Estimating Ensemble weights for Bagging Regressors based on the Mean-Variance portfolio framework

Javier Pérez-Rodríguez[a,*], Francisco Fernández-Navarro[b], Thomas Ashley[a]

[a] *Department of Quantitative Methods, Universidad Loyola Andalucia, Andalucia, 41704, Spain,*
[b] *Department of Computer Languages and Computer Science, University of Málaga, Spain*

Abstract

This paper presents a novel ensemble learning framework inspired by modern portfolio optimization to address regression problems. This formulation in the ensemble learning field allows determining the ensemble's weights considering the error predictions of the base learners, the variability in their predictions, and the covariance among their predictions' errors, which is completely aligned with the bias-variance-covariance theory. Under the framework proposed, four potential instantiations have also been provided. The first two ensemble models impose the non-negativity constraints on the ensemble's weights (along with the equality constraint that the ensemble's weights sum up to one) and are solved with the active set algorithm. The second two ensemble models do not include the non-negativity constraints on the ensemble's weights (which in the financial literature are called the non-shorting constraints), giving rise to a convex quadratic programming (QP) problem (as the matrix included in the quadratic term is symmetric and positive definite) that is solved by the Lagrangian procedure. Extensive experiments with regression datasets evaluate the proposed ensemble framework. Comparisons with other state-of-the-art ensemble methods confirm that the ensemble framework yields the best overall performance.

Keywords: Ensemble Learning, Bagging, Portfolio Optimization, Mean-variance portfolio

*Corresponding author

Email addresses: jperez@uloyola.es (Javier Pérez-Rodríguez), fafernandez@uloyola.es (Francisco Fernández-Navarro), tiashley@uloyola.es (Thomas Ashley)

July 21, 2025

1. Introduction

In ensemble learning, the bagging approach aims at improving prediction capability by combining $S \in \mathbb{R}$ approximation functions, f_s, estimated on a set of S bootstrapping samples on the training set, $s = 1, \ldots, S$ (Breiman, 1996a). Many empirical studies have already reported performance improvements due to the bagging approach's implementation (Breiman, 1996a; Bühlmann and Yu, 2002; Grandvalet, 2004). The reason for this competitive performance is in terms of the bias-variance trade-off, as bagging reduces the variance of the aggregated predictor maintaining the bias almost constant (Bauer and Kohavi, 1999). In this regard, Friedman and Hall (1999) (Friedman and Hall, 2007) analyzed the effect of bagging in terms of the decomposition of statistical estimators into linear and higher order parts. They pointed out that bagging reduces the variability of the nonlinear component by substituting it with an estimate of its expected value whilst leaving the linear part unchanged.

In regression problems, the bagging predictor, f, is obtained as the average of S regression functions based on a set of bootstrapping samples, $f = \frac{1}{S}\sum_{s=1}^{S} f_s$. Under this formulation, all base learners composing the ensemble have the same importance in the ensemble's final output (Breiman, 1996a). However, several studies have revealed that a simple average of models is not always the best and that a weighted ensemble can provide, in certain circumstances, better prediction results (Ekbal and Saha, 2013; Bhasuran et al., 2016; Peykani et al., 2019). Whereas overfitting with a complex ensemble is also undesirable, as considered in the context of the recent COVID19 pandemic (Benítez-Peña et al., 2021).

In this regard, Perrone and Cooper (1992) (Perrone and Cooper, 1992) compared the performance of two ensemble models: the Basic Ensemble Method (BEM), which combines several regression base learners by averaging their outputs, and the Generalized Ensemble Method (GEM), which combines estimates of base learners by finding the optimal weight to aggregate them in a way that the final ensemble minimizes the prediction error (Perrone and Cooper, 1992; Shahhosseini et al., 2022). The authors demonstrate that BEM can reduce the mean square error of the predictions by a factor of S, the number of estimators. Furthermore, their numerical ex-

periments showed that the GEM model is not only consistently better than the BEM model but also outperforms the best individual regressor, which empirically justified the previously mentioned point stating that a weighted ensemble model can outperform the traditional bagging model (Shahhosseini et al., 2022).

Motivated by this drawback of the original model, several authors have proposed different approaches to combine the base learners differently (Breiman, 1996a). The most straightforward weighting approach for regression problems is assigning weights proportional to each base learner's accuracy performance in a validation set (Opitz and Shavlik, 1995). Another interesting approach for establishing the ensemble's weights is to solve the least squares problem (linear regression) with the base learners' outputs and the desired target variable. This approach is similar to GEM, with the difference that the weights are not constrained to sum to one (Breiman, 1996b). The methodology was referred to by its author as Stacked Regression, SR, and can be incorporated or not, in its formulation, the regularization term (Stacked Ridge Regression, SRR) (Breiman, 1996b).

In bagging approaches, diversity is promoted implicitly in the ensemble by modifying the training data (Reeve and Brown, 2018; Perales-González et al., 2019). Thus, the standard Bagging models BEM, GEM, and SRR mentioned previously achieve diversity implicitly by using resampling to create different training sets for each base model. Similarly, in the Random Subspace method (Barandiaran, 1998), the individuals composing the ensemble are trained on randomly chosen subspaces of the original attribute space, i.e., individual training sets are sampled from the attribute space. On the other hand, Random Forest encourages diversity by training different trees with a different bootstrap of the data and splitting the branches along different feature subsets (Breiman, 2001).

Another alternative approach for combining base learners' output is to implement a neural network model that takes the base learners' output as input and tries to minimize the mean squared error of the combiner concerning the desired output (Yang and Browne, 2004). The main difference between this approach and the previously described approaches lies in the nonlinear nature of the combiner. It is important to clarify that the base learners included in the first ensemble level can be any combination of machine learning regressors. From a different perspective, Pham and Olafsson (2020) (Pham and Olafsson, 2020) proposed to replace the regular average with a Cesáro average, and the approach was tested in the field of Random

Forest, although it is extensible to other types of methods. The Ambiguity Bagging Method (ABM) proposed in Krogh and Vedelsby (1995) (Krogh and Vedelsby, 1994) included in the computation of the weights the concepts of generalization error and ambiguity (diversity), being novel for incorporating diversity both in the data sampling but also in the weights' estimation. Thus, the authors empirically showed the importance of including diversity also in the weights' computation. This approach is similar to the one proposed in (Zhou et al., 2002), in which the ensemble's weights are determined through a genetic algorithm to minimize a function that estimates the generalization error of the ensemble. The minimization function is the same as the ambiguity term included in (Krogh and Vedelsby, 1994). In addition, in (Zhou et al., 2002), the ensemble's weights are constrained to be greater than zero and sum to one.

All those research studies are closely related to the approaches included under the umbrella of Negative Correlation Learning (NCL) ensembles (Liu and Yao, 1999c; Liu et al., 2000; Perales-González et al., 2020; Perales-González et al., 2021). Under this framework, base learners are trained to produce outputs negatively correlated (diversity) with the ensemble outputs (in conjunction with the performance criteria, the minimization of the mean squared error). This penalty term is included to encourage specialization and cooperation among the base learners. Thus, diversity is fostered explicitly in the NCL (unlike bagging) as it is directly included in the error functions of the ensemble's regressors. The framework has been implemented in neural networks (Liu and Yao, 1999a,b) and support vector machines (SVM)(J. Zhou and Wang, 2020). In any case, it is important to clarify that the ensemble's weights in the NCL framework are all assumed to be equal ($w_s = \frac{1}{S}, s = 1, \ldots, S$), and the training stage is focused on the estimation of base learners' parameters. The importance of minimizing correlation among base learners was also explored in (Hashem, 1997), which illustrates theoretically and empirically how collinearity among the base learners of the ensemble can have harmful effects on the estimation of the optimal weights when base learners are linearly combined. The author proposed two approaches to improve the ensemble's performance by dropping some of the collinear regressors. The first methodology considers collinearity between the outputs of the base learners, and the second one collinearity between their errors. From a similar approach, the authors in (Dutta, 2009) proposed several metrics to measure diversity in ensemble models for regression. Specifically, they proposed the correlation coefficient, covariance, chi-square, entropy, and

a slightly modified disagreement measure (Kuncheva and Whitaker, 2001).

In the previously described studies, the ensemble's weights are fixed and determined from the training or validation data (using one of the ideas described in the previous paragraphs). Another possibility for setting the weights is to do it dynamically; namely, the weights are determined for each pattern in the test set. For example, in the study by Jimenez (1998) (Jiménez, 1998), the weights are recomputed dynamically from the respective certainties of the base learners' outputs. The more certain a base learner seems to be of its decision, the higher the weight. Under the same paradigm, Shen and Kong (2004) (Shen and Kong, 2004) presented another dynamically weighted ensemble for regression problems using the natural idea that the more accurate a base learner seems to be of its prediction, the higher the weight.

As seen in previous manuscripts (Krogh and Vedelsby, 1994; Perales-González et al., 2021), ensemble diversity, is a central issue in ensemble learning. It is usual to claim (in the scientific literature) that the success of an ensemble model lies in achieving a good tradeoff between the individual performance of its component and the diversity among them (Hansen and Salamon, 1990; Brown et al., 2005a; Kadkhodaei et al., 2020). In this regard, Krogh and Vedelsby (1995) (Krogh and Vedelsby, 1994) theoretically prove that at a single data point, the quadratic error of the ensemble estimator is guaranteed to be less than or equal to the average quadratic error of the component estimators due to the diversity component:

$$(f-y)^2 = \sum_{s=1}^{S} w_s (f_s - y)^2 - \sum_{s=1}^{S} w_s (f_s - f)^2, \tag{1}$$

where y is the target value, $\sum_{s=1}^{S} w_s = 1$, the first term of the decomposition is the weighted average error of the individuals, and the second one the ambiguity term (diversity) (the larger the ambiguity term, the larger the ensemble error reduction).

There is also a theory within the ensemble learning literature called the bias variance-covariance decomposition theory, commonly implemented in practice to generate competitive ensemble models. This theory breaks the mean squared error (MSE) into three components and the optimum "diversity" optimally balances the components to reduce the overall MSE (Brown et al., 2005a,b). This theory, for example, was employed to propose a seminal

ensemble model, the previously described ABM model, in which the weights to combine the individuals are obtained through an optimization procedure that aims at achieving models with high mean prediction performance, and that minimize the covariance matrix associated with the prediction of the models in the training set (Krogh and Vedelsby, 1994).

This way of computing the weights that allow the combination of the outputs of the individual base learners (maximizing the mean performance and minimizing the covariance matrix) is highly similar to what is done in modern portfolio optimization theory. In the mean-variance (MV) framework, the assets' weights are estimated assuming that a rational investor aims to maximize returns and minimize risks. The mathematical formulation of the MV optimization problem estimates the returns through the mean and the risk through the covariance matrix. This formulation in the ensemble learning field allows determining the ensemble's weights considering the error predictions of the base learners, the variability in their predictions, and the covariance among their predictions' errors, which is completely aligned with the bias-variance-covariance theory. For this reason, we will explore potential ensemble methods that estimate the ensemble's weights based on the bias-variance-covariance decomposition and the mean-variance optimization framework from the financial literature. Specifically, we have proposed four different ensemble learning models based on the MV optimization framework, and the models are tested with 75 regression datasets and compared with state-of-the-art bagging regression models.

The proposal has several advantages concerning state-of-the-art methods, such as the diversity is promoted both in data and in the error function (the NCL fosters diversity only in the error function while the BEM, GEM, SR, and SRR only in the data level), and takes into account the individual performance of the models and the diversity among them (the ABM method only considers the diversity component). Whilst a similar framework has been implemented in the literature in an unsupervised learning scenario to adjust weights in a consensus scheme (R Ünlü, 2021), in this work it is adapted ad hoc for a supervised learning scenario in a regression problem context.

The manuscript is organized as follows: an explanation of the Mean-Variance optimization framework for portfolio problems is provided in Section 2. The proposed ensemble model inspired by the previously mentioned framework is fully described in Section 3. The experimental framework is presented in Section 4, and the empirical results are provided in Section 5.

Finally, conclusions and discussion are in the final part of the manuscript, Section 6.

2. Mean-variance portfolios: Mathematical formulation, shorting and diversification

Markowitz (1952) (Markowitz, 1952, 2014) proposed the well-known mean-variance (MV) portfolio model under the hypothesis that a rational investor aims at maximizing returns and minimizing risks. The MV portfolio framework is a bi-objective optimization problem with an efficient frontier composed of all combinations of assets that are not dominated by any other in expected return and risk simultaneously (Masmoudi and Abdelaziz, 2018). The MV portfolio has been widely implemented in the financial community and accepted by professionals (Lim and Zhou, 2002; Yin and Zhou, 2004). The main advantages of the approach are: (i) its ease of use since the approach presents the concept of return and risk in a straightforward manner and (ii) the ease with which the optimization problem is introduced (Fernández-Navarro et al., 2021).

Mathematically speaking, the MV model estimates, for a portfolio consisting of S assets ($s = 1, \ldots, S$), the portfolio's weights (which represent the percentage of the investment of each asset), $w_1, \ldots, w_S \geq 0$, $\sum_{s=1}^{S} w_s = 1$, using as inputs of the optimization problem the expected excess returns (μ_s), risks (σ_s) and covariances between assets (σ_{sm}). Specifically, the MV portfolio is defined as:

$$\begin{aligned} \min_{w_1,\ldots,w_S} \quad & \frac{1}{2}\sum_{s,m=1}^{S} w_s w_m \sigma_{sm} - \lambda \sum_{s=1}^{S} w_s \mu_s. \\ \text{s.t.} \quad & \sum_{s=1}^{S} w_s = 1. \\ & w_1, \ldots, w_S \geq 0, \end{aligned} \tag{2}$$

where the term $\sum_{s=1}^{S} w_s \mu_s$ is the portfolio mean return, $\sum_{s,m=1}^{N} w_s w_m \sigma_{sm}$, ($\sigma_{ss} = \sigma_s^2$) is the portfolio risk and $\lambda \in [0, 1]$ is an hyperparameter of the problem that weights the relative importance of the mean return with respect to the risk.

The last constraint of the MV optimization problem, $w_1, \ldots, w_S \geq 0$, is called the no-shorting constraint. Short selling occurs when an investor borrows an asset and sells it on the open market, planning to repurchase it later for less money. As seen in the optimization problem, Markowitz (1952) (Markowitz, 1952) considered the static MV portfolio selection formulation in a market where shorting is not allowed in its original manuscript. In addition, he developed a numerical scheme, the critical line algorithm, to solve the static mean-variance model with no-shorting (Markowitz, 1952). In any case, it is important to clarify that considering markets with shorting allowed leads to the formulation of an unconstrained mean-variance portfolio optimisation problem and facilitates the derivation of an analytical solution.

