Supplementary Information: Ranking and combining multiple predictors without labeled data

Fabio Parisi*, Francesco Strino*, Boaz Nadler and Yuval Kluger

January 7, 2014

Contents

1 Covariance between classifiers

Proof of Lemma 1. To prove the lemma we first compute the mean $\mu_i = \mathbb{E}[f_i(X)]$ and variance $Var[f_i(X)]$ of the i-th classifier. We then use these results to compute the entries of the population covariance matrix, $q_{ij} = \mathbb{E}[(f_i(X) - \mu_i) \cdot (f_j(X) - \mu_j)]$.

Under the assumption of independence between instances, the population mean of the i-th classifier is

$$\begin{aligned} \mathbb{E}\left[f_i(X)\right] &= \Pr[f_i(X) = 1] - \Pr[f_i(X) = -1] \\ &= \textstyle\sum_{y\in\{-1,1\}} \Pr[f_i(X) = 1|Y = y] \Pr[Y = y] - \sum_{y\in\{-1,1\}} \Pr[f_i(X) = -1|Y = y] \Pr[Y = y]. \end{aligned}$$

Using the definitions of sensitivity $\psi_i = \Pr[f_i(X) = 1|Y = 1]$, specificity $\eta_i = \Pr[f_i(X) = -1|Y = -1]$, and class imbalance $b = \Pr[Y = 1] - \Pr[Y = -1]$, the equation above can be expressed as follows,

$$\begin{aligned} \mu_i = \mathbb{E}\left[f_i(X)\right] &= \psi_i \left(\tfrac{1+b}{2}\right) + (1 - \eta_i) \left(\tfrac{1-b}{2}\right) - (1 - \psi_i) \left(\tfrac{1+b}{2}\right) - \eta_i \left(\tfrac{1-b}{2}\right) \\ &= \psi_i - \eta_i + b\left(\psi_i + \eta_i - 1\right) = 2\delta_i + b(2\pi_i - 1) \end{aligned} \tag{1}$$

where $\pi_i = (\psi_i + \eta_i)/2$ is the balanced accuracy of the i-th classifier and $\delta_i = (\psi_i - \eta_i)/2$. Similarly, the population variance of the i-th classifier is

$$\mathrm{Var}\left[f_i(X)\right] = \mathbb{E}\left[f_i(X)^2\right] - \mathbb{E}\left[f_i(X)\right]^2 = 1 - \mathbb{E}\left[f_i(X)\right]^2 = 1 - \left(2\delta_i + b(2\pi_i - 1)\right)^2. \tag{2}$$

Next, consider $\mathbb{E}[f_i(X) \cdot f_j(X)]$ for $i \neq j$. Under the assumption of independence of errors between different instances and between different classifiers,

$$\begin{aligned} \mathbb{E}[f_i(X) \cdot f_j(X)] &= \Pr[f_i(X) = f_j(X)] - \Pr[f_i(X) = -f_j(X)] \\ &= \left(\tfrac{1+b}{2}\right) \psi_i\psi_j + \left(\tfrac{1+b}{2}\right) (1 - \psi_i)(1 - \psi_j) + \left(\tfrac{1-b}{2}\right) (1 - \eta_i)(1 - \eta_j) + \left(\tfrac{1-b}{2}\right) \eta_i\eta_j \\ &\quad - \left(\tfrac{1+b}{2}\right) \psi_i(1 - \psi_j) - \left(\tfrac{1+b}{2}\right) (1 - \psi_i)\psi_j - \left(\tfrac{1-b}{2}\right) \eta_i(1 - \eta_j) - \left(\tfrac{1-b}{2}\right) (1 - \eta_i)\eta_j \end{aligned} \tag{3}$$

Combining Eq. 1 and Eq. 3 yields that for $i \neq j$

$$\mathbb{E}[f_i(X) \cdot f_j(X)] - \mathbb{E}[f_i(X)] \cdot \mathbb{E}[f_j(X)] = (1 - b^2)(\psi_i + \eta_i - 1)(\psi_j + \eta_j - 1) = (1 - b^2)(2\pi_i - 1)(2\pi_j - 1).$$

Thus, the entries q_{ij} of the $M \times M$ covariance matrix of the M classifiers are

$$q_{ij} = \begin{cases} 1 - \mu_i^2 & i = j \\ (2\pi_i - 1)(2\pi_j - 1)\left(1 - b^2\right) & i \neq j \end{cases} \tag{4}$$

□.

2 Rank-one Eigenvector Estimation

In this section we describe four approaches to estimate the eigenvector $\mathbf{v}$ of the rank one matrix R from the sample covariance matrix $\hat{Q}$. We term these methods (i) **linear system** approach; (ii) **weighted linear system** approach; (iii) **SDP** approach; and (iv) **direct eigendecomposition** approach. In our simulations we found that all four approaches gave comparable rankings, though the latter was slightly less accurate (Fig. S1). The linear system approach (i) had computational complexity comparable to the fastest method of direct eigendecomposition, while providing a ranking of quality comparable to the much more computationally heavy SDP method. Method (i) was also slightly faster to compute than its weighted counterpart, method (ii), so we chose it for our benchmarks.

2.1 Linear system

As discussed in the main text, one approach to rank the M classifiers is to construct an estimator $\hat{R}$ of the rank-one matrix R, compute its leading eigenvector $\hat{\mathbf{v}}$ and rank the M classifiers by sorting its entries. Given that $\mathbb{E}[\hat{Q}] = Q$, we estimate the off-diagonal entries of $\hat{R}$ by those of $\hat{Q}$, and only need a consistent method to estimate the diagonal entries. To this end, note that upon the change of variables $|r_{ij}| = e^{t_i} \cdot e^{t_j}$, it follows that in the population setting, for all $i \neq j$,

$$\log|r_{ij}| - t_i - t_j = \log|q_{ij}| - t_i - t_j = 0.$$

In the finite sample setting, we replace the unknown q_{ij} by $\hat{q}_{ij}$ and look for an M-dimensional vector $\mathbf{t}$ such that the relation above holds approximately for all pairs $i \neq j$,

$$\hat{\mathbf{t}} = \arg\min \sum_{j>i} (\log|\hat{q}_{ij}| - \hat{t}_i - \hat{t}_j)^2. \tag{5}$$

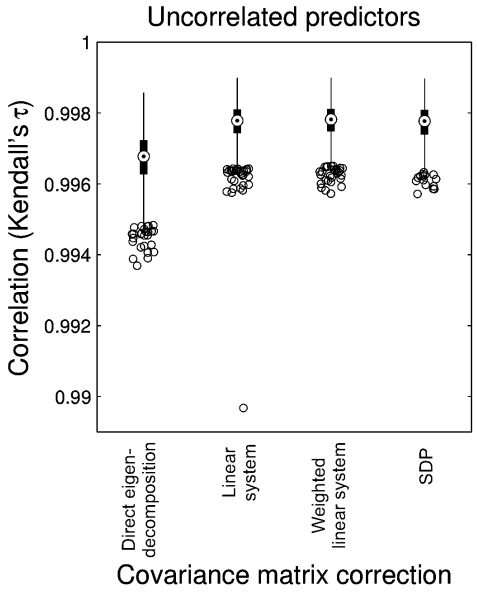

Figure S1: Comparison of the four different approaches to estimate the eigenvector of the rank-one matrix R. The simulated data was constructed as described in section 7.1. The reconstruction quality is measured by Kendall's τ correlation coefficient between the entries of the eigenvector estimated by each approach and the true eigenvector of the rank-one matrix.

From the vector $\mathbf{t}$ we estimate the diagonal entries of R as $\hat{r}_{ii} = \exp(2 \cdot \hat{t}_i)$.

As the functional in Eq. (5) is quadratic, the vector $\hat{\mathbf{t}}$ is efficiently found by solving a system of linear equations with M unknowns. Since $\hat{q}_{ij} \to q_{ij}$ as sample size $S \to \infty$, it follows that $\hat{\mathbf{t}}$ is an asymptotically consistent estimate of $\mathbf{t}$. Consequently the resulting $\hat{\mathbf{v}}$ is a consistent estimate of $\mathbf{v}$, and asymptotically it yields a perfectly correct ranking of the M classifiers, according to their balanced accuracies.

In practice, to avoid the singularity at zero of the logarithm function, we modify Eq. 5 by summing only over indices i, j for which $|\hat{q}_{ij}| > 2\sqrt{Var[\hat{q}_{ij}]}$, where $Var[\hat{q}_{ij}]$ is a plug-in estimator of the true variance, given by Eq. 9 from the main text, and the factor 2 is arbitrary.

2.2 Weighted linear system

Similar to the linear system approach presented above, we can instead consider the following weighted least square problem, where $\mathrm{Var}[\hat{q}_{ij}]$ is given by Eq. 9 from the main text.

$$\hat{\mathbf{t}} = \arg\min \sum_{j>i} \frac{\hat{q}_{ij}^2}{\mathrm{Var}[\hat{q}_{ij}]} \cdot (\log(|\hat{q}_{ij}|) - \hat{t}_i - \hat{t}_j)^2. \tag{6}$$

The resulting estimator $\hat{\mathbf{t}}$ is also solved via a system of linear equations.

2.3 SDP approach

Here we look for a rank-one matrix $\hat{R} = \hat{\lambda}\hat{\mathbf{v}}\hat{\mathbf{v}}^T$, whose off-diagonal terms are closest to those of $\hat{Q}$. While the rank-one constraint is non-convex, its standard relaxation to a trace constraint yields

$$\hat{R} = \arg\min \sum_{i \neq j} (\hat{q}_{ij} - R_{ij})^2 + \theta \,\mathrm{Trace}(R) \tag{7}$$

subject to $R = R^T$, $R \succeq 0$ and where θ is a suitably chosen regularization parameter. This is a convex problem, which can be solved via semi-definite programming [1]. We thus term it **SDP approach**. While in principle SDP problems can be solved to arbitrary accuracy in polynomial time in M, this approach is significantly slower than the two previous ones, which require solutions to systems of linear equations.

2.4 Direct eigendecomposition

Finally, an even simpler approach is to rank the classifiers by directly computing the leading eigenvector of $\hat{Q}$. For a finite number of classifiers M, it follows from Lemma 1 that as $S \to \infty$, this **direct eigen-decomposition**

approach is generally *not* consistent. However, as the following lemma shows, if the rank one matrix R has a large spectral gap, $\lambda \gg 1$, then this leading eigenvector is close to the true one.

