REGULAR ARTICLE

View selection in multi-view stacking: choosing the meta-learner

Wouter van Loon[1] · Marjolein Fokkema[1] · Botond Szabo[2,3,4] ·
Mark de Rooij[1]

Abstract

Multi-view stacking is a framework for combining information from different views (i.e. different feature sets) describing the same set of objects. In this framework, a *base-learner* algorithm is trained on each view separately, and their predictions are then combined by a *meta-learner* algorithm. In a previous study, stacked penalized logistic regression, a special case of multi-view stacking, has been shown to be useful in identifying which views are most important for prediction. In this article we expand this research by considering seven different algorithms to use as the meta-learner, and evaluating their view selection and classification performance in simulations and two applications on real gene-expression data sets. Our results suggest that if both view selection and classification accuracy are important to the research at hand, then the nonnegative lasso, nonnegative adaptive lasso and nonnegative elastic net are suitable meta-learners. Exactly which among these three is to be preferred depends on the research context. The remaining four meta-learners, namely nonnegative ridge regression, nonnegative forward selection, stability selection and the interpolating predictor, show little advantages in order to be preferred over the other three.

✉ Wouter van Loon
   w.s.van.loon@fsw.leidenuniv.nl

1   Department of Methodology and Statistics, Leiden University, Leiden, The Netherlands

2   Department of Mathematics, VU Amsterdam, Amsterdam, The Netherlands

3   Present Address: Department of Decision Sciences, Bocconi University, Milan, Italy

4   Present Address: Bocconi Institute for Data Science and Analytics, Bocconi University, Milan, Italy

🍎 Springer

1 Introduction

In high-dimensional biomedical studies, a common goal is to create an accurate classification model using only a subset of the features (Li et al. 2018). A popular approach to this type of joint classification and feature selection problem is to apply penalized methods such as the *lasso* (Tibshirani 1996). These methods promote sparsity by imposing a penalty on the coefficient vector so that, for a sufficiently large value of the tuning parameter(s), some coefficients will be set to zero during the model fitting process. The tuning parameter decides on the relative importance of the penalty term, and is typically chosen by minimizing the cross-validation error (Friedman et al. 2009). However, biomedical features are often naturally grouped into distinct feature sets. In genomics, for example, genes may be grouped into gene sets or genetic pathways (Wang et al. 2010), while in neuroimaging, different sets of anatomical markers may be calculated from MRI scans (De Vos et al. 2016). Features may also be grouped at a higher level, for example because they correspond to a certain imaging modality or data source (Fratello et al. 2017). Such naturally occurring groups of features describing the same set of objects are known as different *views* of the data, and integrating the information in these different views through machine learning methods is known as *multi-view learning* (Zhao et al. 2017; Sun et al. 2019). In a multi-view setting, it is often more desirable to select or discard entire views rather than individual features, turning the feature selection problem into a view selection problem.

Stacked penalized logistic regression (StaPLR) (Van Loon et al. 2020) is a method specifically developed to tackle the joint classification and view selection problem. Compared with a variant of the lasso for selecting groups of features (the so-called *group lasso* (Yuan and Lin 2007)), StaPLR was empirically shown to be more accurate in view selection, producing sparser models with an often comparable classification accuracy, and offering computational advantages (Van Loon et al. 2020). StaPLR is a special case of a more general framework called *multi-view stacking* (MVS) (Van Loon et al. 2020; Li et al. 2011; Garcia-Ceja et al. 2018). In MVS, a learning algorithm (the *base-learner*) is trained on each view separately, and another algorithm (the *meta-learner*) is then trained on the cross-validated predictions of the view-specific models. The meta-learner thus learns how to best combine the predictions of the individual views. If the meta-learner is chosen to be an algorithm that returns sparse models, MVS performs view selection. This is the case as proposed by Van Loon et al. (2020), where the meta-learner was chosen to be a nonnegative logistic lasso.

A particular challenge of the aforementioned joint classification and view selection problem is its inherent trade-off between accuracy and sparsity. For example, the most accurate model may not perform the best in terms of view selection. In fact, the prediction-optimal amount of regularization causes the lasso to select superfluous features even when the sample size goes to infinity (Meinshause and Bühlmann 2006; Benner et al. 2010). This leads to a consideration of how much predictive accuracy a researcher is prepared to sacrifice for increased sparsity.

Another relevant factor is interpretability of the set of selected views. Although sparser models are typically considered more interpretable, a researcher may be interested in interpreting not only the model and its coefficients, but also the set of selected views. For example, one may wish to make decisions on which views to measure in

the future based on the set of views selected using the current data. For this purpose, one would ideally like to use an algorithm that provides sparsity, but also *algorithmic stability* in the sense that given two very similar data sets, the set of selected views should vary little. However, sparse algorithms are generally not stable, and vice versa (Xu et al. 2012).

An example of the trade-off between sparsity and interpretability of the set of selected views occurs when different views, or combinations of views, contain the same information. If the primary concern is sparsity, a researcher may be satisfied with just one of these combinations being selected, preferably the smallest set which contains the relevant information. But if there is also a desire to interpret the relationships between the views and the outcome, it may be more desirable to identify all of these combinations, even if this includes some redundant information. If one wants to go even further and perform formal statistical inference on the set of selected views, one may additionally be interested in theoretically controlling, say, the *family-wise error rate* (FWER) or *false discovery rate* (FDR) of the set of selected views. However, strict control of such an error rate could end up harming the predictive performance of the model, thus leading to a trade-off between the interpretability of the set of selected views and classification accuracy. In order to evaluate the relative merits of different feature selection algorithms in non-asymptotic settings, empirical comparisons are typically performed. Recent work on this topic includes Wah et al. (2018) comparing different filter and wrapper methods for feature selection using both simulations and real data examples, Bommert et al (2020) comparing 22 different filter methods on various benchmark data sets, as well as Hastie et al. (2020) comparing best subset selection, forward selection and two variants of the lasso using extensive simulations. However, results of previous studies do not directly translate to the multi-view setting, and it is thus important to perform empirical comparisons tailored specifically to this setting, so that recommendations can be formulated with regard to which meta-learner is suitable for which problem.

In MVS, different meta-learners may behave differently with respect to the trade-offs between accuracy, sparsity, and interpretability. For example, in a general feature selection setting with correlated features, the lasso is known to select only a small subset, while the so-called *elastic net* (Zou and Hastie 2005) is more likely to select all of them but with smaller coefficients. Which kind of behavior is desirable heavily depends on the research question at hand. In this article we investigate how the choice of meta-learner affects the view selection and classification performance of MVS. We consider seven different view-selecting meta-learners, and evaluate their performance using simulations and two real gene expression data sets.

2 Multi-view stacking

Multi-view stacking (Van Loon et al. 2020; Li et al. 2011; Garcia-Ceja et al. 2018) is an algorithm for learning from multi-view data based on the stacking (Wolpert 1992) procedure for combining the predictions of different models. Although in general MVS can be applied with any number of base-learners, we will assume the use of a single base-learner throughout this article. Consider a multi-view data set for binary

classification, consisting of views $X^{(v)} = (x_{ij}) \in \mathbb{R}^{n \times m_v}$, $v = 1, \ldots, V$, with $x_i^{(v)}$ the ith row of $X^{(v)}$, and outcome vector $y = (y_1, \ldots, y_n)^T \in \{0, 1\}^n$. Then the MVS algorithm can be described as follows:

1. Train the base-learner separately on the pairs $(X^{(v)}, y)$, $v = 1, \ldots, V$, to obtain view-specific classifiers $\hat{f}_1, \ldots, \hat{f}_V$.
2. Apply K-fold cross-validation to obtain a vector of predictions $z^{(v)} \in [0, 1]^n$ for each of the $\hat{f}_v$, $v = 1, \ldots, V$.
3. Collect the $z^{(v)}$, $v = 1, \ldots, V$, column-wise into the $n \times V$ matrix Z.
4. Train the meta-learner on the pair (Z, y) to obtain a meta-classifier $\hat{f}_{\text{meta}}$.
5. Define the final stacked classifier as $\hat{f}_{\text{meta}}(\hat{f}_1(X^{(1)}), \ldots, \hat{f}_V(X^{(V)}))$.

MVS was originally developed as a procedure for improving classification performance in multi-view learning (Li et al. 2011; Garcia-Ceja et al. 2018). However, the method can also be used for view selection, by choosing a meta-learner that returns sparse models (Van Loon et al. 2020). The special case of MVS where both the base-learner and meta-learner are chosen to be logistic regression with some penalty on the coefficient vector is known as StaPLR (Van Loon et al. 2020).

3 Choosing the meta-learner

MVS is a very flexible method, since one can choose any suitable learning algorithm for the base- and meta-learner. Van Loon et al. (2020) chose the base-learner to be logistic ridge regression and the meta-learner to be the nonnegative logistic lasso, in order to obtain a model most similar to the group lasso. In this article we will further build upon this setting, by using the same base-learner but considering different meta-learners.

In MVS, the meta-learner takes as input the matrix of cross-validated predictions Z. To perform view selection, the meta-learner should be chosen such that it returns (potentially) sparse models. The matrix Z has a few special characteristics which can be exploited, and which distinguishes it from standard settings. First, assuming that the $\hat{f}_v$, $v = 1, \ldots, V$ are probabilistic classifiers (such as logistic regression models), the features in Z are all in the same range $[0, 1]$. Second, the dispersion of each feature contains information about the magnitude of the class probabilities predicted by the corresponding base classifier. To preserve this information it is reasonable to omit the usual step of standardizing all features to zero mean and unit variance before applying penalized regression. Third, since the features in Z correspond to predictions of models trained using the same outcomes y, it is likely that at least some of them are highly correlated. Different penalization methods lead to different behavior in the presence of highly correlated features (Friedman et al. 2009). Finally, it is sensible to constrain the parameters of the meta-learner to be nonnegative (Breiman 1996; Ting and Witten 1999; Van Loon et al. 2020). There are several arguments for inducing such constraints. One intuitive argument is that a negative coefficient leads to problems with interpretation, since this would suggest that if the corresponding base classifier predicts a *higher* probability of belonging to a certain class, then the meta-learner would translate this to a *lower* probability of belonging to that same class.

Additionally, from a view selection perspective, nonnegativity constraints are crucial in preventing unimportant views from entering the model (Van Loon et al. 2020).

In this article we investigate how the choice of meta-learner affects the view selection and classification performance of MVS. We compare the following meta-learners: (1) the interpolating predictor of Breiman (1996), (2) nonnegative ridge regression (Hoerl and Kennard 1970; Van Le Cessie and Houwelingen 1992), (3) the nonnegative elastic net (Zou and Hastie 2005), (4) the nonnegative lasso (Tibshirani 1996), (5) the nonnegative adaptive lasso (Zou 2006), (6) stability selection with the nonnegative lasso (Hofner et al. 2015), and (7) nonnegative forward selection. All of these meta-learners provide models with nonnegative coefficients. In addition, they can all set some coefficients to zero, thus potentially obtaining sparse models and performing view selection. Although not an exhaustive comparison of all possible meta-learners, six of these are popular feature selection methods in their own right, and would most likely end up high on many researchers' list of candidate meta-learners. A likely exception to this is nonnegative ridge regression, since ridge regression without nonnegativity constraints would not set any coefficients to zero. However, this method is included because it provides an indication of the view selection effect of just the addition of nonnegativity constraints on the meta-learner. Each of the seven candidate meta-learners is described in more detail below.

3.1 The interpolating predictor

In the meta-learning problem, we have the binary outcome $\boldsymbol{y}$, and a matrix of cross-validated predictions $\boldsymbol{Z} = (z_i^{(v)}) \in [0, 1]^{n \times V}$. Consider a multi-view stacking model where the final prediction is a simple linear combination of the base classifiers:

$$\hat{y}_i = \sum_{v=1}^{V} \beta_v \hat{f}_v(\boldsymbol{x}_i^{(v)}). \tag{1}$$

We can obtain a so-called *interpolating predictor* (Breiman 1996) by computing the parameter estimates as

$$\hat{\beta}_1, \ldots, \hat{\beta}_V = \underset{\beta_1, \ldots, \beta_V}{\arg\min} \ \sum_{i=1}^{n} \left(y_i - \sum_{v=1}^{V} \beta_v z_i^{(v)} \right)^2, \tag{2}$$

subject to the constraints $\beta_v \geq 0$, $v = 1, \ldots, V$ and $\sum_v \beta_v = 1$. The resulting prediction function *interpolates* in the sense that the final prediction $\hat{y}_i$ can never go outside the range of the predictions of the base classifiers $[\min_v \hat{f}_v(\boldsymbol{x}_i^{(v)}), \ \max_v \hat{f}_v(\boldsymbol{x}_i^{(v)})]$ (Breiman 1996). Although originally proposed in the context of linear regression, it can also be used in binary classification, since if we have probabilistic base-classifiers making predictions in [0, 1], then the final prediction will also be in [0, 1]. Additionally the model is easy to interpret, since the final prediction is a weighted mean of the base classifiers' predictions. Note that replacing the sum-to-one constraint with the constraint $\sum_v \beta_v \leq t$, with $t \geq 0$ a tuning parameter, leads to the nonnegative

lasso (Wu et al. 2014). The interpolating predictor can thus be thought of as a (linear) nonnegative lasso with a fixed amount of regularization.