The MV optimization model can be also formulated in matrix form as follows:

$$\begin{aligned} \min_{\mathbf{w}\in\mathbb{R}^S} \quad & \frac{1}{2}\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w} - \lambda\mathbf{w}^T\boldsymbol{\mu}. \\ \text{s.t.} \quad & \mathbf{1}_S^T\mathbf{w} = 1. \\ & \mathbf{w} \geq \mathbf{0}_S, \end{aligned} \tag{3}$$

where $\mathbf{w} = (w_1 \ldots w_S)^T \in \mathbb{R}^S$ is the vector of weights for the portfolio assets, $\boldsymbol{\Sigma} \in \mathbb{R}^{S\times S}$ is the covariance matrix of asset's returns, $\boldsymbol{\mu} = (\mu_1 \ldots \mu_S)^T \in \mathbb{R}^S$ is the vector with the expected excess returns and $\mathbf{0}_S$ and $\mathbf{1}_S$ are S-dimensional vectors with zeros and ones in all the rows respectively.

The two objectives associated with the framework can be addressed separately as two independent optimization problems giving rise to the global maximum return (GMR) (Zhou and Palomar, 2020) and global minimum variance (GMV) (Coqueret, 2015; Maillet et al., 2015; Bodnar et al., 2018) portfolios. The global maximum return (GMR) portfolio is a convex optimization problem that can be defined as:

$$\begin{aligned} \max_{\mathbf{w}\in\mathbb{R}^S} \quad & \mathbf{w}^T\boldsymbol{\mu}. \\ \text{s.t.} \quad & \mathbf{1}_S^T\mathbf{w} = 1. \\ & \mathbf{w} \geq \mathbf{0}_S, \end{aligned} \tag{4}$$

with the trivial solution of allocating all the budget to the asset of maximum return. This strategy does not include diversification in the investment and traditionally performs poorly due to this fact.

The global minimum variance (GMV) portfolio does not include the maximization of expected return in its formulation and focuses only on minimizing

risks (Coqueret, 2015). The GMV portfolio has been widely implemented in scholarly manuscripts to evaluate and compare different proposals of estimators for the covariance matrix (Bodnar et al., 2018; Coqueret, 2015; Maillet et al., 2015). The problem is formulated as QP:

$$\begin{aligned} \min_{\mathbf{w}\in\mathbb{R}^S} \quad & \frac{1}{2}\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}. \\ \text{s.t.} \quad & \mathbf{1}_S^T\mathbf{w} = 1. \\ & \mathbf{w} \geq \mathbf{0}_S, \end{aligned} \tag{5}$$

Several authors have claimed that one of the most relevant limitations associated with the MV portfolio is its high concentration (Klein and Bawa, 1977; Bird and Tippett, 1986; DeMiguel et al., 2009). MV-optimized portfolios are highly concentrated on a few assets with high return and low-risk (Lin, 2013). Assets with high expected returns will be overweighted; therefore, the benefits of diversification, which optimization is assumed to provide, are reduced (Schmidt, 2019).

Several researchers have presented alternative strategies to overcome the previously mentioned limitation of the MV model (Klein and Bawa, 1977; Lin, 2013; Schmidt, 2019). Those approaches are denoted in the financial literature as diversification strategies for portfolio optimization (Sankaran and Patil, 1999). Diversification reduces portfolio risk by distributing the capital in different assets (Bird and Tippett, 1986; Kuhle, 1987). In a concentrated portfolio (with, for example, two assets), the unexpected fall of one asset return causes a direct effect on the final portfolio return. Contrarily, if the portfolio is diversified, the effect of the unexpected fall is compensated by the other asset returns.

A simple and naive way to solve this limitation of the MV model is to attribute the same weight to all the assets included in the portfolio. Equally weighted (EW) or "$1/S$" portfolios are widely used in the financial literature (Benartzi and Thaler, 2001; Li et al., 2020; Windcliff and Boyle, 2004) as they have been shown as a competitive alternative to the MV portfolios with respect to out-of-sample performance (DeMiguel et al., 2009; Duchin and Levy, 2009; Tu and Zhou, 2011). Another straightforward approach to overcome the concentration-related limitation of the portfolio is to impose upper or lower bounds on the asset weights (Lin, 2013). For example, Abdelaziz et al. (2017) (Abdelaziz et al., 2007), under this approach for diversification, proposed a portfolio optimization in which the amount invested in each asset

is bounded between zero and ten percent of the total budget, and the total amount invested in banking leasing and insurance must be less than thirty percent.

The last way to promote portfolio diversification is by explicitly fostering it in the cost function of the optimization problem. In this context, for example, Schmidt (2019) (Schmidt, 2019) proposed a diversification approach in which a diversity booster term is incorporated explicitly into the objective function of the MV problem, giving rise to the following QP problem:

$$\begin{array}{ll} \min\limits_{\mathbf{w}\in\mathbb{R}^S} & \frac{1}{2}\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w} - \lambda\mathbf{w}^T\boldsymbol{\mu} + \delta\mathbf{w}^T\mathbf{w}. \\ \text{s.t.} & \mathbf{1}_S^T\mathbf{w} = 1. \\ & \mathbf{w} \geq \mathbf{0}_S, \end{array} \tag{6}$$

where $\delta\mathbf{w}^T\mathbf{w}$ is the diversity booster, and $\delta \in \mathbb{R}$ is a parameter that specifies the importance of the previously mentioned term. The strategy generates EW portfolios (DeMiguel et al., 2009) when the δ parameter is set to high values and the MV portfolio when δ is set to zero.

Before providing the necessary details to estimate the parameters of the models, we show in Table 1 the complete list of symbols employed in the section, aiming to clarify the understanding of the algorithmic procedure.

3. Methodology proposed: Ensemble learning based on the mean-variance framework

The methodologies proposed are all under the umbrella of Bagging methods, and the goal is the traditional one, that is, to learn a function, f, that predicts the dependent variable $y \in \mathbb{R}$ in terms of the attributes $\mathbf{x} \in \mathbb{R}^K$ (being K the dimension of the input space) using a set of training set $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$, where N is the number of patterns in the training set. As expected, the proposed ensemble methods' parameter estimation procedure can be decomposed into two parts: bootstrapping and aggregation. These two parts are in line with the two typologies of parameters of the ensemble model, f: (i) the parameters associated with the different base learners and (ii) the parameters to combine the output of these base learners. Due to this reason, the original training set, $\mathcal{D}$, is randomly split in two sets: the training set, $\mathcal{T} = \{(\mathbf{x}_n^t, y_n^t)\}_{n=1}^{N^{\mathrm{t}}}$, which is used to fit the parameters of the S base learners (N^{t} is the number of instance of the subset) and the validation

Variable	Meaning
	Related to the input data
K	Dimension of the input space.
N	Number of patterns in the original input dataset.
S	Ensemble size
$\mathcal{D}$	Original input dataset.
$y \in \mathbb{R}$	Dependent variable (regression problem).
$\mathbf{x} \in \mathbb{R}^K$	Independent variable.
$\mathcal{T}_1, \ldots, \mathcal{T}_S$	Training sets of the S baseline regressors models.
$\mathcal{V}_1, \ldots, \mathcal{V}_S$	Validation sets of the S baseline regressors models.
N^t	Number of patterns in the different training sets.
N^v	Number of patterns in the different validation sets.
	Related to the optimization problem
$\{f_1, \ldots, f_S\}$	Optimized base learners.
$\mathbf{E} \in \mathbb{R}^{N^v \times S}$	Squared error of each base learner in each pattern belonging to the validation set.
$\boldsymbol{\mu} \in \mathbb{R}^S$	Arithmetic mean by columns of the error matrix.
$\boldsymbol{\Sigma} \in \mathbb{R}^{S \times S}$	Covariance matrix of the error matrix.
$\lambda \in \mathbb{R}$	Hyperparameter that weights the relative importance of the mean error with respect to the covariance.
$\delta \in \mathbb{R}$	Hyperparameter that specifies the importance of the diversity booster term.
$\beta \in \mathbb{R}$	Lagrange multiplier.
	Optimization variables
$\mathbf{w} \in \mathbb{R}^S$	Ensemble weights.

Table 1: The complete list of variables employed in the algorithmic procedure description and corresponding meanings.

set, $\mathcal{V} = \{(\mathbf{x}_n^v, y_n^v)\}_{n=1}^{N^{\mathrm{v}}}$, which is employed to determine the parameters that combine the outputs of the base learners (N^{v} is the number of instance of the validation set).

Below, the two parts for the parameter estimation of the ensemble models are described in detail.

3.1. Generation of the base learners: Bootstrap Sampling ($\mathcal{T}_1, \ldots, \mathcal{T}_S$)

In the first stage of the parameter estimation, called the bootstrapping stage, different training sets should be generated to fit the parameters of the

individual base learners. With the number of base learners denoted as S, the original training set, $\mathcal{D}$, is split into the training set, $\mathcal{T}$ and the validation set, $\mathcal{V}$. S training sets, $\mathcal{T}_s$, of size N^t, by sampling from $\mathcal{T}$ uniformly and with replacement are generated, $\mathcal{T}_1, \ldots, \mathcal{T}_S$. These training sets are employed to fit the parameters of the S base learners composing the regression ensemble. Thus, the parameters of the s-th base learner, f_s, are estimated with $\mathcal{T}_s$, the s-th training set. Sampling with replacement ensures each subset, $\mathcal{T}_s$, is independent of its peers, as it does not depend on previously chosen samples when sampling, $s = 1, \ldots, S$. To sum up, in the proposed ensemble models, the S base learners are all built using the same learning algorithm from a different bootstrap sample of the original training set.

Figure 1 shows the algorithmic procedure of this first stage. The function BootstrapResampling($\mathcal{T}$) selects N^t random samples with replacement from the training set, $\mathcal{T}$, data set and the function fit($\mathcal{T}_s$) estimates the parameters of the f_s base learner from the $\mathcal{T}_s$ training set.

Bootstrapping stage ($\mathcal{T}$):

Require: Training set: $\mathcal{T} = \{(\mathbf{x}_n^t, \mathbf{y}_n^t)\}_{n=1}^{N^t}$, where $\mathbf{x}_n^t \in \mathbb{R}^K$ and $\mathbf{y}_n^t \in \mathbb{R}$.
Ensure: Optimized base learners: $\{f_1, \ldots, f_S\}$.

1: **for** $s = 1$ until S **do**
2: $\quad \mathcal{T}_s \leftarrow$ BootstrapResampling($\mathcal{T}$).
3: $\quad f_s \leftarrow$ fit($\mathcal{T}_s$).
4: **end for**
5: **return** $\{f_1, \ldots, f_S\}$.

Figure 1: The algorithmic procedure of the ensemble model: bootstrapping stage

3.2. Aggregation ($\mathcal{V}$)

The goal of this second stage of the algorithmic procedure is to estimate the parameters that allow aggregating the outputs from all the separate base learners into a single prediction as part of the final model. In the case under study, regression problems, the ensemble output is obtained by simply weighted averaging of estimated outputs of the different base learners. Specifically, the ensemble output for a pattern $\mathbf{x}$, $f(\mathbf{x})$, is obtained by convexly combining the outputs of the base regressors, $f_1(\mathbf{x}), \ldots, f_S(\mathbf{x})$, as:

$$f(\mathbf{x}) = \sum_{s=1}^{S} w_s f_s(\mathbf{x}),\ w_s > 0,\ \sum_{s=1}^{S} w_s = 1, \tag{7}$$

where w_s is the weight of the s-th regressor in the ensemble, $\mathbf{w} = (w_1 \dots w_S)^T \in \mathbb{R}^S$.

As seen in the previous equation, each component of the $\mathbf{w}$ vector weights each base learner's importance in the ensemble models' final output. This vector will be estimated under the hypothesis that promising base learners are those who report consistently low estimation errors. Thus, the two performance metrics to be evaluated in base learners are the mean squared error and the variance of those squared errors. The performance of base learners will be assessed in the validation set, $\mathcal{V}$, in order to promote generalization performance in the ensemble model. Accordingly, the aggregation stage begins with the construction of an error matrix, $\mathbf{E}$, that reports the squared error of each base learner in each pattern belonging to the validation set. This matrix is defined as $\mathbf{E} = (e_{n,s})_{i=1\dots,N^v,s=1,\dots S} \in \mathbb{R}^{N^v \times S}$, with $e_{n,s} = (f_s(\mathbf{x}_n^v) - y_n^v)^2$.

Once the error matrix has been computed, the two inputs of the MV optimization problem are estimated: (i) the mean error, $\boldsymbol{\mu}$, which is the arithmetic mean by columns of the error matrix, and (ii) the covariance matrix of the error matrix, $\boldsymbol{\Sigma}$. The first element measures the individual performance of the base learners, and the second one the consistency in performance and correlation among errors of base learners. With these two inputs, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, along with the hyperparameter $\lambda \in \mathbb{R}$, that weights the importance of the mean squared error with respect to the variance and covariance, the ensemble weights, $\mathbf{w}$, can be determined using the ideas of portfolio weights defined in the previous section.

Figure 2 helps to understand the algorithmic procedure of this second stage. The function mean($\mathbf{E}$) computes the arithmetic mean by columns of the input matrix, and the function cov($\mathbf{E}$) estimates the covariance matrix of the input matrix. Finally, the function estimateWeights($\boldsymbol{\mu}, \boldsymbol{\Sigma}, \lambda$) determines the ensemble weights using as input parameters the mean squared error of the base learners, the covariance matrix for those squared errors and the hyperparameter λ. In this research study, four different implementations for the function estimateWeights($\boldsymbol{\mu}, \boldsymbol{\Sigma}, \lambda$) have been proposed. The four implementations of the function lead to the four proposed ensemble models: (i) Ensemble Mean-Variance with No-Shorting Constraints (EMVNSC), (ii) Ensemble Diversified Mean-Variance with No-Shorting Constraints (EDMVNSC), (iii) Ensemble Unconstrained Mean-Variance (EUMV) and (iv) Ensemble Unconstrained Diversified Mean-Variance (EUDMV).

Section 3.3 describes the mathematical formulation for the first two ap-

Aggregation stage ($\mathcal{V}$):

Require: Validation set: $\mathcal{V} = \{(\mathbf{x}_n^v, \mathbf{y}_n^v)\}_{n=1}^{N^v}$, where $\mathbf{x}_n^v \in \mathbb{R}^K$ and $\mathbf{y}_n^v \in \mathbb{R}$.
Trained base learners: $f_1, \ldots, f_S$. Hyperparameter $\lambda \in \mathbb{R}$.