Lemma S1. *Let* $\mathbf{w}$ *be the leading unit-norm eigenvector of the population matrix* Q*, and let* λ *be given by Eq. 7 in the main text. Then,*

$$\left(\mathbf{w}^T\mathbf{v}\right)^2 \geq 1 - \frac{2}{\lambda}. \tag{8}$$

Proof : Let $\lambda(Q)$ be the leading eigenvalue of Q with corresponding unit-norm eigenvector $\mathbf{w}$. Let λ be the eigenvalue of the rank-one matrix R with corresponding unit-norm eigenvector $\mathbf{v}$. First, note that

$$Q = R + D \tag{9}$$

where D is a diagonal matrix with entries

$$d_{ii} = 1 - \mu_i^2 - (1-b^2)(2\pi_i - 1)^2.$$

Hence $\|D\|_2 = \max_i |d_{ii}| \leq 1$. It thus readily follows from Weyl's theorem that

$$|\lambda(Q) - \lambda| \leq \|D\|_2 \leq 1. \tag{10}$$

Now, multiplying the eigenvector equation $Q\mathbf{w} = \lambda(Q)\mathbf{w}$ from the left by $\mathbf{w}^T$, and inserting the relation (9) gives that

$$\lambda(Q) = \lambda\left(\mathbf{w}^T\mathbf{v}\right)^2 + \mathbf{w}^T D\mathbf{w}.$$

The lemma follows by combining Eq. 10 with the bound $|\mathbf{w}^T D\mathbf{w}| \leq 1$. □.

Note that if all classifiers in the ensemble have a balanced accuracy bounded away from 1/2, then $\lambda = O(M)$ and then for $M \gg 1$, the angle between $\mathbf{v}$ and $\mathbf{w}$ is small.

Finally, we note that this direct eigendecomposition approach is equivalent to ranking classifiers by a singular value decomposition (SVD) of the $S \times M$ mean-centered matrix of predicted labels $f_i(x_k)$. This approach, although apparently without the mean-centering operation, was recently suggested in [2], which proposed the j-th entry in the leading right singular vector as a proxy for the reliability of the j-th classifier. Our work provides a probabilistic interpretation to this approach, as it shows that the entries of $\mathbf{w}$, which is also the leading right singular vector of the (mean-centered) matrix $f_i(x_k)$, are approximately those of $\mathbf{v}$, which in turn are proportional to the balanced accuracies of the classifiers.

2.5 Asymptotic Eigenvector Stability

We now consider the asymptotic stability of the estimated eigenvector to small perturbations due to finite sample fluctuations in our estimate $\hat{Q}$. First note that for all $i \neq j$, $\hat{q}_{ij} - q_{ij} = O(1/\sqrt{S})$. It thus follows that upon solving the linear system for the vector $\mathbf{t}$, asymptotically its errors are also $O(1/\sqrt{S})$, and hence for all $i \neq j$, we may assume that $\hat{r}_{ij} - r_{ij} = O(1/\sqrt{S})$.

To understand how these fluctuations affect the estimation of the leading eigenvector of the rank one matrix R, we consider the one-parameter family of matrices $\hat{R}(\epsilon) = R + \epsilon B$ where $B = \sqrt{S}(\hat{R} - R)$ is a matrix whose entries are all $O(1)$. By definition, at $\epsilon = 1/\sqrt{S}$ we have that $\hat{R}(\epsilon) = \hat{R}$. We thus view ϵ as a small parameter, study the dependence of the leading eigenvector of $\hat{R}(\epsilon)$ on ϵ, and eventually plug in $\epsilon = 1/\sqrt{S}$.

Given that both R and B are symmetric, standard results from matrix perturbation theory [3] imply that for sufficiently small ϵ the leading eigenvector and eigenvalue of $\hat{R}(\epsilon)$ are analytic functions of ϵ. At $\epsilon = 0$, these resort to the eigenvector $\mathbf{v}$ and eigenvalue λ of the exact rank one matrix R. For small $\epsilon > 0$ we may thus expand

$$\begin{aligned} \hat{\lambda}(\epsilon) &= \lambda + \epsilon\lambda^{(1)} + \epsilon^2\lambda^{(2)} + \ldots \\ \hat{\mathbf{v}}(\epsilon) &= \mathbf{v} + \epsilon\mathbf{v}^{(1)} + \epsilon^2\mathbf{v}^{(2)} + \ldots \end{aligned}$$

Inserting this expansion into the eigenvalue-eigenvector equation $\hat{R}(\epsilon)\hat{\mathbf{v}}(\epsilon) = \hat{\lambda}(\epsilon)\hat{\mathbf{v}}(\epsilon)$, and equating powers of ϵ gives that the $O(\epsilon)$ equation reads

$$R\mathbf{v}^{(1)} + B\mathbf{v} = \lambda\mathbf{v}^{(1)} + \lambda^{(1)}\mathbf{v}. \tag{11}$$

Since the eigenvector $\hat{\mathbf{v}}(\epsilon)$ is defined only up to a normalization constant, we conveniently chose it to be that $\mathbf{v}^T\hat{\mathbf{v}}(\epsilon) = 1$ for all ϵ, which in particular implies that $\mathbf{v}^T\mathbf{v}^{(1)} = 0$.

Now multiplying Eq. 11 from the left by $\mathbf{v}^T$ gives that $\lambda^{(1)} = \mathbf{v}^T B\mathbf{v}$ and

$$\mathbf{v}^{(1)} = \frac{1}{\lambda}(I - \mathbf{v}\mathbf{v}^T)^\dagger \left(B - \mathbf{v}^T B\mathbf{v}\right)\mathbf{v} \tag{12}$$

where $A^\dagger$ denotes the Moore-Penrose pseudo-inverse of A.

The key point from Eq. 12 is that for a given spectral gap of size λ of the rank-one matrix R, asymptotically in S, the perturbation in the leading eigenvector estimate is $\hat{\mathbf{v}} - \mathbf{v} = O(\frac{1}{\lambda}\frac{1}{\sqrt{S}})$.

3 Spectral Meta-Learner

In this section we present the derivation of the Spectral Meta-Learner (SML) as a linearization of the maximum likelihood estimator (MLE) of the vector of true class labels around $(\psi^*, \eta^*) = (1/2, 1/2)$.

3.1 Maximum Likelihood Estimator (MLE)

Under the assumption of independence between classifier errors and between instances, given the specificities and sensitivities of the M classifiers, the overall likelihood of the labels of all S instances is a product of the likelihood of each individual instance label. Hence, for each instance x_k its class label y_k can be estimated independently of the class labels of all other instances. The MLE $\hat{y}_k^{(\mathrm{ML})}$ of y_k is

$$\begin{aligned}
\hat{y}_k^{(\mathrm{ML})} &= \operatorname{argmax}\{\log\, \mathfrak{L}(f_1(x_k),\ldots,f_M(x_k); y_k = 1), \log\, \mathfrak{L}(f_1(x_k),\ldots,f_M(x_k); y_k = -1)\} \\
&= \operatorname{sign}\left(\log\, \mathfrak{L}(f_1(x_k),\ldots,f_M(x_k); y_k = 1) - \log\, \mathfrak{L}(f_1(x_k),\ldots,f_M(x_k); y_k = -1)\right) \\
&= \operatorname{sign}\left(\sum_{i|f_i(x_k)=1} \log\psi_i + \sum_{i|f_i(x_k)=-1} \log(1-\psi_i) - \sum_{i|f_i(x_k)=1} \log(1-\eta_i) - \sum_{i|f_i(x_k)=-1} \log\eta_i\right) \\
&= \operatorname{sign}\left(\sum_{i|f_i(x_k)=1} (\log\psi_i - \log(1-\eta_i)) + \sum_{i|f_i(x_k)=-1} (\log(1-\psi_i) - \log\eta_i)\right)
\end{aligned}$$

Next, note that the conditions $f_i(x_k) = 1$ and $f_i(x_k) = -1$ in the two sums above can be represented by the following two indicator functions,

$$\frac{1+f_i(x_k)}{2} = \begin{cases} 0 & f_i(x_k) = -1 \\ 1 & f_i(x_k) = 1 \end{cases} \quad \text{and} \quad \frac{1-f_i(x_k)}{2} = \begin{cases} 1 & f_i(x_k) = -1 \\ 0 & f_i(x_k) = 1 \end{cases}.$$

Using these indicator functions allows to express the MLE as a function of ψ_i and η_i as follows

$$\begin{aligned}
\hat{y}_k^{(\mathrm{ML})} &= \operatorname{sign}\left(\sum_i \frac{1+f_i(x_k)}{2}(\log\psi_i - \log(1-\eta_i)) + \sum_i \frac{1-f_i(x_k)}{2}(\log(1-\psi_i) - \log\eta_i)\right) \\
&= \operatorname{sign}\left(\sum_{i=1}^{M} f_i(x_k)\log\alpha_i + \log\beta_i\right) \qquad (13)
\end{aligned}$$

where

$$\alpha_i = \frac{\psi_i\eta_i}{(1-\psi_i)(1-\eta_i)} \quad \text{and} \quad \beta_i = \frac{\psi_i(1-\psi_i)}{\eta_i(1-\eta_i)}. \qquad (14)$$

3.2 The SML: A first-order approximation of the MLE estimator

Combining Eqs. 13 and 14, the maximum likelihood estimate $\hat{y}_k^{(\mathrm{ML})}$ of the label y_k of the instance x_k is

$$\hat{y}_k^{(\mathrm{ML})} = \operatorname{sign}\left(\sum_i f_i(x_k)\log\left(\frac{\psi_i\eta_i}{(1-\psi_i)(1-\eta_i)}\right) + \log\left(\frac{\psi_i(1-\psi_i)}{\eta_i(1-\eta_i)}\right)\right). \qquad (15)$$

A first-order Taylor expansion of the logarithms, around specificity and sensitivity values (ψ_i^*, η_i^*) gives

$$\begin{aligned}
\sum_{i=1}^{M} f_i(x_k)\log\alpha_i + \log\beta_i &= \sum_i f_i(x_k)\log\left(\frac{\psi_i^*\eta_i^*}{(1-\psi_i^*)(1-\eta_i^*)}\right) + \log\left(\frac{\psi_i^*(1-\psi_i^*)}{\eta_i^*(1-\eta_i^*)}\right) \\
&\quad + f_i(x_k)\left(\frac{\psi_i-\psi_i^*}{\psi_i^*} + \frac{\eta_i-\eta_i^*}{\eta_i^*} + \frac{\psi_i-\psi_i^*}{1-\psi_i^*} + \frac{\eta_i-\eta_i^*}{1-\eta_i^*}\right) \\
&\quad + \left(\frac{\psi_i-\psi_i^*}{\psi_i^*} - \frac{\eta_i-\eta_i^*}{\eta^*} - \frac{\psi_i-\psi_i^*}{1-\psi_i^*} + \frac{\eta_i-\eta_i^*}{1-\eta_i^*}\right) \\
&\quad + O((\psi_i-\psi_i^*)^2, (\eta_i-\eta_i^*)^2, (\psi_i-\psi_i^*)\cdot(\eta_i-\eta_i^*)) \\
&= \sum_i f_i(x_k)\log\left(\frac{\psi_i^*\eta_i^*}{(1-\psi_i^*)(1-\eta_i^*)}\right) + \log\left(\frac{\psi_i^*(1-\psi_i^*)}{\eta_i^*(1-\eta_i^*)}\right) \\
&\quad + (\psi_i-\psi_i^*)\frac{f_i(x_k)-(2\psi_i^*-1)}{\psi_i^*(1-\psi_i^*)} + (\eta_i-\eta_i^*)\frac{f_i(x_k)+(2\eta_i^*-1)}{\eta_i^*(1-\eta_i^*)} \\
&\quad + O((\psi_i-\psi_i^*)^2, (\eta_i-\eta_i^*)^2, (\psi_i-\psi_i^*)\cdot(\eta_i-\eta_i^*))
\end{aligned}$$

At the specific values $(\psi^*, \eta^*) = (1/2, 1/2)$, where $2\psi_i^* - 1 = 2\eta_i^* - 1 = 0$, the Taylor expansion above simplifies considerably. Inserting the resulting expression back into Eq. (15) yields

$$\hat{y}_k^{(\mathrm{SML})} = \mathrm{sign}\left(\sum_i f_i(x_k)\left(\psi_i + \eta_i - 1\right)\right) = \mathrm{sign}\left(\sum_i f_i(x_k)\left(2\pi_i - 1\right)\right) = \mathrm{sign}\left(\sum_i f_i(x_k) v_i\right),$$

where $\mathbf{v} \in \mathbb{R}^M$ is the leading eigenvector of the rank-one matrix R, as described in the main text. We thus call this unsupervised ensemble-classifier the *Spectral Meta-Learner* (SML).