3.2 The elastic net, ridge regression, and the lasso

Instead of a simple linear function to combine the base classifiers, we can also use the logistic function:

$$\hat{y}_i = \frac{\exp\left(\beta_0 + \sum_{v=1}^{V} \beta_v \hat{f}_v(\boldsymbol{x}_i^{(v)})\right)}{1 + \exp\left(\beta_0 + \sum_{v=1}^{V} \beta_v \hat{f}_v(\boldsymbol{x}_i^{(v)})\right)}, \tag{3}$$

with $\beta_0 \in \mathbb{R}$ an intercept. The logistic function restricts predictions to $[0, 1]$ without the need for constraints on the parameters. Parameter estimates in logistic regression are usually obtained through maximizing the log-likelihood or, equivalently, minimizing the negative log-likelihood. Denote by $\boldsymbol{\beta} = (\beta_1, ..., \beta_V)^T$ the vector of regression coefficients, and by $z_i = (z_i^{(1)}, z_i^{(2)}, \ldots, z_i^{(V)})$ the ith row of $\boldsymbol{Z}$. Then the negative log-likelihood corresponding to the logistic regression model is given by

$$\mathcal{L}(\beta_0, \boldsymbol{\beta}) = -\left[\frac{1}{n} \sum_{i=1}^{n} y_i(\beta_0 + z_i\boldsymbol{\beta}) - \log(1 + \exp(\beta_0 + z_i\boldsymbol{\beta}))\right]. \tag{4}$$

Although constraints on the parameters are not required to keep the predictions in the range $[0, 1]$, we can still employ regularization to perform view selection and obtain more stable models. The elastic net (Zou and Hastie 2005) is a popular regularization method which employs both L_1 and L_2 penalties. To obtain parameter estimates using the nonnegative variant of the elastic net, one optimizes

$$\hat{\beta}_0, \hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta} \geq 0, \beta_0}{\arg\min} \quad \mathcal{L}(\beta_0, \boldsymbol{\beta}) + \lambda\left[(1 - \alpha)\|\boldsymbol{\beta}\|_2^2/2 + \alpha\|\boldsymbol{\beta}\|_1\right], \tag{5}$$

with tuning parameters $\lambda \geq 0$ and $\alpha \in [0, 1]$. Ridge regression ($\alpha = 0$) and the lasso ($\alpha = 1$) are both special cases of the elastic net. Choosing any other value of α leads to a mixture of L_1 and L_2 penalties. Since the columns of $\boldsymbol{Z}$ correspond to predictions of models trained using the same outcomes $\boldsymbol{y}$, it is very likely that at least some of them are highly correlated. When faced with a set of highly correlated views, the lasso may select one of them and discard the others. The addition of an L_2 penalty causes the elastic net to favor solutions where the entire set of correlated views is included with moderate coefficients. Which type of behavior is desirable will depend on the research question at hand. In this paper we will apply ridge regression, the lasso, and the elastic net with $\alpha = 0.5$, all with nonnegativity constraints. Note that, although ridge regression usually does not perform any view selection, the addition of nonnegativity constraints forces some coefficients to be zero, causing view selection even when $\alpha = 0$.

3.3 The adaptive lasso

The *adaptive lasso* (Zou 2006) is a weighted version of the lasso with data-dependent weights. Consider again the negative log-likelihood in (4). Let us define for each $\beta_v \in \boldsymbol{\beta}$ a corresponding weight $\hat{w}_v = 1/|\hat{\beta}_v|^\gamma$, with $\hat{\beta}_v$ an initial estimate of β_v, and $\gamma > 0$ a tuning parameter. Then the adaptive lasso estimates are given by

$$\hat{\beta}_0^*, \hat{\boldsymbol{\beta}}^* = \underset{\beta_0, \boldsymbol{\beta}}{\arg\min} \quad \mathcal{L}(\beta_0, \boldsymbol{\beta}) + \lambda \sum_{v=1}^{V} \hat{w}_v |\beta_v|. \tag{6}$$

In the context of linear models, Zou (2006) suggested to use OLS or ridge regression to obtain the initial $\hat{\beta}_v$'s. We use (logistic) ridge regression with nonnegativity constraints to obtain the initial estimates. Due to the nonnegativity constraints, views which would have otherwise obtained a negative coefficient thus obtain an infinitely large penalty (i.e. are removed from the model) in the weighted lasso.

3.4 Stability selection

Stability selection is an ensemble learning framework originally proposed for use with the lasso (Meinshausen and Bühlmann 2010), although it can be used with a wide variety of feature selection methods (Hofner et al. 2015). The basic idea of stability selection is to apply a feature selection method on subsamples of the data, and then incorporate in the final model only those features which were chosen by the feature selection method on a sufficiently large proportion of the subsamples. In this study, we specifically use complementary pairs stability selection (Shah and Samworth 2013) with the nonnegative lasso, as implemented in the R package `stabs` (Hofner and Hothorn 2017). This procedure can be described as follows (Hofner et al. 2015; Hofner and Hothorn 2017; Shah and Samworth 2013):

1. Let $\{(D_{2b-1}, D_{2b}) : b = 1, \ldots, B\}$ be randomly chosen independent pairs of subsets of $\{1, \ldots, n\}$ of size $\lfloor n/2 \rfloor$ such that $D_{2b-1} \cap D_{2b} = \emptyset$.
2. For each $b = 1, \ldots, 2B$, fit a nonnegative lasso path using only the observations in D_b, i.e. the pair $(\mathbf{Z}_{i \in D_b}, \mathbf{y}_{i \in D_b})$. Start with a high value of the penalty parameter, then decrease the value until q views are selected, with q a pre-defined positive integer. Let $\hat{S}(D_b)$ be the index set of selected views.
3. Compute for each view $v = 1, \ldots, V$, the relative selection frequency:

$$\hat{\pi}_v := \frac{1}{2B} \sum_{b=1}^{2B} 1_{\{v \in \hat{S}(D_b)\}} \tag{7}$$

4. Select the set of views $\hat{S}_{\text{stable}} := \{v : \hat{\pi}_v \geq \pi_{\text{thr}}\}$, with $\pi_{\text{thr}} \in (0.5, 1]$ a pre-defined threshold.

Typically B is set to 50, but the choice of q and π_{thr} is somewhat more involved. In particular, one can obtain a bound on the expected number of falsely selected variables,

the so-called *per-family error rate* (PFER), for a given value of q and π_{thr} (Meinshausen and Bühlmann 2010; Shah and Samworth 2013; Hofner et al. 2015). For our particular choices of these parameters, see Sect. 4.2. The meta-classifier is obtained by fitting a nonnegative logistic regression model using only the cross-validated predictions corresponding to the set of stable views. Note that the computational cost of stability selection is several times larger than that of the regular lasso with cross-validation: in the linear case, stability selection has been estimated to be approximately 3 times more expensive than ten-fold cross-validation when $n < V$, and approximately 5.5 times more expensive when $n > V$ (Meinshausen and Bühlmann 2010).

3.5 Nonnegative forward selection

Forward selection is a simple, greedy feature selection algorithm (Guyon and Elisseeff 2003). It is a so-called *wrapper* method, which means it can be used in combination with any learner (Guyon and Elisseeff 2003). The basic strategy is to start with a model with no features, and then add the single feature to the model which is "best" according to some criterion. One then proceeds to sequentially add the next "best" feature at every step until some stopping criterion is met. Here we consider forward selection based on the Akaike Information Criterion (AIC). In order to impose nonnegativity of the coefficients, we will use a slightly modified procedure which we will call *nonnegative forward selection* (NNFS). This procedure can be described as follows:

1. Start with a model containing only an intercept.
2. Calculate for each candidate view the reduction in AIC if this view is added to the model.
3. Consider the view corresponding to the largest reduction in AIC. If the coefficients (excluding the intercept) of the resulting model are all nonnegative, update the model and repeat starting at step 2.
4. If some of the coefficients (excluding the intercept) of the resulting model are negative, remove the view (from step 3) from the list of candidates and repeat starting at step 3.
5. Stop when none of the remaining candidate views show a reduction in AIC, or there are no more views remaining.

4 Simulations

4.1 Design

In order to compare the different meta-learners in terms of classification and view selection performance, we perform a series of simulations. We generate multi-view data with $V = 30$ or $V = 300$ disjoint views, where each view $X^{(v)}$, $v = 1, \ldots,$ V, is an $n \times m_v$ matrix of normally distributed features scaled to zero mean and unit variance. For each number of views, we consider two different view sizes. In any single simulated data set, all views are always set to be the same size. If $V = 300$, then either $m_v = 25$ or $m_v = 250$. If $V = 30$, then either $m_v = 250$ or $m_v = 2500$. We use

two different sample sizes: $n = 200$ or $n = 2000$. In addition, we apply different correlation structures defined by the population correlation between features from the same view ρ_w, and the population correlation between features from different views ρ_b. We use six different parameterizations: ($\rho_w = 0.1$, $\rho_b = 0$), ($\rho_w = 0.5$, $\rho_b = 0$), ($\rho_w = 0.9$, $\rho_b = 0$), ($\rho_w = 0.5$, $\rho_b = 0.4$), ($\rho_w = 0.9$, $\rho_b = 0.4$), and ($\rho_w = 0.9$, $\rho_b = 0.8$). This leads to a total of $2 \times 2 \times 2 \times 6 = 48$ different experimental conditions.

For each experimental condition, we simulate 100 multi-view data training sets. For each such data set, we randomly select 10 views. In 5 of those views, we determine all of the features to have a relationship with the outcome. In the other 5 views, we randomly determine 50% of the features to have a relationship with the outcome. The relationship between features and response is determined by a logistic regression model, where each feature related to the outcome is given a regression weight. In the setting with 30 views, we use the same regression weight as a similar simulation study in Van Loon et al. (2020). This regression weight is either 0.04 or −0.04, each with probability 0.5. In the setting with 300 views, the number of features per view is reduced by a factor 10. To compensate for the reduction in the number of features, the aforementioned regression weights are multiplied by $\sqrt{10}$ in this setting.

We apply multi-view stacking to each simulated training set, using logistic ridge regression as the base-learner. Once we obtain the matrix of cross-validated predictions $\mathbf{Z}$, we apply the seven different meta-learners. To assess classification performance, we generate a matching test set of 1000 observations for each training set, and calculate the classification accuracy of the stacked classifiers on this test set. To assess view selection performance we calculate three different measures: (1) the *true positive rate* (TPR), i.e. the average proportion of views truly related to the outcome that were correctly selected by the meta-learner; (2) the *false positive rate* (FPR), i.e. the average proportion of views not related to the outcome that were incorrectly selected by the meta-learner; and (3) the *false discovery rate* (FDR), i.e. the average proportion of the selected views that are not related to the outcome.

Although we can average over the 100 replications within each condition, with 7 different meta-learners and 48 experimental conditions, this would still lead to 336 averages for each of the outcome measures. In our reporting of the results we will therefore focus only on the most important interactions between the meta-learners and the different experimental conditions. To determine which interactions are most important in a data-driven way, we perform a mixed analysis of variance (ANOVA), and calculate a standardized measure of effect size (partial η^2) for each interaction. We do this separately for each of the four outcome measures. A common rule of thumb is that $\eta^2 \geq 0.06$ corresponds to a moderate effect size, and $\eta^2 \geq 0.14$ corresponds to a large effect size (Cohen 1988; Rovai et al. 2013). We discuss only the interactions that have at least a moderate effect size $\eta^2 \geq 0.06$. Note that we use ANOVA only to calculate a measure of effect size, and do not report test statistics or *p*-values. This is because (1) these tests would rely too heavily on the assumptions of ANOVA, and (2) in a simulation study any arbitrarily small difference can be artificially made "significant" by increasing the number of replications.

4.2 Software

All simulations are performed in R (version 3.4.0) (R Core Team 2017) on a high-performance computing cluster running Ubuntu (version 14.04.6 LTS) with Open Grid Scheduler/Grid Engine (version 2011.11p1). All pseudo-random number generation is performed using the Mersenne Twister (Matsumoto and Nishimura 1998), R's default algorithm. The training of the base-learners, and the generation of cross-validated predictions is performed using an early development version of package `mvs` (Van Loon 2022). Optimization of the nonnegative ridge, elastic net, and lasso is performed using coordinate descent through the package `glmnet` 1.9–8 (Friedman et al. 2010). Nonnegativity constraints are implemented by setting a coefficient to zero if it becomes negative during the update cycle (Friedman et al. 2010; Hastie et al. 2015). To select the tuning parameter λ, a sequence of 100 candidate values of λ is adaptively chosen by the software (Friedman et al. 2010). In particular, the 100 candidate values are decreasing on a log scale from λ_{max} to λ_{min}, where λ_{max} is the smallest value such that the entire coefficient vector is zero, and $\lambda_{min} = \epsilon \lambda_{max}$, with $\epsilon = 10^{-4}$. The value of λ is then selected by minimizing the 10-fold cross-validation error. For the nonnegative elastic net we set $\alpha = 0.5$. We choose to fix the value of α at 0.5 rather than tune it so that we can compare the performance of the equal mixture of the two penalties with using only an L_1 or only an L_2 penalty. The code used for fitting the nonnegative adaptive lasso is also based on `glmnet`, were we use 10-fold cross-validation to select both λ and γ. For γ we consider the possible values {0.5, 1, 2}, for each of which we fit a path of 100 candidate values for λ. We then choose the combination of γ and λ which has the lowest cross-validation error. For stability selection we use the package `stabs` 0.6–3 (Hofner and Hothorn 2017; Hofner et al. 2015). We adopt the recommendations of Hofner et al. (2015) for choosing the parameters, by specifying q and a desired bound $PFER_{max}$, and then calculating the associated threshold π_{thr}. The parameter q should be chosen large enough that in theory all views corresponding to signal can be chosen (Hofner et al. 2015). We therefore choose $q = 10$, since we have 10 views corresponding to signal. Note that this means that the procedure has additional information about the true model unavailable to the other meta-learners. We choose a desired bound of $PFER_{max} = 1.5$, which is equivalent to controlling the *per-comparison error rate* (PCER) at $1.5/30 = 0.05$ when $V = 30$, or $1.5/300 = 0.005$ when $V = 300$. Under the *unimodality assumption* of Shah and Samworth (2013), this leads to $\pi_{thr} = 0.9$ when $V = 30$, and $\pi_{thr} = 0.57$ when $V = 300$. The code used to perform nonnegative forward selection is based on `stepAIC` from `MASS` 7.3–47 (Venables and Ripley 2002). The optimization required for fitting the interpolating predictor is performed using the package `lsei` 1.2–0 (Wang et al. 2017). After optimization, coefficients smaller than $\max(10^{-2}/V, 10^{-8})$ are set to zero.

4.3 Results

4.3.1 Effect sizes

The values of partial η^2 obtained from the mixed ANOVAs for each of the four outcome measures are given in Table 1. Note that we are primarily interested in the extent to which differences between the meta-learners are moderated by the experimental factors of sample size, view size, number of views, and correlation structure. In Table 1 we therefore show only the interaction terms including the meta-learner factor.

Large or moderate effect sizes can be observed across all four outcome measures for the main effect of the meta-learner, as well as for the interactions with sample size and correlation structure. When accuracy or TPR is used as the outcome, the three-way interaction between meta-learner, sample size and correlation structure also shows a moderate effect size. In Sects. 4.3.2 through 4.3.5, we therefore show the results split by sample size and correlation structure, and use fixed levels of the other experimental factors. In particular, we use $V = 300$ and $m_v = 25$, since this structure is the most similar to our real data examples (Sect. 5). The results for other combinations of V and m_v can be found in the Appendix.

