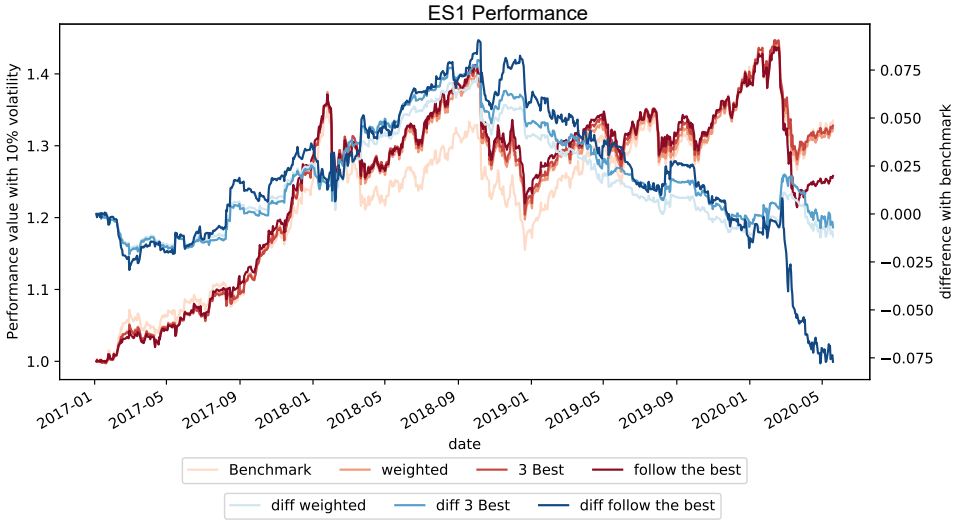

Fig. 7: *ES1* strategy performance on the test set

	Benchmark	*Weighted*	*3 Best*	*FtB without TC*	*FtB with TC*
Ann. ret	0.088	0.091	**0.094**	0.084	0.071
Ann. vol	**9.9%**	10.6%	10.8%	12.1%	12.1%
Sharpe ratio	**0.891**	0.861	0.872	0.695	0.583
Sortino	**1.198**	1.147	1.165	0.938	0.802
Drawdown	**14.7%**	15.2%	15.4%	18.4%	18.5%
Ann. Turnover	**8.24**	35.98	39.18	62.06	62.06

Table 6: *ES1* strategy performance for the test period (2017-2020)

3.4 Future Works

This study has shown how to construct strategies where the best models are selected by a supervised learning overlay. This adaptive supervised learning methodology leads to good results on the U.S. 10-year Treasury Note futures contract at the cost of a higher turnover. A direct extension of this work is therefore related to explore several techniques that can reduce turnover. In addition, we have found that this approach leads to less efficient results on the E-mini S&P 500 index futures contract. It is therefore worthwhile to better understand the major differences when applying this methodology to stock and bond futures. Finally, these two financial instruments, albeit among the most liquid in the world, represent a very small sample that could be extended to dozens of other financial futures contracts. More conclusive statistics could be reached if applied to a more complete set of generic futures contracts. All these steps are for further exploration in future works.

4 Conclusion

Volatility targeting strategies rely on precise forecasts of future volatility. Hence, they require reliable models for predicting the volatility dynamic, which is, obviously, a crucial and complex task given the inherently noisy and complex nature of financial markets. In practice, deciding which model to use is challenging due to market regime changes, the non-stationarity nature of these markets, and the high correlation between volatility models. In this paper, we present a supervised learning task to determine at each timestep which model(s) to overweight. In terms of input features, we not only use past performances of the different models and their forecasted volatility, but additional macro and financial data to determine the optimal model(s). We apply this methodology, with mixed success, to the case of volatility targeting strategies for the U.S. 10-year Treasury Note and the E-mini S&P 500 index futures contracts. We show that GBDT is an effective method to

predict the best model(s) in the fixed income case while further studies must be conducted for equity indices. We filter our features by importance and observe that the selected features are mostly model-related such as the distance of the volatility forecast to its mean.