4 Comparison between SML and Majority Voting

In the present section we provide insights into the potential advantages of SML over majority voting. To this end, we study the performance of these two unsupervised ensemble learners in the specific case where all classifiers, except one, have equal sensitivities and specificities. We prove that the resulting balanced accuracy of the weighted voting scheme employed by SML is greater than or equal to the balanced accuracy of majority voting. It is also greater than the balanced accuracy of the best algorithm in the ensemble, up to a small constant.

Lemma S2. *Consider an ensemble of M conditionally independent classifiers such that the first classifier has sensitivity and specificity $\psi_1 = \eta_1 = \pi_1$, and the remaining $M-1$ classifiers have all the same specificity and sensitivity $\psi = \eta = \pi$. Let π_{Vo} be the resulting balanced accuracy of majority voting and let π_{SML} be the balanced accuracy of an (oracle) SML classifier, whose weights assume perfect knowledge of the values π_1 and π. Then, for any value of π_1 and π (with $\pi > 1/2$),*

(i) The balanced accuracy of SML is always greater than or equal to that of majority voting,

$$\pi_{SML} \geq \pi_{Vo}. \tag{16}$$

(ii) The balanced accuracy of SML is always greater than or equal to that of the $M-1$ classifiers,

$$\pi_{SML} \geq \pi. \tag{17}$$

(iii) The balanced accuracy of SML is greater than or equal to that of the first classifier, up to a small constant,

$$\pi_{SML} \geq \pi_1 - \exp\left(-2\epsilon^2(M-1)\right), \tag{18}$$

with $\epsilon = \left(1 + (M-1)(2\pi-1)^2\right) / \left(2(M-1)(2\pi-1)\right)$.

Remarks: Albeit for the specific case where $\pi_1 = \pi$, this lemma yields five insights:

(i) The performance of SML is higher than that of majority voting. Intuitively, this is expected since SML, being a Taylor approximation of the MLE, has weights closer to the optimal ones, in contrast to the equal weights employed by majority voting.

(ii) The second insight is that SML is more accurate than most classifiers in the ensemble. This is not necessarily true for majority voting. For example, in a challenging classification problem where most classifiers in an ensemble are slightly better than random and one classifier is much worse than random, majority voting can have a balanced accuracy smaller than 1/2.

(iii) Eq. 18 may seem disappointing at first sight, as it states that there may be cases where SML has a lower accuracy than the best classifier in the ensemble. However, this is to be expected, since SML follows from a Taylor expansion of the maximum likelihood solution at specificity and sensitivity values of 1/2 (e.g., close to being totally random). Thus, SML is a *conservative* meta-classifier. For example, if the first classifier had perfect balanced accuracy, $\pi_1 = 1$, then its weight in the maximum likelihood solution would be infinite, with effectively zero weights for all other classifiers, see Eq. 14. In contrast, SML gives finite and non-zero weights to all classifiers, provided they are not totally random ($\pi \neq 1/2$). Hence, it may in general be worse than the best classifier in the ensemble. Eq. 18 states, however, that even in this extreme case, the difference in performance between SML and the best classifier is small and it decreases exponentially with the number of classifiers.

(iv) For simplicity we state and prove the lemma assuming that the exact values of π and π_1 are provided by an oracle. As discussed in Section 2.5, with a finite unlabeled dataset consisting of S samples, these values can be estimated with accuracy $O(1/\sqrt{S})$. These estimation errors affect only the SML classifier (as majority voting gives equal weights to all classifiers), and imply that claims (i), (ii) and (iii) hold, up to additional small $O(1/\sqrt{S})$ terms.

(v) Both in the statement of the lemma and in its proof, when a weighted ensemble classifier of the form $\mathrm{sign}(\sum_j a_j f_j(x))$ gives a result of zero for the argument inside the sign, to output a ± 1 class label, we flip a coin at random with probability 1/2 and output its result.

Proof: Under the assumptions of the lemma, it follows that for the corresponding majority voting classifier $\pi_{\text{Vo}} = \psi_{\text{Vo}} = \eta_{\text{Vo}}$, and similarly, $\pi_{\text{SML}} = \psi_{\text{SML}} = \eta_{\text{SML}}$. Hence, it suffices to show that claims (i), (ii) and (iii) hold only for the respective sensitivities.

We start by proving claim (i). To this end, we first consider the case where $\pi_1 = \pi$, or equivalently $\psi_1 = \psi$. In this case SML and majority voting yield the same classifier, whose sensitivity is given by the probability that more than half of the classifiers make a correct prediction. This probability is given by the tail of the binomial cumulative distribution function,

$$\psi_{\text{SML}}\Big|_{\psi_1=\psi} = \psi_{\text{Vo}}\Big|_{\psi_1=\psi} = \psi_{\text{equal}} = \sum_{j>\lfloor M/2 \rfloor}^{j\leq M} \psi^j (1-\psi)^{M-j} \binom{M}{j} = 1 - F\left(\frac{M}{2}; M, \psi\right), \tag{19}$$

where $\lfloor M/2 \rfloor$ denotes the floor (or integer truncation) operation and $F(k; n, p)$ is the probability of at most $\lfloor k \rfloor$ successes in a Binomial distribution with n independent trials of success probability p,

$$F(k; n, p) = \sum_{i=0}^{\lfloor k \rfloor} \binom{n}{i} p^i (1-p)^{n-i}. \tag{20}$$

Next, we analyze the sensitivity of majority voting when $\psi_1 \neq \psi$. By conditioning on the outcome of the first algorithm (giving either a correct or incorrect prediction), it follows from Eq. 19 that

$$\begin{aligned} \psi_{\text{Vo}} &= \psi_1 \left[1 - F\left(\tfrac{M}{2} - 1; M-1, \psi\right)\right] + (1-\psi_1)\left[1 - F\left(\tfrac{M}{2}; M-1, \psi\right)\right] \\ &= 1 - F\left(\tfrac{M}{2}; M-1, \psi\right) + \psi_1 \left[F\left(\tfrac{M}{2}; M-1, \psi\right) - F\left(\tfrac{M}{2} - 1; M-1, \psi\right)\right]. \end{aligned} \tag{21}$$

Importantly, ψ_{Vo} depends linearly on ψ_1 and thus its partial derivative $\partial\psi_{\text{Vo}}/\partial\psi_1$ is constant

$$\frac{\partial \psi_{\text{Vo}}}{\partial \psi_1} = F\left(\tfrac{M}{2}; M-1, \psi\right) - F\left(\tfrac{M}{2} - 1; M-1, \psi\right) = F(\tfrac{M-1}{2} + \tfrac{1}{2}; M-1, \psi) - F(\tfrac{M-1}{2} - \tfrac{1}{2}; M-1, \psi) \tag{22}$$

Now we consider the sensitivity of the SML classifier. Recall that in SML the first classifier is weighted differently from the other classifiers in the ensemble, proportionally to $2\psi_1 - 1$. We define its **relative weight** as

$$\theta = \frac{2\psi_1 - 1}{2\psi - 1}. \tag{23}$$

Again conditioning on the outcome of the first classifier, we have that

$$\psi_{\text{SML}} = \psi_1 \left[1 - F(\tfrac{M-1}{2} - \tfrac{\theta}{2}; M-1, \psi)\right] + (1-\psi_1)\left[1 - F(\tfrac{M-1}{2} + \tfrac{\theta}{2}; M-1, \psi)\right] \tag{24}$$

Note that due to the floor operation in computing the stair-case cumulative distribution function F, ψ_{SML} is a piecewise linear function of ψ_1. It is thus not differentiable at values of ψ_1 for which $(M-1)/2 \pm \theta/2$ is an integer. In addition, some values of ψ_1 for which $(M-1)/2 \pm \theta/2$ is an integer correspond to isolated local minima in the function of ψ_{SML}. These local minima can be effectively replaced by their left (or right) limit $\lim_{\theta\to\theta_\pm} \psi_{\text{SML}}$, thus obtaining a piecewise linear function without isolated points.

At any other value of ψ_1, ψ_{SML} can be differentiated w.r.t. ψ_1. Since the cumulative distribution is constant for sufficiently small positive or negative changes in ψ_1, it follows that

$$\frac{\partial \psi_{\text{SML}}}{\partial \psi_1} = F(\tfrac{M-1}{2} + \tfrac{\theta}{2}; M-1, \psi) - F(\tfrac{M-1}{2} - \tfrac{\theta}{2}; M-1, \psi) \tag{25}$$

Comparing Eq. 25 to Eq. 22, we note that for $\psi_1 > \psi$, for which $\theta > 1$, we have that $\frac{\partial \psi_{\text{SML}}}{\partial \psi_1} \geq \frac{\partial \psi_{\text{Vo}}}{\partial \psi_1}$, whereas for $\psi_1 < \psi$, for which $\theta < 1$, it follows that $\frac{\partial \psi_{\text{SML}}}{\partial \psi_1} \leq \frac{\partial \psi_{\text{Vo}}}{\partial \psi_1}$. Since at $\psi_1 = \psi$, the two ensemble classifiers coincide, it follows that claim (i) holds (see figure S2 for an illustrative example).