Note that for the false positive rate only, a moderate effect size can also be observed for the interaction between the choice of meta-learner and the number of views. For the false discovery rate, a borderline moderate effect size can be observed for the three-way interaction between the choice of meta-learner, V, and n. However, these interactions appear to be dominated by the interpolating predictor. In particular, when $V < n$ the interpolating predictor generally produces sparse models with around 3 nonzero coefficients on average. However, when $V > n$ (i.e. when the view selection problem is high-dimensional) the interpolating predictor produces dense models with around 90 nonzero coefficients, leading to a large increase in FPR and FDR.

4.3.2 Test accuracy

Classification accuracy on the test set for each of the meta-learners can be observed in Fig. 1. Based on these results, the meta-learners can be divided into two groups: On the one hand, the nonnegative lasso, adaptive lasso, elastic net and NNFS generally have very similar classification performance; on the other hand, nonnegative ridge regression, the interpolating predictor and stability selection all perform noticeably worse than the other meta-learners in a subset of the experimental conditions. In particular, ridge regression and the interpolating predictor perform worse when the features in different views are uncorrelated ($\rho_b = 0$), or when the correlation between the features in different views is much lower than the correlation between features in the same view ($\rho_b = 0.4$, $\rho_w = 0.9$). Stability selection performs worse when $n = 200$ and the correlation between features from different views is of a similar magnitude as the correlation between features from the same view (i.e. $\rho_b = 0.4$, $\rho_w = 0.5$ or $\rho_b = 0.8$, $\rho_w = 0.9$). These results appear even more pronounced in the case when $V = 30$, see Figs. 10 and 14 in the Appendix.

Table 1 Standardized measures of effect size (partial η^2) for the interactions between the choice of meta-learner and the other experimental factors, for each of the four outcome measures of true positive rate, false positive rate, false discovery rate and classification accuracy

	η^2 (accuracy)	η^2 (TPR)	η^2 (FPR)	η^2 (FDR)
Meta-learner	**0.208**	**0.552**	**0.483**	**0.466**
Meta-learner*V	0.038	0.011	*0.096*	0.030
Meta-learner*m_v	0.019	0.006	0.011	0.010
Meta-learner*n	*0.130*	**0.421**	**0.164**	**0.188**
Meta-learner*cor	**0.236**	**0.386**	**0.243**	*0.064*
Meta-learner*V*n	0.008	0.028	0.045	*0.060*
Meta-learner*v*cor	0.028	0.028	0.040	0.030
Meta-learner*m_v*n	0.001	0.003	0.003	0.006
Meta-learner*mv*cor	0.035	0.013	0.009	0.011
Meta-learner*n*cor	*0.080*	*0.094*	0.022	0.045
Meta-learner*V*n*cor	0.024	0.010	0.007	0.020
Meta-learner*m_v*n*cor	0.004	0.008	0.004	0.013

Large effect sizes ($\eta^2 \geq 0.14$) are printed in bold. Moderate effect sizes ($0.06 \leq \eta^2 < 0.14$) are printed in italics. V denotes the number of views, m_v the number of features per view, n the sample size, and cor the correlation structure

4.3.3 View selection: true positive rate

The true positive rate in view selection for each of the meta-learners can be observed in Fig. 2. Ignoring the interpolating predictor for now, nonnegative ridge regression has the highest TPR, which is unsurprising seeing as it performs feature selection only through its nonnegativity constraints. Nonnegative ridge regression is followed by the elastic net and then lasso. The lasso is followed by the adaptive lasso, NNFS and stability selection, although the order among these three methods changes somewhat for the different conditions. The interpolating predictor shows behavior that is completely different from the other meta-learners. Whereas for the other meta-learners the TPR increases as sample size increases, the TPR of the interpolating predictor actually decreases in some cases. Although it appears to have the highest TPR in some conditions, it can be observed in the next section that it also has the highest FPR in these conditions.

4.3.4 View selection: false positive rate

The false positive rate in view selection for each of the meta-learners can be observed in Fig. 3. Again ignoring the interpolating predictor for now, the ranking of the different meta-learners is similar to their ranking by TPR. Nonnegative ridge regression has the highest FPR, followed by the elastic net, lasso, adaptive lasso and NNFS (which have almost identical performance), and finally stability selection. It is clear from Fig. 3 that using a meta-learner specifically aimed at view selection can decrease
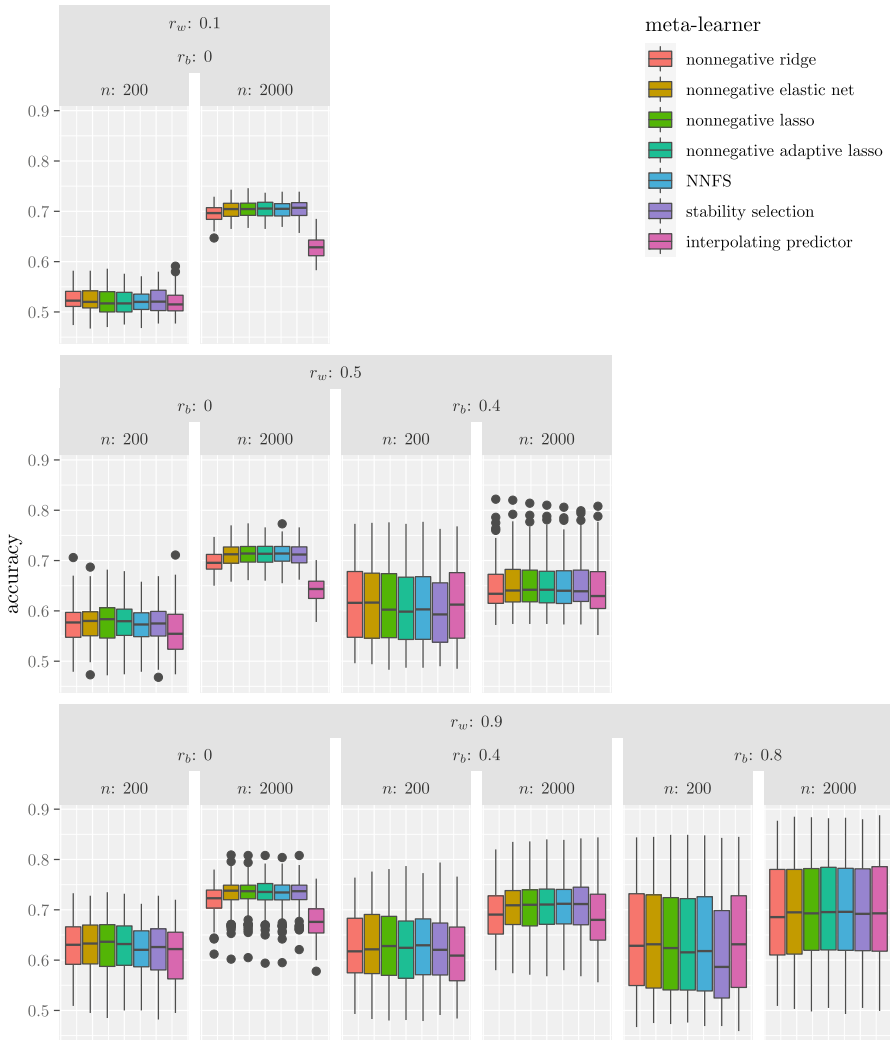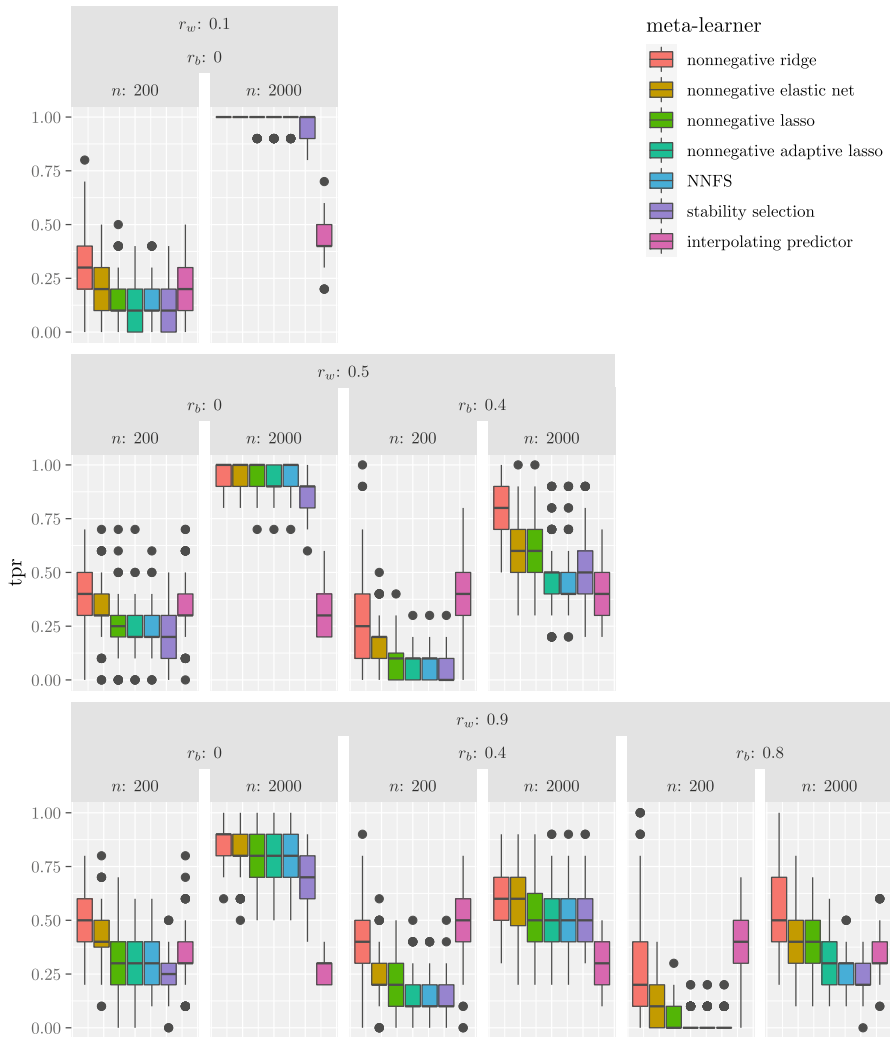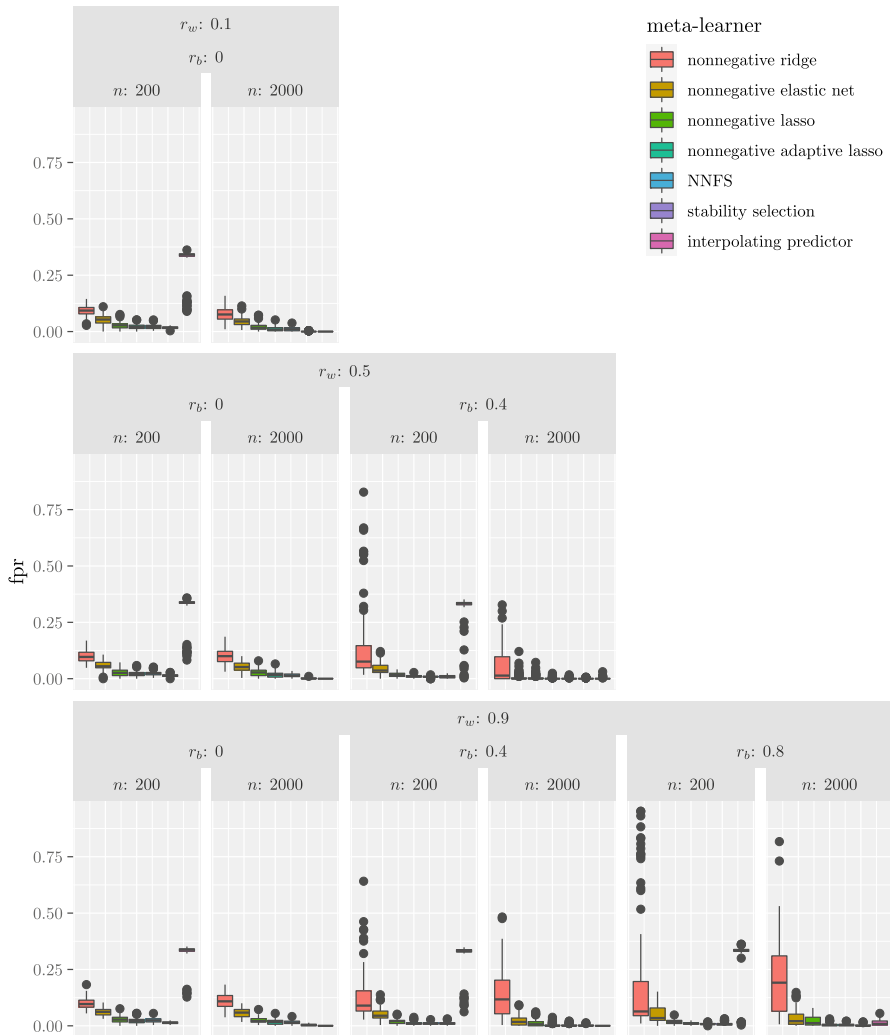
Fig. 1 Boxplots of test accuracy for the different meta-learners, with 300 views and 25 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications

the FPR substantially compared to using only nonnegativity constraints. Again the interpolating predictor shows different behavior. In particular, it has the highest FPR whenever $n = 200$. This appears to be caused by the interpolating predictor producing very dense models whenever the view selection problem is high-dimensional.
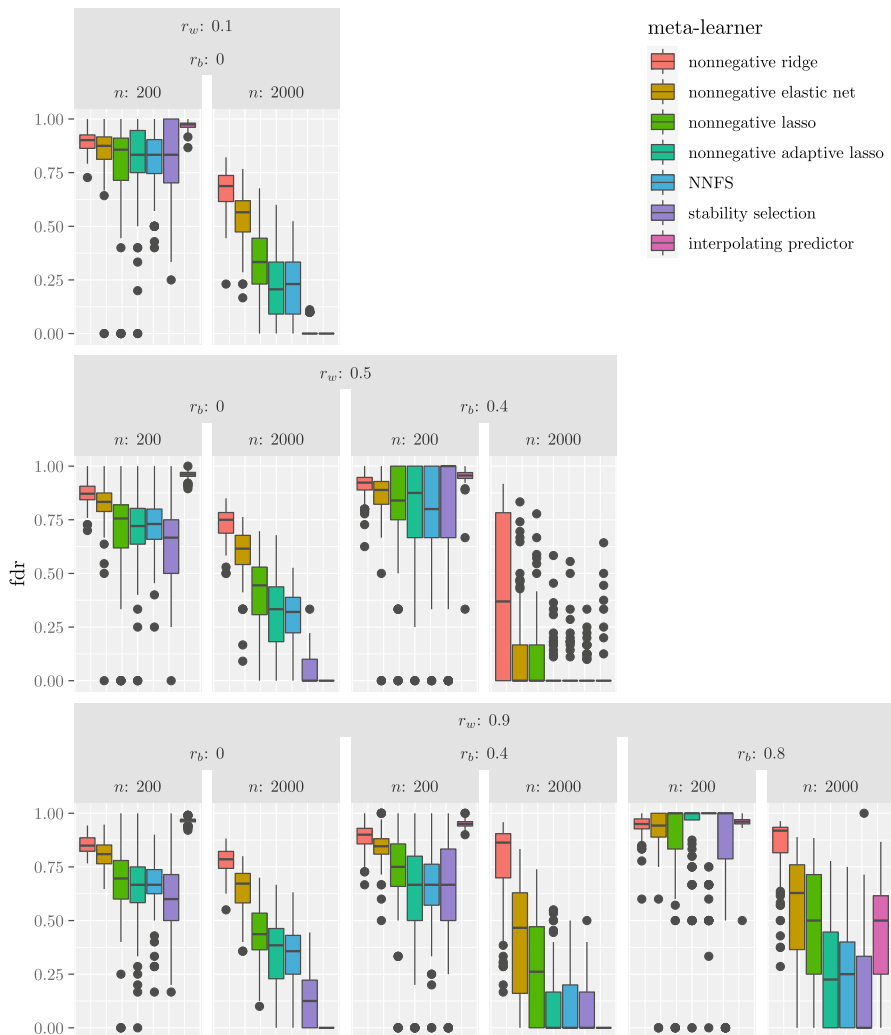
Fig. 2 Boxplots of the true positive rate (TPR) for the different meta-learners, with 300 views and 25 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications

