Mach Learn (2017) 106:863–886
DOI 10.1007/s10994-016-5616-2

High-probability minimax probability machines

Simon Cousins[1] · John Shawe-Taylor[1]

Received: 29 April 2016 / Accepted: 18 November 2016 / Published online: 2 February 2017

Abstract In this paper we focus on constructing binary classifiers that are built on the premise of minimising an upper bound on their future misclassification rate. We pay particular attention to the approach taken by the minimax probability machine (Lanckriet et al. in J Mach Learn Res 3:555–582, 2003), which directly minimises an upper bound on the future misclassification rate in a worst-case setting: that is, under all possible choices of class-conditional distributions with a given mean and covariance matrix. The validity of these bounds rests on the assumption that the means and covariance matrices are known in advance, however this is not always the case in practice and their empirical counterparts have to be used instead. This can result in erroneous upper bounds on the future misclassification rate and lead to the formulation of sub-optimal predictors. In this paper we address this oversight and study the influence that uncertainty in the moments, the mean and covariance matrix, has on the construction of predictors under the minimax principle. By using high-probability upper bounds on the deviation between true moments and their empirical counterparts, we can re-formulate the minimax optimisation to incorporate this uncertainty and find the predictor that minimises the *high-probability*, worst-case misclassification rate. The moment uncertainty introduces a natural regularisation component into the optimisation, where each class is regularised in proportion to the degree of moment uncertainty. Experimental results would support the view that in the case of with limited data availability, the incorporation of moment uncertainty can lead to the formation of better predictors.

Keywords Classification · Regularisation · Minimax program

Editors: Nathalie Japkowicz and Stan Matwin.

✉ Simon Cousins
s.cousins@cs.ucl.ac.uk

[1] Department of Computer Science, University College London, Gower Street, London WC1E 6BT, UK

1 Introduction

In this paper we examine the problem of constructing classifiers that are built to minimise upper bounds on the future misclassification rate of a predictor. This is a fundamental problem in machine learning providing practitioners with a guarantee on the future performance of a trained predictor. Given its importance, there has been a significant amount of research that seeks to address this problem, one of the most prominent directions being a theory on the uniform convergence of empirical quantities to their mean (Vapnik and Chervonenkis 1971; Vapnik 1995). This theory provides a way of estimating the future misclassification rate of a predictor based on its empirical performance and some measure of the complexity of the predictor function e.g. the Vapnik-Chervonenkis dimension (Vapnik and Chervonenkis 1971) or the fat-shattering dimension (Alon et al. 1997). Further work in this direction (Marchand and Shawe-Taylor 2002; Sokolova et al. 2002) has been based on the prior assumption that the decision boundary can be constructed as a logical combination of a small set of data derived features. The analysis of these algorithms considers class-conditional error bounds that can be used for unequal loss functions. Class-conditional error bounds also inspire the approach described in the next paragraph.

In this paper we view the problem of generalisation from a different perspective by building upon the minimax probability machine (MPM) framework introduced in Lanckriet et al. (2003). In this setting, rather than trying to trade off the error over the training sample with the complexity of the function, we directly minimise an upper bound on the future misclassification rate. This minimisation takes place in a worst-case setting by considering all possible class-conditional distributions that have a particular mean and covariance matrix. These class-conditional means and covariance matrices play a key role in determining the optimal predictor and deriving upper bounds on its future misclassification rate. However when it comes to implementing these algorithms in practice, the true moments are not known and their empirical counterparts have to be used instead.

In this paper we seek to address the problems caused by the uncertainty of empirical moments by presenting the high-probability minimax probability machine (HP-MPM). The HP-MPM incorporates high probability upper bounds on the deviation of true moments from their empirical counterparts into the minimax problem to ensure that the future misclassification rate guarantees hold true with high-probability. The incorporation of moment uncertainty introduces a natural regularisation component into the optimisation scheme. We see that a smaller number of observations for a particular class results in greater uncertainty regarding its distribution, thus warranting additional regularisation. This is an often overlooked component of binary classifiers, where the class-conditional distributions are traditionally jointly regularised, ignoring the relative amount of information that is available for each class.

This paper follows with an introduction to the original MPM in Sect. 2, providing much of the technical details that will be required for the formulation of the HP-MPM. In Sect. 3 we present high-probability bounds on the deviations of true moments from their empirical counterparts, and show how they give rise to the HP-MPM optimisation scheme. In Sect. 4 we discuss the alternating optimisation that was designed to solve the problem, and deal with the kernelisation of the algorithm in Sect. 5. We present the results of our experiments in Sect. 6, and conclude in Sect. 7 with some final remarks regarding how best to use the newly proposed algorithm and where future research should focus.

2 Minimax probability machines

We consider the problem of constructing a binary classifier (predictor) by using some labelled training set consisting of inputs $\{\mathbf{x}_1, \dots, \mathbf{x}_m\} \in \mathcal{X} \subseteq \mathbb{R}^d$, and their corresponding class labellings $\{y_1, \dots, y_m\} \in \{0, 1\}$. For each class $j = 0, 1$, we assume that the input observations belonging to this class are generated according to some underlying distribution $\mathcal{D}_j$ where the mean $\bar{\mathbf{x}}_j \in \mathbb{R}^d$ and covariance matrix $\Sigma_j \in \mathbb{R}^{d\times d}$ of the distribution are known, but is otherwise arbitrary. The goal of the MPM is to find the linear decision boundary (hyperplane) that minimises the probability that future observations from these distributions will lie on the wrong side of this boundary.

Central to the derivation of the minimax program used in the MPM is the following theorem, a multivariate extension of the Chebyshev Inequality (Marshall and Olkin 1960), which was popularised for convex optimisation in Bertsimas and Popescu (2005):

Theorem 1 (Marshall and Olkin 1960; Bertsimas and Popescu 2005)

$$\sup_{\mathbf{x}\sim\mathcal{D}} \mathbf{Pr}\{\mathbf{x} \in \mathcal{S}\} = \frac{1}{1+d^2}, \quad \textit{with} \quad d^2 = \inf_{\mathbf{x}\in\mathcal{S}} (\mathbf{x}-\bar{\mathbf{x}})^T \Sigma^{-1} (\mathbf{x}-\bar{\mathbf{x}}),$$

where $\mathbf{x}$ *is a random vector,* $\mathcal{S}$ *is a given convex set, and where the supremum is taken over all distributions* $\mathcal{D}$ *for* $\mathbf{x}$ *that have mean* $\bar{\mathbf{x}}$ *and covariance matrix* Σ.