Ensure: Ensemble weights: $\mathbf{w} = (w_1 \ldots w_S)^T \in \mathbb{R}^S$.

for $n = 1$ until N^v **do**
  for $s = 1$ until S **do**
    $e_{n,s} \leftarrow (f_s(\mathbf{x}_n^v) - y_n^v)^2$.
  end for
end for
$\boldsymbol{\mu} \leftarrow \text{mean}(\mathbf{E})$.
$\boldsymbol{\Sigma} \leftarrow \text{cov}(\mathbf{E})$.
$\mathbf{w} \leftarrow \text{estimateWeights}(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \lambda)$.
return $\mathbf{w} = (w_1 \ldots w_S)^T \in \mathbb{R}^S$.

Figure 2: The algorithmic procedure of the ensemble model: aggregation stage

proaches (EMVNSC and EDMVNSC), while Section 3.3.1 details the last two ensemble approaches (EUMV and EUDMV).

3.3. Aggregation for the Mean-Variance formulation with No-Shorting Constraints

In this section, we show that the MV optimization and its philosophy are quite similar to the one traditionally used in ensemble modeling in machine learning. Thus, MV theory could be implemented to optimize the weights of the base learners within a bagging regression model. This section details two potential ensemble approaches that implement the original MV framework: the EMVNSC and EDMVNSC models. Both models take as input parameters: (i) the covariance matrix of the errors, $\boldsymbol{\Sigma} \in \mathbb{R}^{S \times S}$, and (ii) the vector of mean errors of the different base learners, $\boldsymbol{\mu} \in \mathbb{R}^S$, and (iii) the hyperparamter that defines the importance of the second term concerning the first one, $\lambda \in \mathbb{R}$. In addition, both ensemble models incorporate the no-shorting constraints in their optimization problems, $\mathbf{w} \geq \mathbf{0}_S$, precisely as it was included in the pioneered study by Markowitz (Markowitz, 1952).

The EMVNSC ensemble model. The optimization problem of the EMVNSC model is defined as:

$$\begin{aligned} \min_{\mathbf{w}\in\mathbb{R}^S} \quad & \frac{1}{2}\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w} + \lambda\mathbf{w}^T\boldsymbol{\mu}. \\ \text{s.t.} \quad & \mathbf{1}_S^T\mathbf{w} = 1. \\ & \mathbf{w} \geq \mathbf{0}_S. \end{aligned} \tag{8}$$

It is important to stress that the optimization problem of the ensemble model is an inequality-constrained QP that has been solved with the active set algorithm provided by the MATLAB programming environment.

The EDMVNSC ensemble model. The optimization problem associated with the EDMVNSC model is the same as the EMVNSC model but explicitly includes the diversification term in the objective function. In portfolio literature, this is one of the possibilities to overcome the limitation of the portfolio's weights of high concentration. In ensemble learning, the diversification terms would help avoid a few base learners dominating the ensemble's output. The mathematical formulation of the model is:

$$\begin{aligned} \min_{\mathbf{w}\in\mathbb{R}^S} \quad & \frac{1}{2}\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w} + \lambda\mathbf{w}^T\boldsymbol{\mu} + \delta\mathbf{w}^T\mathbf{w}. \\ \text{s.t.} \quad & \mathbf{1}_S^T\mathbf{w} = 1. \\ & \mathbf{w} \geq \mathbf{0}_S, \end{aligned} \tag{9}$$

The objective function of the EDMVNSC model can also be written as:

$$\min_{\mathbf{w}\in\mathbb{R}^S} \quad \mathbf{w}^T\left(\frac{1}{2}\boldsymbol{\Sigma} + \delta\mathbf{I}\right)\mathbf{w} + \lambda\mathbf{w}^T\boldsymbol{\mu}, \tag{10}$$

As can be seen in equation (26), diversification is achieved by adding δ to the diagonal of the covariance matrix, which is done to improve diversification in the resulting portfolio and also helps to improve the numerical stability of the matrix inversion through regularization. Specifically, the previously shown diversification strategy is called the l-2 regularization. Additionally, the cardinality constrained problem of the portfolio selection problem is known as the l-0 regularization (Sadigh et al., 2012; Yaman and Dalkılıç, 2021).

Finally, the optimization problem of the EDMVNSC model is again an inequality-constrained QP that has also been solved with the active set algorithm provided by the MATLAB programming environment.

3.3.1. Aggregation for the Unconstrained Mean-Variance formulation

This section fully describes the ensemble models that do not include the short-selling constraints in their formulations: the EUMV and EUDMV ensemble models. Fortunately, the optimization problems of the two models included in this section are particularly simple since the quadratic terms of both problems are positive definite, and there is only one equality constraint in their optimization problems. Due to this fact, the solution process is linear and is obtained using the Lagrange multiplier approach.

The EUMV ensemble model. The optimization problem associated with the EUMV model is very similar to that of the EMVNSC but without including the short-selling constraints, and therefore, it is defined as:

$$\begin{aligned} \min_{\mathbf{w}\in\mathbb{R}^S} \quad & \frac{1}{2}\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w} + \lambda\mathbf{w}^T\boldsymbol{\mu}. \\ \text{s.t.} \quad & \mathbf{w}^T\mathbf{1}_S = 1. \end{aligned} \tag{11}$$

The EUMV optimization problem can be solved using the Lagrange Multiplier approach. Particularly, the Lagrangian of the optimization problem is defined as:

$$\mathcal{L} = \frac{1}{2}\left(\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}\right) + \lambda\mathbf{w}^T\boldsymbol{\mu} - \beta(\mathbf{w}^T\mathbf{1}_S - 1), \tag{12}$$

enforcing $\nabla\mathcal{L} = 0$:

$$\begin{aligned} \frac{\partial\mathcal{L}}{\partial\mathbf{w}} =& \boldsymbol{\Sigma}\mathbf{w} + \lambda\boldsymbol{\mu} - \beta\mathbf{1}_S = 0 \\ \frac{\partial\mathcal{L}}{\partial\beta} =& \mathbf{w}^T\mathbf{1}_S - 1 = 0 \Rightarrow \mathbf{w}^T\mathbf{1}_S = 1 \Rightarrow \mathbf{1}_S^T\mathbf{w} = 1. \end{aligned} \tag{13}$$

Therefore the solution satisfies that:

$$\begin{aligned} \boldsymbol{\Sigma}\mathbf{w} =& \beta\mathbf{1}_S - \lambda\boldsymbol{\mu} \\ \mathbf{1}_S^T\mathbf{w} =& 1. \end{aligned} \tag{14}$$

In this way, the solution to the optimization problem is:

$$\mathbf{w} = \boldsymbol{\Sigma}^{-1}\left(\beta\mathbf{1}_S - \lambda\boldsymbol{\mu}\right), \tag{15}$$

with the following equation for the Lagrange multiplier β:

$$\beta = \frac{1 + \mathbf{1}_S^T\boldsymbol{\Sigma}^{-1}\lambda\boldsymbol{\mu}}{\mathbf{1}_S^T\boldsymbol{\Sigma}^{-1}\mathbf{1}_S}. \tag{16}$$

The EUDMV ensemble model. The optimization function associated with the EUDMV model is the same as that of EUMV but includes the diversification term explicitly in the cost function and consequently is defined as:

$$\begin{aligned} \min_{\mathbf{w}\in\mathbb{R}^S} \quad & \frac{1}{2}\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w} + \lambda\mathbf{w}^T\boldsymbol{\mu} + \delta\mathbf{w}^T\mathbf{w}. \\ \text{s.t.} \quad & \mathbf{w}^T\mathbf{1}_S = 1, \end{aligned} \tag{17}$$

and, again, can be solved using Lagrange multipliers (Boyd et al., 2004). The Lagrangian is:

$$\mathcal{L} = \frac{1}{2}\mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w} + \lambda\mathbf{w}^T\boldsymbol{\mu} + \delta\mathbf{w}^T\mathbf{w} - \beta(\mathbf{w}^T\mathbf{1}_S - 1), \tag{18}$$

where β is the Lagrange multiplier associated with the equality constraint. enforcing $\nabla\mathcal{L} = 0$:

$$\begin{aligned} \frac{\partial\mathcal{L}}{\partial\mathbf{w}} =& \boldsymbol{\Sigma}\mathbf{w} + \lambda\boldsymbol{\mu} + 2\delta\mathbf{w} - \beta\mathbf{1}_S = 0 \\ \frac{\partial\mathcal{L}}{\partial\beta} =& \mathbf{w}^T\mathbf{1}_S - 1 = 0 \end{aligned} \tag{19}$$

so:

$$\begin{aligned} \boldsymbol{\Sigma}\mathbf{w} + 2\delta\mathbf{w} =& \beta\mathbf{1}_S - \lambda\boldsymbol{\mu}, \\ \mathbf{w} =& (\boldsymbol{\Sigma} + 2\delta\mathbf{I})^{-1}(\beta\mathbf{1}_S - \lambda\boldsymbol{\mu}), \end{aligned} \tag{20}$$

where the Lagrange multiplier β is defined as:

$$\beta = \frac{1 + \mathbf{1}_S^T(\boldsymbol{\Sigma} + 2\delta\mathbf{I})^{-1}\lambda\boldsymbol{\mu}}{\mathbf{1}_S^T(\boldsymbol{\Sigma} + 2\delta\mathbf{I})^{-1}\mathbf{1}_S}. \tag{21}$$

4. Experimental Framework

An extensive experimental study was carried out to show the competitive performance of the methodological approaches outlined in Section 3. Section 4.1 explains the structure and the stages of the computational experiments. The description of the datasets employed and the adopted experimental design is included in Section 4.2. The definition of the ensemble models of the experimental study, along with the configuration of their parameters, are given in Section 4.3, whereas the measure used to evaluate the performance is detailed in Section 4.4. Finally, statistical tests implemented to validate the results are specified in Section 4.5.

4.1. Structure of the computational experiments

A comparative study of the performance of all ensemble methods inspired by portfolio optimization was conducted as a first step. This phase aims to show which of the combiner philosophies introduced in Section 3 is compared with the performances reported by the state-of-the-art ensemble methodologies. As expected, this phase aims to demonstrate that the best method found in the first stage is a competitive approach compared to other state-of-the-art ensemble methods specifically designed to deal with regression datasets. Additionally, the robustness of the different ensemble methods in the rankings of the standard deviations of the corresponding performance metric and the computational complexity of those methods are also analyzed.

Regression Datasets				
ID	Dataset	#Patterns	#Attr.	Repository
1	tic	9823	85	KEEL Repository
2	casp	45730	9	UCI ML Repository
3	friedman	40768	9	LIACC Regression Repository
4	ailerons	7154	40	LIACC Regression Repository
5	compactiv	8192	21	KEEL Repository
6	electrical-grid	10000	12	UCI ML Repository
7	parkinsons-total	5875	16	UCI ML Repository
8	delta_elv	9517	6	KEEL Repository
9	winequality-white	4898	11	UCI ML Repository
10	abalone	4177	10	UCI ML Repository
11	ANACALT	4052	7	KEEL Repository
12	student-performance-por	649	43	UCI ML Repository
13	parkinsons-speech	1040	26	UCI ML Repository
14	usopen-men-2013a	126	168	UCI ML Repository
15	usopen-men-2013b	126	168	UCI ML Repository
16	frenchopen-men-2013a	123	170	UCI ML Repository
17	wimbledon-women-2013a	118	170	UCI ML Repository
18	wimbledon-women-2013b	118	170	UCI ML Repository
19	wimbledon-men-2013a	113	163	UCI ML Repository
20	wimbledon-men-2013b	113	163	UCI ML Repository
21	winequality-red	1599	11	UCI ML Repository
22	frenchopen-women-2013a	111	155	UCI ML Repository
23	frenchopen-women-2013b	111	155	UCI ML Repository
24	student-performance-mat	395	43	UCI ML Repository
25	wankara	1609	9	KEEL Repository
26	forestfires	517	28	UCI ML Repository
27	ausopen-men-2013a	103	138	UCI ML Repository
28	ausopen-women-2013a	99	141	UCI ML Repository
29	ausopen-women-2013b	99	141	UCI ML Repository
30	automobile	160	62	UCI ML Repository
31	usopen-women-2013a	74	106	UCI ML Repository
32	usopen-women-2013b	74	106	UCI ML Repository
33	housing	506	13	UCI ML Repository
34	auto-mpg	392	7	UCI ML Repository
35	autoMPG8	392	7	KEEL Repository
36	dee	365	6	KEEL Repository
37	servo	167	12	UCI ML Repository
38	autoMPG6	392	5	KEEL Repository
39	lpga2009	146	11	Larry Winner Repository
40	machineCPU	209	6	KEEL Repository
41	brazilian-logistic	60	20	KEEL Repository
42	slump-mpa	103	7	UCI ML Repository
43	lpga2008	140	5	Larry Winner Repository
44	beer	23	7	Larry Winner Repository
45	diabetes	43	2	LIACC Regression Repository

Table 2: Characteristics of the selected regression datasets

4.2. Datasets

For the empirical validation of the ensemble methods proposed, seventy-five datasets were selected. The selection of the datasets was made aiming to include in the test sets datasets of different nature and with different characteristics (in terms of size and number of attributes). Furthermore, the datasets were obtained from different sources (repositories). Specifically, the repositories of data considered were: the UCI repository (Dheeru and Karra Taniskidou, 2019), LIACC[1], Larry Winner repository[2], and, finally, the KEEL repository (Alcalá-Fdez et al., 2009).

Table 2 reports the main characteristics of the datasets employed in the experimental study: the identifier of the dataset (ID), name (Dataset), number of patterns (#Patterns), number of attributes (#Attr.) and the repository in which the dataset was obtained (Repository). Datasets are sorted from the largest value resulting from multiplying #Patterns and #Attr to the lowest, with IDs from 1 to 75.

The experimental design was conducted using a 5-fold cross-validation procedure, with ten repetitions per fold. The partitions were the same for all compared models. Thus, a total of 50 error measures were obtained for all the models compared, which assures a proper statistical significance of the results. Finally, the original input variables were all normalized to have zero mean and unit variance.