4.3.5 View selection: false discovery rate

The false discovery rate in view selection for each of the meta-learners can be observed in Fig. 4. Note that the FDR is particularly sensitive to variability since its denominator is the number of selected views, which itself is a variable quantity. In particular, when the number of selected views is small, the addition or removal of a single view may cause large increases or decreases in FDR. This happens especially whenever $\rho_b > 0$,

Fig. 3 Boxplots of the false positive rate (FPR) for the different meta-learners, with 300 views and 25 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications

as can be observed in Fig. 4. The ranking of the different meta-learners is similar to their ranking by TPR and FPR. When $n = 200$, the interpolating predictor has the highest FDR due to its tendency to select very dense models when $n < V$. When $n = 2000$, the interpolating predictor often has a very low FDR, but in these settings it also has considerably lower TPR and test accuracy than the other meta-learners. Of the other meta-learners nonnegative ridge regression has the highest FDR, followed by the elastic net, lasso, adaptive lasso and NNFS, and stability selection.

Fig. 4 Boxplots of the false discovery rate (FDR) for the different meta-learners, with 300 views and 25 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications

4.3.6 Summary of simulation results

In summary, the nonnegative lasso, adaptive lasso, elastic net and NNFS generally showed comparable classification performance in our simulations, while nonnegative ridge regression, stability selection and the interpolating predictor performed notice-ably worse in a subset of the experimental conditions. Among the meta-learners that performed well in terms of accuracy, model sparsity was generally associated with a

lower false positive rate in terms of view selection, but also with a lower true positive rate. Nevertheless, there are situations when the sparser meta-learners obtained both a low FPR and a high TPR, particularly when the features from different views were uncorrelated. However, even when the FPR was very low, the FDR was often high, especially in the setting with a sample size of 200.

5 Gene expression application

5.1 Design

We apply MVS with the seven different meta-learners to two gene expression data sets, namely the colitis data of Burczynski et al. (2006), and the breast cancer data of Ma et al. (2004). These data sets were previously used to compare the group lasso with the sparse group lasso (Simon et al. 2013), and to compare the group lasso with StaPLR (Van Loon et al. 2020). The colitis data (Burczynski et al. 2006) consists of 85 colitis cases and 42 healthy controls for which gene expression data was collected using 22,283 probe sets. As in (Van Loon et al. 2020), we matched this data to the C1 cytogenetic gene sets from MSigDB 6.1 (Subramanian et al. 2005), and removed any duplicate probes, genes not included in the C1 gene sets, and gene sets which consisted of only a single gene after matching. This led to a multi-view data set consisting of 356 views (gene sets), with an average view size of 33 features (genes). A boxplot of the distribution of the view sizes is included in Appendix 1. The total number of features was 11,761. All features were $\log_2$-transformed, then standardized to zero mean and unit variance before applying the MVS procedure.

The breast cancer data (Ma et al. 2004) consists of 60 tumor samples labeled according to whether cancer did (28 cases) or did not (32 cases) recur. The data was matched to the C1 gene sets using the same procedure as in the colitis data, leading to a multi-view data set of 354 views, with an average view size of 36 features. A boxplot of the distribution of the view sizes is included in Appendix 1. The total number of features was 12,722. The features were already $\log_2$-transformed, but were further standardized to zero mean and unit variance before applying the MVS procedure. To assess classification performance for each of the data sets, we perform 10-fold cross-validation. We repeat this procedure 10 times and average the results to account for random differences in the cross-validation partitions. The hold-out data in the cross-validation procedure is used only for model evaluation, not for parameter tuning. We again report classification accuracy using a standard threshold of 0.5. Because the colitis data is somewhat unbalanced in terms of class membership, one might additionally be interested in the performance of the methods across multiple possible thresholds. Due to different opinions regarding which metric is most suitable for comparing performance across multiple thresholds (Hand 2009; Flach et al. 2011; Hernández-Orallo et al. 2012), we report two popular metrics, namely the area under the receiver operating characteristic curve (AUC), and the H measure (Hand 2009). Both metrics can take values in [0, 1], but the AUC more typically takes values in [0.5, 1], with a value of 0.5 denoting a noninformative classifier. For the H measure, a noninformative classifier is associated with a value of zero.

In terms of view selection, each of the 10×10 fitted models is associated with a set of selected views. However, quantities like TPR, FPR and FDR cannot be computed since the true status of the views is unknown. We therefore report the number of selected views, since this allows assessment of model sparsity. In addition, we report a measure of the stability of the set of selected views. In particular, we use the feature selection stability measure of Nogueira et al. (2018). This stability measure, $\hat{\Phi}$, has an upper bound of 1 and a lower bound which is asymptotically zero but depends on the number of fitted models (in our case it is approximately -0.01). Higher values indicate increased stability, and $\hat{\Phi}$ attains its maximum of 1 if-and-only-if the set of selected views is the same for each fitted model (Nogueira et al. 2018). A particularly desirable property of this stability measure is that it is corrected for chance in the sense that its expected value is zero if a selection algorithm would select sets of views of a certain size randomly rather than systematically (Nogueira et al. 2018). The measure can additionally be considered a special case of the Fleiss' Kappa (Fleiss 1971) measure of inter-rater agreement, where each of the fitted models is a "rater" classifying the views into whether or not they are relevant (Nogueira et al. 2018). Rules of thumb for interpreting the strength of agreement associated with a certain value of Fleiss' Kappa have been formulated by Landis and Koch (1977):.00-.20=slight, $0.21-0.40$=fair, $0.41-0.60$=moderate, $0.61-0.80$=substantial, $0.81-1.0$=almost perfect. However, these rules of thumb are largely arbitrary (Landis and Koch 1977), and we will focus on relative comparisons.

5.2 Software

We use the same software as described in Sect. 4.2. All cross-validation loops used for parameter tuning are nested within the outer loop used for evaluating classification performance. We again use the recommendations of Hofner et al. (2015) for choosing the parameters, by specifying q and a desired bound $PFER_{\max}$, and then calculating the associated threshold π_{thr}. We again specify a desired bound of 1.5. The parameter q should be large enough so that all views corresponding to signal can be selected (Hofner et al. 2015), but in real data the number of views corresponding to signal is unknown. However, domain knowledge or results from previous experiments can be used to obtain an estimate of the number of views corresponding to signal, and thus assist in choosing the value of q. The colitis and breast cancer data sets have previously been analyzed using StaPLR with the nonnegative lasso as a meta-learner (Van Loon et al. 2020). We set the value of q to the maximum number of views selected by the StaPLR method in the colitis and breast cancer data sets as reported in Van Loon et al. (2020). This leads to $q = 15$, $\pi_{\mathrm{thr}} = 0.62$ for the colitis data, and $q = 11$, $\pi_{\mathrm{thr}} = 0.57$ for the breast cancer data. Note that this means the stability selection procedure has access to information from a previous experiment that the other meta-learners do not have access to. The AUC is calculated using the package AUC 0.3.0 (Ballings and Van 2013), and the H measure using the package hmeasure 1.0–2 (Anagnostopoulos and Hand 2019). The stability measure $\hat{\Phi}$ was calculated using a custom script, which is included in the supplementary materials.

5.3 Results

The results of applying MVS with the seven different meta-learners to the colitis data can be observed in Table 2. In terms of raw test accuracy the nonnegative lasso is the best performing meta-learner, followed by the nonnegative elastic net and the nonnegative adaptive lasso. In terms of AUC and H, the best performing meta-learners are the elastic net and ridge regression. However, the elastic net selects on average almost 4 times as many views as the lasso, and nonnegative ridge regression selects on average almost 13 times as many views, and both have lower raw test accuracy than the lasso. Since feature selection often comes at a cost in terms of stability (Xu et al. 2012), it is to be expected that view selection stability ($\hat{\Phi}$) is higher for meta-learners that select more views. The results of two meta-learners do not align with this pattern, namely those for the interpolating predictor and NNFS. The interpolating predictor is very dense but has lower stability than several sparse models, while NNFS is less sparse than stability selection and also less stable. Note that we mean sparsity in terms of the number of selected views, but this corresponds to sparsity in terms of the number of selected features (see Appendix 1, Table 4).

The results for the breast cancer data can be observed in Table 3. The interpolating predictor and the lasso are the best performing meta-learners in terms of all three classification measures, with the interpolating predictor having higher test accuracy and H, and the lasso having higher AUC. However, the interpolating predictor selects over 80 times as many views as the lasso, and is less stable. Again, the interpolating predictor and NNFS do not align with the pattern that less sparsity is associated with higher stability.

Table 2 Results of applying MVS with different meta-learners to the colitis data

Meta-learner	Accuracy	AUC	H	ANSV	$\hat{\Phi}$
Lasso	**0.963** $\pm$ 0.010	0.985 $\pm$ 0.009	0.892 $\pm$ 0.027	10.7 $\pm$ 1.9	0.539
Elastic net	0.956 $\pm$ 0.008	0.989 $\pm$ 0.006	**0.896** $\pm$ 0.017	40.0 $\pm$ 4.6	0.606
Adaptive lasso	0.955 $\pm$ 0.015	0.981 $\pm$ 0.010	0.883 $\pm$ 0.028	7.4 $\pm$ 1.6	0.488
Ridge	0.948 $\pm$ 0.008	**0.992** $\pm$ 0.003	0.894 $\pm$ 0.014	132.7 $\pm$ 18.1	**0.641**
Interpolating predictor	0.940 $\pm$ 0.006	0.988 $\pm$ 0.001	0.863 $\pm$ 0.010	242.0 $\pm$ 1.6	0.463
NNFS	0.928 $\pm$ 0.019	0.943 $\pm$ 0.026	0.788 $\pm$ 0.063	2.6 $\pm$ 0.5	0.198
Stability selection	0.923 $\pm$ 0.013	0.951 $\pm$ 0.011	0.788 $\pm$ 0.040	2.0 $\pm$ 0.9	0.375

ANSV denotes the average number of selected views. H denotes the H measure (Hand 2009). In computing the H measure we assume that the misclassification cost is the same for each class. $\hat{\Phi}$ denotes the feature selection stability measure of Nogueira et al. (2018). For accuracy, AUC and H we show the mean and standard deviation across the 10 replications. For the number of selected views we show the mean and standard deviation across the 10×10 different fitted models. The total number of views for this data set is 356, and 67% of observations belong to the majority class

The highest values of accuracy, AUC, H, and $\hat{\Phi}$ are printed in bold

Table 3 Results of applying MVS with different meta-learners to the breast cancer data

Meta-learner	Accuracy	AUC	H	ANSV	$\hat{\Phi}$
Lasso	0.660 ± 0.031	$\mathbf{0.681} \pm 0.024$	0.236 ± 0.047	3.6 ± 1.8	0.334
Elastic net	0.647 ± 0.037	0.665 ± 0.026	0.201 ± 0.033	13.1 ± 2.8	0.396
Adaptive lasso	0.652 ± 0.034	0.675 ± 0.034	0.222 ± 0.058	2.6 ± 1.2	0.313
Ridge	0.653 ± 0.031	0.672 ± 0.021	0.207 ± 0.024	52.1 ± 15.9	$\mathbf{0.476}$
Interpolating predictor	$\mathbf{0.682} \pm 0.038$	0.676 ± 0.023	$\mathbf{0.239} \pm 0.060$	298.6 ± 1.6	0.118
NNFS	0.633 ± 0.024	0.644 ± 0.045	0.162 ± 0.058	3.1 ± 1.0	0.257
Stability selection	0.540 ± 0.045	0.542 ± 0.045	0.073 ± 0.042	1.5 ± 1.1	0.108

ANSV denotes the average number of selected views. H denotes the H measure (Hand 2009). In computing the H measure we assume that the misclassification cost is the same for each class. $\hat{\Phi}$ denotes the feature selection stability measure of Nogueira et al. (2018). For accuracy, AUC and H we show the mean and standard deviation across the 10 replications. For the number of selected views we show the mean and standard deviation across the 10×10 different fitted models. The total number of views for this data set is 354, and 53% of observations belong to the majority class

The highest values of accuracy, AUC, H, and $\hat{\Phi}$ are printed in bold

6 Discussion

In this article we investigated how different view-selecting meta-learners affect the performance of multi-view stacking. In our simulations, the interpolating predictor often performed worse than the other meta-learners on at least one outcome measure. For example, when the sample size was larger than the number of views, the interpolating predictor often had the lowest TPR in view selection, as well as the lowest test accuracy, particularly when there was no correlation between the different views. When the sample size was smaller than the number of views, the interpolating predictor had an FPR in view selection that was considerably higher than that of all other meta-learners. In terms of accuracy it performed very well in the breast cancer data, but less so in the colitis data. However, in both cases it produced very dense models, which additionally had low view selection stability. The fact that its behavior varied considerably across our experimental conditions, combined with its tendency to select very dense models when the meta-learning problem is high-dimensional, suggests that the interpolating predictor should not be used when view selection is among the goals of the study under consideration. However, it may have some use when its interpretation as a weighted mean of the view-specific models is of particular importance.