This theorem relates the maximum probability of a random vector $\mathbf{x} \sim \mathcal{D}$ belonging to a convex set $\mathcal{S}$ to the minimum Mahalanobis distance d^2 from the centre of the distribution $\bar{\mathbf{x}}$ to that set. Motivated by finding a linear decision boundary, Lanckriet et al. (2003) showed that when $\mathcal{S}$ is the upper half-space defined the separating hyperplane $\mathcal{H}(\mathbf{w}, b) := \{\mathbf{x} \mid \mathbf{w}^T\mathbf{x} = b\}$, the distance d^2 admits a closed form expression given by

$$d^2 = \inf_{\mathbf{w}^T\mathbf{x}\geq b} (\mathbf{x}-\bar{\mathbf{x}})^T \Sigma^{-1} (\mathbf{x}-\bar{\mathbf{x}}) = \begin{cases} \frac{(b-\mathbf{w}^T\bar{\mathbf{x}})^2}{\mathbf{w}^T\Sigma\mathbf{w}} & \text{if } \mathbf{w}^T\bar{\mathbf{x}} < b \\ 0 & \text{if } \mathbf{w}^T\bar{\mathbf{x}} \geq b \end{cases}. \tag{1}$$

This expression enables us to upper bound the probability that an observation drawn from a class-conditional distribution will lie on the wrong side of the separating hyperplane, alternatively it provides a lower bound on the probability that the observation will lie on the correct side of the hyperplane. This results in the following optimisation problem:

$$\begin{aligned} \max_{\mathbf{w},b,\alpha} \ \alpha \quad \text{s.t.} \quad & \inf_{\mathbf{x}_1\sim\mathcal{D}_1} P(\mathbf{w}^T\mathbf{x}_1 \geq b) \geq \alpha \\ & \inf_{\mathbf{x}_0\sim\mathcal{D}_0} P(\mathbf{w}^T\mathbf{x}_0 \leq b) \geq \alpha, \end{aligned}$$

where $\alpha \in [0, 1]$ is the minimum probability that examples are labelled correctly in the future. To see this, let our classifier predict that $\mathbf{x}$ belongs to class 1 if $\mathbf{w}^T\mathbf{x} \geq b$. The maximum probability that a point drawn from $\mathcal{D}_1$ resides on the wrong side of this hyperplane $\mathcal{H}(\mathbf{w}, b)$ is given by

$$\sup_{\mathbf{x}_1\sim\mathcal{D}_1} P(\mathbf{w}^T\mathbf{x}_1 < b) = \frac{1}{1+d^2} = 1-\alpha.$$

Therefore the minimum probability that a random vector $\mathbf{x}_1$ resides on the correct side of the hyperplane is greater than α. Using the closed form expression for the Mahalanobis distance given in (1), and assuming that $\mathbf{w}^T\bar{\mathbf{x}}_1 > b$, we derive the following key equivalence statement

$$\inf_{\mathbf{x}_1\sim\mathcal{D}_1} P(\mathbf{w}^T\mathbf{x}_1 \geq b) \geq \alpha \iff -b + \mathbf{w}^T\bar{\mathbf{x}}_1 \geq \kappa(\alpha)\sqrt{\mathbf{w}^T\Sigma_1\mathbf{w}}, \tag{2}$$

where $\kappa(\alpha) = \sqrt{\alpha/(1-\alpha)}$. A similar but opposite formulation for class 0 allows the optimisation problem to be written as

$$\max_{\mathbf{w},b,\alpha} \alpha \quad \text{s.t.} \quad -b + \mathbf{w}^T\bar{\mathbf{x}}_1 \geq \kappa(\alpha)\sqrt{\mathbf{w}^T\Sigma_1\mathbf{w}}$$
$$b - \mathbf{w}^T\bar{\mathbf{x}}_0 \geq \kappa(\alpha)\sqrt{\mathbf{w}^T\Sigma_0\mathbf{w}}.$$

Further reductions led to the second-order cone program given by the following theorem:

Theorem 2 (Lanckriet et al. 2003) *If $\bar{\mathbf{x}}_1 = \bar{\mathbf{x}}_0$ then the minimax probability decision problem does not have a meaningful solution and the worst case misclassification probability is given by $1-\alpha_* = 1$. Otherwise an optimal hyperplane $\mathcal{H}(\mathbf{w}_*, b_*)$ exists and can be determined by solving the convex optimisation problem*

$$\kappa_*^{-1} := \min_{\mathbf{w}} \sqrt{\mathbf{w}^T\Sigma_1\mathbf{w}} + \sqrt{\mathbf{w}^T\Sigma_0\mathbf{w}} \quad s.t. \quad \mathbf{w}^T(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_0) = 1, \tag{3}$$

and setting b to the value

$$b_* = \mathbf{w}_*^T\bar{\mathbf{x}}_1 - \kappa_*\sqrt{\mathbf{w}_*^T\Sigma_1\mathbf{w}_*}, \tag{4}$$

where $\mathbf{w}_*$ is the optimal solution to (3). The optimal worst-case misclassification probability is given by

$$1-\alpha_* = \frac{1}{1+\kappa_*^2} = \frac{\left(\sqrt{\mathbf{w}_*^T\Sigma_1\mathbf{w}_*} + \sqrt{\mathbf{w}_*^T\Sigma_0\mathbf{w}_*}\right)^2}{1+\left(\sqrt{\mathbf{w}_*^T\Sigma_1\mathbf{w}_*} + \sqrt{\mathbf{w}_*^T\Sigma_0\mathbf{w}_*}\right)^2}. \tag{5}$$

If either Σ_1 or Σ_0 is positive definite, the optimal hyperplane is unique.

Lanckriet et al. (2003) showed that it was possible to solve the optimisation (3) using an iterative least-squares scheme, which has a worst-case complexity $\mathcal{O}(d^3)$. Furthermore, their empirical results show that the MPM approach to classification is competitive with the state-of-the-art support vector machine (SVM) (Boser et al. 1992; Cortes and Vapnik 1995), thus providing evidence in support of the MPM as an efficient and effective approach for binary classification. These encouraging results resulted in the MPM approach being applied to both novelty-detection (Ghaoui et al. 2002) and regression (Strohmann and Grudic 2002), with some degree of success. In Huang et al. (2004) the authors identified an oversight in the original MPM formulation: that is, it implicitly assumes that the prior probability of each class is the same. The authors showed that if the prior probabilities of the classes differed, then it was no longer optimal to minimise a single worst-case future misclassification rate but rather one should minimise a weighted combination of class specific worst-case misclassification rates. The weights correspond to their prior class probability and this formulation became known as the minimum error MPM (ME-MPM). An alternative approach for dealing with imbalanced data was presented in Osadchy et al. (2015), where the authors optimised an objective that used the SVM hinge loss for the less frequent class and the minimax loss formulation for the abundant class. A transductive minimax probability machine was proposed in Huang et al. (2014), here unlabelled test points were assigned classes based upon their ability to minimise the worst case error bound and it was shown to be especially competitive with the transductive SVM on semi-supervised learning tasks. Similar to the transductive setting, in Huang et al. (2015) the authors focus on the problem of clustering by assigning unlabelled data to clusters in an attempt to optimise criterion defined by the MPM framework.

3 High-probability MPMs

The bounds on the future misclassification rate presented in (5) are valid when the true moments of the class-conditional distributions are known in advance. However this is not often the case, and in practice the true moments have to be substituted for empirical ones during the algorithm's implementation. This can result in the derivation of sub-optimal predictors and lead to bounds on the future misclassification rate that are invalid. In this section we make use of high-probability bounds on the deviation of true moments from their empirical counterparts, and use this to derive an optimisation scheme that directly takes into consideration the uncertainty of the empirical moments when constructing the predictor.