References

1. Aguilar-Rivera, R., Valenzuela-Rendón, M., Rodríguez-Ortiz, J.: Genetic algorithms and darwinian approaches in financial applications: A survey. Expert Systems with Applications **42**(21), 7684–7697 (2015)
2. Akiba, T., Sano1, S., Yanase1, T., Ohta1, T., Koyama, M.: Optuna: A next-generation hyperparameter optimization framework. arxiv (2019)
3. Atsalakis, G.S., Valavanis, K.P.: Surveying stock market forecasting techniques – part ii: Soft computing methods. Expert Systems with Applications **36**(3, Part 2), 5932–5941 (2009)
4. Benhamou, E., Saltiel, D., Ohana, J.J., Atif, J., Laraki, R.: Deep reinforcement learning (drl) for portfolio allocation. In: Dong, Y., Ifrim, G., Mladenić, D., Saunders, C., Van Hoecke, S. (eds.) Machine Learning and Knowledge Discovery in Databases. Applied Data Science and Demo Track. pp. 527–531. Springer International Publishing, Cham (2021)
5. Benhamou, E., Saltiel, D., Tabachnik, S., Wong, S.K., Chareyron, F.: Adaptive learning for financial markets mixing model-based and model-free rl for volatility targeting. In: AAAMAS: ALA. AAAI Press (2021)
6. Benhamou, E., Saltiel, D., Ungari, S., Mukhopadhyay, A.: Bridging the gap between markowitz planning and deep reinforcement learning. In: Proceedings of the 30th International Conference on Automated Planning and Scheduling (ICAPS): PRL. AAAI Press (2020)
7. Benhamou, E., Saltiel, D., Ungari, S., Mukhopadhyay, A.: Time your hedge with deep reinforcement learning. In: Proceedings of the 30th International Conference on Automated Planning and Scheduling (ICAPS): FinPlan. AAAI Press (2020)
8. Benjamin Bruder, T.R.: Managing risk exposures using the risk budgeting approach. SSRN Electronic Journal 2009778 (2012)
9. Bollerslev, T.: Generalized autoregressive conditional heteroskedasticity. Journal of Econometrics **31**, 307–327 (1986)
10. Bollerslev, T., Chou, R.Y., Kroner, K.F.: Arch modeling in finance: A review of the theory and empirical evidence. Journal of Econometrics **52**, 5–59 (1992)
11. Bollerslev, T., Hood, B., Huss, J., Pedersen, L.H.: Risk everywhere: Modeling and managing volatility. Review of Financial Studies **31**(7), 2729–2773 (2018)
12. Bollerslev, T., Zhou, H.: Volatility puzzles: a simple framework for gauging return-volatility regressions. Journal of Econometrics **131**, 123–150 (2006)
13. Corsi, F.: A simple approximate long-memory model of realized volatility. Journal of Financial Econometrics **7**, 174–196 (2009)
14. Denis Chaves, Jason Hsu, F.L., Shakernia, O.: Risk parity portfolio vs. other asset allocation heuristic portfolios. The Journal of Investing **20**(1), 108–118 (2011)
15. Dreyer, A.A., Hubrich, S.: Tail-risk mitigation with managed volatility strategies. Journal of Investment Strategies **8**(2956), 1 (2019)
16. Fleming, J., Kirby, C., Ostdiek, B.: The economic value of volatility timing. Journal of Finance **56**, 329–352 (2001)
17. Fleming, J., Kirby, C., Ostdiek, B.: The economic value of volatility timing using realized volatility. Journal of Financial Economics **67**, 473–509 (2003)
18. Glosten, L.R., Jagannathan, R., Runkle, D.E.: On the relation between the expected value and the volatility of the nominal excess return on stocks. Journal of Finance **48**(5), 1779–1801 (1993)
19. Harvey, C.R., Hoyle, E., Korgaonkar, R., Rattray, S., Sargaison, M., Van Hemert, O.: The impact of volatility targeting. Journal of Portfolio Management **45**(1), 14–33 (2018)
20. Jaeger, M., Krügel, S., Marinelli, D., Papenbrock, J., Schwendner, P.: Interpretable machine learning for diversified portfolio construction. The Journal of Financial Data Science (2021)
21. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.Y.: Lightgbm: A highly efficient gradient boosting decision tree. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30, pp. 3146–3154. Curran Associates, Inc. (2017)
22. Kirby, C., Ostdiek, B.: Its all in the timing: simple active portfolio strategies that outperform naive diversification. Journal of Financial and Quantitative Analysis **47**, 437–467 (2012)
23. Li, Y., Ma, W.: Applications of artificial neural networks in financial economics: A survey. In: 2010 International Symposium on Computational Intelligence and Design. vol. 1, pp. 211–214 (2010)
24. Liang, C., Wei, Y., Zhang, Y.: Is implied volatility more informative for forecasting realized volatility: An international perspective. Journal of Forecasting **39**, 1253–1276 (2020)
25. Lommers, K., Harzli, O.E., Kim, J.: Confronting machine learning with financial research. The Journal of Financial Data Science (2021)
26. Lu, Y.K.: Modeling and forecasting stock return volatility using a random level shift model. Journal of Empirical Finance pp. 138–156 (2009)
27. Markowitz, H.: Portfolio selection. The Journal of Finance **7**(1), 77–91 (Mar 1952)

28. Molyboga, M.: A modified hierarchical risk parity framework for portfolio management. The Journal of Financial Data Science **2**(3), 128–139 (2020)
29. Mylnikov, G.: Volatility targeting: It's complicated! The Journal of Portfolio Management (2021)
30. Nti, I.K., Adekoya, A.F., Weyori, B.A.: A systematic review of fundamental and technical analysis of stock market predictions. Artificial Intelligence Review pp. 1–51 (2019)
31. Pardo, F.D.M., López, R.C.: Mitigating overfitting on financial datasets with generative adversarial networks. The Journal of Financial Data Science **2**(1), 76–85 (2020)
32. Qian, E.: Risk parity portfolios. Panagora Asset Managemen (2005)
33. Rundo, F., Trenta, F., di Stallo, A.L., Battiato, S.: Machine learning for quantitative finance applications: A survey. Applied Sciences **9**(24), 5574 (2019)
34. Schnaubelt, M.: A comparison of machine learning model validation schemes for non-stationary time series data. FAU Discussion Papers in Economics p. 42 (2019)
35. Sezer, O.B., Gudelek, M.U., Ozbayoglu, A.M.: Financial time series forecasting with deep learning: A systematic literature review: 2005-2019. arXiv preprint arXiv:1911.13288 (2019)
36. Shah, D., Isah, H., Zulkernine, F.: Stock market analysis: A review and taxonomy of prediction techniques. International Journal of Financial Studies **7**(2), 26 (2019)
37. Shephard, N., Sheppard, K.: Realising the future: forecasting with high-frequency-based volatility (heavy) models. Journal of Applied Econometrics **25**(2), 197–231 (2010)
38. Taylor, N.: The economic value of volatility forecasts: A conditional approach. Journal of Financial Econometrics **12**, 433–478 (2014)
39. Xing, F.Z., Cambria, E., Welsch, R.E.: Natural language based financial forecasting: a survey. Artificial Intelligence Review **50**(1), 49–73 (2018)
40. Zhang, Z., Zohren, S., Roberts, S.: Deep learning for portfolio optimization. The Journal of Financial Data Science **2**(4), 8–20 (2020)