Excluding the interpolating predictor, nonnegative ridge regression produced the least sparse models. This is not surprising considering it performs view selection only through its nonnegativity constraints. Its high FPR in view selection appeared to negatively influence its test accuracy, as there was generally at least one sparser model with better accuracy in both our simulations and real data examples. Although nonnegative ridge regression shows that the nonnegativy constrains alone already cause many coefficients to be set to zero, if one assumes the true underlying model to

be sparse, one should probably choose one of the meta-learners specifically aimed at view selection.

The nonnegative elastic net, with its additional L_1 penalty compared with ridge regression, is one such method. In our simulations it produced sparser models than nonnegative ridge regression, usually with better or comparable accuracy. These sparser models were associated with a reduction in FPR and FDR, but in some setting also with a reduction in TPR, particularly when there are correlations between the views. However, we fixed the mixing parameter α at 0.5 to observe a specific setting in between ridge regression and the lasso. In practice, one can tune α, for example through cross-validation. This may allow the elastic net to better adapt to different correlation structures. In the colitis data, the elastic net performed better than nonnegative ridge regression in terms of test accuracy, whereas in the breast cancer data it performed slightly worse. However, in both cases it produced much sparser models, demonstrating its use in view selection.

The nonnegative lasso, utilizing only an L_1 penalty, produced even sparser models than the elastic net. Interestingly, in our simulations this increased sparsity did not appear to have a substantial negative effect on accuracy, although some minor reductions were observed in some low sample size cases. In the colitis data it performed best in terms of raw test accuracy, second in terms of H measure, and third in terms of AUC. In the breast cancer data it performed second-best in accuracy and H measure, and best in terms of AUC. Notably, it selected on average only 3.7 views out of 354, whereas the only better performing meta-learner in terms of accuracy, the interpolating predictor, selected on average 298.6 views. Our results indicate that using the nonnegative lasso as a meta-learner can substantially reduce the number of views while still providing accurate prediction models.

Our implementation of the nonnegative adaptive lasso produced slightly sparser models than the regular nonnegative lasso. This did not appear to substantially reduce classification accuracy in our simulations, although there were some minor reductions in some low sample size cases. In both gene expression data sets the adaptive lasso performed worse on average than the lasso in all three classification metrics, but the observed differences were small. The main difference between these two meta-learners appears to be that the regular lasso slightly favors classification performance, whereas the adaptive lasso slightly favors sparsity. Note that the adaptive lasso is a flexible method, and one can change the way in which its weights are initialized, which will likely affect performance. Additionally, one could consider a larger set of possible values for the tuning parameter γ. However, this flexibility also means that the method is less straightforward to use than the regular lasso.

The NNFS algorithm performed surprisingly well in our simulations given its simple and greedy nature, showing performance very similar to that of the adaptive lasso. However, in both gene expression data sets it was among the two worst performing methods, both in terms of accuracy and view selection stability. If one additionally considers that NNFS does not scale well with larger problems there is generally no reason to choose this algorithm over the nonnegative (adaptive) lasso.

Excluding the interpolating predictor, stability selection produced the sparsest models in our simulations. However, this led to a reduction in accuracy whenever the correlation within features from the same view was of a similar magnitude as the

correlations between features from different views. In both gene expression data sets stability selection also produced the sparsest models, but it also had the worst classification accuracy of all meta-learners. In applying stability selection, one has to specify several parameters. We calculated the values of these parameters in part by specifying a desired bound on the PFER (in our case 1.5). This kind of error control is much less strict than the typical family-wise error rate (FWER) or FDR control one would apply when doing statistical inference. In fact, one can observe in Figs. 3 and 4 that although stability selection has a low FPR, for a sample size of 200 its FDR is still much higher than one would typically consider acceptable when doing inference (common FDR control levels are 0.05 or 0.1). Additionally, we gave the meta-learner information about the number of views containing signal in the data (parameter q), which the other meta-learners did not have access to. It is also worth noting that the sets of views selected by stability selection in both gene expression data sets had low view selection stability. Ideally, selecting views based on their stability would lead to a set of selected views that is itself highly stable, but evidently this is not the case. It follows then that stability selection may produce a set of selected views which is neither particularly useful for prediction, nor for inference. One could add additional assumptions (Shah and Samworth 2013), which may increase predictive performance, but may also increase FDR. Or one could opt for stricter error control, but this would likely reduce classification performance even further. This implies that performing view selection for both the aims of prediction and inference using a single procedure may produce poor results, since the resulting set of selected views may not be suitable for either purpose.

In this study we only considered different meta-learners within the MVS framework. Of course, many other algorithms for training classifiers exist. Some of those classifiers may be expected to perform better in terms of classification performance than the classifiers presented here, but not many have the embedded view selection properties of MVS-based methods. For example, a random forest would probably perform very well in terms of classification, but the resulting classifier is hard to interpret and does not automatically select the most important views for prediction. One non-MVS method which does automatically select views is the group lasso (Yuan and Lin 2007), but we did not include it here as an extensive comparison between StaPLR/MVS and the group lasso has already been performed elsewhere (Van Loon et al. 2020).

Any simulation study is limited by its choice of experimental factors. In particular, in our simulations we assumed that all features corresponding to signal have the same regression weight, and that all views contain an equal number of features. The correlation structures we used are likely simpler than those encountered in real data sets. Additionally, we defined the view selection problem in such a way that we want to select any view which contains at least some (in our simulations at least 50%) features truly related to the outcome. In practice, the amount of signal present in a view may be lower, leading to considerations of exactly how much signal should be present in a view for a researcher to consider it worth selecting. Additionally, we only considered settings where views are mutually exclusive, but in practice views may overlap (Yuan et al. 2011; Park et al. 2015), meaning that a single feature may correspond to multiple views. In general, the MVS algorithm can handle overlapping views by simply 'copying' a feature for each additional view in which it occurs.

However, an exploration of the implications of overlapping views for view selection, both in MVS and in general, would make an interesting topic for future research. We also did not include the possibility of missing data. In multi-view data, it is quite likely that if missing data occurs, all features within a view will be simultaneously missing. Future work may focus on developing optimal strategies for handling missing data in the multi-view context.

In this study, we evaluated the performance of the different meta-learners across a variety of settings, including high-dimensional and highly correlated settings. Most of these settings were not easy problems, as evident by the absolute accuracy values obtained by the meta-learners. Additionally we considered two real data examples, one considerably harder than the other. Across all our experiments, the relative performance of the nonnegative lasso, nonnegative adaptive lasso and nonnegative elastic net remained remarkably stable. Our results show that MVS can be used with one of these meta-learners to obtain models which are substantially sparser at the view level than those obtained with other meta-learners, without incurring a major penalty in classification accuracy. The nonnegative elastic net is particularly suitable if it is important to the research that, out of a set of correlated features, more than one should be selected. If this is not of particular importance, the nonnegative lasso and nonnegative adaptive lasso can provide even sparser models.

Declarations

Appendix 1 Additional information on colitis and breast cancer data

See Fig. 5 and Table 4.

Fig. 5 Boxplots of the number of features per view for the breast cancer and colitis data sets
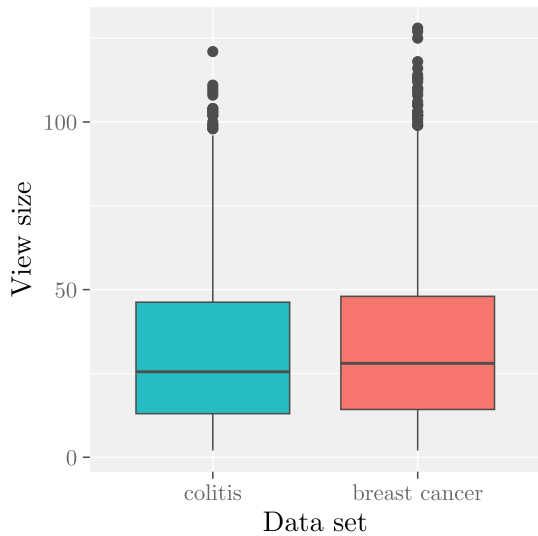

Table 4 Comparing the view and feature dimension resulting from applying MVS with different metalearners to the breast cancer and colitis data. ANSV denotes the average number of selected views, ANSF denotes the average number of selected features. We show the mean and standard deviation across the 10×10 different fitted models

Metalearner	Colitis		Breast cancer	
	ANSV	ANSF	ANSV	ANSF
Lasso	10.7 ± 1.9	678.4 ± 119.6	3.6 ± 1.8	99.3 ± 65.0
Elastic net	40.0 ± 4.6	1978.2 ± 282.8	13.1 ± 2.8	388.6 ± 115.5
Adaptive lasso	7.4 ± 1.6	482.6 ± 90.9	2.6 ± 1.2	71.4 ± 42.0
Ridge	132.7 ± 18.1	5349.6 ± 749.6	52.1 ± 15.9	1643.8 ± 489.3
Interpolating predictor	242.0 ± 1.6	9418.2 ± 114.6	298.6 ± 1.6	9778.9 ± 172.1
NNFS	2.6 ± 0.5	180.1 ± 49.7	3.1 ± 1.0	99.3 ± 56.6
Stability selection	2.0 ± 0.9	161.7 ± 76.9	1.5 ± 1.1	46.7 ± 49.6