4.3. Algorithms used for comparison purposes

The proposed ensemble models were evaluated by comparing their empirical results to those provided by state-of-the-art Bagging models (as the methods proposed are all included in this umbrella of methods). The comparison methods behave all very similarly to the proposed methods, and the main difference lies in the implementation of the function estimateWeights$(\cdot)$ (the estimation of the ensemble's weights). Before detailing the comparison methods, it is necessary to denote the vector with the outputs of the s-th base learner in the validation set as $\mathbf{f}_s = (f_s(\mathbf{x}_1^v) \dots f_s(\mathbf{x}_{N^v}^v))^T \in \mathbb{R}^{N^v}$ (the outputs of the s-th base learner in the validation set), the matrix with the outputs of all base learners in the validation set as $\mathbf{f} = (\mathbf{f}_1 \dots \mathbf{f}_S) \in \mathbb{R}^{N^v \times S}$, and, finally, the vector with desired outputs in the validation set as $\mathbf{y}^v = (y_1^v \dots y_{N^v}^v)^T \in$

[1]LIACC URL: `http://www.ncc.up.pt/liacc/ML/statlog/datasets.html`
[2]Larry winner URL: `http://users.stat.ufl.edu/~winner/datasets.html`

$\mathbb{R}^{N^v}$. Once those vectors and matrices are defined, the comparison methods can be briefly described in continuation:

- **Basic Ensemble Method (BEM)** (Breiman, 1996a). The original implementation of the Bagging approach in which diversity is promoted only implicitly by random sampling. In this ensemble model, the function estimateWeights$(\cdot)$ assigns the same weights to all base learners composing the ensemble, $w_s = \frac{1}{S}, s = 1, \ldots, S$. The BEM model is the only state-of-the-art model which does not include information from the validation set to estimate the ensemble's weights.

- **Generalized Ensemble Method (GEM)** (Perrone and Cooper, 1992). In this approach, the ensemble's weights are estimated by minimizing the squared error of the ensemble model in the validation set, so the optimization problem is defined as:

$$\begin{aligned} \min_{\mathbf{w}\in\mathbb{R}^S} \quad & \|\mathbf{y}^v - \mathbf{f}\mathbf{w}\|^2. \\ \text{s.t.} \quad & \mathbf{1}_S^T\mathbf{w} = 1. \\ & \mathbf{w} \geq \mathbf{0}_S. \end{aligned} \tag{22}$$

- **Stacked Regression (SR)** (Breiman, 1996b). This ensemble model is very similar to the previous one but removes the equality and inequality constraints from that model. Consequently, the ensemble's weights are obtained as a solution to the following optimization problem:

$$\min_{\mathbf{w}\in\mathbb{R}^S} \quad \|\mathbf{y}^v - \mathbf{f}\mathbf{w}\|^2, \tag{23}$$

  and, therefore, the weights are computed as $\mathbf{w} = \left(\mathbf{f}^T\mathbf{f}\right)^{-1}\mathbf{f}^T\mathbf{y}^v$.

- **Stacked Ridge Regression (SRR)** (Breiman, 1996b). This ensemble model is the same as the previous one but incorporating the regularization term in the objective function:

$$\min_{\mathbf{w}\in\mathbb{R}^S} \quad \|\mathbf{y}^v - \mathbf{f}\mathbf{w}\|^2 + \delta\|\mathbf{w}\|^2, \tag{24}$$

  and the weights are determined as $\mathbf{w} = \left(\mathbf{f}^T\mathbf{f} + \delta\mathbf{I}\right)^{-1}\mathbf{f}^T\mathbf{y}^v$.

- **Ambiguity Bagging Method (ABM)** (Krogh and Vedelsby, 1994). The AEM incorporates in its optimization function the concept of ambiguity, measured as the variation of the output of ensemble members averaged, so it quantifies the disagreement among the networks. In this regard, this model is the first one that promotes diversity in the sampling and the objective function. As suggested by the authors, the optimal weights can be found by Linear Programming (LP).

The important contribution of the manuscript is related to the implementation of the function estimateWeights$(\cdot)$ and therefore, aiming to focus on that specific part, we have selected a simple base learner for the construction of the ensemble. Specifically, in this experimental study, the individuals of the ensemble methods implemented are all baseline linear regression models. The number of individuals in the ensemble, S, was set to 10 in all the ensembles tested, as recommended in (Brown et al., 2005b). Additionally, the regularization parameter, δ, was determined in the SRR method and the methods proposed EDMVNSC and EDMVNSC with the grid: $\delta \in \{10^{-3}, \ldots, 10^{3}\}$. Similarly, the diversity coefficient, λ, in the proposed models was also determined by cross-validation using the following grid values: $\lambda \in \{10^{-3}, \ldots, 10^{3}\}$.

4.4. Performance metric

The ensemble models proposed and those implemented for comparison purposes were evaluated with the root mean squared percentage error (RMSPE), the standard deviation of the differences between predicted and target values in percentage (Göçken et al., 2016). The metric was employed instead of the traditional root mean squared error (RMSE), as the RMSPE is not scale-dependent (Shcherbakov et al., 2013). It allows comparing the error in predictions for different datasets in percentage terms. The metric is defined as:

$$\text{RMSPE} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(\frac{f(\mathbf{x}_n) - y_n}{y_n}\right)^2}. \tag{25}$$

Additionally, the time required to estimate the parameters of each method has also been considered. The time (T) is the simplest method for measuring the empirical efficiency of a method. The average time elapsed (in seconds) is analyzed by every method, considering the aggregation stage (as the estimation one is the same compared to state-of-the-art-methods).

4.5. Statistical tests

In the presented empirical research, hypothesis testing was employed to provide statistical support in discussing the results. It is important to mention that a performance analysis through parametric tests could lead to incorrect conclusions in this research study since a previous evaluation of the $RMSPE$ values provided by the comparison and proposed methods resulted in rejecting the normality and equality of the variance hypothesis. Furthermore, as noted by Demšar (Demšar, 2006; Luengo et al., 2009; García et al., 2010), the independence condition is not truly verified in a 5-fold cross-validation. For these reasons, two nonparametric Friedman tests (with the ranking of $RMSPE$ of the models as the test variables) were carried out to determine the statistical significance of the rank differences in the two stages of the experimental study (Friedman, 1940). Consequently, two nonparametric Holm post hoc tests were employed to ascertain which models were distinctive among the multiple comparisons performed in the two stages of the study (Holm, 1979). The first test was used to select the best-proposed method and the second one to ascertain if there are statistical differences in mean ranking between the proposed method selected in the first stage and the ensemble models used for comparison purposes.

5. Results

This section will be destined to analyse the results obtained by the different methods involved in the experimental stage on the selected datasets. Thus, in Section 5.1, the performance results on the proposed regression problems are shown and discussed, and a statistically supported comparison is conducted. In Section 5.2, an analysis of the stability of the experimental results of the methods is included. Furthermore, the computational complexity of the proposed methods has been compared to state-of-the-art methods with both the big O notation and empirically (analyzing the running time in seconds) in Section 5.3. Finally, a detailed study of the sensitivity of the performance of the proposed method to the setting of its hyperparameters is presented in Section 5.4.

5.1. Performance Analysis

As aforementioned, the experimental validation is divided into two phases. In the first phase, the algorithms proposed are compared among them, aiming to define the most promising approach, while in the second phase, the selected

model is compared in terms of prediction performance against the state-of-the-art models. Thus, the section is also divided into these two parts for clarity.

5.1.1. Comparison between our proposals

Table 3 presents the average generalisation results for the $RMSPE$ metric obtained by the methods involved in the first stage of the experimental design on the selected benchmark datasets. The best result from each dataset is highlighted in bold-face and the second one in italics.

As seen in Table 3, the EUDMV model appears to be the most promising model as it achieves the best performance (if compared to the other models) in thirty-five out of the seventy-five datasets considered in the experimental study. The second most promising method seems to be the EDMVNSC model, which yields the best performance in fifteen regression datasets. These two models incorporate the diversification term in their formulations, which is connected with regularization, stressing the importance of incorporating that component as the path to bias-variance trade-off.

Another attractive property of the methodologies that include the diversification term in their formulations is that it allows weighting differently the variance and covariance elements, as the quadratic terms can be grouped as:

$$\min_{\mathbf{w}\in\mathbb{R}^S} \quad \mathbf{w}^T \left(\frac{1}{2}\boldsymbol{\Sigma} + \delta\mathbf{I}\right)\mathbf{w} + \lambda\mathbf{w}^T\boldsymbol{\mu}, \tag{26}$$

which helps to provide different weights to the variance components than the covariance elements (according to the specificities of the problem under study).

The last noteworthy aspect of this method is that it generalises the bagging approach when the values of the hyperparameter δ are large. When this occurs, the model replicates the behaviour of EW, a model methodologically equivalent to the strategy proposed by bagging. In line with the underlying theory of ensemble bagging approaches, the δ hyperparameter incorporates elements in the main diagonal of Σ that make the procedure focus on minimising the variance of individuals.

A non-parametric Friedman test was performed using the $RMSPE$ rankings of the implemented models in order to determine the statistical significance of the experimental results shown above. For the 4 methods involved in this stage on the 75 regression problems, the confidence interval for the Friedman test was $C_0 = (0, F_{0.05} = 2.6453)$, and the rank differences set