Now we turn to prove claim (ii), that $\psi_{SML} \geq \psi$. To this end, note that when the first algorithm is random, $\pi_1 = \psi_1 = 1/2$, according to Eq. 23 we have $\theta = 0$, and thus, from Eq. 25 it follows that

$$\frac{\partial \psi_{\text{SML}}}{\partial \psi_1}\Big|_{\psi_1=1/2} = 0.$$

Furthermore, for $\psi_1 > 1/2$ this derivative is positive, whereas for $\psi_1 < 1/2$ the derivative is negative. We thus conclude that $\psi_1 = 1/2$ is a *global minima* of ψ_{SML} as a function of ψ_1. Furthermore, when $\psi_1 = 1/2$, the first algorithm has no weight, and the sensitivity of the SML classifier is the same as that of majority voting, based on $M-1$ conditionally independent classifiers, all with balanced accuracy equal to ψ. It thus readily follows that claim (ii) holds.

To finish the proof of the lemma, we now consider claim (iii). First, observe that if $\psi > \psi_1$, then $\psi_{\text{SML}} > \psi_1$. We therefore focus on the case $\psi_1 > \psi$. When $\psi_1 = 1$, $\theta = 1/(2\psi - 1)$ and

$$\psi_{\text{SML}}\Big|_{\psi_1=1} = 1 - F\left(\tfrac{M-1}{2} - \tfrac{1}{2}\tfrac{1}{2\psi-1}; M-1, \psi\right). \tag{26}$$

As discussed in remark (iii) after the lemma, the value of ψ_{SML} can be strictly smaller than one in this case, meaning that SML is not always as good as the best classifier in the ensemble.

However, we now show that if the $M-1$ remaining classifiers have balanced accuracy better than random, then SML has balanced accuracy close to π_1. To prove Eq. 18 of claim (iii), first note that according to Eq. 25, $\partial\psi_{\text{SML}}/\partial\psi_1 \in [0,1]$ for all $\psi_1 \geq \psi$. Hence, as ψ_1 is decreased from a value of 1, ψ_{SML} decreases *slower* than ψ_1 itself. Thus, to prove the claim, it suffices to show that at the extreme case $\psi_1 = 1$,

$$F\left(\tfrac{M-1}{2} - \tfrac{1}{2}\tfrac{1}{2\psi-1}; M-1, \psi\right) \leq \exp(-2\epsilon^2(M-1)).$$

To this end, we apply Hoeffding's inequality for i.i.d. Bernoulli random variables (namely that for a random variable $X \sim Bin(n,\psi)$, $\Pr[X \leq n(\psi-\epsilon)] = F(n(\psi-\epsilon); n, \psi) \leq \exp(-2\epsilon^2 n)$). In our case, $n = M-1$, and comparing

$$\frac{M-1}{2} - \frac{1}{2}\frac{1}{2\psi-1} = (M-1)(\psi-\epsilon) \tag{27}$$

gives $\epsilon = \left(1 + (M-1)(2\psi-1)^2\right) / \left(2(M-1)(2\psi-1)\right)$. Plugging this into Hoeffding's inequality concludes the proof. □.

5 Covariance between classifiers in presence of a cartel

Proof of Lemma 2. As in the proof of Lemma 1, for each classifier f_i we first compute its mean and variance, $\mu_i = \mathbb{E}[f_i(X)]$ and $Var[f_i(X)]$, respectively. We then use these results to compute the entries of the population covariance matrix, $q_{ij} = \mathbb{E}[(f_i(X) - \mu_i) \cdot (f_j(X) - \mu_j)]$.

The mean and variance of honest classifiers with indices $i \in P$ have already been computed in the proof of Lemma 1. We now consider the mean and variance of classifiers $i \in C$ that belong to the cartel. For brevity, we denote by ψ_c, η_c and π_c the specificity, sensitivity and balanced accuracy of the cartel target with respect to the ground truth,

$$\psi_c = \Pr[T=1|Y=1], \quad \eta_c = \Pr[T=-1|Y=-1], \quad \pi_c = \tfrac{1}{2}(\psi_c + \eta_c).$$

Furthermore, for each $i \in C$, we denote by p_i and n_i its specificity and sensitivity w.r.t. the cartel target,

$$p_i = \Pr[f_i(X)=1|T=1], \qquad n_i = \Pr[f_i(X)=-1|T=-1]. \tag{28}$$

Under the assumption of independence between instances, the mean of a cartel member with $i \in C$ is

$$\begin{array}{rcl} \mathbb{E}\left[f_i(X)\right] & = & \Pr[f_i(X)=1] - \Pr[f_i(X)=-1] \\ & = & \sum_{t,y\in\{-1,1\}} \Pr[f_i(X)=1|T=t]\Pr[T=t|Y=y]\Pr[Y=y] \\ & & -\sum_{t,y\in\{-1,1\}} \Pr[f_i(X)=-1|T=t]\Pr[T=t|Y=y]\Pr[Y=y] \end{array}$$

which, after simple algebraic manipulations, simplifies to

$$\mathbb{E}\left[f_i(X)\right] = b(1-\psi_c-\eta_c+n_i(\psi_c+\eta_c-1)+p_i(\psi_c+\eta_c-1)) + n_i(\psi_c-\eta_c-1) + p_i(\psi_c-\eta_c+1) + \eta_c - \psi_c \tag{29}$$

Similarly, as in Lemma 1, the population variance of the i-th classifier is

$$\text{Var}\left[f_i(X)\right] = \mathbb{E}\left[f_i(X)^2\right] - \mathbb{E}\left[f_i(X)\right]^2 = 1 - \mathbb{E}\left[f_i(X)\right]^2. \tag{30}$$

Next, we compute $\mathbb{E}[f_i(X) \cdot f_j(X)]$. The case $i,j \in P$ was already considered in the proof of Lemma 1, whereas the case $i,j \in C$ can be deduced from it, with the truth replaced by the cartel's target T. Thus,

$$\mathbb{E}[f_i(X) \cdot f_j(X)] = \begin{cases} (2\pi_i - 1)(2\pi_j - 1)(1-b^2) & i \neq j,\ i \in P, j \in P \\ (2\xi_i - 1)(2\xi_j - 1)(1-b^2) & i \neq j,\ i \in C, j \in C \end{cases} \tag{31}$$

It thus remains to compute $\mathbb{E}[f_i(X) \cdot f_j(X)]$ for the mixed case with $i \in P$ and $j \in C$. Under the assumption of independence of errors between different instances and between different classifiers,

$$\begin{array}{rcl} \mathbb{E}[f_i(X) \cdot f_j(X)] & = & \Pr[f_i(X) = f_j(X)] - \Pr[f_i(X) = -f_j(X)] \\ & = & ((2\psi_i - 1)((1-2n_j)(1-\psi_c) - (1-2p_j)\psi_c))(1+b)/2 \\ & & +((2\eta_i - 1)((1-2p_j)(1-\eta_c) - (1-2n_j)f))(1-b)/2 \end{array}$$

Combining the three equations above yields that for $i \in P, j \in C$

$$\begin{aligned}\mathbb{E}[f_i(X) \cdot f_j(X)] - \mathbb{E}[f_i(X)] \cdot \mathbb{E}[f_j(X)] &= (1-b^2)(\psi_i + \eta_i - 1)(\psi_c + \eta_c - 1)(n_j + p_j - 1) \\ &= (1-b^2)(2\pi_i - 1)(2\pi_c - 1)(2\xi_j - 1)\end{aligned}$$

Thus, the entries q_{ij} of the $M \times M$ covariance matrix between the M classifiers are

$$q_{ij} = \begin{cases} 1 - \mu_i^2 & i = j \\ (2\pi_i - 1)(2\pi_j - 1)\left(1 - b^2\right) & i \neq j, i \in P, j \in P \\ (2\pi_i - 1)(2\pi_c - 1)(2\xi_j - 1)\left(1 - b^2\right) & i \in P, j \in C \\ (2\xi_i - 1)(2\xi_j - 1)\left(1 - b^2\right) & i \neq j, i \in C, j \in C \end{cases} \tag{32}$$

□.

6 Matrix rank and leading eigenvectors in presence of a cartel

Proof of Theorem 1. To simplify notation, we make the following convenient change of variables:

$$\rho_i = 2\pi_i - 1, \quad \tau_i = 2\xi_i - 1, \quad u = (1 - b^2), \quad \text{and} \quad \rho_c = 2\pi_c - 1$$

where π_c is the balanced accuracy of the cartel with respect to the truth. In this notation, for indices $i \in P, j \in C$ as an example, we have the compact representation $q_{ij} = u\rho_i\tau_j\rho_c$.

Our proof of the theorem is constructive: we explicitly construct $\lambda_1, \lambda_2 \in \mathbb{R}$ and two orthonormal vectors $\mathbf{e}_1, \mathbf{e}_2 \in \mathbb{R}^M$ such that for all $i \neq j$

$$q_{ij} = \lambda_1 e_{1i} e_{1j} + \lambda_2 e_{2i} e_{2j}. \tag{33}$$

Furthermore, as we prove below, these eigenvectors have in fact the following specific form

$$\sqrt{\lambda_1} e_{1i} = \begin{cases} \sqrt{u} \cdot a_{11}\rho_i & i \in P \\ \sqrt{u} \cdot a_{21}\tau_i & i \in C \end{cases} \qquad \sqrt{\lambda_2} e_{2i} = \begin{cases} \sqrt{u} \cdot a_{12}\rho_i & i \in P \\ \sqrt{u} \cdot a_{22}\tau_i & i \in C \end{cases} \tag{34}$$

where $a_{11}, a_{12}, a_{21}, a_{22}$ are scalars yet to be determined.

The requirement that the eigenvectors $\mathbf{e}_1, \mathbf{e}_2$ are orthogonal, namely that $\sum_i e_{1i}e_{2i} = 0$, implies that

$$a_{11}a_{12} \sum_{i \in P} \rho_i^2 + a_{21}a_{22} \sum_{j \in C} \tau_j^2 = 0. \tag{35}$$

Next, comparing the exact values of q_{ij}, Eq. 32, with our assumed form above gives the following set of equations,

$$\begin{cases} u\rho_i\rho_j = u \cdot a_{11}^2\rho_i\rho_j + u \cdot a_{12}^2\rho_i\rho_j & i \in P, j \in P \\ u\rho_i\rho_c\tau_j = u \cdot a_{11}a_{21}\rho_i\tau_j + u \cdot a_{12}\rho_i \cdot a_{22}\tau_j & i \in P, j \in C \\ u\tau_i\tau_j = u \cdot a_{21}\tau_i \cdot a_{21}\tau_j + u \cdot a_{22}\tau_i \cdot a_{22}\tau_j & i \in C, j \in C \end{cases} \tag{36}$$

Hence, for Eq. 33 to hold, a_{11}, a_{12}, a_{21} and a_{22} should satisfy the following set of equations

$$\begin{cases} a_{11}^2 + a_{12}^2 &= 1 \\ a_{11}a_{21} + a_{12}a_{22} &= \rho_c \\ a_{21}^2 + a_{22}^2 &= 1 \\ (a_{11}a_{12})/(a_{21}a_{22}) &= -\left(\sum_{j \in C} \tau_j^2\right)/\left(\sum_{i \in P} \rho_i^2\right) = -\lambda_C/\lambda_P \end{cases}, \tag{37}$$

where $\lambda_C = (1 - b^2)\sum_{i \in C}(2\xi_i - 1)^2$ and $\lambda_P = (1 - b^2)\sum_{i \in P}(2\pi_i - 1)^2$.