Appendix 2 Simulation results for $V = 300$ and $m_v = 250$

See Figs. 6, 7, 8, 9.
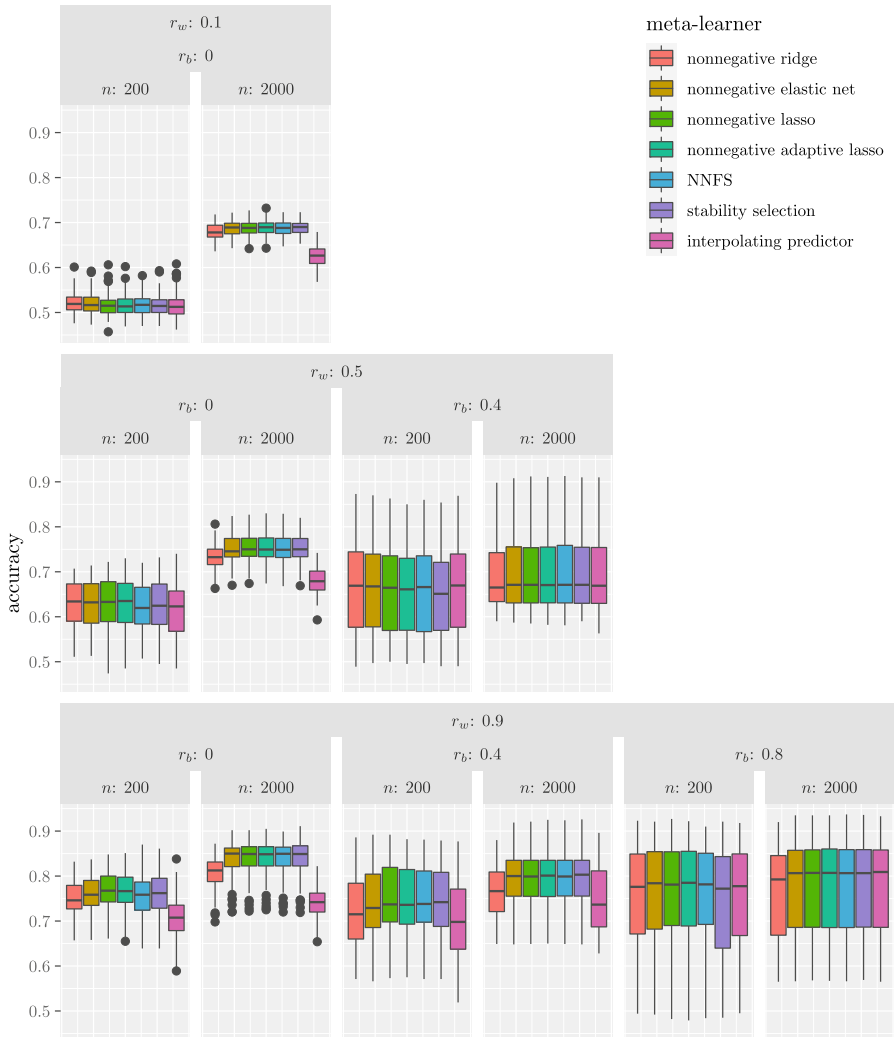
Fig. 6 Boxplots of test accuracy for the different meta-learners, with 300 views and 250 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications
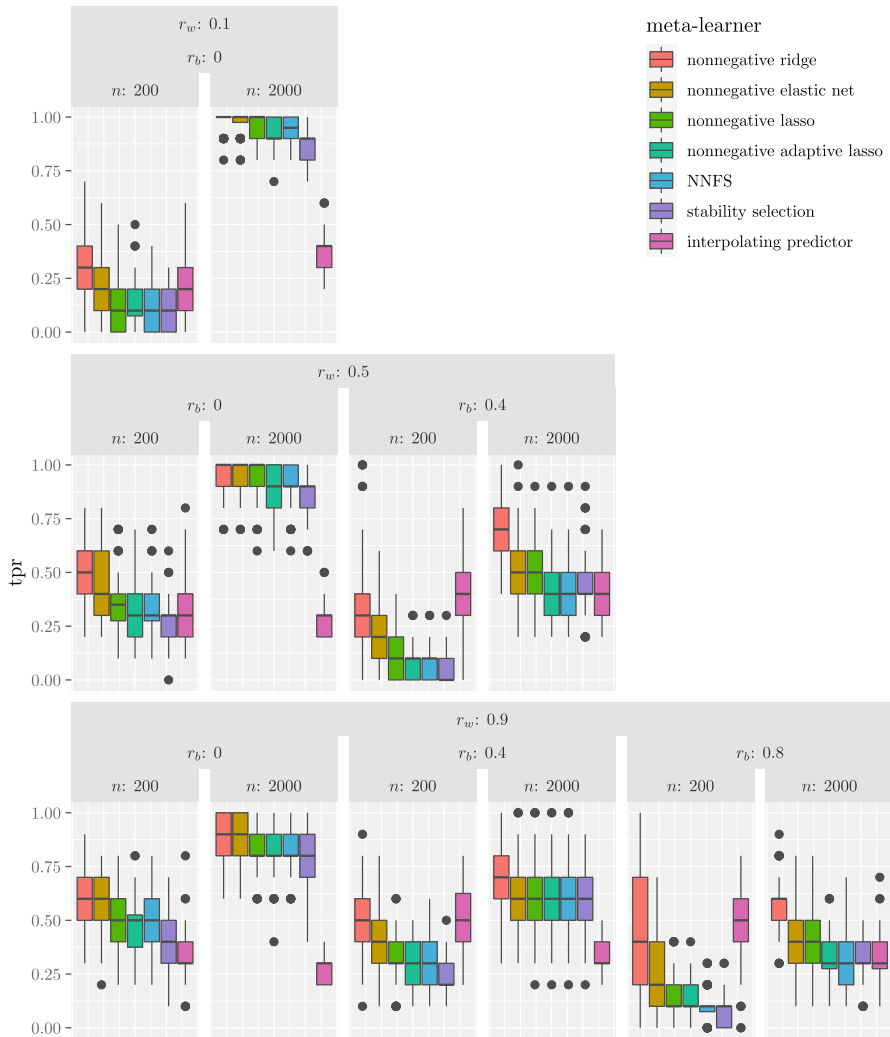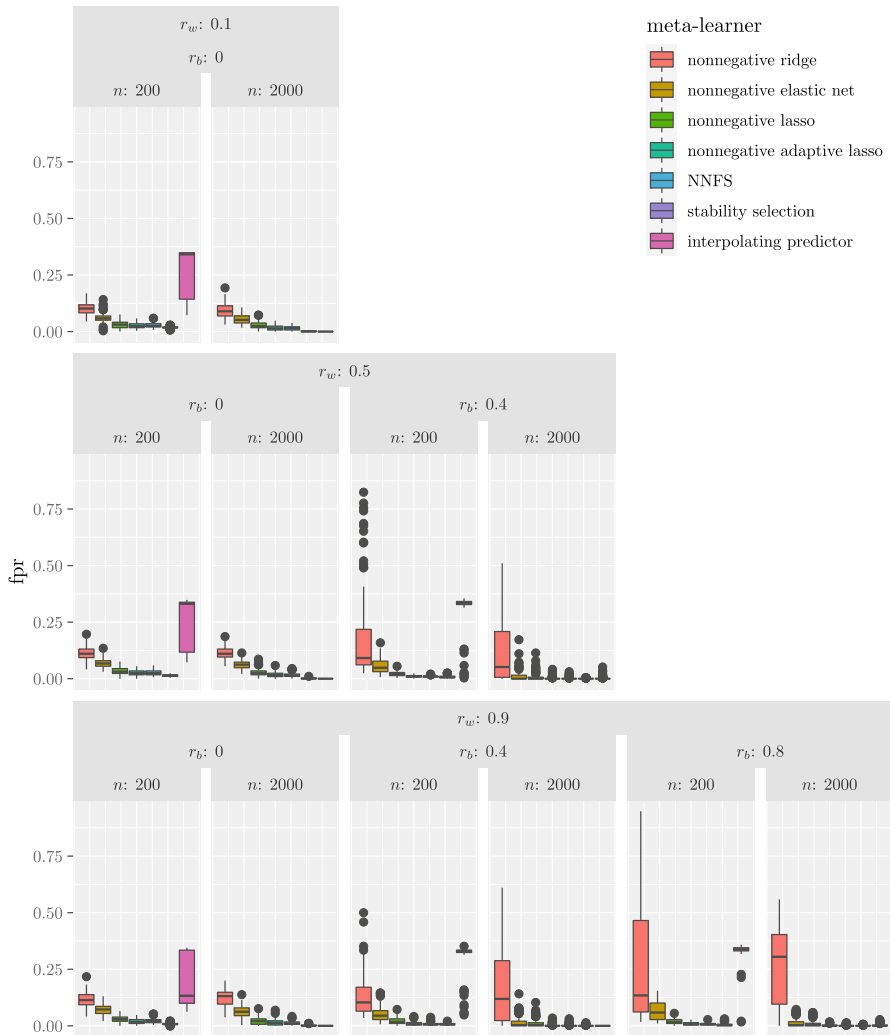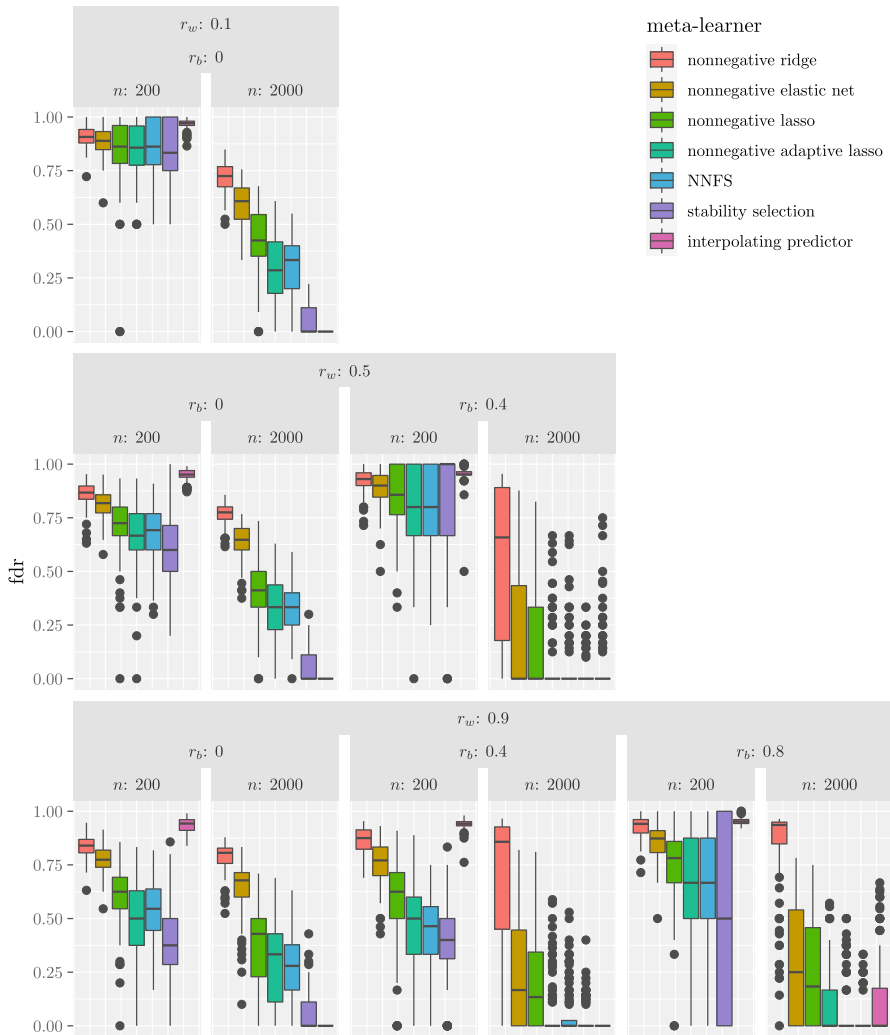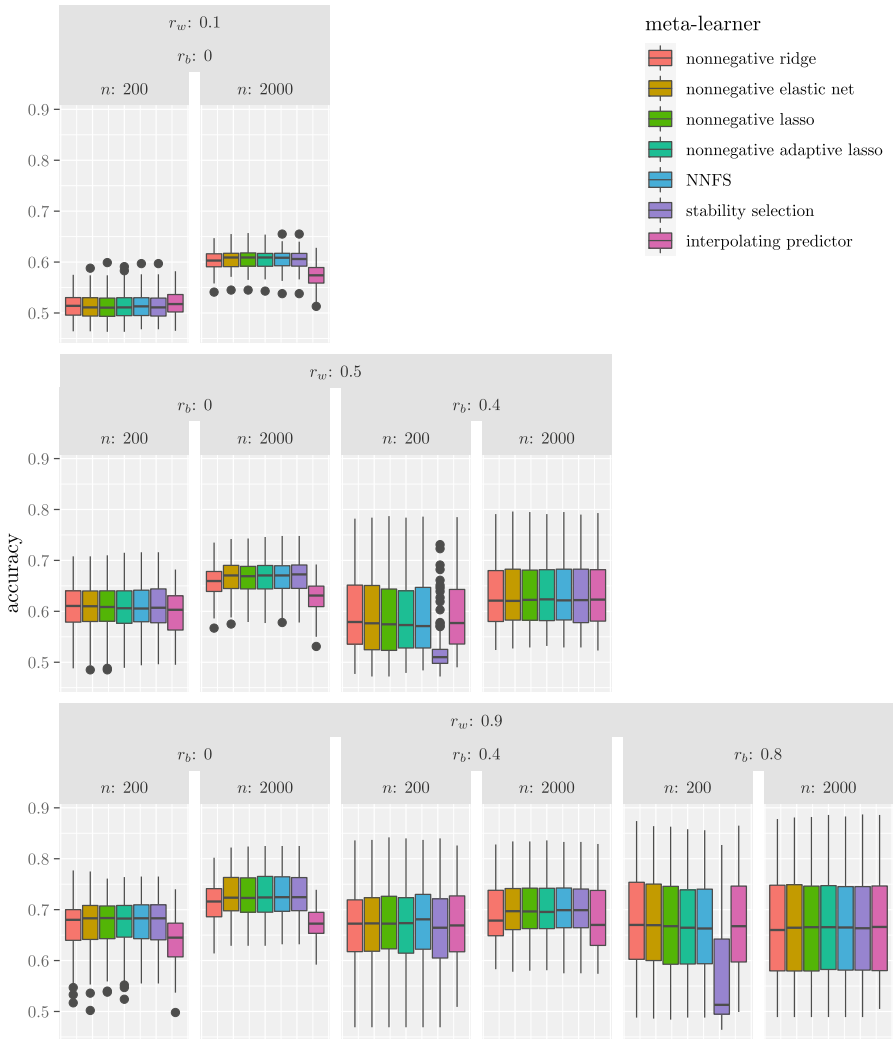
Fig. 7 Boxplots of the true positive rate (TPR) for the different meta-learners, with 300 views and 250 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications
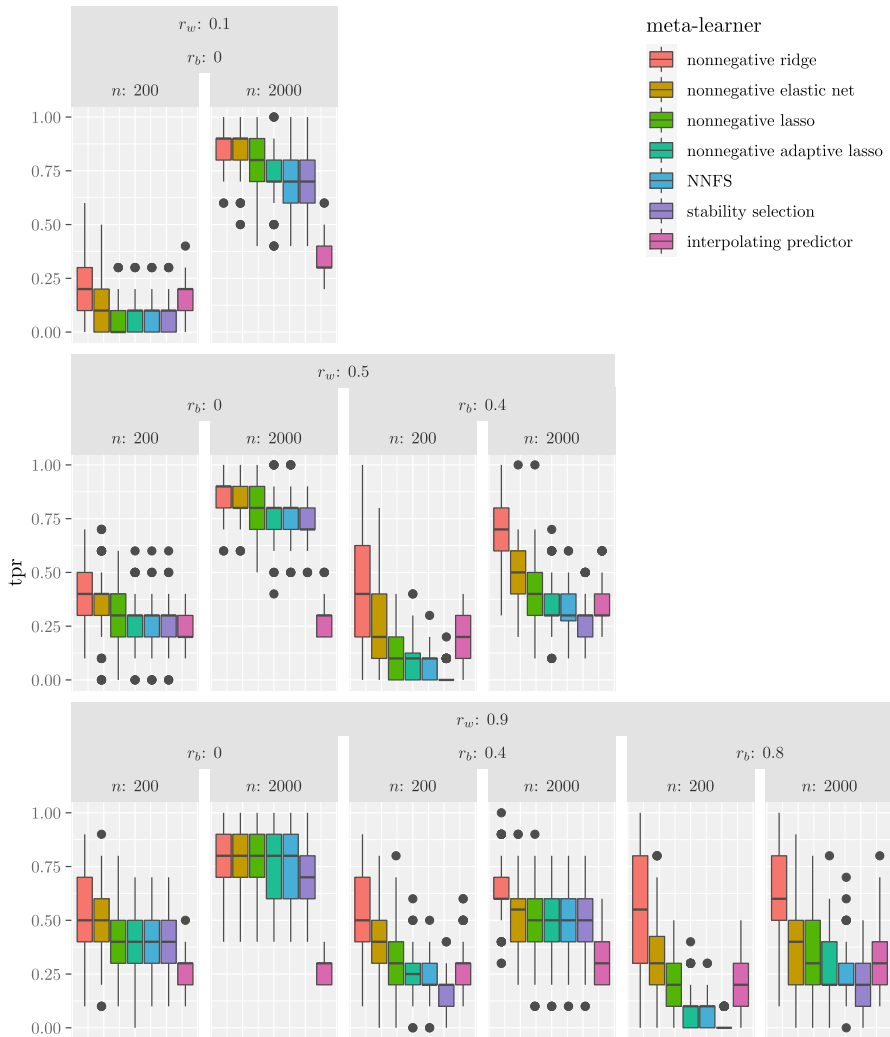
Fig. 8 Boxplots of the false positive rate (FPR) for the different meta-learners, with 300 views and 250 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications

Fig. 9 Boxplots of the false discovery rate (FDR) for the different meta-learners, with 300 views and 250 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications

Appendix 3 Simulation results for $V = 30$ and $m_v = 250$

See Figs. 10, 11, 12, 13.