ID	EMVNSC	EDMVNSC	EUMV	EUDMV
1	$1.0025_{0.2083}$	$0.9001_{0.1714}$	$\mathbf{0.2442}_{0.0070}$	$\mathit{0.8398}_{0.1188}$
2	$1.4151_{0.1812}$	$1.4164_{0.1691}$	$\mathit{0.9515}_{0.0015}$	$\mathbf{0.8387}_{0.0770}$
3	$\mathit{0.2105}_{0.0055}$	$\mathbf{0.2097}_{0.0058}$	$0.9992_{0.0002}$	$0.3910_{0.0026}$
4	$2.4163_{0.0920}$	$2.4152_{0.1112}$	$\mathbf{0.6142}_{0.0082}$	$\mathit{1.6359}_{0.0439}$
5	$\mathit{0.2344}_{0.0080}$	$\mathbf{0.2339}_{0.0081}$	$0.9995_{0.0002}$	$0.4002_{0.0010}$
6	$\mathbf{0.2800}_{0.2304}$	$\mathit{0.4359}_{0.1275}$	$0.9998_{0.0002}$	$0.5807_{0.2650}$
7	$0.5849_{0.0484}$	$\mathit{0.5838}_{0.0474}$	$0.9970_{0.0014}$	$\mathbf{0.4611}_{0.0125}$
8	$\mathit{0.7355}_{0.0608}$	$0.7616_{0.0651}$	$0.9819_{0.0014}$	$\mathbf{0.5960}_{0.0287}$
9	$\mathit{0.6000}_{0.0335}$	$0.6322_{0.0397}$	$0.9930_{0.0006}$	$\mathbf{0.5985}_{0.1164}$
10	$\mathit{0.1522}_{0.0338}$	$\mathbf{0.1519}_{0.0332}$	$0.9999_{0.0002}$	$0.4398_{0.0389}$
11	$0.5410_{0.1620}$	$\mathit{0.5379}_{0.1932}$	$0.9274_{0.0207}$	$\mathbf{0.4088}_{0.0594}$
12	$14.9782_{11.0734}$	$14.4820_{12.7741}$	$\mathbf{0.9995}_{0.0005}$	$\mathit{9.1090}_{7.8539}$
13	$1.2336_{0.8793}$	$1.2374_{0.8930}$	$\mathit{0.9819}_{0.0270}$	$\mathbf{0.8602}_{0.4231}$
14	$1.0114_{0.6789}$	$1.0303_{0.7159}$	$\mathit{0.9816}_{0.0347}$	$\mathbf{0.7219}_{0.3789}$
15	$\mathbf{0.1447}_{0.0329}$	$\mathit{0.1453}_{0.0341}$	$0.9998_{0.0003}$	$0.4082_{0.0265}$
16	$\mathbf{0.5820}_{0.1216}$	$0.6715_{0.0694}$	$0.9980_{0.0012}$	$\mathit{0.6578}_{0.1883}$
17	$1.1153_{0.6238}$	$1.0197_{0.6504}$	$\mathit{0.5911}_{0.0947}$	$\mathbf{0.5121}_{0.2595}$
18	$\mathit{0.4987}_{0.1845}$	$0.6676_{0.3045}$	$0.9962_{0.0070}$	$\mathbf{0.3142}_{0.1207}$
19	$\mathit{0.1478}_{0.0359}$	$\mathbf{0.1477}_{0.0368}$	$0.9998_{0.0003}$	$0.3714_{0.0406}$
20	$1.1554_{0.1939}$	$1.1655_{0.2005}$	$\mathit{0.9832}_{0.0056}$	$\mathbf{0.8261}_{0.1126}$
21	$\mathit{0.7026}_{0.6779}$	$0.7205_{0.7126}$	$0.9876_{0.0200}$	$\mathbf{0.6164}_{0.3465}$
22	$2.0910_{0.8264}$	$2.0350_{1.0089}$	$\mathbf{0.5248}_{0.4959}$	$\mathit{1.4237}_{0.4995}$
23	$1.5203_{0.5646}$	$1.5434_{0.3756}$	$\mathbf{0.8885}_{0.0339}$	$\mathit{0.9823}_{0.2238}$
24	$\mathit{0.4284}_{0.0799}$	$\mathbf{0.3640}_{0.0986}$	$0.9936_{0.0088}$	$0.4314_{0.0828}$
25	$1.4461_{0.5572}$	$1.4623_{0.5370}$	$\mathbf{0.8335}_{0.0595}$	$\mathit{0.9635}_{0.3269}$
26	$0.6950_{0.3282}$	$\mathit{0.6661}_{0.3178}$	$0.9811_{0.0188}$	$\mathbf{0.5013}_{0.0857}$
27	$1.8447_{0.7370}$	$1.7585_{0.7181}$	$\mathbf{0.8371}_{0.0701}$	$\mathit{1.1725}_{0.4729}$
28	$0.7469_{0.3683}$	$\mathit{0.7454}_{0.3983}$	$0.9942_{0.0093}$	$\mathbf{0.5337}_{0.2464}$
29	$1.2984_{0.3757}$	$1.4474_{0.3374}$	$\mathbf{0.8226}_{0.0462}$	$\mathit{0.9160}_{0.2762}$
30	$0.7302_{0.2036}$	$\mathit{0.6987}_{0.1488}$	$0.9802_{0.0195}$	$\mathbf{0.5542}_{0.1534}$
31	$\mathbf{0.4444}_{0.1621}$	$\mathit{0.4458}_{0.1673}$	$0.9969_{0.0033}$	$0.4837_{0.0473}$
32	$1.8840_{0.3865}$	$1.8004_{0.2660}$	$\mathbf{0.8045}_{0.0536}$	$\mathit{1.1471}_{0.1841}$
33	$\mathit{0.7607}_{0.7769}$	$0.7853_{0.6812}$	$1.0021_{0.0027}$	$\mathbf{0.6565}_{0.3925}$
34	$1.3920_{0.7366}$	$1.3862_{0.7335}$	$\mathbf{0.9502}_{0.0332}$	$\mathit{0.9673}_{0.4021}$
35	$\mathbf{0.0705}_{0.0314}$	$\mathit{0.0722}_{0.0352}$	$0.9585_{0.0067}$	$0.3605_{0.0294}$
36	$\mathit{0.0964}_{0.0237}$	$\mathbf{0.0910}_{0.0157}$	$0.9421_{0.0089}$	$0.3544_{0.0134}$
37	$\mathit{0.0815}_{0.0255}$	$\mathbf{0.0811}_{0.0246}$	$0.9985_{0.0016}$	$0.4182_{0.0166}$
38	$0.2945_{0.0766}$	$0.2939_{0.0803}$	$\mathbf{0.1702}_{0.1580}$	$\mathit{0.2366}_{0.0813}$
39	$1.9840_{0.7000}$	$2.0216_{0.8159}$	$\mathbf{0.8804}_{0.0659}$	$\mathit{1.3286}_{0.4999}$
40	$0.5877_{0.1178}$	$\mathit{0.5685}_{0.1732}$	$0.9859_{0.0170}$	$\mathbf{0.4369}_{0.1333}$
41	$2.2094_{0.9266}$	$2.0928_{0.9887}$	$\mathbf{0.8578}_{0.0380}$	$\mathit{1.4254}_{0.5832}$
42	$0.7168_{0.4391}$	$\mathit{0.7131}_{0.4244}$	$0.9744_{0.0279}$	$\mathbf{0.5134}_{0.0990}$
43	$\mathit{0.1218}_{0.0487}$	$\mathbf{0.1215}_{0.0469}$	$0.9983_{0.0019}$	$0.4363_{0.0320}$
44	$1.0317_{0.3601}$	$1.3026_{0.3503}$	$\mathit{0.9956}_{0.0129}$	$\mathbf{0.7934}_{0.2168}$
45	$1.3714_{0.7990}$	$1.1381_{0.7458}$	$\mathit{1.1146}_{0.2981}$	$\mathbf{0.7935}_{0.5977}$
46	$\mathit{0.3227}_{0.0714}$	$\mathbf{0.3224}_{0.0776}$	$0.9869_{0.0099}$	$0.4150_{0.0237}$
47	$0.5233_{0.0626}$	$\mathit{0.5139}_{0.0561}$	$0.9892_{0.0088}$	$\mathbf{0.4509}_{0.0242}$
48	$1.4627_{0.3000}$	$1.5797_{0.4336}$	$\mathbf{0.7938}_{0.0696}$	$\mathit{0.9664}_{0.2461}$
49	$0.7072_{0.2463}$	$\mathit{0.5760}_{0.2556}$	$0.9866_{0.0271}$	$\mathbf{0.5244}_{0.0668}$
50	$\mathbf{0.4448}_{0.0727}$	$\mathit{0.4678}_{0.0706}$	$0.9978_{0.0030}$	$0.5754_{0.0561}$
51	$0.8190_{0.2723}$	$\mathit{0.7624}_{0.2220}$	$0.9867_{0.0182}$	$\mathbf{0.6497}_{0.1395}$
52	$1.0302_{0.0303}$	$1.0020_{0.0279}$	$\mathit{0.8009}_{0.0101}$	$\mathbf{0.7086}_{0.0130}$
53	$0.9966_{0.2376}$	$0.9391_{0.1412}$	$\mathit{0.7732}_{0.0366}$	$\mathbf{0.5341}_{0.0331}$
54	$\mathit{0.1511}_{0.0306}$	$\mathbf{0.1502}_{0.0313}$	$0.8061_{0.0138}$	$0.3282_{0.0241}$
55	$\mathit{0.6679}_{0.0834}$	$0.6719_{0.1044}$	$0.9142_{0.0100}$	$\mathbf{0.5329}_{0.0698}$
56	$0.7965_{0.0641}$	$\mathit{0.7904}_{0.0626}$	$0.9128_{0.0073}$	$\mathbf{0.5964}_{0.0157}$
57	$0.5233_{0.2927}$	$\mathbf{0.5085}_{0.2810}$	$0.9768_{0.0423}$	$\mathit{0.5148}_{0.0790}$
58	$\mathit{0.4407}_{0.0596}$	$\mathbf{0.4197}_{0.0917}$	$0.9876_{0.0133}$	$0.4655_{0.0744}$
59	$\mathit{0.2965}_{0.0596}$	$\mathbf{0.2950}_{0.0623}$	$0.9960_{0.0037}$	$0.4214_{0.0248}$
60	$1.1676_{0.2183}$	$1.0898_{0.2158}$	$\mathit{0.7828}_{0.0593}$	$\mathbf{0.6421}_{0.1157}$
61	$\mathbf{0.4389}_{0.0438}$	$\mathit{0.4520}_{0.0428}$	$0.9885_{0.0060}$	$0.4845_{0.0366}$
62	$1.1519_{0.2631}$	$1.0196_{0.0497}$	$\mathit{0.6789}_{0.0283}$	$\mathbf{0.5114}_{0.0276}$
63	$\mathit{0.7980}_{0.2866}$	$0.8526_{0.2923}$	$0.9456_{0.0383}$	$\mathbf{0.6323}_{0.1570}$
64	$\mathit{0.3445}_{0.1377}$	$\mathbf{0.3096}_{0.1757}$	$0.9996_{0.0009}$	$0.4276_{0.0970}$
65	$0.8878_{0.4025}$	$0.8435_{0.3685}$	$\mathit{0.5970}_{0.0050}$	$\mathbf{0.4409}_{0.1678}$
66	$\mathit{0.4563}_{0.2566}$	$0.4564_{0.2566}$	$0.9922_{0.0175}$	$\mathbf{0.3920}_{0.1043}$
67	$\mathit{0.6059}_{0.2910}$	$0.6156_{0.3124}$	$0.9379_{0.0127}$	$\mathbf{0.3556}_{0.0448}$
68	$1.5057_{0.5879}$	$1.8302_{0.8406}$	$\mathbf{0.9499}_{0.0748}$	$\mathit{1.1213}_{0.4987}$
69	$1.8552_{1.3980}$	$1.5991_{1.0320}$	$\mathbf{0.9076}_{0.0467}$	$\mathit{1.0733}_{0.5397}$
70	$\mathit{0.3182}_{0.0616}$	$\mathbf{0.2768}_{0.0561}$	$0.9836_{0.0283}$	$0.3796_{0.0526}$
71	$\mathit{0.9550}_{0.4834}$	$0.9571_{0.4625}$	$0.9846_{0.0172}$	$\mathbf{0.6733}_{0.1876}$
72	$\mathit{1.6790}_{1.0532}$	$1.7546_{1.0700}$	$8.9691_{18.3243}$	$\mathbf{1.0577}_{0.7241}$
73	$\mathbf{2.1695}_{1.3559}$	$\mathit{2.6002}_{2.2541}$	$2.6075_{1.4684}$	$2.6764_{2.5852}$
74	$2.9506_{3.4063}$	$2.6300_{3.0253}$	$\mathbf{0.7536}_{0.2586}$	$\mathit{1.7563}_{1.8167}$
75	$\mathit{0.7447}_{0.5229}$	$0.7882_{0.5729}$	$1.0048_{0.0108}$	$\mathbf{0.6330}_{0.2495}$

Table 3: First stage experimental results: Comparison between our proposals, in terms of RMSPE.

the F-distribution statistic value as $F^* = 12.51 \notin C_0$. Therefore, the null hypothesis stating that all models have the same performance for regression in mean $RMSPE$ rankings is rejected. Based on such rejection, the Holm post-hoc test is used to compare in $RMSPE$.

Table 4 shows the results of the Holm test along with the mean $RMSPE$ ($\overline{RMSPE}$) and mean RMSPE ranking ($\overline{\mathrm{R}}_{RMSPE}$), z-statistic, p-values for the ensemble models used for comparison purposes, and adjusted α values ($\alpha_{0.10}$ and $\alpha_{0.05}$). The main conclusion that can be drawn from the results shown in Table 4 is that the EUDMV method is significantly better in performance than methods of its same nature. From a strictly descriptive point of view, EUDMV obtains the best average performance in generalisation, $\overline{RMSPE} = 0.8071$, and the best mean ranking, $\overline{R}_{RMSPE} = 1.8267$, in regression problems. This fact leads to the choice of this method to compare its performance in the second phase with the state-of-the-art methods.

Method	$\overline{RMSPE}$	$\overline{R}_{RMSPE}$	z-statistic	p-value	$\alpha_{0.05}$	$\alpha_{0.10}$
EUMV$_\bullet$	*1.0360*	2.9867	5.5024	0.0	0.0167	0.0333
EMVNSC$_\bullet$	1.0773	2.6800	4.0477	5.0E-5	0.025	0.05
EDMVNSC$_\bullet$	1.0703	*2.5067*	3.2255	0.0013	0.05	0.1
EUDMV	**0.8071**	**1.8267**	-	-	-	-

$\bullet$: Statistical difference with $\alpha = 0.05$

Table 4: Statistical results for the comparisons of the first stage.

5.1.2. Comparison against state-of-the-art algorithms

Analogous to the first phase, Table 5 presents the average generalisation results for the $RMSPE$ metric obtained by the methods involved in the second stage of the experimental design on the selected benchmark datasets. The best result from each dataset again is highlighted in bold-face and the second one in italics. Again, the EUDMV model seems to be the best performing model reaching the best performance in fifty regression datasets out of the seventy-five considered.

Following the same statistical procedure as in the first phase, a new non-parametric Friedman test was performed using the $RMSPE$ rankings of the EUDMV and state-of-the-art bagging models in order to determine the statistical significance of the experimental results shown in Table 5. For the 6 methods selected to this stage on the used regression problems, the