We now show that this set of equations indeed has a unique solution, up to the trivial sign ambiguities in the definition of the two eigenvectors. To this end, note that the following change of variables, $a_{11} = \cos\alpha, a_{12} = \sin\alpha, a_{21} = \sin\beta$, and $a_{22} = \cos\beta$, reduces the system in Eq. 37 to

$$\begin{cases} \sin(\alpha + \beta) &= k_1 \\ \sin(2\alpha) / \sin(2\beta) &= -k_2 \end{cases} \tag{38}$$

where $k_1 = \rho_c$ and $k_2 = \lambda_C/\lambda_P$.

To solve this system, note that standard trigonometric equalities applied to the first equation above give that

$$\sin 2(\alpha + \beta) = 2k_1\sqrt{1 - k_1^2} \quad \text{and} \quad \cos 2(\alpha + \beta) = 1 - 2k_1^2. \tag{39}$$

Next, rewrite the second equation as $\sin(2(\alpha + \beta) - 2\beta) + k_2 \sin(2\beta) = 0$, and expand the first term. This gives

$$\sin(2\alpha) + k_2 \sin 2(\alpha + \beta)\cos(2\alpha) - k_2 \cos(2(\alpha + \beta))\sin(2\alpha) = 0$$

or equivalently,

$$\tan(2\alpha) = -\frac{k_2 \sin(2(\alpha+\beta))}{1 - k_2 \cos(2(\alpha+\beta))}$$

Combining this with Eq. 39 gives

$$\alpha = \frac{1}{2} \arctan\left(\frac{k_1 k_2}{k_2(1-2k_1^2)-1}\right).$$

Similarly, writing the second equation as $\sin(2(\alpha+\beta)-2\beta) + k_2 \sin(2\beta) = 0$ and expanding gives

$$\tan(2\beta) = -\frac{\sin(2\delta)}{k_2 - \cos(2\delta)}$$

whose solution is

$$\beta = \frac{1}{2} \arctan\left(\frac{2k_1\sqrt{1-k_1^2}}{1-k_2-2k_1^2}\right).$$

Consistent with the sign ambiguity of the eigenvectors, these solutions for α and β are unique up to a rotation with periodicity $\frac{\pi}{2}$. The expressions for the eigenvectors and their respective eigenvalues readily follow by back-substitution into Eq. 34. □

7 Simulations and benchmarks

The following section describes how we generated the simulated data and how we performed the benchmarks. For each component of the simulation we also provide pseudo-code.

7.1 Simulated data: Ensembles of statistically independent predictions

We generated ensembles of statistically independent predictions using the random detector with fixed balanced accuracy (RDFBA) algorithm [4]. A generic RDFBA predictor with pre-determined empirical balanced accuracy π on a test set with T samples, is denoted as RDFBA(π). Given a test data with T samples, a collection of RDFBAs is constructed such that any two classifiers are conditionally independent and such that their *empirical* balanced accuracy on the test data is equal to π. Note that two RDFBAs with the same balanced accuracy π may nonetheless have different sensitivity ψ and specificity η.

To briefly describe the construction we use the following standard notation: Let P be the number of positives, i.e. the number of instances whose true class label is $+1$; N is the number of negatives, where $T = \mathrm{P} + \mathrm{N}$; FP is the number of false positives, i.e. the number of negatives that have been mistakenly predicted as positives; FN is the number of false negatives. An RDFBA(π) classifier is constructed from the ground truth vector y as follows:

1. Initialize the entries of the prediction vector $f(x)$ with the corresponding entries in the ground truth $\mathbf{y}$.
2. Under the constraint that $\mathrm{FN} = (2 - 2\pi - \mathrm{FP}/\mathrm{N}) \cdot \mathrm{P}$ is an integer, draw a random integer FP with uniform probability from $[0, \mathrm{N}]$.
3. FP randomly chosen instances in $f(x)$, whose true label is -1, are assigned the wrong class label, $+1$.
4. FN randomly chosen instances in $f(x)$, whose true label is $+1$, are assigned the wrong class label, -1.

In our simulations, we used $\pi \sim \mathbb{U}(0.3, 0.8)$ and a total of $T = 10000$ samples, from which we randomly sampled 300 positive and 300 negative instances, to form our test data D of size $S = 600$ samples. Hence, the empirical balanced accuracy of the RDFBA classifiers on the test data D may be slighly higher than 0.8 or lower than 0.3.

7.2 Simulated data: Ensembles of independent predictors with one cartel present

To generate datasets of conditionally independent predictors which include a cartel with $r \cdot M$ predictors, we applied the following steps: First, we generated an ensemble P of $(1-r)M$ independent predictions as described above for the ground truth vector $\mathbf{y}$. Then, using another RDFBA predictor, we constructed the cartel's target vector $\mathbf{c}$, such that it had an empirical balanced accuracy π_c with respect to the ground truth. Next, using this vector $\mathbf{c}$ we constructed an ensemble C of independent predictions, as in the procedure described above, with the only difference that the balanced accuracies of all members of the cartel relative to the cartel's target were set to be equal to 0.7. The dataset is obtained by the union of the two ensembles of predictions, P and C. In our simulations we used $\pi_c = 0.5$ thus obtaining a cartel's target that is orthogonal to the ground truth.

7.3 Real data: Ensembles of predictions from standard machine-learning classifiers

To generate ensemble of predictions from standard machine-learning classifiers on real data, we trained the classifiers on partially overlapping training data and collected their predictions obtained on the same test data, which was independent from all the training data. In detail, from each dataset we sampled 600 instances (or all the instances if less than 600 were available), half of which (up to 300) were used for testing. Independently for each classifier, we selected a random subset comprising of 90% of the instances reserved for training and used this subset as a "private" training set. The purpose of this procedure was to produce training data that was slightly different between the different classifiers, while at the same time allowing to have a significantly large number of training samples even in the smaller datasets. We chose to use at most 600 instances to reduce computational time. To determine the empirical distribution of performances of each classifier and of the ensemble approaches discussed in the manuscript, for each dataset we repeated this procedure 1000 times, unless otherwise specified in the figure caption.

7.4 Custom Real datasets: Ensembles of predictions from standard machine-learning classifiers

To generate ensemble of predictions from standard machine-learning classifiers on custom datasets from big-data repositories, we trained the classifiers on non-overlapping training data and collected their predictions obtained on the same testing data, which was independent from all the training data. In detail, from each dataset we sampled 50,000 instances (or half of the instances if less than 50,000 were available), and, independently for each classifier, we selected a random subset comprising of 500 instances for training. The purpose of this procedure was to produce training data that had the potential to be markedly different between the different classifiers. We chose to use at most 500 instances to reduce the computational time and memory usage required for training. To determine the empirical distribution of performances of each classifier and of the ensemble approaches discussed in the manuscript, for each dataset we repeated this procedure 30 times, unless otherwise specified in the figure caption.

8 Custom datasets

In addition to eight standard machine learning datasets from the UCI repository, which are described in the first part of Table S1, we created nine additional datasets from publicly available data in the fields of economics, sociology, geography, semantics, ecology, and finance (see second part of this table).

As these datasets are not readily available, we provide scripts to generate the corresponding matrices of features and class labels. These matrices can be used to train the set of 33 standard machine learning algorithms described in Table S2 and subsequently apply the SML and iMLE approaches described in the main text.

The scripts are available at http://sourceforge.net/projects/klugerlab/files/SML_customdatasets

8.1 ACS

This dataset was constructed from surveys conducted by the American Community Survey in 2009. The data provides information about a geographical area, including education levels, household income, demographics, household size, gender statistics and age groups. The classification task was to predict the geographical location of an area based on sociological and economical parameters of the region. The class label was equal to 1 if the center of the geographical unit had a decimal latitude above 39.09916, which corresponds to the latitude of the 16^{th} Circuit Court of Jackson County in Missouri, USA.

8.2 AMEX

The dataset was constructed from the daily opening, closing, high and low prices, as well as traded volumes, for stocks at the American Stock Exchange between 1970 and 2010. For each stock, we divided the time series into segments of 10 days. The task was to identify whether the highest price at the tenth day had a 5% increase over the highest price at the ninth day, using only information from day 1 to day 9. A class label of one indicated that

$$\frac{\text{high}_{day10} - \text{high}_{day9}}{\text{high}_{day9}} > 1.05$$

8.3 ENRON

This dataset was constructed based on the email exchanges from employees at ENRON. The collection contains emails from about 150 users, mostly senior management of Enron, made public and posted to the web by the Federal Energy Regulatory Commission during its investigation. For each email we constructed a feature space corresponding to the histogram of occurrences of manually selected keywords. The task was to predict whether

an email included email addresses from a domain that is different from enron.com. A class label of 1 indicated that at least one of the addresses in the To, CC or BCC fields of the email contained a different domain than enron.com

8.4 GEO

The dataset was constructed from Sea-viewing Wide Field-of-view Sensor (SeaWiFS) data on the Indicators of Coastal Water Quality Collection, originally collected to determine concentrations of chlorophyll-a in the coastal water. The data consists of gridded satellite measurements of chlorophyll-a concentrations (in nanogram/cubic meter) in a band extending between 10 and 100 km from the shoreline [14]. The grids are annual composites at a resolution of 5 arc-minutes (approximately 9 x 9 km at the equator). The gridding was done by the Columbia University Center for International Earth Science Information Network (CIESIN). In our dataset, features correspond to measurements from previous years for the same geographical unit, as well as convolution of yearly measurements, the latest being in 2007, using different random kernels of increasing size. The task was to predict whether coastal chlorophyll-a increased in 2008 relative to 2007. A class label of 1 indicated that chlorophyll-a indeed increased in 2008 relative to 2007.

8.5 LASTFM

The dataset was constructed from tags assigned by listeners to songs broadcast by the online service Last.FM in 2007 (http://musicmachinery.com/2010/11/10/lastfm-artisttags2007/). We selected the most common 995 tags in the entire dataset and described each song as the histogram of counts for these 995 tags. The task was to identify whether a song was ever tagged, at least once, with a tag containing the word "favorite". A class label of 1 indicated that the song had at least one user assigning a tag containing the word "favorite".