We begin by reviewing the robustness results presented in the original MPM (Lanckriet et al. 2003), where the authors sought to address the use of empirical moment estimates using a specific uncertainty sets, $\mathcal{U}_0$ and $\mathcal{U}_1$, for the value of the true moments. More specifically, for each $j = 0, 1$, the optimisation scheme considered all values of the true moments, $\bar{\mathbf{x}}_j$ and Σ_j, that resided within the uncertainty set

$$\mathcal{U}_j = \left\{ (\bar{\mathbf{x}}_j, \hat{\Sigma}_j) \; : \; (\bar{\mathbf{x}}_j - \hat{\mathbf{x}}_j)^T \Sigma_j^{-1} (\bar{\mathbf{x}}_j - \hat{\mathbf{x}}_j) \leq \nu^2, \;\; ||\Sigma_j - \hat{\Sigma}_j||_F \leq \rho \right\},$$

where $\hat{\mathbf{x}}_j$ and $\hat{\Sigma}_j$, are the empirical estimates of the mean and covariance matrix of class j, derived from the training sample i.e.

$$\hat{\mathbf{x}}_j = \frac{1}{m_j} \sum_{k=1}^{m_j} \mathbf{x}_j \quad \text{and} \quad \hat{\Sigma}_j = \frac{1}{m_j} \sum_{k=1}^{m_j} (\mathbf{x}_k - \hat{\mathbf{x}}_j)(\mathbf{x}_k - \hat{\mathbf{x}}_j)^T$$

where m_j is the number of observations belonging to class j. The value of $\nu \geq 0$ and $\rho \geq 0$ control the size of the uncertainty set, and have to be set at the practitioner's discretion. It can be argued that this specific uncertainty set chosen more for its numerical tractability rather than its statistical accuracy, and in what follows of this section, we derive a statistically motivated approach for incorporating the uncertainty of the moments into the optimisation scheme.

Shawe-Taylor and Cristianini (2003) present high-probability upper bounds on the deviation of true moments $(\bar{\mathbf{x}}, \Sigma)$ from their empirical counterparts $(\hat{\mathbf{x}}, \hat{\Sigma})$: high-probability in the sense that the probability that the true value of the moment deviates from the empirical one by more than $\epsilon \in \mathbb{R}$ is less than $\delta \geq 0$. They showed that the following holds true with probability at least $1 - \delta$:

$$||\bar{\mathbf{x}} - \hat{\mathbf{x}}||_2 \leq \frac{R}{\sqrt{m}} \left(2 + \sqrt{2 \log \frac{1}{\delta}} \right) \quad \text{and} \quad ||\Sigma - \hat{\Sigma}||_F \leq \frac{2R^2}{\sqrt{m}} \left(2 + \sqrt{2 \log \frac{2}{\delta}} \right), \tag{6}$$

where $|| \cdot ||$ and $|| \cdot ||_F$ denote the L_2 and Frobenius norms, respectively, $R > 0$ is the radius of the smallest sphere containing the support of $\mathcal{X}$ i.e. for all $\mathbf{x} \in \mathcal{X}$, $||\mathbf{x}|| \leq R$, and m is the number of observations that were used to construct the empirical moment. The authors examined the implications of the these deviation bounds on the MPM guarantees, showing that the high-probability worst-case estimate for the future misclassification errors can differ significantly from that found using (5). Their focus was on finding what the high-probability worst-case future misclassification rate was, given that the predictor was constructed using the original MPM formulation (3). Whereas our focus is on designing an optimisation scheme that directly minimises the high-probability bound on future misclassification by taking into consideration the uncertainty in the empirical moments.

To do this we begin by reviewing how moment uncertainty affects the key equivalence relationship given in (2). To simplify the analysis, we assume that the weight vector lies within the unit-ball defined by the L_2-norm i.e. $||\mathbf{w}|| \leq 1$. We want to find a high-probability bound on the deviation of the values in the inequality (2) when using empirical and true moments in the expression. We do this by using the following adaptation of the proposition presented in Shawe-Taylor and Cristianini (2003).

Proposition 1 *Let $\hat{\mathbf{x}}$ and $\hat{\Sigma}$ be the empirical mean and covariance matrix of a sample of m points drawn independently according some probability distribution $\mathcal{D}$ with mean $\bar{\mathbf{x}}$ and covariance matrix Σ. The weight vector $||\mathbf{w}|| \leq 1$ where $\mathbf{w} \neq \mathbf{0}$, and $b \in \mathbb{R}$ are given such that $\mathbf{w}^T\hat{\mathbf{x}} \leq b$. Then if*

$$b - \mathbf{w}^T\hat{\mathbf{x}} \geq \sqrt{\kappa(\alpha)^2\mathbf{w}^T\hat{\Sigma}\mathbf{w} + T} \tag{7}$$

where

$$T = \frac{4R^2}{\sqrt{m}}\left(2 + \sqrt{2\ln\frac{2}{\delta}}\right) + \kappa(\alpha)^2\frac{2R^2}{\sqrt{m}}\left(2 + \sqrt{2\ln\frac{2}{\delta}}\right)$$

then with probability at least $1 - \delta$ over the draw of the random sample

$$b - \mathbf{w}^T\bar{\mathbf{x}} \geq \kappa(\alpha)\sqrt{\mathbf{w}^T\Sigma\mathbf{w}} \quad \textit{and} \quad \inf_{\mathbf{x}\sim\mathcal{D}} P(\mathbf{w}^T\mathbf{x} \geq b) \geq \alpha.$$

Proof To prove this we show that if

$$(b - \mathbf{w}^T\hat{\mathbf{x}})^2 - \kappa(\alpha)^2\mathbf{w}^T\hat{\Sigma}\mathbf{w} \geq T$$

then with probability at least $1 - \delta$

$$(b - \mathbf{w}^T\bar{\mathbf{x}})^2 - \kappa(\alpha)^2\mathbf{w}^T\Sigma\mathbf{w} \geq 0.$$

We do this by bounding the high-probability differences in the value of the expressions on the left hand side of the inequalities

$$\begin{aligned}
&\left|(b - \mathbf{w}^T\hat{\mathbf{x}})^2 - \kappa(\alpha)^2\mathbf{w}^T\hat{\Sigma}\mathbf{w} - (b - \mathbf{w}^T\bar{\mathbf{x}})^2 + \kappa(\alpha)^2\mathbf{w}^T\Sigma\mathbf{w}\right| \\
&\leq ||\hat{\mathbf{x}} - \bar{\mathbf{x}}||\left(2b + ||\hat{\mathbf{x}} + \bar{\mathbf{x}}||\right) + \kappa(\alpha)^2\left|\mathbf{w}^T\hat{\Sigma}\mathbf{w} - \mathbf{w}\Sigma\mathbf{w}\right| \\
&\leq ||\hat{\mathbf{x}} - \bar{\mathbf{x}}||4R + \kappa(\alpha)^2||\hat{\Sigma} - \Sigma||_F.
\end{aligned}$$