Fig. 10 Boxplots of test accuracy for the different meta-learners, with 30 views and 250 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications
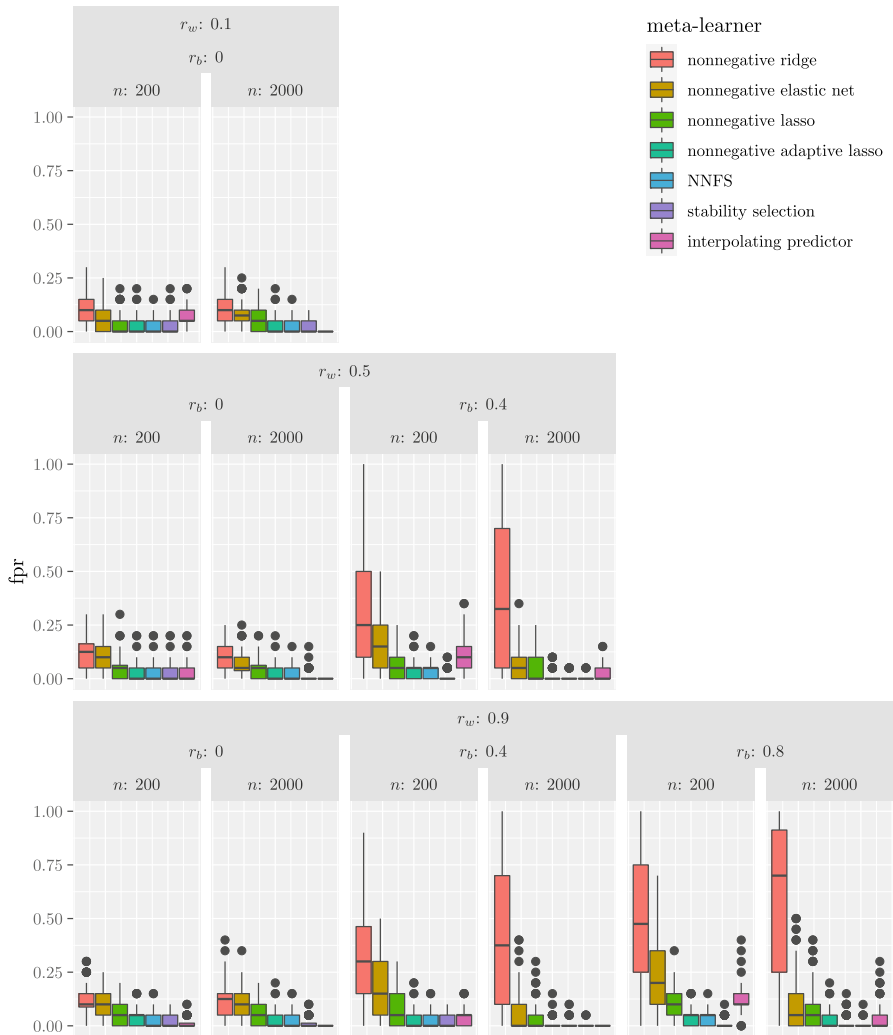
Fig. 11 Boxplots of the true positive rate (TPR) for the different meta-learners, with 30 views and 250 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications
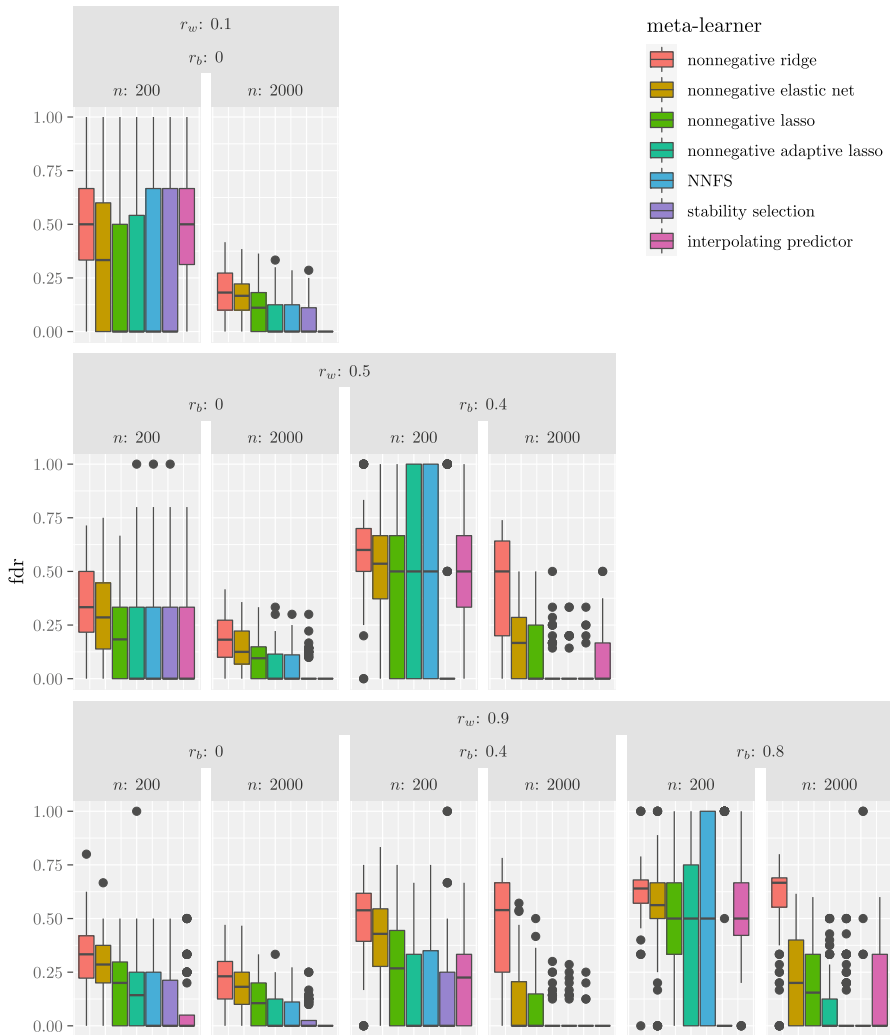
Fig. 12 Boxplots of the false positive rate (FPR) for the different meta-learners, with 30 views and 250 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications

Fig. 13 Boxplots of the false discovery rate (FDR) for the different meta-learners, with 30 views and 250 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications

Appendix 4 Simulation results for $V = 30$ and $m_v = 2500$
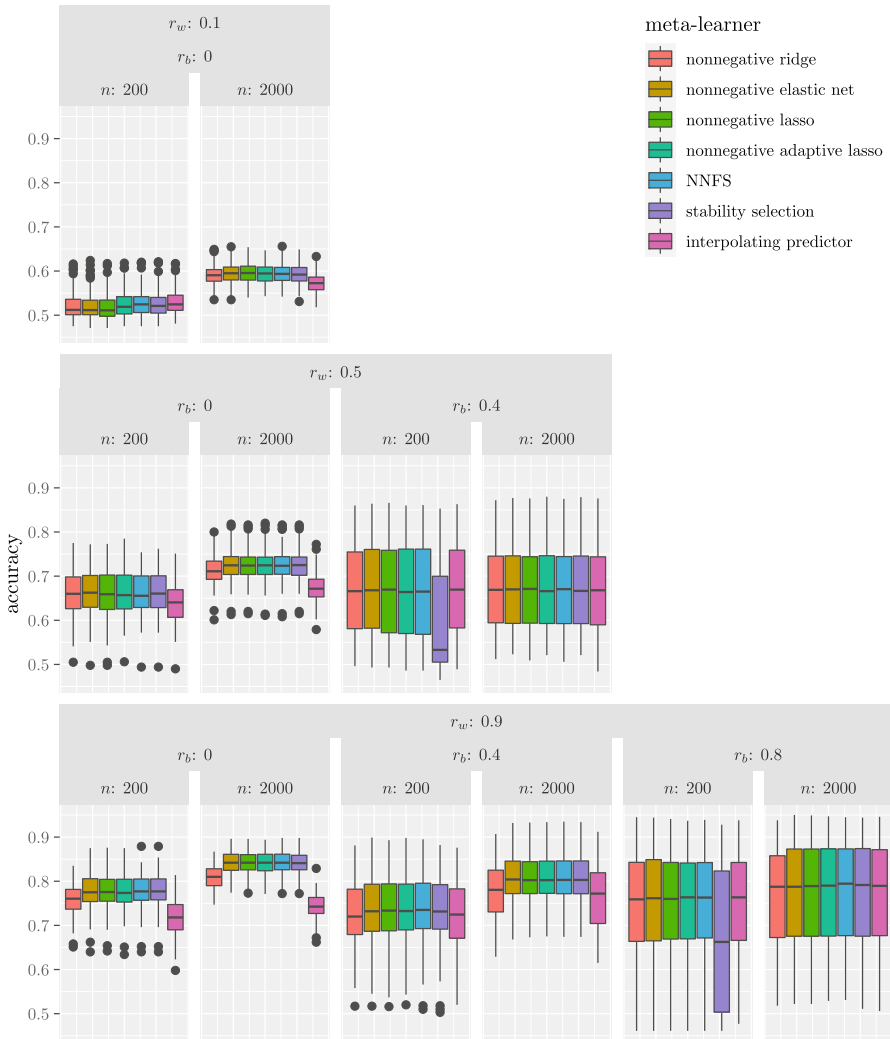
See Figs. 14, 15, 16, 17.

Fig. 14 Boxplots of test accuracy for the different meta-learners, with 30 views and 2500 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications
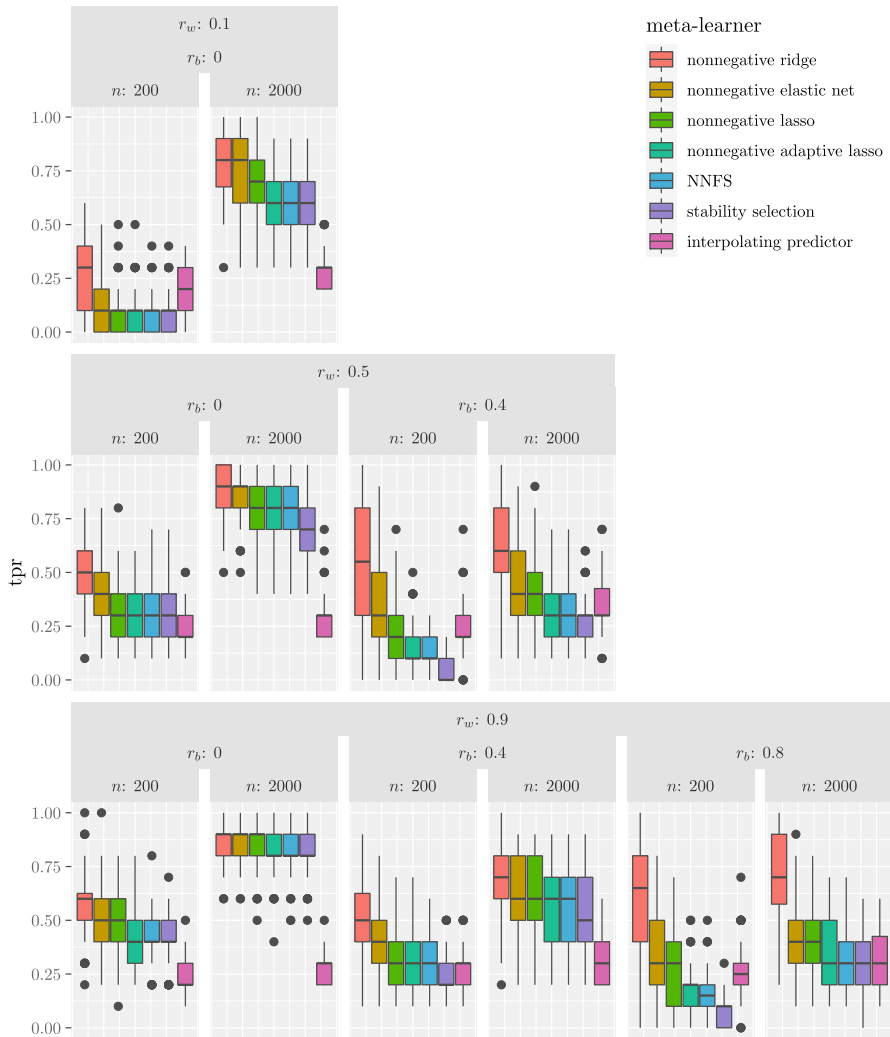
Fig. 15 Boxplots of the true positive rate (TPR) for the different meta-learners, with 30 views and 2500 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications
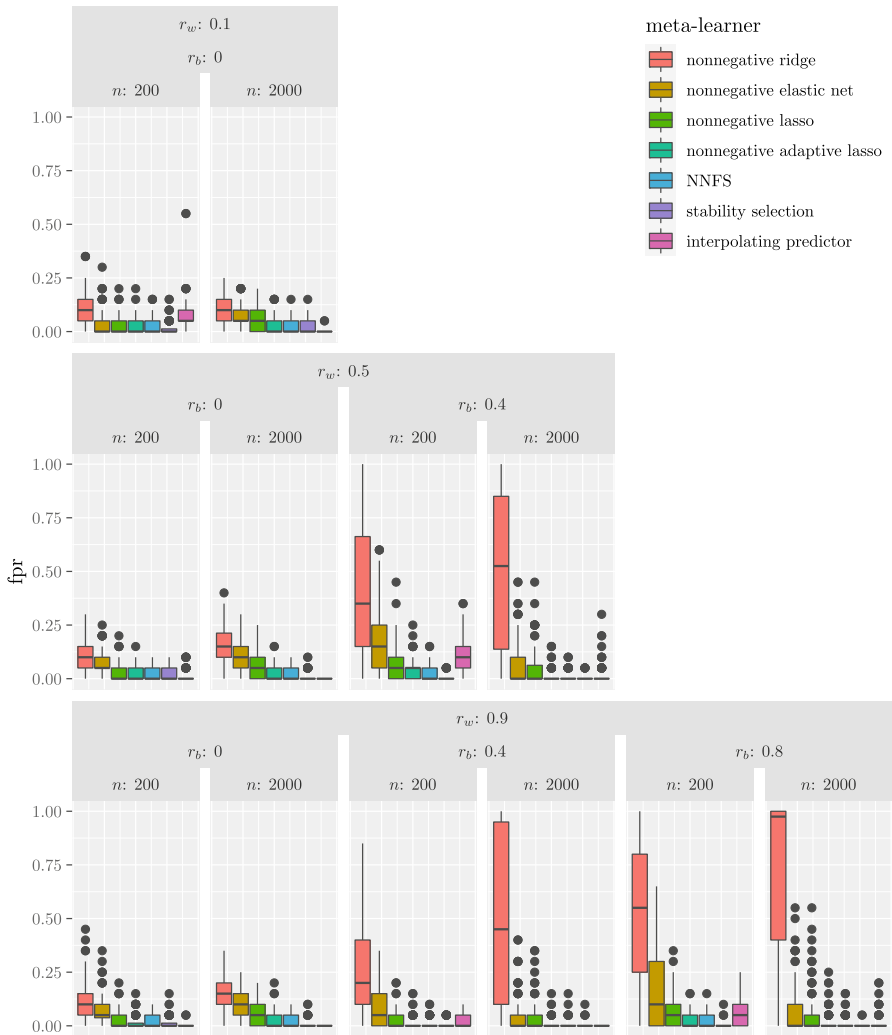
Fig. 16 Boxplots of the false positive rate (FPR) for the different meta-learners, with 30 views and 2500 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications
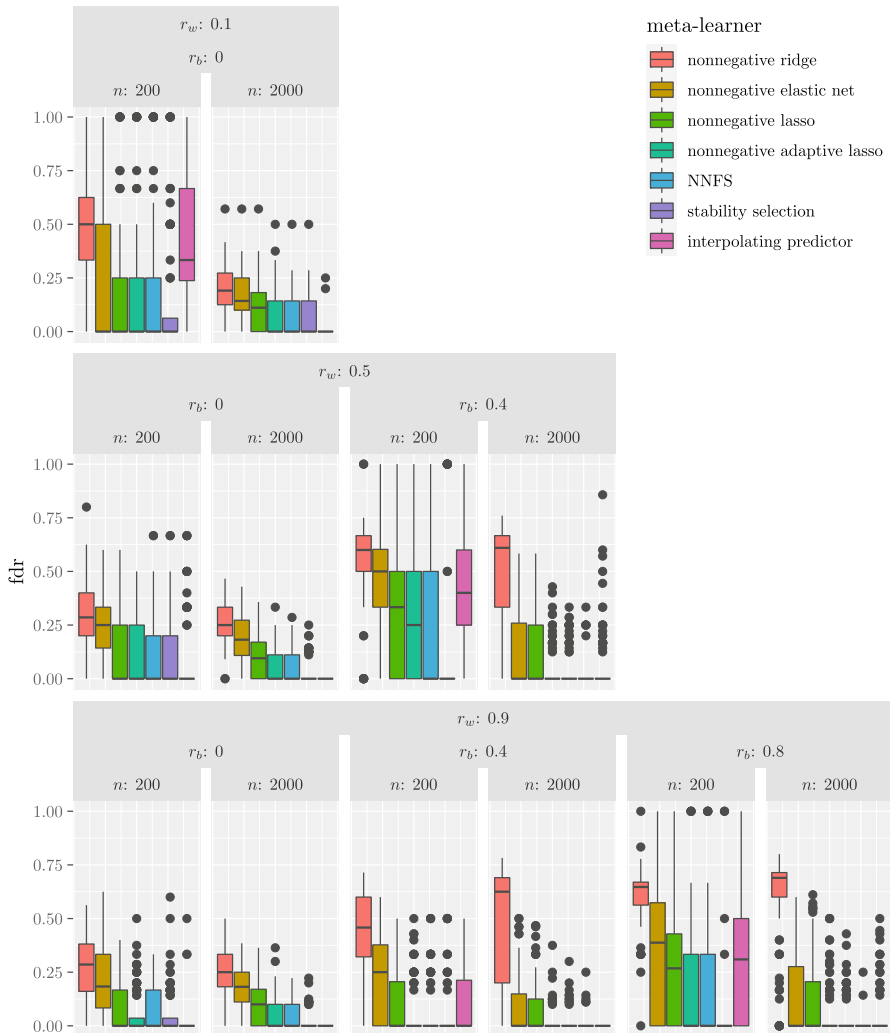
Fig. 17 Boxplots of the false discovery rate (FDR) for the different meta-learners, with 30 views and 2500 features per view. The results are shown for all combinations of the correlation between features within the same view (ρ_w), the correlation between features from different views (ρ_b), and sample size (n). Each plot is based on 100 replications