8.6 NASDAQ

The dataset was constructed from the daily opening, closing, high and low prices, as well as traded volumes, for stocks at the National Association of Securities Dealers Automated Quotations Stock Exchange between 1970 and 2010. For each stock, we divided the time series into segments of 10 days. The task was to identify whether the opening price at the tenth day was higher than the closing price at the ninth day, using only information from day 1 to day 9. A class label of one indicated that

$$\text{open}_{day10} > \text{close}_{day9}$$

8.7 NYSE

The dataset was constructed from the daily opening, closing, high and low prices, as well as traded volumes, for stocks at the New York Stock Exchange between 1970 and 2010. For each stock, we divided the time series into segments of 10 days. The task was to identify whether the highest price at the tenth day had a 5% increase over the highest price at the ninth day, using only information from day 1 to day 9. A class label of one indicated that

$$\frac{\text{high}_{day10} - \text{high}_{day9}}{\text{high}_{day9}} > 1.05$$

8.8 PNS

The dataset was constructed from a list of common place names. The task was to determine whether the first letter of a place is a vowel, excluding the letter y, based on the histogram of the letters composing the rest of the place name. A class label of 1 indicated that the letter was a vowel. PNS is an acronym for Place Name Strings.

8.9 SP500

The dataset was constructed from the daily opening, closing, high and low prices, as well as traded volumes, for S&P 500 stocks. For each stock, we divided the time series into segments of 8 days. The task was to identify whether the opening price at the eighth day had an increase over the closing price at the seventh day, using only information from day 1 to day 7. A class label of one indicated that

$$\text{open}_{day8} > \text{close}_{day7}$$

9 Supplemental Tables

Table S1: **Summary of the datasets.**

Datasets from the UCI repository [5]

Dataset	**Instances**	**Features**	**Class**	**Reference**
AD (Abalone data)	4,177	8	male/female	[12]
ID (Ionosphere data)	351	34	good return/bad return	[10]
MGT (MAGIC Gamma Telescope)	19,020	11	signal/background	[9]
MM (Mammographic masses)	961	6	disease severity (2 classes)	[11]
PD (Parkinson data)	197	23	affected/unaffected	[8]
SD (Spambase data)	4,601	57	spam/not spam	[5]
WBC (Wisconsin breast cancer data)	699	10	benign/malignant	[7]
YBC (Yale breast cancer data)	650	6	nodal status	[6]

Custom datasets

Dataset	**Instances**	**Features**	**Field**	**Reference**
ACS	321,583	53	sociology/economy/geography	[15]
AMEX	190,769	45	finance	[16]
ENRON	517,424	64	text analysis	[17]
GEO	494,268	45	ecology/geography	[14]
LASTFM	20,908	995	recommendation systems	[18]
NASDAQ	847,427	45	finance	[19]
NYSE	919,792	45	finance	[20]
PNS	10,196	27	text analysis	[21]
SP500	15,028	35	finance	[22]

Table S2: **Summary of the machine learning classifiers from Weka [13].**

classifier/meta-learner	**Weka class**	**Description**
KNN (k=1, odd)	lazy/IBk	k-nearest neighbor classifier with k=1
KNN (k=2, even)	lazy/IBk	k-nearest neighbor classifier with k=2
KNN (k=5)	lazy/IBk	k-nearest neighbor classifier with k=5
k-Star	lazy/KStar	Instance-based learner using entropy-based distance
DecisionStump	trees/DecisionStump	One-level decision tree
J48	trees/J48	Decision tree with pruning
REPTree	trees/REPTree	Decision tree using information gain
JRip	Rules/JRip	Propositional rule learner
LMT	trees/LMT	Logistic model trees
LWL	lazy/LWL	Locally weighted learning algorithm
Logistic regression	functions/SimpleLogistic	Logistic regression
Regularized Logistic regression	functions/Logistic	Regularized logistic regression
Sequential Minimal Optimization	function/SMO	Sequential minimal optimization for SVM
NaiveBayes	bayes/NaiveBayes	Naïve Bayes classifier
M5P	rules/M5P	M5 Model trees and rules
OneR	rules/OneR	Minimum-error attribute classifier
PART	rules/PART	Partial decision trees classifier
RandomForest (n=10 trees)	trees/RandomForest	Random Forest classifier with n=10
RandomForest (n=20 trees)	trees/RandomForest	Random Forest classifier with n=20
Multilayer Perceptron	functions/MultilayerPerceptron	Multilayer neural network using backpropagation
Voted Perceptron	functions/VotedPerceptron	Voted perceptron classifier
SGD	functions/SGD	Stochastic gradient descent
Voting	meta/Vote	Majority voting of an ensemble of J48 classifiers
Stacking	meta/Stacking	Stacking of an ensemble of J48 classifiers
AdaBoost + NaiveBayes	meta/AdaBoostM1	AdaBoost of an ensemble of Naïve Bayes classifiers
AdaBoost + Logistic Regression	meta/AdaBoostM1	AdaBoost of an ensemble of logistic regressions
AdaBoost + J48	meta/AdaBoostM	AdaBoost of an ensemble of J48 classifiers
Bagging + REPTree	meta/Bagging	Bagging of an ensemble of REPTrees
Bagging + RandomTree	meta/Bagging	Bagging of an ensemble of Random Trees
Bagging + RandomForest	meta/Bagging	Bagging of an ensemble of Random Forests
LogitBoost + ZeroR	meta/LogitBoost	ZeroR classifiers use the mode as prediction
LogitBoost + KNN	meta/LogitBoost	LogitBoost of an ensemble of KNN classifiers
LogitBoost + DecisionStump	meta/LogitBoost	LogitBoost of an ensemble of Decision Stumps

Table S3: **Characteristics of the ensemble of predictions for the real world datasets.** The deviation from the assumption of conditional independence is expressed as the absolute value of the difference Δ between the two sides of Eq. 3 in the main text. With the exception of the median true rank of the best inferred predictor, noted as r_{best}, all quantities are averages over all runs.

Dataset	$\lvert\Delta\rvert$	$\lambda_1/\sum\lambda$ (%)	$\lambda_2/\sum\lambda$ (%)	r_{best}
ACS	0.0067	70.0	9.0	1
AD	0.0082	37.4	12.8	8
AMEX	0.0001	59.0	14.2	1
ENRON	0.0035	47.2	12.7	2
GEO	0.0034	37.1	15.7	2
ID	0.0158	75.9	4.2	2
LASTFM	0.0063	78.1	5.4	3
MGT	0.0107	70.4	6.3	4
MM	0.0138	80.4	5.2	3
NASDAQ	0.0015	50.4	13	1
NYSE	0.0001	57.3	17.3	2
PD	0.0101	51.4	9.8	5
PNS	0.0150	57.9	23.1	2
SD	0.0059	79.8	3.5	2
SP500	0.0016	38.8	15.7	3
WBC	0.0033	92.8	1.4	2
YBC	0.0209	50.0	10.2	4

10 Supplemental Figures

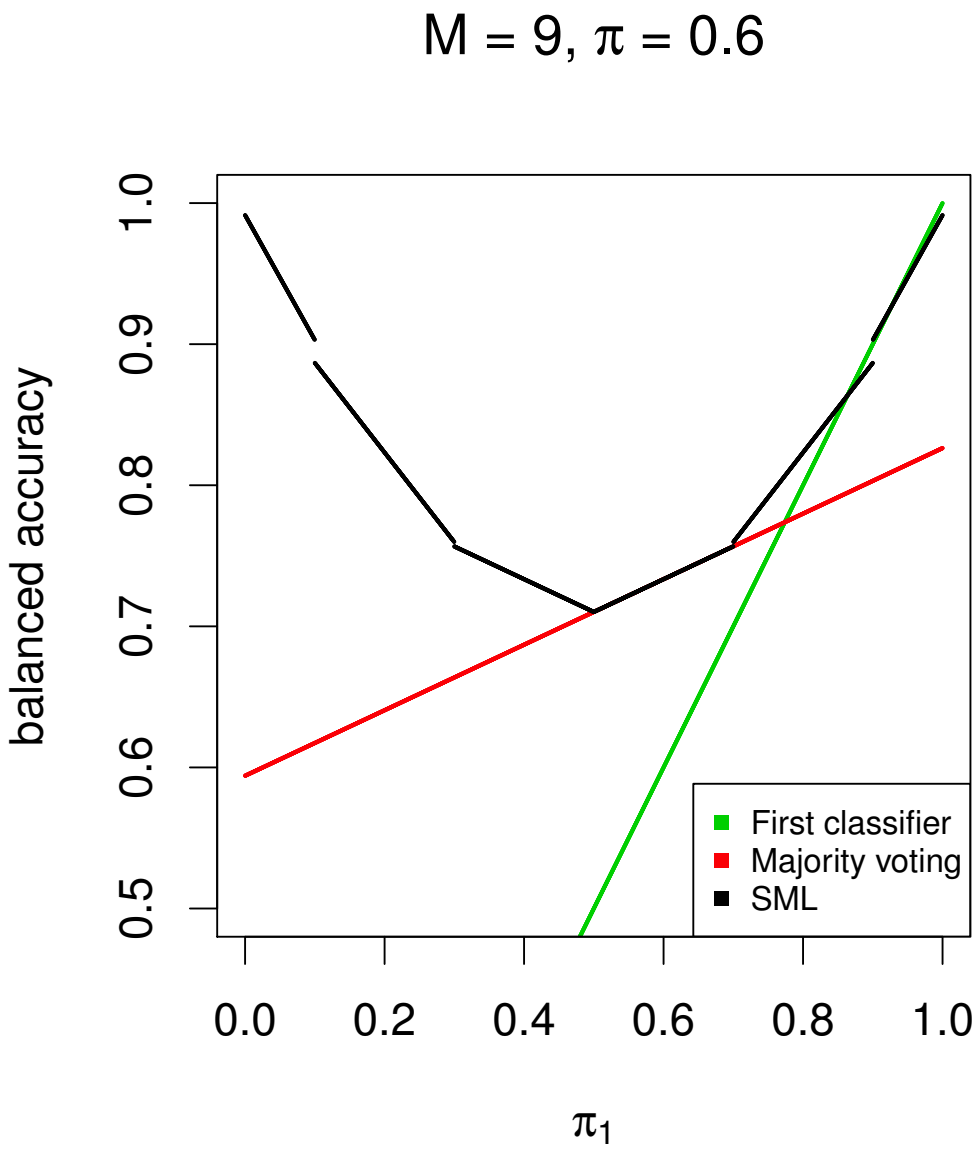

Figure S2: Performance of the first algorithm, majority voting and SML as a function of the balanced accuracy of the first algorithm when all other algorithms in the ensemble have identical sensitivities $\pi = 0.6$. In this illustrative example, $M = 9$. The performance of majority voting (in red) changes linearly with π_1, albeit with partial derivative smaller than 1. The balanced accuracy of SML (in black) is a piecewise linear function of π_1. The jumps in the balanced accuracy of SML occur when the value $(M-1)/2 - 1/2 \cdot 1/(2\pi - 1)$ is an integer.