ID	EUDMV	BEM	GEM	SR	SRR	ABM
1	**0.8398**$_{0.1188}$	0.9045$_{0.1304}$	0.8836$_{0.1727}$	*0.8590*$_{0.1509}$	0.8863$_{0.1259}$	1.0560$_{0.1804}$
2	**0.8387**$_{0.0770}$	1.4087$_{0.1797}$	1.4111$_{0.1763}$	*1.4075*$_{0.1723}$	1.4109$_{0.1840}$	1.4124$_{0.1752}$
3	0.3910$_{0.0026}$	0.2101$_{0.0054}$	**0.2095**$_{0.0056}$	0.2097$_{0.0056}$	*0.2097*$_{0.0061}$	0.2107$_{0.0075}$
4	**1.6359**$_{0.0439}$	*2.4432*$_{0.0625}$	2.4603$_{0.1292}$	2.4643$_{0.0654}$	2.5141$_{0.0890}$	2.5273$_{0.1320}$
5	0.4002$_{0.0010}$	0.2341$_{0.0079}$	0.2341$_{0.0081}$	0.2342$_{0.0081}$	**0.2339**$_{0.0081}$	*0.2339*$_{0.0078}$
6	0.5807$_{0.2650}$	0.5122$_{0.1828}$	**0.1665**$_{0.1258}$	*0.2246*$_{0.2578}$	0.2991$_{0.2103}$	0.6934$_{0.1699}$
7	**0.4611**$_{0.0125}$	0.5839$_{0.0484}$	*0.5830*$_{0.0455}$	0.5838$_{0.0473}$	0.5835$_{0.0481}$	0.5870$_{0.0471}$
8	**0.5960**$_{0.0287}$	0.7479$_{0.0400}$	0.7362$_{0.0511}$	0.7379$_{0.0541}$	0.7418$_{0.0612}$	*0.7229*$_{0.0519}$
9	*0.5985*$_{0.1164}$	0.6320$_{0.0345}$	0.6264$_{0.0371}$	0.6205$_{0.0344}$	0.6379$_{0.0543}$	**0.5862**$_{0.0707}$
10	0.4398$_{0.0389}$	0.1521$_{0.0332}$	0.1518$_{0.0336}$	0.1519$_{0.0336}$	*0.1514*$_{0.0337}$	**0.1504**$_{0.0335}$
11	**0.4088**$_{0.0594}$	0.5434$_{0.1943}$	0.5394$_{0.1857}$	0.5344$_{0.1721}$	0.5293$_{0.1689}$	*0.5266*$_{0.1633}$
12	**9.1090**$_{7.8539}$	13.4370$_{10.1498}$	15.6487$_{12.4735}$	15.6373$_{10.7781}$	14.5277$_{11.9302}$	*12.7563*$_{10.2249}$
13	**0.8602**$_{0.4231}$	1.2774$_{0.9178}$	1.2512$_{0.8886}$	1.2756$_{0.9245}$	*1.2276*$_{0.8958}$	1.2401$_{0.8440}$
14	**0.7219**$_{0.3789}$	1.0480$_{0.7660}$	1.0388$_{0.7597}$	1.0251$_{0.7210}$	1.0430$_{0.7801}$	*1.0129*$_{0.7268}$
15	0.4082$_{0.0265}$	**0.1453**$_{0.0339}$	*0.1453*$_{0.0340}$	0.1454$_{0.0340}$	0.1455$_{0.0342}$	0.1458$_{0.0347}$
16	0.6578$_{0.1883}$	0.6925$_{0.1219}$	*0.6387*$_{0.1314}$	0.6406$_{0.1240}$	0.7005$_{0.1188}$	**0.6194**$_{0.0716}$
17	**0.5121**$_{0.2595}$	0.9603$_{0.5685}$	1.0485$_{0.6196}$	1.0945$_{0.6438}$	1.0464$_{0.6403}$	*0.7497*$_{0.4230}$
18	**0.3142**$_{0.1207}$	0.5785$_{0.2704}$	*0.5347*$_{0.2079}$	0.5367$_{0.2145}$	0.7016$_{0.3041}$	0.8401$_{0.6566}$
19	0.3714$_{0.0406}$	0.1484$_{0.0369}$	*0.1477*$_{0.0368}$	0.1478$_{0.0369}$	0.1482$_{0.0373}$	**0.1461**$_{0.0360}$
20	**0.8261**$_{0.1126}$	1.1695$_{0.2114}$	1.1719$_{0.2019}$	1.1705$_{0.2042}$	*1.1693*$_{0.2011}$	1.1870$_{0.2120}$
21	**0.6164**$_{0.3465}$	0.7373$_{0.7357}$	0.7308$_{0.7261}$	*0.7281*$_{0.7220}$	0.7288$_{0.7323}$	0.7350$_{0.7217}$
22	**1.4237**$_{0.4995}$	1.9674$_{0.9462}$	1.9811$_{0.8256}$	2.0661$_{0.9665}$	*1.9035*$_{0.8127}$	2.0518$_{1.1005}$
23	**0.9823**$_{0.2238}$	1.5729$_{0.4202}$	*1.4917*$_{0.4690}$	1.6202$_{0.5813}$	1.5664$_{0.5586}$	1.5530$_{0.4432}$
24	0.4314$_{0.0828}$	0.3918$_{0.1134}$	*0.3851*$_{0.1063}$	0.3974$_{0.0819}$	0.4121$_{0.0867}$	**0.3537**$_{0.0895}$
25	**0.9635**$_{0.3269}$	1.4684$_{0.5405}$	1.4506$_{0.5303}$	*1.4057*$_{0.5243}$	1.4843$_{0.5893}$	1.5355$_{0.5782}$
26	**0.5013**$_{0.0857}$	0.6561$_{0.2885}$	*0.6478*$_{0.3172}$	0.7183$_{0.3487}$	0.6634$_{0.3458}$	0.7177$_{0.3450}$
27	**1.1725**$_{0.4729}$	*1.7592*$_{0.5549}$	1.8601$_{0.7246}$	1.9781$_{0.7713}$	1.8952$_{0.6909}$	1.9926$_{0.8306}$
28	**0.5337**$_{0.2464}$	0.7240$_{0.3298}$	0.7373$_{0.3809}$	0.7398$_{0.4024}$	0.7277$_{0.3860}$	*0.7016*$_{0.3548}$
29	**0.9160**$_{0.2762}$	*1.4208*$_{0.3923}$	1.5200$_{0.3564}$	1.6003$_{0.5182}$	1.6491$_{0.5526}$	1.5016$_{0.4141}$
30	**0.5542**$_{0.1534}$	0.6817$_{0.1485}$	0.7300$_{0.1220}$	0.6894$_{0.1291}$	*0.6482*$_{0.1527}$	0.7104$_{0.1700}$
31	0.4837$_{0.0473}$	0.4504$_{0.1656}$	**0.4476**$_{0.1655}$	*0.4489*$_{0.1672}$	0.4492$_{0.1681}$	0.4499$_{0.1785}$
32	**1.1471**$_{0.1841}$	1.8558$_{0.3333}$	1.8529$_{0.3856}$	1.8983$_{0.3488}$	1.8764$_{0.2763}$	*1.8073*$_{0.3929}$
33	**0.6565**$_{0.3925}$	0.8517$_{0.7752}$	*0.7780*$_{0.7261}$	0.8080$_{0.7784}$	0.7833$_{0.7648}$	0.7940$_{0.7358}$
34	**0.9673**$_{0.4021}$	1.4034$_{0.7387}$	*1.3708*$_{0.7304}$	1.3900$_{0.6907}$	1.3744$_{0.7348}$	1.4108$_{0.7603}$
35	0.3605$_{0.0294}$	0.0731$_{0.0376}$	0.0756$_{0.0402}$	*0.0721*$_{0.0371}$	0.0731$_{0.0357}$	**0.0659**$_{0.0230}$
36	0.3544$_{0.0134}$	0.0923$_{0.0118}$	0.0878$_{0.0124}$	**0.0827**$_{0.0045}$	0.1011$_{0.0168}$	*0.0861*$_{0.0223}$
37	0.4182$_{0.0166}$	0.0815$_{0.0265}$	0.0817$_{0.0254}$	0.0819$_{0.0255}$	**0.0811**$_{0.0262}$	*0.0811*$_{0.0253}$
38	*0.2366*$_{0.0813}$	0.2696$_{0.0740}$	0.2891$_{0.0591}$	0.2883$_{0.0602}$	**0.2048**$_{0.1089}$	0.2661$_{0.1335}$
39	**1.3286**$_{0.4999}$	*1.9868*$_{0.7699}$	2.0995$_{0.9594}$	2.1716$_{0.9495}$	2.0047$_{0.9614}$	2.1728$_{0.9098}$
40	**0.4369**$_{0.1333}$	0.5509$_{0.1735}$	0.5513$_{0.1796}$	0.5697$_{0.1661}$	0.5302$_{0.1402}$	*0.5052*$_{0.1396}$
41	**1.4254**$_{0.5832}$	*2.1328*$_{0.9998}$	2.1399$_{1.0217}$	2.2986$_{0.9969}$	2.1622$_{1.0453}$	2.2979$_{1.1038}$
42	**0.5134**$_{0.0990}$	*0.6881*$_{0.4012}$	0.7072$_{0.3816}$	0.7093$_{0.4065}$	0.7095$_{0.4303}$	0.8244$_{0.4702}$
43	0.4363$_{0.0320}$	0.1208$_{0.0478}$	*0.1207*$_{0.0483}$	0.1222$_{0.0491}$	0.1221$_{0.0484}$	**0.1184**$_{0.0449}$
44	**0.7934**$_{0.2168}$	1.1630$_{0.3258}$	1.1431$_{0.2970}$	*1.1343*$_{0.3677}$	1.1698$_{0.2430}$	1.2275$_{0.3898}$
45	**0.7935**$_{0.5977}$	1.3327$_{0.6308}$	*1.1875*$_{0.9942}$	1.4463$_{1.1325}$	1.2211$_{0.8198}$	2.4882$_{2.3198}$
46	0.4150$_{0.0237}$	0.3199$_{0.0715}$	**0.3192**$_{0.0772}$	*0.3195*$_{0.0762}$	0.3210$_{0.0799}$	0.3219$_{0.0782}$
47	**0.4509**$_{0.0242}$	*0.5121*$_{0.0546}$	0.5161$_{0.0652}$	0.5127$_{0.0698}$	0.5143$_{0.0626}$	0.5378$_{0.0750}$
48	**0.9664**$_{0.2461}$	1.5166$_{0.3750}$	*1.4194*$_{0.4014}$	1.4981$_{0.4744}$	1.4646$_{0.3654}$	1.6672$_{0.3843}$
49	**0.5244**$_{0.0668}$	*0.5971*$_{0.2421}$	0.6693$_{0.2016}$	0.6167$_{0.1992}$	0.6062$_{0.2646}$	0.6293$_{0.2471}$
50	0.5754$_{0.0561}$	0.4688$_{0.0743}$	0.4632$_{0.0691}$	0.4619$_{0.0715}$	*0.4576*$_{0.0731}$	**0.4510**$_{0.0973}$
51	**0.6497**$_{0.1395}$	0.8340$_{0.2918}$	0.7910$_{0.2660}$	0.7865$_{0.2599}$	0.8118$_{0.2522}$	*0.7478*$_{0.1786}$
52	**0.7086**$_{0.0130}$	1.0144$_{0.0249}$	1.0133$_{0.0275}$	1.0145$_{0.0280}$	1.0057$_{0.0276}$	*0.9976*$_{0.0353}$
53	**0.5341**$_{0.0331}$	0.9533$_{0.1535}$	0.9104$_{0.1417}$	0.9229$_{0.0927}$	0.9187$_{0.1475}$	*0.8695*$_{0.1934}$
54	0.3282$_{0.0241}$	*0.1508*$_{0.0326}$	**0.1504**$_{0.0321}$	0.1509$_{0.0321}$	0.1509$_{0.0323}$	0.1522$_{0.0321}$
55	**0.5329**$_{0.0698}$	0.6646$_{0.0863}$	0.6661$_{0.0938}$	0.6660$_{0.0951}$	0.6642$_{0.0872}$	*0.6396*$_{0.0987}$
56	**0.5964**$_{0.0157}$	*0.7887*$_{0.0616}$	0.7915$_{0.0634}$	0.7914$_{0.0631}$	0.7914$_{0.0658}$	0.7994$_{0.0721}$
57	0.5148$_{0.0790}$	*0.5068*$_{0.2717}$	0.5120$_{0.2951}$	0.5076$_{0.2861}$	0.5087$_{0.2894}$	**0.4984**$_{0.2878}$
58	0.4655$_{0.0744}$	0.4229$_{0.0613}$	*0.4179*$_{0.0714}$	0.4196$_{0.0676}$	0.4193$_{0.0642}$	**0.4113**$_{0.0912}$
59	0.4214$_{0.0248}$	*0.2958*$_{0.0662}$	0.2983$_{0.0627}$	**0.2925**$_{0.0602}$	0.3002$_{0.0698}$	0.2972$_{0.0567}$
60	**0.6421**$_{0.1157}$	1.0691$_{0.2322}$	1.0657$_{0.2291}$	1.0710$_{0.2140}$	1.0153$_{0.2185}$	*0.9348*$_{0.0987}$
61	0.4845$_{0.0366}$	0.4398$_{0.0494}$	0.4508$_{0.0468}$	0.4423$_{0.0483}$	*0.4319*$_{0.0462}$	**0.4306**$_{0.0359}$
62	**0.5114**$_{0.0276}$	0.9619$_{0.0723}$	0.9561$_{0.0726}$	*0.9428*$_{0.0785}$	0.9726$_{0.0694}$	0.9762$_{0.1033}$
63	**0.6323**$_{0.1570}$	0.8622$_{0.2916}$	0.8032$_{0.2524}$	0.8365$_{0.3066}$	*0.7939*$_{0.2471}$	0.8226$_{0.2569}$
64	0.4276$_{0.0970}$	0.3210$_{0.1767}$	*0.3155*$_{0.1315}$	0.3162$_{0.1353}$	**0.3050**$_{0.1246}$	0.3223$_{0.1712}$
65	**0.4409**$_{0.1678}$	0.8662$_{0.4203}$	0.8858$_{0.4114}$	0.9168$_{0.4287}$	0.8772$_{0.4148}$	*0.8107*$_{0.4677}$
66	**0.3920**$_{0.1043}$	0.4564$_{0.2566}$	0.4563$_{0.2566}$	0.4563$_{0.2566}$	*0.4561*$_{0.2564}$	0.4563$_{0.2566}$
67	**0.3556**$_{0.0448}$	0.6123$_{0.3042}$	0.6081$_{0.3039}$	0.6112$_{0.3046}$	0.6000$_{0.3012}$	*0.5906*$_{0.2763}$
68	**1.1213**$_{0.4987}$	*1.5762*$_{0.5410}$	1.6772$_{0.6113}$	1.5921$_{0.5730}$	1.7863$_{0.5813}$	2.5360$_{1.6897}$
69	**1.0733**$_{0.5397}$	1.4600$_{0.7675}$	1.7212$_{0.9284}$	1.5143$_{0.9165}$	1.5459$_{0.8846}$	*1.3389*$_{0.6449}$
70	0.3796$_{0.0526}$	0.3424$_{0.0862}$	0.3015$_{0.0554}$	*0.2879*$_{0.0989}$	**0.2878**$_{0.0436}$	0.3086$_{0.0519}$
71	**0.6733**$_{0.1876}$	0.9903$_{0.4073}$	0.9546$_{0.4306}$	0.9814$_{0.4696}$	*0.9344*$_{0.4242}$	1.0641$_{0.4083}$
72	**1.0577**$_{0.7241}$	1.6557$_{1.1512}$	1.6977$_{1.0820}$	1.6048$_{1.0038}$	1.4440$_{0.9031}$	*1.2599*$_{0.6999}$
73	*2.6764*$_{2.5852}$	3.7975$_{3.5599}$	**2.4502**$_{1.9996}$	2.8646$_{3.2645}$	2.8079$_{2.3239}$	9.0154$_{12.5727}$
74	**1.7563**$_{1.8167}$	2.7393$_{3.2121}$	2.7029$_{3.1218}$	2.7800$_{3.1330}$	2.6796$_{3.0206}$	*2.5879*$_{2.8495}$
75	**0.6330**$_{0.2495}$	0.7510$_{0.4985}$	0.7552$_{0.5257}$	0.7754$_{0.5521}$	0.7539$_{0.5333}$	*0.6654*$_{0.3977}$

Table 5: Second stage experimental results: Comparison against the state-of-the-art models, in terms of RMSPE.

confidence interval for the Friedman test was $C_0 = (0, F_{0.05} = 2.2387)$, and the rank differences set the F-distribution statistic value as $F^* = 6.6707 \notin C_0$. Therefore, again the null hypothesis stating that all models have the same performance for regression in mean $RMSPE$ rankings is rejected. Based on such rejection, the Holm post-hoc test is used to compare in $RMSPE$. Table 7 summarizes the output of Holm post-hoc test.

Based on the results of the statistical procedure collated in Table 7, the EUDMV method is significantly better in performance than the state-of-the-arts bagging methods. EUDMV obtains the best average performance in generalisation, $\overline{RMSPE} = 0.8071$, and the best mean ranking, $\overline{R}_{RMSPE} = 2.4864$, in regression problems.

Method	$\overline{RMSPE}$	$\overline{R}_{RMSPE}$	z-statistic	p-value	$\alpha_{0.05}$	$\alpha_{0.10}$
SR$_\bullet$	1.0992	3.9729	4.8331	0.0	0.01	0.02
BEM$_\bullet$	1.0775	3.9459	4.7453	0.0	0.0125	0.025
ABM$_\bullet$	1.1666	3.6216	3.6907	2.2E-4	0.0167	0.0333
SRR$_\bullet$	*1.0719*	3.500	3.2953	9.8E-4	0.025	0.05
GEM$_\bullet$	1.0822	*3.4729*	3.2074	0.0013	0.05	0.1
EUDMV	**0.8071**	**2.4864**	-	-	-	-

•: Statistical difference with $\alpha = 0.05$

Table 6: Statistical results for the comparisons of the second stage.

Method	$\overline{RMSPE}$	$\overline{R}_{RMSPE}$	z-statistic	p-value	$\alpha_{0.05}$	$\alpha_{0.10}$
SR$_\bullet$	1.0992	3.9729	4.8331	0.0	0.01	0.02
BEM$_\bullet$	1.0775	3.9459	4.7453	0.0	0.0125	0.025
ABM$_\bullet$	1.1666	3.6216	3.6907	2.2E-4	0.0167	0.0333
SRR$_\bullet$	*1.0719*	3.500	3.2953	9.8E-4	0.025	0.05
GEM$_\bullet$	1.0822	*3.4729*	3.2074	0.0013	0.05	0.1
EUDMV	**0.8071**	**2.4864**	-	-	-	-

•: Statistical difference with $\alpha = 0.05$

Table 7: Statistical results for the comparisons of the second stage.

The results obtained by the SR method are explained by the fact that the method does not include a constraint that the weights it assigns to each base learner in the ensemble must be greater than 0. This feature leads to the model suffering from overfitting on the training set, which negatively

affects the generalisation performance. In the case of the BEM method, it overcomes this particularity but suffers from assigning each base element the same importance, a problem discussed in Section 1. The inclusion of the concept of ambiguity in the formulation of the ABM method means that this model overcomes these drawbacks. Such a feature could be interpreted as incorporating the concept of diversity. The SRR and GEM models control for overfitting by either including a regularisation term or by defining a lower bound on the weight given by the model to each of the base learner, respectively. Both approaches were already justified in (Chen and Yao, 2009), where the importance of including a regularisation term in ensemble models is highlighted, and in (Breiman, 1996b), where the relevance of controlling the weights assigned by imposing restrictions on the value of the weights is considered. While integrating all the positive aspects outlined in the previous models, the EUDMV model allows the system to adjust for variance and covariance bias simultaneously.