The proof is completed by using the bounds on the empirical moments presented (6) with δ replaced with $\delta/2$,

$$\begin{aligned}
&\left|(b - \mathbf{w}^T\hat{\mathbf{x}})^2 - \kappa(\alpha)^2\mathbf{w}^T\hat{\Sigma}\mathbf{w} - (b - \mathbf{w}^T\bar{\mathbf{x}})^2 + \kappa(\alpha)^2\mathbf{w}^T\Sigma\mathbf{w}\right| \\
&\leq \frac{4R^2}{\sqrt{m}}\left(2 + \sqrt{2\ln\frac{2}{\delta}}\right) + \kappa(\alpha)^2\frac{2R^2}{\sqrt{m}}\left(2 + \sqrt{2\ln\frac{2}{\delta}}\right).
\end{aligned}$$

Note that the bound comes into play when we consider

$$(b - \mathbf{w}^T\hat{\mathbf{x}})^2 - \kappa(\alpha)^2\mathbf{w}^T\hat{\Sigma}\mathbf{w} \geq (b - \mathbf{w}^T\bar{\mathbf{x}})^2 - \kappa(\alpha)^2\mathbf{w}^T\Sigma\mathbf{w},$$

and it holds true regardless of the bound if the inequality is reversed. □

To formulate the HP-MPM optimisation scheme we use each classes corresponding inequality (7), where the class specific uncertainty is captured by the term T_j for $j = 0, 1$ with

$$\begin{aligned} T_j &= \frac{4R^2}{\sqrt{m_j}}\left(2+\sqrt{2\ln\frac{2}{\delta}}\right) + \kappa(\alpha)^2\frac{2R^2}{\sqrt{m_j}}\left(2+\sqrt{2\ln\frac{2}{\delta}}\right) \\ &= 2A_j + \kappa(\alpha)^2 A_j, \end{aligned}$$

where

$$A_j = \frac{2R^2}{\sqrt{m_j}}\left(2+\sqrt{2\ln\frac{2}{\delta}}\right). \tag{8}$$

We can drop the dependence of the optimisation on α by noting the monotonic relationship it has with $\kappa(\alpha)$, and by introducing the constraint that $||\mathbf{w}|| \leq 1$ and using the inequalities (7), the minimax program becomes

$$\begin{aligned} \max_{\mathbf{w},b,\kappa}\ \kappa \quad \text{s.t.} \quad & ||\mathbf{w}|| \leq 1 \\ -b + \mathbf{w}^T\hat{\mathbf{x}}_1 &\geq \sqrt{2A_1 + \kappa^2\left(\mathbf{w}^T\hat{\Sigma}_1\mathbf{w} + A_1\right)} \\ b - \mathbf{w}^T\hat{\mathbf{x}}_0 &\geq \sqrt{2A_0 + \kappa^2\left(\mathbf{w}^T\hat{\Sigma}_0\mathbf{w} + A_0\right)}. \end{aligned}$$

Corollary 1 *For $j = 0, 1$, let $\hat{\mathbf{x}}_j$ and $\hat{\Sigma}_j$ be the empirical mean and covariance matrix of m_j points drawn independently from distributions D_j with true mean $\bar{\mathbf{x}}_j$ and covariance matrix Σ_j, and let A_j be defined according to (8). If $||\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0|| \leq \sqrt{2A_1} + \sqrt{2A_0}$ then the high probability MPM decision problem does not have a meaningful solution and the worst-case misclassification probability is given by $1 - \alpha_* = 1$. Otherwise an optimal hyperplane $\mathcal{H}(\mathbf{w}_*, b_*)$ exists and can be determined by solving the optimisation problem given by*

$$\begin{aligned} \max_{\mathbf{w},\kappa}\ \kappa \quad s.t. \quad & ||\mathbf{w}|| \leq 1 \\ \mathbf{w}^T(\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0) &= \sqrt{2A_1 + \kappa^2\left(\mathbf{w}^T\hat{\Sigma}_1\mathbf{w} + A_1\right)} + \sqrt{2A_0 + \kappa^2\left(\mathbf{w}^T\hat{\Sigma}_0\mathbf{w} + A_0\right)}, \end{aligned} \tag{9}$$

and setting b to the value

$$b_* = \mathbf{w}_*^T\hat{\mathbf{x}}_1 - \sqrt{2A_1 + \kappa_*^2\left(\mathbf{w}_*^T\hat{\Sigma}_1\mathbf{w}_* + A_1\right)} = \mathbf{w}_*^T\hat{\mathbf{x}}_0 + \sqrt{2A_0 + \kappa_*^2\left(\mathbf{w}_*^T\hat{\Sigma}_0\mathbf{w}_* + A_0\right)},$$

where $\mathbf{w}_*$ and κ_* are the optimal solutions to (9). Then with probability at least $1-\delta$ over the draws of the random sample, the optimal worst-case misclassification probability is given by

$$1 - \alpha_* = \frac{1}{1+\kappa_*^2}.$$

When presented with a new input observation $\mathbf{x}'$, we make our prediction according to what side of the optimal hyperplane the point resides i.e. we predict that $y' = 1$ if $\mathbf{w}_*^T\mathbf{x}' - b_* \geq 0$, and that $y' = 0$ otherwise.

4 Optimisation scheme

The optimisation problem given in (9) can not be solved using the same approach taken in Lanckriet et al. (2003) because of the unit L_2-norm restriction on $\mathbf{w}$, and the presence of the uncertainty terms A_j under the square root. To solve this problem we propose the use of an auxiliary function $h(\mathbf{w}, \kappa)$ in conjunction with an alternating update scheme over $\mathbf{w}$ and κ. The auxiliary function is given by

$$h(\mathbf{w}, \kappa) = \mathbf{w}^T(\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0) - \sqrt{2A_1 + \kappa^2\left(\mathbf{w}^T\hat{\Sigma}_1\mathbf{w} + A_1\right)} - \sqrt{2A_0 + \kappa^2\left(\mathbf{w}^T\hat{\Sigma}_0\mathbf{w} + A_0\right)} \tag{10}$$

Note that the class uncertainty terms require the computation of span of the data i.e. find R such that $||\mathbf{x}|| \leq R$ for all $\mathbf{x} \in \mathcal{X}$. During implementation this will have to be estimated from the training sample or can be enforced by some normalisation scheme that is independent of the learning algorithm.