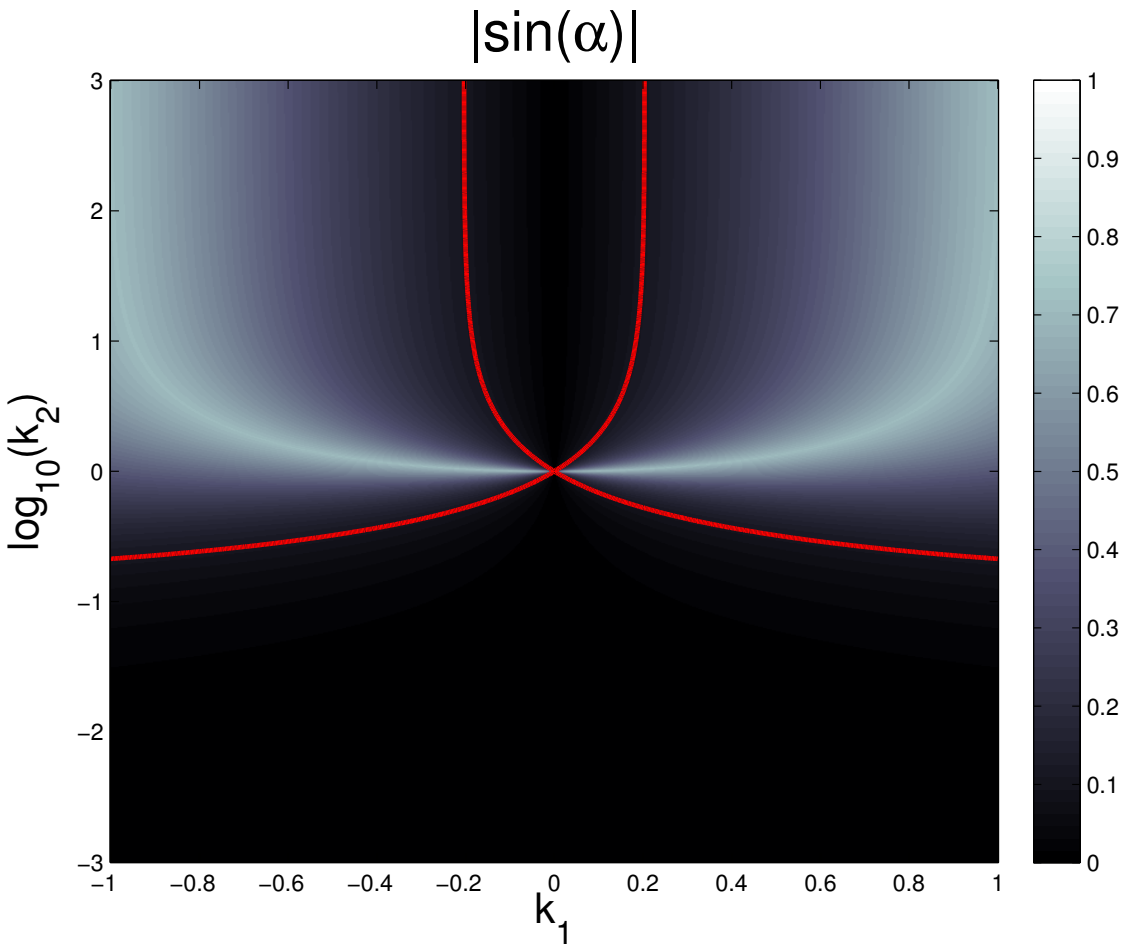

Figure S3: The heatmap shows the absolute value of the angle between the truth and the eigenvector e_1, on which the SML prediction is based. The dark area between the two red lines graphically shows the relationship between k_1 and k_2 such that $|\alpha| \leq 6°$. The figure shows that SML is robust to cartels: when $\alpha \approx 0$, the honest classifiers lie approximatively on the eigenvector e_1.

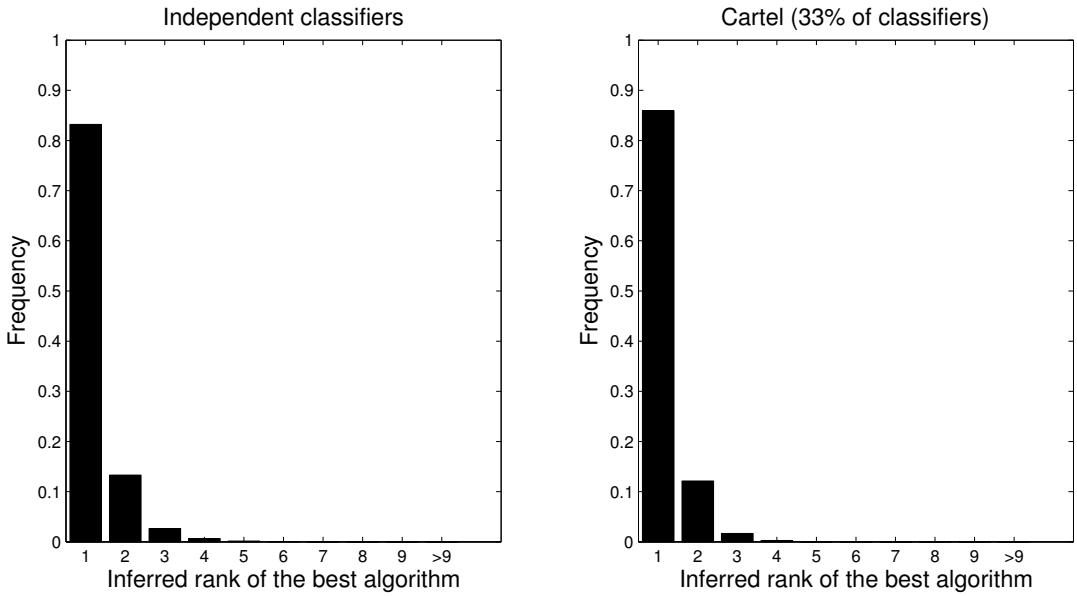

Figure S4: The largest entry in the leading eigenvector often corresponds to the best classifier in the ensemble. In the plots, each bar represents the empirical probability that the entry in the leading eigenvector corresponding to best classifier attained a specific rank.

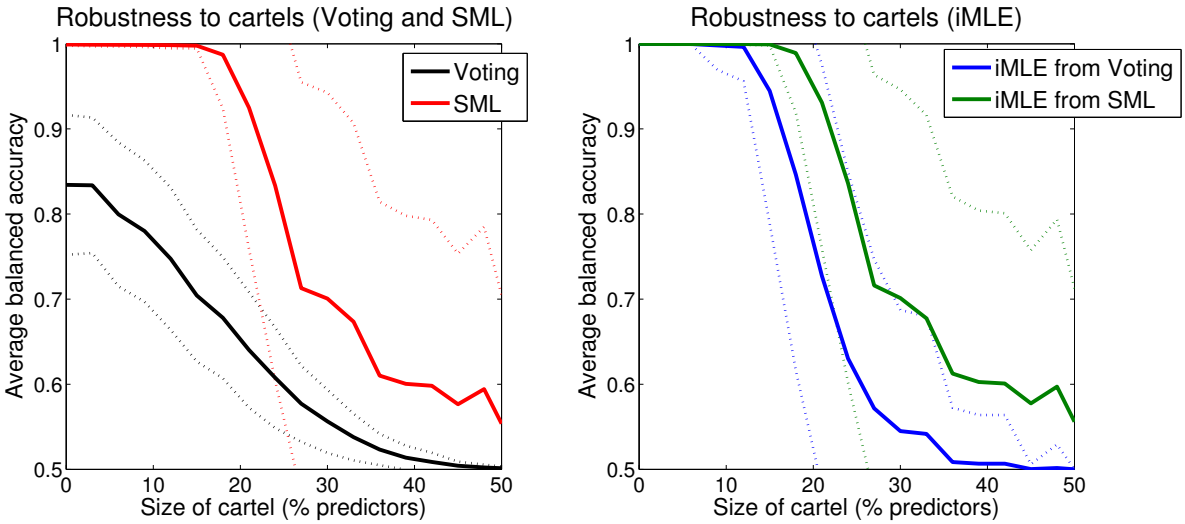

Figure S5: SML is more robust to cartels than majority voting (left panel). iMLE using SML estimates as starting point is also more robust to cartels than iMLE using majority voting as the starting condition (right panel). For each meta-learner prediction the average balanced accuracy is shown (filled lines) together with the standard error (dotted lines, n=500 runs for each cartel's fraction).

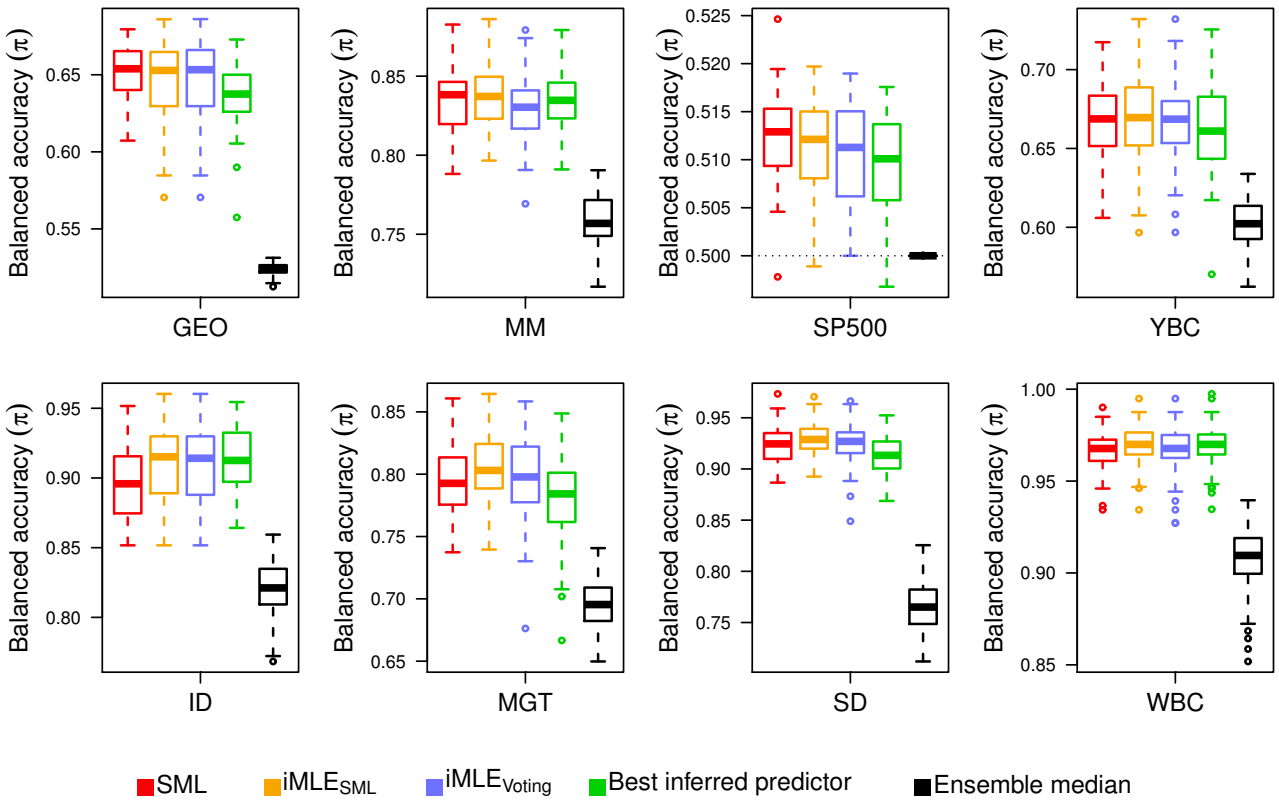

Figure S6: Comparison of several classifiers on real-world datasets where our conditions are nearly satisfied. The median balanced accuracy of all classifiers in the ensemble is shown in black.