5.2. Analysis of the robustness of results

One of the advantages of applying mean-variance models to adjust ensemble weights is not only to improve predictive performance but also to ensure a certain degree of robustness. Models of this nature are expected to exhibit lower standard deviation in the results in terms of their performance metric. Therefore, we have decided to perform a comparative analysis of the stability of the ensemble models' results.

Employing the identical statistical methodology as employed in 5.1 where the predictive performance was analyzed, a non-parametric Friedman test was conducted utilizing the RMSPE standard deviation rankings of both the EUDMV and the state-of-the-art bagging models. The primary aim was to determine the statistical significance of the experimental outcomes standard deviation included in Table 5 as subindices. For the six methods selected, the confidence interval for the Friedman test was $C_0 = (0, F_0.05 = 2.2383)$, and the rank differences established the F-distribution statistic value as $F^* = 24.0552 \notin C_0$. Consequently, the null hypothesis was rejected, which posits that all models demonstrate the same performance robustness in the mean of RMSPE standard deviation rankings. Following this rejection, the Holm post-hoc test was implemented to compare in RMSPE standard deviation. The results of the Holm post-hoc test are summarised in Table 8.

Based on the results of the statistical procedure collated in Table 8, the EUDMV method is significantly better in robustness than the state-

of-the-art bagging methods. EUDMV obtains the best average robustness, $\overline{sd(RMSPE)} = 0.3267$, and the best mean ranking, $\overline{R}_{sd(RMSPE)} = 1.64$, in the selected regression problems. This finding is in line with the literature which pointed out that BEM is typically more robust than other ensemble methods, such as boosting and stacking, because it reduces the variance of the model by decreasing the chance of overfitting to the training data and, as described in the methodological section, the proposed method generalizes the BEM approach (converging to the baseline with high values of δ).

Method	$\overline{sd(RMSPE)}$	$\overline{R}_{sd(RMSPE)}$	z-statistic	p-value	$\alpha_{0.05}$	$\alpha_{0.10}$
ABM$_\bullet$	0.665	4.02	7.7904	0.0	0.01	0.02
SR$_\bullet$	0.5276	4.02	7.7904	0.0	0.0125	0.025
SRR$_\bullet$	0.5191	3.968	7.6813	0.0	0.0167	0.0333
GEM$_\bullet$	0.5231	3.68	6.6775	0.0	0.025	0.05
BEM$_\bullet$	*0.50495*	*3.6533*	6.5902	0.0	0.05	0.1
EUDMV	**0.3267**	**1.64**	-	-	-	-

$\bullet$: Statistical difference with $\alpha = 0.05$

Table 8: Statistical results for the comparisons of RMSPE standard deviation.

5.3. Analysis of the computational burden

This section analyzes and compares the proposed methods' computational complexity with the baseline ensemble bagging models. It is essential to clarify that we will focus only on the aggregation part of the algorithms, as the bootstrapping stage of all methods is the same. The aggregation stage of the methods that include in their formulations the non-shorting constraints (EMVNSC and EDMVNSC) could be decomposed in three steps: (i) to compute the mean of a matrix with size $N^v \times S$, (ii) to calculate the covariance matrix of the same matrix, and (iii) to solve a QP problem in which the matrices involved have sizes $S \times S$ and $S \times 1$. The computational complexity of the algorithms can be estimated using big O notation, which provides an upper bound on the computational burden of the algorithm. The computational burden of computing the mean is $O(N^v \cdot S)$, whereas the complexity of calculating the covariance is $O(N^v S \cdot S)$. Regarding the QP problem (with a quadratic matrix of size $S \times S$ and a linear part with size Sx1), it is essential to clarify that one can use an optimization algorithm such as the interior point method or active set to solve it. The computational complexity of

these algorithms is typical $O(S^3)$. Hence, the total computational complexity of the algorithms is the sum of the complexities of each step, which is $O(N^v S + N^v S \cdot S + S^3)$.

On the other hand, the two methods which do not include the non-shorting constraints (EUMV and EUDMV) estimate the optimal weights analytically, and, consequently, the aggregation stage is the same as the previous one but substituting the QP part by an inversion of a matrix with size $S \times S$. The most common algorithm used for matrix inversion is the Gaussian elimination method with partial pivoting, which has a computational complexity of $O(S^3)$ in the worst case. Thus, the computational complexity of all methods proposed is $O(N^v \cdot S + N^v S \cdot S + S^3)$.

Regarding state-of-the-art methods, the GEM approach solves a non-negative least squares problem of a matrix with size $N^v \times S$, which typically has a computational complexity of $O(N^v \cdot S^2)$. The SR and SRR methods estimate the ensemble' weights by solving a least squares problem. One standard algorithm used to solve least-squares problems is the QR decomposition method, which has a computational complexity of $O(N^v \cdot S^2)$ in the worst case. Finally, the ABM requires the computation of a covariance matrix and the resolution of an LP problem, which have a computational complexity of $O(N^v \cdot S^2 + 2^S)$ (if the LP problem is solved by the simplex method).

To compare execution times among different methods empirically, we have developed an experimental procedure that involves randomly generated regression problems of varying sizes. Thus, we have generated random datasets and configured methods with varying numbers of base learners. Since the aggregation part of all analyzed methods depends primarily on two parameters - the size of the validation set, N^v, and the size of the ensemble, S - the procedure enables a comparative analysis of how algorithms scale with these parameters. Additionally, this approach allows us to observe how different algorithms behave in scenarios not present in the datasets used to test the predictive performance of models, as explained in Section 4.1.

The efficiency of the EUDMV method in determining the coefficients of each base learner is demonstrated in Figure 3, while state-of-the-art methods GEM, SR, SRR, and ABM computational times are shown to perform the same task in Figures 4a, 4b, 4c, and 4d, respectively. Notably, the hyperparameter validation process is not considered for any of the methods. The axes of the figures represent the number of base learners used in the ensemble and the dataset size as a function of the number of patterns, respectively. We chose dataset sizes $N^v \in \{500, \ldots, 10.000\}$ and the number

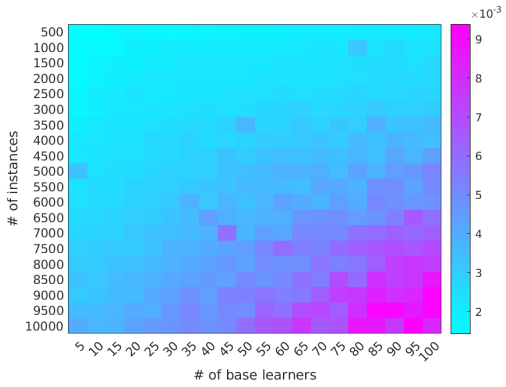


Figure 3: EUDMV method execution times.

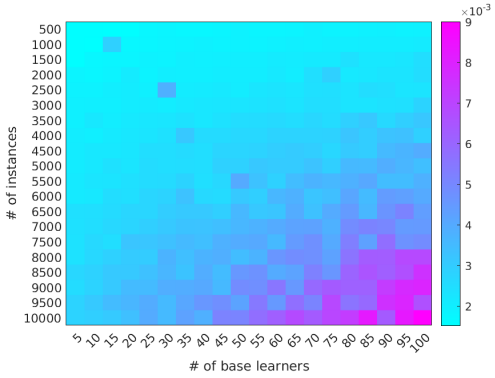
of base learners involved $S \in \{5, \ldots, 100\}$ to demonstrate the scaling mode of the methods as the problem grew. The color scheme used in the figures represents the execution time in seconds. Other MV methods were excluded from this analysis because they have the same computational complexity as EUDMV, and BEM was not included due to its unique simplicity.

From the empirical results, it is clear that all of the methods perform similarly in terms of scaling. This suggests that the computational overhead associated with calculating additional elements of the proposed methods, such as the error matrix, mean vector, and covariance matrix, is negligible in the programming environment of MATLAB, as it does not appear to significantly affect execution times.

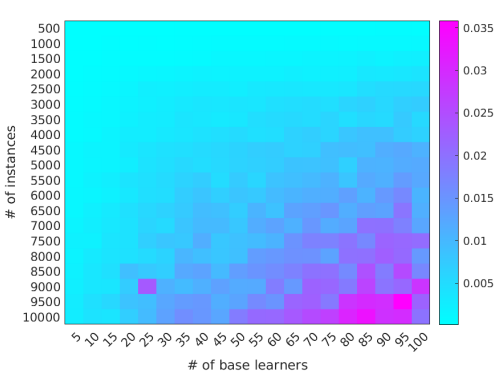
5.4. Hyperparameter Sensitivity Analysis

The proposed EUDMV methodology is based on two hyperparameters that must be set a priori: λ and δ. To study the sensitivity of the method in terms of $RMSPE$ to these two hyperparameters, a behavioural analysis was performed on different datasets. During this hyperparameter sensitivity test, the number of ensemble elements was fixed at 5 (S = 5), all linear regressors. Specifically, the study was conducted considering the regression datasets `wimbledon-men-2013b`, `delta_elv`, `slump-flow` and `parkinsons-speech`. The EUDMV method was run ten times in a 5-fold cross-validation for hyperparameter values ranging in the sets $\lambda, \delta \in \{10^{-3}, 10^{-2} \ldots, 10^{2}, 10^{3}\}$.

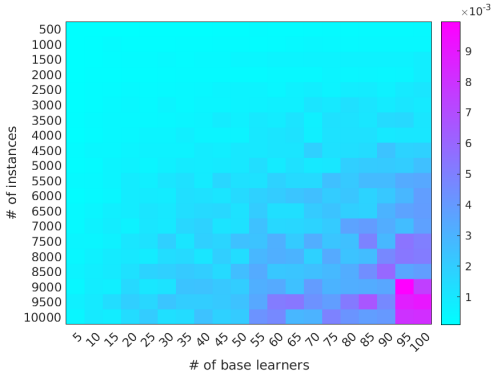
Figure 5 illustrates the mean $RMSPE$ performance of the ten runs per fold in the regression problems mentioned above. The axes where the hyperparameters are located are scaled logarithmically for the sake of visualisation.



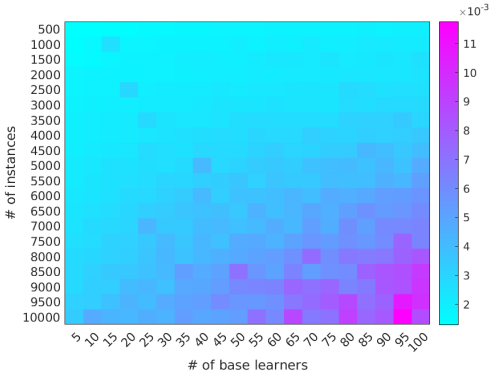
(a) GEM method execution times.



(b) SR method execution times.

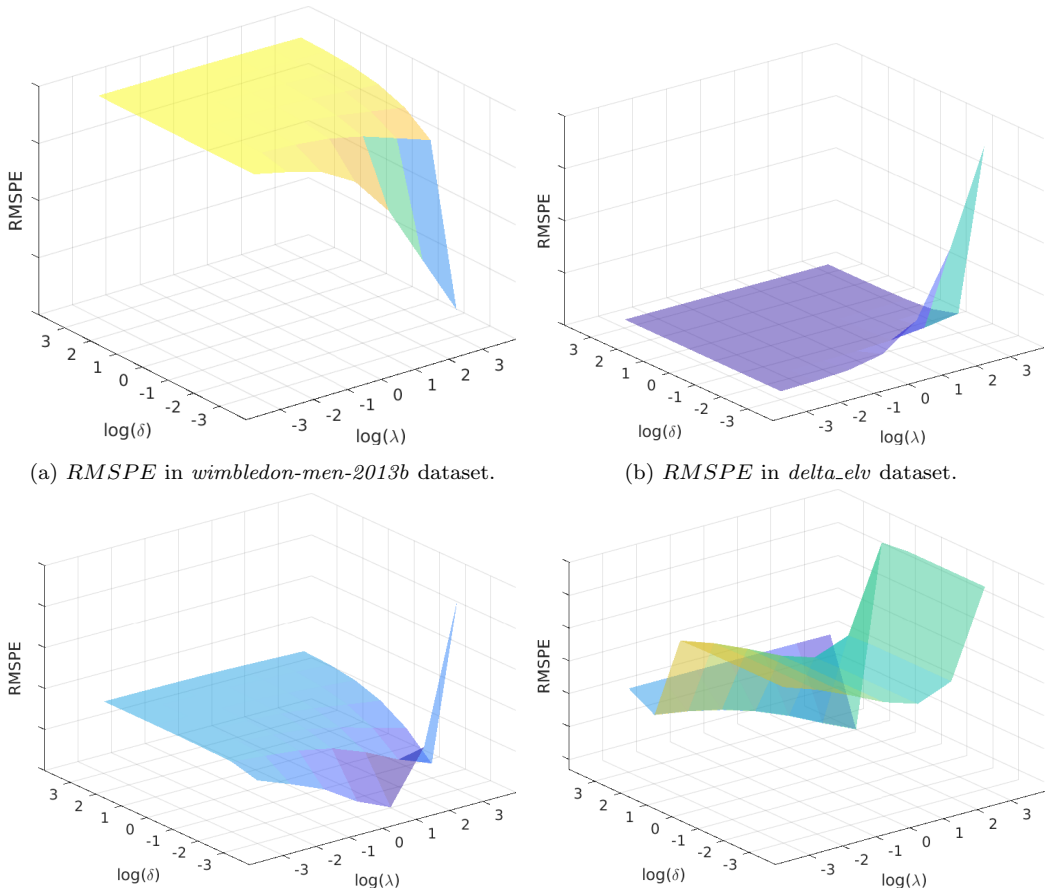


(c) SRR method execution times.



(d) ABM method execution times.

Figure 4: State-of-the-art methods execution times.



(a) $RMSPE$ in *wimbledon-men-2013b* dataset.

(b) $RMSPE$ in *delta_elv* dataset.

(c) $RMSPE$ in *slump-flow* dataset.

(d) $RMSPE$ in *parkinsons-speech* dataset.

Figure 5: Hyperparameters study on $RMSPE$ for the EUDMV method and the parameters λ and δ

Figure 5a and Figure 5b show opposite situations. In the first case, the best performance is obtained for high values of λ and low values of δ. In contrast, this combination is the worst possible combination for the `delta_elv` dataset. The situation posed in Figures 5c and 5d is that for both of these datasets, the combination of hyperparameter values at which the regressor gives optimal performance is at an intermediate point in the grid.