Initialisation: To initialise the optimisation, we begin with $\kappa = 0$, and find the value of $\mathbf{w}$ that maximises $h(\mathbf{w}, \kappa)$ subject to the constraints that $||\mathbf{w}|| \leq 1$. This has a closed form solution $(\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0)/||\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0|| = \arg\max_{||\mathbf{w}||\leq 1} h(\mathbf{w}, 0)$, and provides the conditions necessary for a meaningful solution to the high probability MPM decision problem i.e. we require that $\max_{||\mathbf{w}||\leq 1} h(\mathbf{w}, 0) > 0$ in order to be able to find a positive value of κ in the next step of the optimisation scheme.
w-step: For non-initialisation $\mathbf{w}$-steps, the goal is maximise the value of the auxiliary function by performing gradient ascent subject to our constraint $||\mathbf{w}|| \leq 1$. It is straightforward to show that $h(\mathbf{w}, \kappa)$ is a concave in $\mathbf{w}$ and therefore every local optimum will be a global optimum. Therefore we can use standard constrained optimisation tools to solve this intermediate problem. Note that we do not need to run these constrained optimisations to convergence, we simply need the value of the auxiliary function to increase, in order to allow for a larger value of κ in the next step. We view this as a constrained maximisation subject to some implicit degree of regularisation imposed by the value of κ.
κ-step: In order to continue the optimisation, we require that the $\mathbf{w}$-step results in a strictly positive value for the auxiliary function, $h(\mathbf{w}, \kappa) > 0$. If this is not the case then the optimisation has converged, and we have reached the optimal solution. If $h(\mathbf{w}, \kappa) > 0$, we can increase the value of κ to κ' such that the value of the auxiliary function is zero i.e. $h(\mathbf{w}, \kappa') = 0$. This can be performed using a simple line-search procedure, or by finding the roots of a quadratic expression involving κ. Note that in order for the optimisation to progress we must find κ' such that $\kappa' > \kappa$. To simplify the range of the line-search we observe an upper bound on the value of κ', namely $\kappa' \leq \kappa_u = ||\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0|| / \left(\sqrt{\mathbf{w}^T\hat{\Sigma}_1\mathbf{w}} + \sqrt{\mathbf{w}^T\hat{\Sigma}_0\mathbf{w}}\right)$.
Optimal solution: We prove that the optimal solution for the weight vector $\mathbf{w}_*$ will have a unit L_2-norm i.e. $||\mathbf{w}_*|| = 1$. To do this suppose that $||\mathbf{w}_*|| < 1$, we know that at optimality $h(\mathbf{w}_*, \kappa_*) = 0$ and that $\mathbf{w}' = \mathbf{w}_*/||\mathbf{w}_*||$ is also a feasible solution. We show that $h(\mathbf{w}', \kappa_*) > 0$ and that $\mathbf{w}_*$, where $||\mathbf{w}_*|| < 1$, can not be the optimal solution. To see this observe that

$$\begin{aligned}\sqrt{2A_j + \kappa_*^2\left(\mathbf{w}'^T\hat{\Sigma}_j\mathbf{w}' + A_j\right)} &= \sqrt{2A_j + \kappa_*^2\left(\frac{1}{||\mathbf{w}_*||^2}\mathbf{w}_*^T\hat{\Sigma}_j\mathbf{w}_* + A_j\right)} \\ &< \frac{1}{||\mathbf{w}_*||}\sqrt{2A_j + \kappa_*^2\left(\mathbf{w}_*^T\hat{\Sigma}_j\mathbf{w}_* + A_j\right)}.\end{aligned}$$

Algorithm 1 HP-MPM Optimisation Scheme

```
Input: $\hat{\mathbf{x}}_j, \hat{\Sigma}_j, A_j$ for $j = 0, 1$, tolerance $\epsilon_\kappa > 0$ and $\epsilon_{\mathbf{w}} > 0$
Initialise: $\kappa = 0$, $\mathbf{w} = (\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0)/||\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0||$ and $converged = false$
if $||\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0|| \leq \sqrt{2A_1} + \sqrt{2A_0}$ then
  while: (not $converged$)
    $converged = true$
    Find by line-search $\kappa' \in [\kappa, \kappa_u]$ such that $h(\mathbf{w}, \kappa') = 0$
    $\mathbf{w}' = \arg\max_{||\mathbf{w}|| \leq 1} h(\mathbf{w}, \kappa)$
    if: $(|\kappa' - \kappa| > \epsilon_\kappa) \;\vee\; (h(\mathbf{w}', \kappa') > \epsilon_{\mathbf{w}})$
      then $converged = false$
    end if
    $\mathbf{w} \leftarrow \mathbf{w}', \kappa \leftarrow \kappa'$
  end while
end if
$\mathbf{w}_* = \mathbf{w}$, $\kappa_* = \kappa$ and $b_* = \mathbf{w}_*^T \hat{\mathbf{x}}_1 - \sqrt{2A_1 + \kappa_*^2 \left(\mathbf{w}_*^T \hat{\Sigma}_1 \mathbf{w}_* + A_1\right)}$
```

Using this inequality in the auxiliary function $h(\mathbf{w}', \kappa_*)$ we see that

$$h(\mathbf{w}', \kappa_*) > \frac{1}{||\mathbf{w}_*||} h(\mathbf{w}_*, \kappa_*),$$

where we know by the monotonicity of $h(\mathbf{w}, \kappa)$ with respect to κ, that there exists $\kappa' > \kappa_*$, satisfying the constraints in (9). Therefore $(\mathbf{w}_*, \kappa_*)$ can not be the optimal solution to this problem.

Geometric interpretation: The original MPM can be viewed as looking for the point of intersection between two ellipsoids centered at the class means, where the shape of the ellipsoids are determined by the covariance matrices and their size is controlled by the value of κ i.e. for $j = 0, 1$

$$\mathcal{E}_j(\kappa) = \left\{ \mathbf{x} = \bar{\mathbf{x}}_j + \Sigma_j^{1/2} \mathbf{u} \;:\; ||\mathbf{u}|| \leq \kappa \right\}$$

Clearly as the size of κ increases these ellipsoids will eventually overlap. However, the optimal hyperplane is given by the common tangent to the ellipsoids at the first point of their tangency. During our optimisation scheme, we alternate between allowing these ellipsoids, albeit a penalised verson of them, to intersect i.e. $h(\mathbf{w}, \kappa) = 0$, and rotating $\mathbf{w}$ to provide additional space for the ellipsoids to expand into at the next stage of the optimisation. We can view the moment uncertainty as introducing a regularisation component to the covariance matrices, along with a penalty regarding the location of the means. The regularised ellipsoids we consider in the high-probability setting are given by

$$\hat{\mathcal{E}}_j(\kappa) = \left\{ \mathbf{x} = \hat{\mathbf{x}}_j + \tilde{\Sigma}(\kappa)_j^{1/2} \mathbf{u} : \; ||\mathbf{u}|| \leq \kappa, \; \tilde{\Sigma}(\kappa)_j = \hat{\Sigma}_j + I_d \left(A_j + \frac{2A_j}{\kappa^2} \right) \right\}. \quad (11)$$

Using the geometrical interpretation, we see that as the value of κ grows, the effective regularisation on the covariance matrix decreases. This results from a relative reduction in the role played by the mean uncertainty in the square root term. Intuitively, as we move away from the means, with increasing values of κ, the point of origin becomes less important and we focus more on the underlying shape of the ellipsoid. In Fig. 1 we show how the ellipsoids change as we increase κ up until their point of tangency. The intermediate solutions where the auxiliary function $h(\mathbf{w}, \kappa) = 0$, represent hyerplanes that are tangential to the ellipsoids but where the ellipsoids are not tangential to one another.