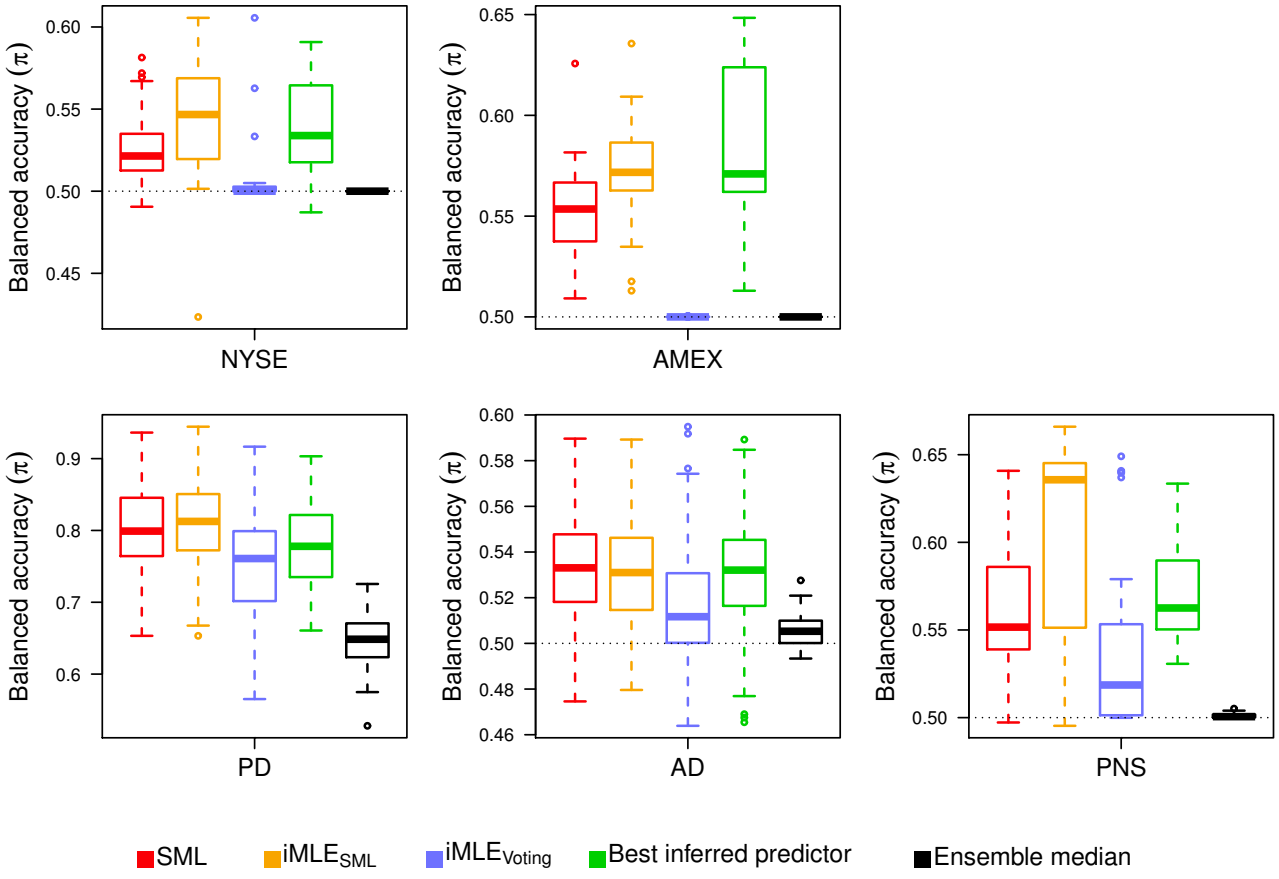

Figure S7: Comparison of several classifiers on real-world datasets, where predictors have structure similar to that of cartels. The median balanced accuracy of all classifiers in the ensemble is shown in black.

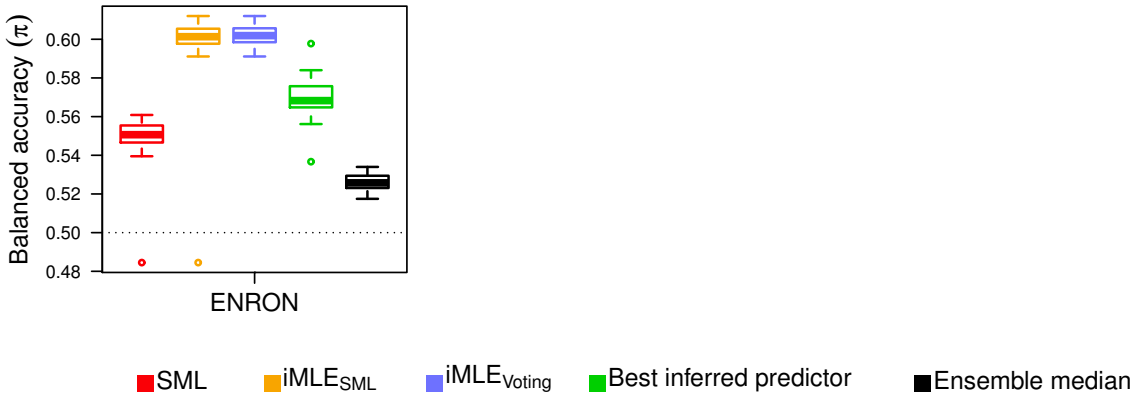

Figure S8: Comparison of several classifiers on the ENRON dataset, characterized by a sparse feature space. The median balanced accuracy of all classifiers in the ensemble is shown in black.

References

[1] Candes, E. J & Recht, B. (2009) Exact matrix completion via convex optimization, *Foundations of Computational Mathematics* **9**, 717–772.

[2] Karger, D. R, Oh, S, & Shah, D. (2011) Budget-optimal crowdsourcing using low-rank matrix approximations. *Proc. of the IEEE Allerton Conf. on Communication, Control, and Computing*, pp. 284–291.

[3] Kato, T. (1995). *Perturbation theory for linear operators*, 2nd edition, Springer-Verlag.

[4] F. Strino, F. Parisi, and Y. Kluger. VDA, a method of choosing a better classifier with fewer validations. *PLoS ONE*, 6(10):e26074, 2011.

[5] A. Frank and A. Asuncion. UCI machine learning repository. Irvine, CA: University of California,

[6] F. Parisi, A. M. González, Y. Nadler, R. L. Camp, D. L. Rimm, H. M. Kluger, and Y. Kluger. Benefits of biomarker selection and clinico-pathological covariate inclusion in breast cancer prognostic models. *Breast Cancer Res*, 12(5):R66, Sep 2010.

[7] W. H. Wolberg and O. L. Mangasarian. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proc Natl Acad Sci U S A*, 87(23):9193–6, Dec 1990.

[8] M. A. Little, P. E. McSharry, S. J. Roberts, D. A. E. Costello, and I. M. Moroz. Exploiting nonlinear recurrence and fractal scaling properties voice disorder detection. *Biomed Eng Online*, 6:23, 2007.

[9] D. Heck, J. Knapp, J. Capdevielle, G. Schatz, and T. Thouw. Report FZKA 6019, Forschungszentrum Karlsruhe, 1998. Technical report, 1986.

[10] V. Sigillito, S. Wing, L. Hutton, and K. Baker. Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Technical Digest*, 10(3):262–266, 1989.

[11] M. Elter, R. Schulz-Wendtland, and T. Wittenberg. The classification of breast cancer biopsy outcomes using two approaches that both emphasize an intelligible decision process. *Med Phys*, 34(11):4164–72, Nov 2007.

[12] P. McShane and J. Reyn. Small-scale spatial variation in growth, size at maturity, and yield-and egg-per-recruit relations in the new zealand Abalone Haliotis *New Zealand Journal of Marine and Freshwater Research*, 29(4):603–612, 1995.

[13] I. Witten, E. Frank, and M. Hall. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2011.

[14] Goddard Space Flight Center (GSFC), and Center for International Earth Science Information Network (CIESIN)/Columbia University. *Indicators of Coastal Water Quality: Annual Chlorophyll-a Concentration 1998-2007* Palisades, NY: NASA Socioeconomic Data and Applications Center (SEDAC). `http://sedac.ciesin.columbia.edu/data/set/icwq-annual-chlorophyll-a-concentration-1998-2007`

[15] Data prepared by Infochimps *US Census (ACS): Income, Age, Housing and Population by Location (2009)* `http://www.infochimps.com/datasets/us-census-acs-income-age-housing-and-population-by-location`

[16] Data prepared by Infochimps *AMEX Daily 1970-2010 Open, Close, High, Low and Volume* `http://www.infochimps.com/datasets/amex-exchange-daily-1970-2010-open-close-high-low-and-volume`

[17] William W. Cohen *Enron Email Dataset* `http://www.cs.cmu.edu/~enron/`

[18] Paul Lamere The LastFM-ArtistTags2007 Data set `http://static.echonest.com/Lastfm-ArtistTags2007.tar.gz`

[19] Data prepared by Infochimps *NASDAQ Exchange Daily 1970-2010 Open, Close, High, Low and Volume* `http://www.infochimps.com/datasets/nasdaq-exchange-daily-1970-2010-open-close-high-low-and-volume`

[20] Data prepared by Infochimps *NYSE Daily 1970-2010 Open, Close, High, Low and Volume* `http://www.infochimps.com/datasets/nyse-daily-1970-2010-open-close-high-low-and-volume`

[21] Data prepared by Infochimps *Word List - 10,000+ Common Place Names* `http://www.infochimps.com/datasets/word-list-10000-common-place-names`

[22] Data prepared by StockWiz, *Historical Data for S&P 500 Stocks* `http://pages.swcp.com/stocks/`

[23] M. Micsinai, F. Parisi, F. Strino, P. Asp, B.D. Dynlacht and Y. Kluger, Picking ChIP-seq peak detectors for analyzing chromatin modification experiments. *Nucleic Acid Research*, 2012 May;40(9):e70. doi: 10.1093/nar/gks048