References

Anagnostopoulos C, Hand DJ (2019) . hmeasure: the H-measure and other scalar classification performance metrics https://CRAN.R-project.org/package=hmeasure R package version 1.0-2

Ballings M, Van den Poel D (2013) AUC: threshold independent performance measures for probabilistic classifiers. https://CRAN.R-project.org/package=AUC R package version 0.3.0

Benner A, Zucknick M, Hielscher T, Ittrich C, Mansmann U (2010) High-dimensional cox models: the choice of penalty as part of the model building process. Biom J 52(1):50–69. https://doi.org/10.1002/bimj.200900064

Bommert A, Sun X, Bischl B, Rahnenführer J, Lang M (2020) Benchmark for filter methods for feature selection in high-dimensional classification data. Comput Stat Data Anal 14(3):106–839. https://doi.org/10.1016/j.csda.2019.106839

Breiman L (1996) Stacked regressions. Mach Learn 24:49–64. https://doi.org/10.1007/bf00117832

Burczynski ME, Peterson RL, Twine NC, Zuberek KA, Brodeur BJ, Casciotti L, Dorner AJ (2006) Molecular classification of Crohn's disease and ulcerative colitis patients using transcriptional profiles in peripheral blood mononuclear cells. J Mol Diagn 81:51–61. https://doi.org/10.2353/jmoldx.2006.050079

Cohen J (1988) Statistical power analysis for the behavioral sciences, 2nd edn. Academic Press, New York

De Vos F, Schouten TM, Hafkemeijer A, Dopper EGP, van Swieten JC, de Rooij M, Rombouts SA (2016) Combining multiple anatomical MRI measures improves Alzheimer's disease classification. Human Brain Mapp 37:1920–1929. https://doi.org/10.1002/hbm.23147

Flach P, Hernández-Orallo J, Ferri C (2011) A coherent interpretation of AUC as a measure of aggregated classification performance . In: Proceedings of the 28th international conference on machine learning, pp 657–664

Fleiss JL (1971) Measuring nominal scale agreement among many raters. Psychol Bull 76(5):378–382. https://doi.org/10.1037/h0031619

Fratello M, Caiazzo G, Trojsi F, Russo A, Tedeschi G, Tagliaferri R, Esposito F (2017) Multi-view ensemble classification of brain connectivity images for neurodegeneration type discrimination. Neuroinformatics 15(2):199–213. https://doi.org/10.1007/s12021-017-9324-2

Friedman J, Hastie T, Tibshirani R (2009) The elements of statistical learning, 2nd edn. Springer-Verlag, New York. https://doi.org/10.1007/978-0-387-84858-7

Friedman J, Hastie T, Tibshirani R (2010) Regularization paths for generalized linear models via coordinate descent. J Stat Softw 33:11–22. https://doi.org/10.18637/jss.v033.i0

Garcia-Ceja E, Galván-Tejada CE, Brena R (2018) Multi-view stacking for activity recognition with sound and accelerometer data. Inf Fusion 40:45–56. https://doi.org/10.1016/j.inffus.2017.06.004

Guyon I, Elisseeff A (2003) An introduction to variable and feature selection. J Mach Learn Res 3:1157–1182. https://doi.org/10.1007/978-3-540-35488-8_1

Hand DJ (2009) Measuring classifier performance: a coherent alternative to the area under the ROC curve. Mach Learn 77:103–123. https://doi.org/10.1007/s10994-009-5119-5

Hastie T, Tibshirani R, Tibshirani R (2020) Best subset, forward stepwise, or lasso? Analysis and recommendations based on extensive comparisons. Stat Sci 35(4):579–592. https://doi.org/10.1214/19-sts733

Hastie T, Tibshirani R, Wainwright M (2015) Statistical learning with sparsity: the lasso and generalizations. CRC Press, Boca Raton. https://doi.org/10.1201/b18401

Hernández-Orallo J, Flach P, Ferri C (2012) A unified view of performance metrics: translating threshold choice into expected classification loss. J Mach Learn Res 13(1):2813–2869. https://doi.org/10.1145/1015330.1015395

Hoerl AE, Kennard RW (1970) Ridge regression: biased estimation for nonorthogonal problems. Technometrics 12(1):55–67. https://doi.org/10.1080/00401706.1970.10488634

Hofner B, Boccuto L, Göker M (2015) Controlling false discoveries in high-dimensional situations: boosting with stability selection. BMC Bioinf 16:144. https://doi.org/10.1186/s12859-015-0575-3

Hofner B, Hothorn T (2017) Stabs: stability selection with error control. https://CRAN.R-project.org/package=stabs R package version 0.6-3

Landis JR, Koch GG (1977) The measurement of observer agreement for categorical data. Biometrics 33(1):159–174. https://doi.org/10.2307/2529310

Li, R, Hapfelmeier A, Schmidt J, Perneczky R, Drzezga A, Kurz A, Kramer S (2011) A case study of stacked multi-view learning in dementia research . In: 13th conference on artificial intelligence in medicine, pp 60–69

Li Y, Wu FX, Ngom A (2018) A review on machine learning principles for multi-view biological data integration. Brief Bioinform 19(2):325–340. https://doi.org/10.1093/bib/bbw113

Ma XJ, Wang Z, Ryan PD, Isakoff SJ, Barmettler A, Sgroi Fuller ADC (2004) A two-gene expression ratio predicts clinical outcome in breast cancer patients treated with tamoxifen. Cancer Cell 56:607–616. https://doi.org/10.1016/j.ccr.2004.05.015

Matsumoto M, Nishimura T (1998) Mersenne twister: a 623-dimensionally equidistributed uniform pseudorandom number generator. ACM Trans Model Comput Simul 8:13–30. https://doi.org/10.1145/272991.272995

Meinshausen N, Bühlmann P (2006) High-dimensional graphs and variable selection with the lasso. Ann Stat 34(3):1436–1462. https://doi.org/10.1214/009053606000000281

Meinshausen N, Bühlmann P (2010) Stability selection. J Royal Stat Soc B 72(4):417–473. https://doi.org/10.1111/j.1467-9868.2010.00740.x

Nogueira S, Sechidis K, Brown G (2018) On the stability of feature selection algorithms. J Mach Learn Res 18(17):41–54. https://doi.org/10.1007/978-3-030-46150-8_20

Park H, Niida A, Miyano S, Imoto S (2015) Sparse overlapping group lasso for integrative multi-omics analysis. J Comput Biol 22(2):73–84. https://doi.org/10.1089/cmb.2014.0197

R Core Team (2017) R : a language and environment for statistical computing. Vienna, Austria. https://www.R-project.org/

Rovai AP, Baker JD, Ponton MK (2013) Social science research design and statistics: a practitioner's guide to research methods and IBM SPSS. Watertree Press LLC, Chesapeake

Shah RD, Samworth RJ (2013) Variable selection with error control: another look at stability selection. J Royal Stat Soc Series B (Stat Methodol) 75(1):55–80. https://doi.org/10.1111/j.1467-9868.2011.01034.x

Simon N, Friedman J, Hastie T, Tibshirani R (2013) A sparse-group lasso. J Comput Graph Stat 22(2):231–245. https://doi.org/10.1080/10618600.2012.681250

Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, Mesirov JP (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. Proc Natl Acad Sci USA 102(43):15545–15550. https://doi.org/10.1073/pnas.0506580102

Sun S, Mao L, Dong Z, Wu L (2019) Multiview machine learning. Springer-Verlag, Berlin. https://doi.org/10.1007/978-981-13-3029-2

Tibshirani R (1996) Regression shrinkage and selection via the lasso. J Royal Stat Soc B 58(1):267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

Ting KM, Witten IH (1999) Issues in stacked generalization issues in stacked generalization. J Artif Intell Res 10:271–289. https://doi.org/10.1613/jair.594

Van Le Cessie S, Houwelingen JC (1992) Ridge estimators in logistic regression. J Royal Stat Soc C 41(1):191–201. https://doi.org/10.2307/2347628

Van Loon W (2022) MVS: methods for high-dimensional multi-view learning. https://CRAN.R-project.org/package=mvs R package version 1.0.2

Van Loon W, Fokkema M, Szabo B. De, Rooij M (2020) Stacked penalized logistic regression for selecting views in multi-view learning. Inf Fusion 61:113–123. https://doi.org/10.1016/j.inffus.2020.03.007

Venables WN, Ripley BD (2002) Modern applied statistics with S. (4th Edn), Springer-Verlag, New York. ISBN 0-387-95457-0

Wah YB, Ibrahim N, Hamid HA, Abdul-Rahman S, Fong S (2018) Feature selection methods: case of filter and wrapper approaches for maximising classification accuracy. Pertanika J Sci Technol. https://doi.org/10.1109/icecct.2019.8869518

Wang K, Li M, Hakonarson H (2010) Analysing biological pathways in genome-wide association studies. Nat Rev Genet 11(12):843–854. https://doi.org/10.1038/nrg2884

Wang Y, Lawson CL, Hanson RJ (2017) Lsei: solving least squares or uadratic programming problems under equality/inequality constraints. https://CRAN.R-project.org/package=lsei R package version 1.2-0

Wolpert DH (1992) Stacked generalization. Neural Netw 5:241–259. https://doi.org/10.1016/s0893-6080(05)80023-1

Wu L, Yang Y, Hanzhong L (2014) Nonnegative-lasso and application in index tracking. Comput Stat Data Anal 70:116–126. https://doi.org/10.1016/j.csda.2013.08.012

Xu H, Caramanis C, Mannor S (2012) Sparse algorithms are not stable: a no-free-lunch theorem. IEEE Trans Pattern Anal Mach Intell 34(1):187–193. https://doi.org/10.1109/tpami.2011.177

Yuan L, Liu J, Ye J (2011) Efficient methods for overlapping group lasso. Adv Neural Inf Process Syst 24:352–360. https://doi.org/10.1109/tpami.2013.17

Yuan M, Lin Y (2007) Model selection and estimation in regression with grouped variables. J Royal Stat Soc B 68(1):49–67. https://doi.org/10.1111/j.1467-9868.2005.00532.x

Zhao J, Xie X, Xu X, Sun S (2017) Multi-view learning overview: recent progress and new challenges. Inf Fusion 38:43–54. https://doi.org/10.1016/j.inffus.2017.02.007

Zou H (2006) The adaptive lasso and its oracle properties. J Am Stat Assoc 101(476):1418–1429. https://doi.org/10.1198/016214506000000735

Zou H, Hastie T (2005) Regularization and variable selection via the elastic net. J Royal Stat Soc B 67(2):301–320. https://doi.org/10.1111/j.1467-9868.2005.00503.x

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.