Therefore, as might be expected, it can be observed that the EUDMV method's behaviour proves to be very sensitive to different values of the hyperparameters λ and δ, and the recommendation is to estimate them by a complete cross-validation procedure with each dataset in particular.

6. Conclusions

Inspired by the principles of portfolio optimisation from the field of Finance, this paper presents an innovative methodological approach in the framework of ensemble learning to address regression problems. In full alignment with the bias-variance-covariance theory, this formulation allows determining the weights of the ensemble elements by considering the errors and variance of the predictions of the base learners and the covariance of these predictions. Consequently, four different models have been developed under these tenets. The first two ensemble models, EMVNSC and EDMVNSC, impose non-negativity constraints on the weights assigned to the ensemble elements and that they all sum one. They differ in that EDMVNSC includes a regularisation term. The second two ensemble models, EUMV and EUDMV, do not impose non-negativity constraints on the weights of the ensemble elements, resulting in a convex quadratic programming (QP) problem because the matrix included in the quadratic term is symmetric and positive definite. Likewise to the first ones, EUDMV incorporates regularisation.

The proposed methodologies were applied on 75 benchmark regression datasets in a comprehensive two-stage experimental process. In the first stage, the performances of the models implemented under the new paradigm were compared in a thorough statistical procedure, resulting EUDMV model as the best in terms of $RMSPE$ metric. In the second stage, comparing the EUDMV model with five state-of-the-art proven Bagging models confirmed that it achieves the best overall performance.

Several different directions might also be explored after the findings of this study. Based on the ideas reported in (Kim et al., 2014; Chen et al., 2020), one promising line would be to propose a new mean-variance approach

that could control for larger moments such as skewness and kurtosis. Another intriguing research path that could extend the ideas of this work would be exploring an alternative diversification strategy as an $l - 1$ regularisation. Finally, it might be interesting to quantitatively examine the importance of diversity on the generalisation performance of the presented models.

References

Abdelaziz, F.B., Aouni, B., El Fayedh, R., 2007. Multi-objective stochastic programming for portfolio selection. European Journal of Operational Research 177, 1811–1823.

Alcalá-Fdez, J., Sanchez, L., Garcia, S., del Jesus, M.J., Ventura, S., Garrell, J.M., Otero, J., Romero, C., Bacardit, J., Rivas, V.M., et al., 2009. Keel: a software tool to assess evolutionary algorithms for data mining problems. Soft Computing 13, 307–318.

Barandiaran, I., 1998. The random subspace method for constructing decision forests. IEEE Trans. Pattern Anal. Mach. Intell. 20, 1–22.

Bauer, E., Kohavi, R., 1999. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. Machine learning 36, 105–139.

Benartzi, S., Thaler, R.H., 2001. Naive diversification strategies in defined contribution saving plans. American economic review 91, 79–98.

Benítez-Peña, S., Carrizosa, E., Guerrero, V., Jiménez-Gamero, M.D., Martín-Barragán, B., Molero-Río, C., Ramírez-Cobo, P., Morales, D.R., Sillero-Denamiel, M.R., 2021. On sparse ensemble methods: An application to short-term predictions of the evolution of covid-19. European Journal of Operational Research 295.

Bhasuran, B., Murugesan, G., Abdulkadhar, S., Natarajan, J., 2016. Stacked ensemble combined with fuzzy matching for biomedical named entity recognition of diseases. Journal of biomedical informatics 64, 1–9.

Bird, R., Tippett, M., 1986. Note—naive diversification and portfolio risk—a note. Management Science 32, 244–251.

Bodnar, T., Parolya, N., Schmid, W., 2018. Estimation of the global minimum variance portfolio in high dimensions. European Journal of Operational Research 266, 371–390.

Boyd, S., Boyd, S.P., Vandenberghe, L., 2004. Convex optimization. Cambridge university press.

Breiman, L., 1996a. Bagging predictors. Machine learning 24, 123–140.

Breiman, L., 1996b. Stacked regressions. Machine learning 24, 49–64.

Breiman, L., 2001. Random forests. Machine learning 45, 5–32.

Brown, G., Wyatt, J., Harris, R., Yao, X., 2005a. Diversity creation methods: a survey and categorisation. Information fusion 6, 5–20.

Brown, G., Wyatt, J., Tino, P., 2005b. Managing Diversity in Regression Ensembles. Journal of Machine Learning Research 6, 1621–1650.

Bühlmann, P., Yu, B., 2002. Analyzing bagging. The annals of Statistics 30, 927–961.

Chen, B., Zhong, J., Chen, Y., 2020. A hybrid approach for portfolio selection with higher-order moments: Empirical evidence from shanghai stock exchange. Expert Systems with Applications 145, 113104.

Chen, H., Yao, X., 2009. Regularized negative correlation learning for neural network ensembles. IEEE Transactions on Neural Networks 20, 1962–1979.

Coqueret, G., 2015. Diversified minimum-variance portfolios. Annals of Finance 11, 221–241.

DeMiguel, V., Garlappi, L., Uppal, R., 2009. Optimal versus naive diversification: How inefficient is the 1/n portfolio strategy? The review of Financial studies 22, 1915–1953.

Demšar, J., 2006. Statistical comparisons of classifiers over multiple data sets. Journal of Machine learning research 7, 1–30.

Dheeru, D., Karra Taniskidou, E., 2019. UCI machine learning repository. URL: `http://archive.ics.uci.edu/ml`.

Duchin, R., Levy, H., 2009. Markowitz versus the talmudic portfolio diversification strategies. The Journal of Portfolio Management 35, 71–74.

Dutta, H., 2009. Measuring diversity in regression ensembles. Proc. IICAI 9, 17.

Ekbal, A., Saha, S., 2013. Stacked ensemble coupled with feature selection for biomedical entity extraction. Knowledge-Based Systems 46, 22–32.

Fernández-Navarro, F., Martínez-Nieto, L., Carbonero-Ruz, M., Montero-Romero, T., 2021. Mean squared variance portfolio: A mixed-integer linear programming formulation. Mathematics 9.

Friedman, J.H., Hall, P., 2007. On bagging and nonlinear estimation. Journal of statistical planning and inference 137, 669–683.

Friedman, M., 1940. A comparison of alternative tests of significance for the problem of m rankings. The Annals of Mathematical Statistics 11, 86–92.

García, S., Fernández, A., Luengo, J., Herrera, F., 2010. Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. Information sciences 180, 2044–2064.

Grandvalet, Y., 2004. Bagging equalizes influence. Machine Learning 55, 251–270.

Göçken, M., Özçalıcı, M., Boru, A., Dosdoğru, A.T., 2016. Integrating metaheuristics and artificial neural networks for improved stock price prediction. Expert Systems with Applications 44, 320–331.

Hansen, L.K., Salamon, P., 1990. Neural network ensembles. IEEE transactions on pattern analysis and machine intelligence 12, 993–1001.

Hashem, S., 1997. Optimal linear combinations of neural networks. Neural networks 10, 599–614.

Holm, S., 1979. A simple sequentially rejective multiple test procedure. Scandinavian journal of statistics 6, 65–70.

J. Zhou, Z. Jiang, F.L.C., Wang, S., 2020. Formulating ensemble learning of svms into a single svm formulation by negative agreement learning. IEEE Trans. Syst. Man Cybern. Syst. .

Jiménez, D., 1998. Dynamically weighted ensemble neural networks for classification, in: 1998 IEEE International Joint Conference on Neural Networks Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98CH36227), IEEE. pp. 753–756.

Kadkhodaei, H.R., Moghadam, A.M.E., Dehghan, M., 2020. Hboost: A heterogeneous ensemble classifier based on the boosting method and entropy measurement. Expert Systems with Applications 157, 113482.

Kim, W.C., Fabozzi, F.J., Cheridito, P., Fox, C., 2014. Controlling portfolio skewness and kurtosis without directly optimizing third and fourth moments. Economics Letters 122, 154–158.

Klein, R.W., Bawa, V.S., 1977. The effect of limited information and estimation risk on optimal portfolio diversification. Journal of Financial Economics 5, 89–111.

Krogh, A., Vedelsby, J., 1994. Neural network ensembles, cross validation, and active learning, in: Tesauro, G., Touretzky, D., Leen, T. (Eds.), Advances in Neural Information Processing Systems, MIT Press. pp. 231–238.

Kuhle, J., 1987. Portfolio diversification and return benefits—common stock vs. real estate investment trusts (reits). Journal of Real Estate Research 2, 1–9.

Kuncheva, L.I., Whitaker, C.J., 2001. Ten measures of diversity in classifier ensembles: Limits for two classifiers. Proc. DERA/IEE Workshop Intell. Sensor Process. , 1–10.

Li, H., Huang, Q., Wu, B., 2020. Improving the naive diversification: An enhanced indexation approach. Finance Research Letters , 101661.

Lim, A.E., Zhou, X.Y., 2002. Mean-variance portfolio selection with random parameters in a complete market. Mathematics of Operations Research 27, 101–120.

Lin, J.L., 2013. On the diversity constraints for portfolio optimization. Entropy 15, 4607–4621.

Liu, Yao, X., 1999a. Ensemble learning via negative correlation. Neural Netw. 12, 1399–1404.

Liu, Yao, X., 1999b. Simultaneous training of negatively correlated neural networks in an ensemble. IEEE Trans. Syst. 29, 716–725.

Liu, Y., Yao, X., 1999c. Ensemble learning via negative correlation. Neural networks 12, 1399–1404.

Liu, Y., Yao, X., Higuchi, T., 2000. Evolutionary ensembles with negative correlation learning. IEEE Transactions on Evolutionary Computation 4, 380–387.

Luengo, J., García, S., Herrera, F., 2009. A study on the use of statistical tests for experimentation with neural networks: Analysis of parametric test conditions and non-parametric tests. Expert Systems with Applications 36, 7798–7808.

Maillet, B., Tokpavi, S., Vaucher, B., 2015. Global minimum variance portfolio optimisation under some model risk: A robust regression-based approach. European Journal of Operational Research 244, 289–299.

Markowitz, H., 1952. Portfolio selection. The Journal of Finance 7, 77–91.

Markowitz, H., 2014. Mean–variance approximations to expected utility. European Journal of Operational Research 234, 346–355.

Masmoudi, M., Abdelaziz, F.B., 2018. Portfolio selection problem: a review of deterministic and stochastic multiple objective programming models. Annals of Operations Research 267, 335–352.

Opitz, D., Shavlik, J., 1995. Generating accurate and diverse members of a neural-network ensemble, in: Touretzky, D., Mozer, M., Hasselmo, M. (Eds.), Advances in Neural Information Processing Systems, MIT Press. pp. 535–541.

Perales-González, C., Carbonero-Ruz, M., Becerra-Alonso, D., Pérez-Rodríguez, J., Fernández-Navarro, F., 2019. Regularized ensemble neural

networks models in the extreme learning machine framework. Neurocomputing 361, 196–211.

Perales-González, C., Carbonero-Ruz, M., Pérez-Rodríguez, J., Becerra-Alonso, D., Fernández-Navarro, F., 2020. Negative correlation learning in the extreme learning machine framework. Neural Computing and Applications 32, 13805–13823.

Perales-González, C., Fernández-Navarro, F., Carbonero-Ruz, M., Pérez-Rodríguez, J., 2021. Global negative correlation learning: A unified framework for global optimization of ensemble models. IEEE Transactions on Neural Networks and Learning Systems , 1–12doi:10.1109/TNNLS.2021.3055734.

Perrone, M., Cooper, L., 1992. When networks disagree: Ensemble methods for hybrid neural networks. chapter 1. pp. 342–358.

Peykani, P., Mohammadi, E., Emrouznejad, A., Pishvaee, M.S., Rostamy-Malkhalifeh, M., 2019. Fuzzy data envelopment analysis: an adjustable approach. Expert Systems with Applications 136, 439–452.

Pham, H., Olafsson, S., 2020. On cesaro averages for weighted trees in the random forest. Journal of Classification 37, 223–236.

R Ünlü, P.X., 2021. A reduced variance unsupervised ensemble learning algorithm based on modern portfolio theory. Expert Systems with Applications 180.

Reeve, H.W., Brown, G., 2018. Diversity and degrees of freedom in regression ensembles. Neurocomputing 298, 55–68.

Sadigh, A.N., Mokhtari, H., Iranpoor, M., Ghomi, S., 2012. Cardinality constrained portfolio optimization using a hybrid approach based on particle swarm optimization and hopfield neural network. Advanced Science Letters 17, 11–20.

Sankaran, J.K., Patil, A.A., 1999. On the optimal selection of portfolios under limited diversification. Journal of banking & Finance 23, 1655–1666.

Schmidt, A.B., 2019. Managing portfolio diversity within the mean variance theory. Annals of Operations Research 282, 315–329.

Shahhosseini, M., Hu, G., Pham, H., 2022. Optimizing ensemble weights and hyperparameters of machine learning models for regression problems. Machine Learning with Applications 7, 100251.

Shcherbakov, M.V., Brebels, A., Shcherbakova, N.L., Tyukov, A.P., Janovsky, T.A., Kamaev, V.A., et al., 2013. A survey of forecast error measures. World applied sciences journal 24, 171–176.

Shen, Z.Q., Kong, F.S., 2004. Dynamically weighted ensemble neural networks for regression problems, in: Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No. 04EX826), IEEE. pp. 3492–3496.

Tu, J., Zhou, G., 2011. Markowitz meets talmud: A combination of sophisticated and naive diversification strategies. Journal of Financial Economics 99, 204–215.

Windcliff, H., Boyle, P.P., 2004. The 1/n pension investment puzzle. North American Actuarial Journal 8, 32–45.

Yaman, I., Dalkılıç, T.E., 2021. A hybrid approach to cardinality constraint portfolio selection problem based on nonlinear neural network and genetic algorithm. Expert Systems with Applications 169, 114517.

Yang, S., Browne, A., 2004. Neural network ensembles: combining multiple models for enhanced performance using a multistage approach. Expert Systems 21, 279–288.

Yin, G., Zhou, X.Y., 2004. Markowitz's mean-variance portfolio selection with regime switching: From discrete-time models to their continuous-time limits. IEEE Transactions on automatic control 49, 349–360.

Zhou, R., Palomar, D.P., 2020. Understanding the quintile portfolio. IEEE Transactions on Signal Processing 68, 4030–4040.

Zhou, Z.H., Wu, J., Tang, W., 2002. Ensembling neural networks: many could be better than all. Artificial intelligence 137, 239–263.