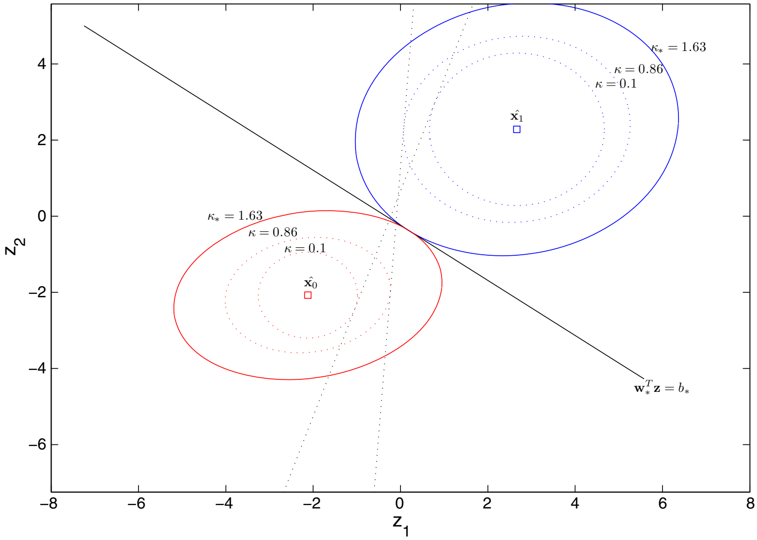


Fig. 1 Geometric interpretation of the high-probability MPM and the intermediate solutions produced during the optimisation scheme. We can see that in the beginning, for small values of κ, the penalised (regularised) covariance matrices are almost spherical. As the value of κ increases, and we move away from the class means, the ellipsoids begin to take on a shape increasingly determined by the sampled covariance matrix, however there still remains the regularisation caused by the uncertainty in the value of the covariance matrix. We see that the intermediate solutions $h(\mathbf{w}, \kappa) = 0$ result in hyperplanes that are tangential to the each classes ellipsoid, however these ellipsoids are only tangential to one another at the optimal solution. In the samples used to generate this solution, $m_1 = 20$ and $m_0 = 200$, explaining the larger size of the ellipsoid for class 1

5 Kernelisation

So far we have explored the notion of finding an optimal linear decision boundary. Geometrically we saw that the worst-case bound on the future misclassification rate depends on both the distance between the class means, and the shape of the ellipsoids that their covariance matrices determine. However, it is often the case that by mapping the inputs into some higher-dimensional feature space there is a greater degree of separation between the two classes, and thus we should be able to reduce the worst-case future misclassification rate. Kernel methods (Vapnik 1998; Shawe-Taylor and Cristianini 2004) are able to take advantage of these higher-dimensional feature spaces without having to explicitly compute them, and have proven a useful tool for many classification algorithms. In this section we show that our optimisation problem can be re-written in terms of the kernel function, which allows us to efficiently use higher-dimensional feature spaces to represent the input space. To do this we closely follow the approach taken in Lanckriet et al. (2003).

We begin by introducing the feature mapping $\phi : \mathcal{X} \rightarrow \mathcal{F}$ where the linear decision boundary in this space is given by hyperplane $\mathcal{H}(\mathbf{w}, b) = \{\phi(\mathbf{x}) \in \mathcal{F} : \mathbf{w}^T\phi(\mathbf{x}) = b\}$. Note that the linear decision boundary in feature space corresponds to a non-linear decision boundary in the original space $\mathcal{X}$. The data is mapped according to

$$\mathbf{x}_1 \rightarrow \phi(\mathbf{x}_1) \sim \mathcal{D}_1^{\phi}$$
$$\mathbf{x}_0 \rightarrow \phi(\mathbf{x}_0) \sim \mathcal{D}_0^{\phi},$$

where distribution $\mathcal{D}_j^{\phi}$, has mean $\bar{\phi}_j$ and covariance matrix Σ_j^{ϕ} defined in the feature space $\mathcal{F}$. To find the optimal hyperplane in $\mathcal{F}$ we follow the same optimisation problem given in (9), where we substitute the original empirical moments with their feature space counterparts $\hat{\phi}_j$ and $\hat{\Sigma}_j^{\phi}$ for each class $j = 0, 1$. In order to efficiently use the feature mappings and make use of the kernel-trick, we have to show that the feature mappings enter the optimisation scheme only in terms of their inner-product $\langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle = K(\mathbf{x}, \mathbf{x}')$, where $K : \mathcal{F} \times \mathcal{F}$ is the kernel function corresponding to the feature mapping ϕ. This allows us to use high-dimensional feature spaces without having to explicitly compute them, thus making them tractable to work with.

To do this, first we have to show that any optimal solution to (9) must lie in the space spanned by the input data. To prove this, suppose the optimal solution is given by $\mathbf{w}_* = \mathbf{w}_s + \mathbf{w}_o$, where $\mathbf{w}_s$ is the projection of $\mathbf{w}$ onto the span of the input data and $\mathbf{w}_o$ is orthogonal to the space spanned by the input data. We show that the value of $\mathbf{w}_o$ plays no role in our ability to satisfy the first constraint in (9), however it does play a part in the unit L_2-norm restriction $||\mathbf{w}|| \leq 1$. Therefore if we removed this orthogonal component and scaled our $\mathbf{w}_s$ so that it resided on the unit-ball, we have already showed that this will increase the value of the auxiliary function, thus permitting an increase in the value of κ in the next round of the optimisation scheme. Therefore a solution containing an orthogonal component can never be optimal.

The empirical means and covariances are linear combinations of the input data, and it is straightforward to show that

$$\mathbf{w}^T(\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0) = \mathbf{w}_s^T(\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0)$$

$$\mathbf{w}^T\hat{\Sigma}_j\mathbf{w} = \mathbf{w}_s^T\hat{\Sigma}_j\mathbf{w}_s \quad \text{for} \quad j = 0, 1.$$

Therefore the value of the auxiliary function evaluated at $\mathbf{w}$ and $\mathbf{w}_s$ are the same i.e. $h(\mathbf{w}, \kappa) = h(\mathbf{w}_s, \kappa)$. We know that if we replace $\mathbf{w} = \mathbf{w}_s + \mathbf{w}_o$ with $\mathbf{w}_s/||\mathbf{w}_s||$, where $||\mathbf{w}_s|| < 1$,then the value of our auxiliary function increases, and allows for a larger value of κ at optimality. Therefore a solution containing a component orthogonal to the span of the input data can not be optimal, and the optimal solution must be given by a linear combination of the input data

$$\mathbf{w}_* = \sum_{i=1}^{m} \gamma_i \mathbf{x}_i,$$

where $\gamma_i \in \mathbb{R}$ for all $i = 1, \ldots, m$. To take full advantage of the kernel-trick, and avoid having to explicitly evaluate the feature mappings, we now have to show that the feature mappings only appear in the optimisation problem as inner-products.

Let us denote the kernel matrix $\mathbf{K}$ where $\mathbf{K}_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$ for all $i, j = 1, \ldots, m$. The first m_1 rows and last m_0 rows of $\mathbf{K}$ are denoted $\mathbf{K}_1$ and $\mathbf{K}_0$, respectively:

$$\mathbf{K} = \begin{pmatrix} \mathbf{K}_1 \\ \mathbf{K}_0 \end{pmatrix},$$

where $y_i = 1$ for $i = 1, \ldots m_1$, and $y_i = 0$ for $i = m_1 + 1, \ldots, m_1 + m_0$. The class row averages, $\mathbf{l}_1^T$ and $\mathbf{l}_0^T$, are m-dimensional vectors given by

$$\left(\mathbf{l}_1^T\right)_i = \frac{1}{m_1}\sum_{j=1}^{m_1} K(\mathbf{x}_j, \mathbf{x}_i) \text{ and } \left(\mathbf{l}_0^T\right)_i = \frac{1}{m_0}\sum_{j=m_1+1}^{m} K(\mathbf{x}_j, \mathbf{x}_i).$$

We create the block-row-averaged kernel matrix $\mathbf{L}$ by setting the row average of $\mathbf{K}_1$ and $\mathbf{K}_0$ equal to zero by:

$$\mathbf{L} = \begin{pmatrix} \mathbf{K}_1 - \mathbf{1}_{m_1}\mathbf{l}_1^T \\ \mathbf{K}_0 - \mathbf{1}_{m_0}\mathbf{l}_0^T \end{pmatrix} = \begin{pmatrix} \sqrt{m_1}\ \mathbf{L}_1 \\ \sqrt{m_0}\ \mathbf{L}_0 \end{pmatrix},$$

where $\mathbf{1}_m$ is a column vector of ones of dimension m. The empirical moment estimates in the feature space are given by

$$\hat{\phi}_1 = \frac{1}{m_1}\sum_{i=1}^{m_1}\phi(\mathbf{x}_i) \text{ and } \hat{\Sigma}_1^{\phi} = \frac{1}{m_1}\sum_{i=1}^{m_1}\left(\phi(\mathbf{x}_i) - \hat{\hat{\mathbf{x}}}_1^{\phi}\right)\left(\phi(\mathbf{x}_i) - \hat{\hat{\mathbf{x}}}_1^{\phi}\right)^T$$

$$\hat{\phi}_0 = \frac{1}{m_0}\sum_{i=m_1+1}^{m}\phi(\mathbf{x}_i) \text{ and } \hat{\Sigma}_0^{\phi} = \frac{1}{m_0}\sum_{i=m_1+1}^{m}\left(\phi(\mathbf{x}_i) - \hat{\hat{\mathbf{x}}}_0^{\phi}\right)\left(\phi(\mathbf{x}_i) - \hat{\hat{\mathbf{x}}}_0^{\phi}\right)^T$$

We saw earlier that the solution is given by $\mathbf{w} = \sum_{i=1}^{m}\gamma_i\phi(\mathbf{x}_i)$, and therefore the components of the optimisation become

$$\mathbf{w}^T(\hat{\phi}_1 - \hat{\phi}_0) = \boldsymbol{\gamma}^T(\mathbf{l}_1 - \mathbf{l}_0),\quad \mathbf{w}^T\hat{\Sigma}_1^{\phi}\mathbf{w} = \boldsymbol{\gamma}^T\mathbf{L}_1^T\mathbf{L}_1\boldsymbol{\gamma} \quad\text{and}\quad \mathbf{w}^T\hat{\Sigma}_0^{\phi}\mathbf{w} = \boldsymbol{\gamma}^T\mathbf{L}_0^T\mathbf{L}_0\boldsymbol{\gamma}.$$

This allows us to write the kernelised version of the HP-MPM as

$$\max_{\boldsymbol{\gamma},\kappa}\ \kappa \ \text{ s.t. } ||\mathbf{w}||^2 = \boldsymbol{\gamma}^T\mathbf{K}\boldsymbol{\gamma} \le 1$$

$$\boldsymbol{\gamma}^T(\mathbf{l}_1 - \mathbf{l}_0) = \sqrt{2A_1 + \kappa^2\left(\boldsymbol{\gamma}^T\mathbf{L}_1^T\mathbf{L}_1\boldsymbol{\gamma} + A_1\right)} + \sqrt{2A_0 + \kappa^2\left(\boldsymbol{\gamma}^T\mathbf{L}_0^T\mathbf{L}_0\boldsymbol{\gamma} + A_1\right)}.$$

The same alternating optimisation procedure can be used to find the optimal values κ_* and $\boldsymbol{\gamma}_*$, and the optimal value of the bias term is given by

$$b_* = \boldsymbol{\gamma}_*^T\mathbf{l}_1 - \sqrt{2A_1 + \kappa_*^2\left(\boldsymbol{\gamma}_*^T\mathbf{L}_1^T\mathbf{L}_1\boldsymbol{\gamma}_* + A_1\right)} = \boldsymbol{\gamma}_*^T\mathbf{l}_0 + \sqrt{2A_0 + \kappa_*^2\left(\boldsymbol{\gamma}_*^T\mathbf{L}_0^T\mathbf{L}_0\boldsymbol{\gamma}_* + A_0\right)}$$

As with the linear case, when presented with a new input observation $\mathbf{x}'$, we predict that $y' = 1$ if $\boldsymbol{\gamma}_*^T\mathbf{k}_{\mathbf{x}'} - b_* \ge 0$, where $(\mathbf{k}_{\mathbf{x}'})_i = k(\mathbf{x}_i, \mathbf{x}')$, and $y' = 0$ otherwise. We should point out that we have no reason to expect that the solution $\boldsymbol{\gamma}_*$ will be sparse i.e. many $(\gamma_*)_i = 0$, and therefore the computational cost at prediction will be linear in the size of the training sample.

6 Experiments

In this section we examine the performance of the proposed HP-MPM and compare it to the original MPM, and two other popular binary classification algorithms, Fisher's discriminant (FDA) (Fisher 1936), and the support vector machine (SVM). In Table 1 we provide a summary of the datasets taken from the UCI repository, http://archive.ics.uci.edu/ml/, and the toy dataset used in Lanckriet et al. (2003), that we have used in our experiments. We have included details regarding the number of observations, the dimension of the input space and the relative class frequencies to help support our argument regarding the importance of including information regarding the moment uncertainty into the derivation of the predictor. All of the datasets were normalised so that each feature had zero mean and unit variance. To handle missing values, as in the *vote* dataset, we simply computed the means and standard deviations of each feature using the available data, performed standard normalisation on them and then set the values of the missing data to zero post-normalisation. Each dataset was randomly partitioned 50 times into training, validation and test samples, and we report the average performance over all test samples. During the experiments we varied the size of the training sample between 10 and 70%, in increments of 10%, of the full dataset to investigate how the algorithms performed with various amounts of information. The size of the validation set was fixed at 20% and the remaining data was used for testing. The goal of these experiments was to evaluate the benefits of considering moment uncertainty in the

Table 1 Overview of the UCI datasets used during the experiments

Dataset	Observations	Features	Class 1
Adult	48,844	123	23.93
Australian	690	14	44.49
BCI	400	117	50.00
Breast	682	10	64.96
Diabetes	768	8	65.10
Digit1	1500	241	48.93
German	1000	24	70.00
Heart	920	13	44.67
Ionosphere	351	32	64.10
Ringnorm	7400	20	49.51
Sonar	208	60	53.37
Splice	3175	60	51.91
Toy	120	2	50.00
Twonorm	7400	20	50.04
Vote	435	16	38.62

construction of the predictor, and to understand the relative gains that its inclusion have as we change the number of training points.

One of the main motivations of the formulation of the HP-MPM was to correct for overly confident estimates on the worst-case future misclassification rate. This was done through the introduction of high-probability bounds on the deviation of the true moments from the empirical counterparts. However, we noticed that during the experiments that these high-probability bounds appeared to be too restrictive in many settings and we were unable to generate meaningful solutions i.e. $\alpha_* = 0$. To overcome this deficiency we propose to use the moment uncertainty terms as a form of regularisation, and during the experiments we use a validation procedure to choose what fraction of the true moment uncertainty we should use. More precisely, rather than using A_j we used some fractional amount $\hat{A}_j = \nu A_j$ of the full uncertainty, where $\nu \in \{0.05, 0.1, 0.15, 0.2, 0.3, 0.5, 1\}$. For the parameter selection process, in each training and test sample we had a distinct validation set that was used to evaluate the performance of the predictor generated for the particular regularisation parameter. For each of these training, validation and test sets we evaluated the performance of the predictor (parameter) with the best validation set accuracy on the test sample. The same method to choose the regularisation parameter for the SVM and FDA, where SVM's *capacity* parameter was selected from $C \in \{10^{-3}, \ldots, 10^{3}\}$, and the FDA's regularisation term chosen from $\lambda \in \{10^{-3}, \ldots, 10^{3}\}$.

In Table 2 we examine the performance of the linear based classification algorithms and show how their performance varies as we change the size of the training sample used to construct the predictor. As one would expect, in general the performance on the test samples improves as more training examples are presented to the algorithm during training. However in the sonar dataset we see a drop in the performance of the MPM and FDA predictors as we increase the fraction of the dataset used in training from 0.1 to 0.3. This can be explained by the relatively small number of observations that were used to construct the empirical moments, which determine each algorithms decision boundary. On the other hand we see that the HP-MPM and the SVM are relatively robust to the use of small training samples, and we observe the benefits of the regularisation scheme implemented by the HP-MPM, and note the benefits of constructing the decision boundary using peripheral points, as advocated by the SVM, rather than poorly estimated empirical moments.

Table 2 Linear experiments: we show how the performance of the classification algorithms on the datasets vary as the amount of data used during training changes

	Training proportion						
	0.1	0.2	0.3	0.4	0.5	0.6	0.7
Australian							
MPM	84.21	85.88	85.89	86.08	86.04	86.25	86.39
HP-MPM	**85.01**	**86.29**	**86.23**	**86.18**	**86.43**	86.31	86.55
SVM	84.81	86.15	86.07	85.80	85.63	85.87	86.02
FDA	83.85	85.66	85.92	86.13	86.20	**86.60**	**86.87**
BCI							
MPM	58.30	62.25	56.55	70.81	76.49	79.74	81.55
HP-MPM	**61.70**	**66.84**	**71.50**	**77.30**	**80.18**	**82.33**	**82.96**
SVM	60.39	66.25	70.83	74.17	75.90	78.82	79.41
FDA	58.66	57.72	53.80	57.12	60.88	63.97	68.03
Breast							
MPM	96.20	97.04	97.10	97.22	97.22	**97.33**	**97.23**
HP-MPM	**97.12**	**97.10**	**97.18**	**97.24**	**97.24**	97.29	97.22
SVM	96.58	96.77	96.85	96.97	97.05	97.07	97.03
FDA	92.54	93.76	94.50	94.50	94.50	94.58	94.46
Diabetes							
MPM	72.74	74.12	75.02	75.11	74.99	74.97	74.86
HP-MPM	73.14	73.86	74.76	74.75	74.41	74.49	74.53
SVM	**74.40**	**75.97**	**76.30**	76.29	76.48	76.55	76.74
FDA	73.97	75.32	75.94	**76.61**	**76.51**	**76.68**	**76.91**
Digit1							
MPM	73.47	75.08	85.92	89.53	91.38	92.16	92.90
HP-MPM	**92.99**	**93.80**	**94.30**	94.53	94.63	94.62	94.60
SVM	91.66	93.38	94.20	**94.69**	**95.10**	**95.25**	**95.70**
FDA	80.09	75.77	85.82	89.35	91.42	92.17	92.83
German							
MPM	68.96	70.84	71.55	72.08	72.24	72.49	72.70
HP-MPM	69.54	71.26	71.86	72.34	72.26	72.50	72.79
SVM	**71.66**	**73.82**	**74.55**	**74.99**	75.51	75.82	76.03
FDA	71.50	73.64	74.45	74.85	**75.58**	**76.10**	**76.47**
Heart							
MPM	78.02	79.54	80.20	80.72	81.33	**81.84**	**82.21**
HP-MPM	**79.36**	**80.15**	**80.62**	80.87	**81.34**	81.56	81.96
SVM	78.77	79.62	80.36	**80.98**	81.29	81.69	81.97
FDA	77.47	79.33	80.01	80.61	81.01	81.64	81.99
Ionosphere							
MPM	72.45	78.73	80.49	81.21	81.70	82.29	82.62
HP-MPM	**82.18**	**82.93**	83.35	82.88	83.18	83.39	83.11
SVM	80.68	82.91	**83.51**	**83.83**	**84.18**	**84.64**	**84.19**
FDA	68.60	74.09	76.41	78.04	79.63	80.06	80.38

Table 2 continued

	Training proportion						
	0.1	0.2	0.3	0.4	0.5	0.6	0.7
Ringnorm							
MPM	74.43	74.70	74.88	74.93	74.94	74.94	74.86
HP-MPM	76.51	76.71	76.81	76.80	76.86	76.91	76.79
SVM	**76.55**	**76.88**	**76.88**	**77.05**	**77.03**	77.09	77.06
FDA	76.19	76.63	76.74	76.82	76.91	**77.12**	**77.07**
Sonar							
MPM	63.59	61.80	57.42	67.61	70.36	71.76	75.47
HP-MPM	**69.88**	**73.84**	**74.71**	74.93	**76.71**	**76.01**	**77.41**
SVM	67.51	72.95	74.08	**75.31**	75.84	75.32	77.20
FDA	62.37	59.88	55.11	64.49	68.28	70.44	73.73
Splice							
MPM	81.52	83.21	83.83	**84.23**	84.36	84.55	**84.89**
HP-MPM	**82.15**	**83.31**	**83.87**	84.20	84.35	**84.67**	84.80
SVM	81.74	82.96	83.55	84.14	**84.37**	84.40	84.84
FDA	81.12	82.98	83.68	84.11	84.25	84.56	84.71
Toy							
MPM	89.92	93.17	94.00	94.41	**94.36**	94.45	94.50
HP-MPM	**91.16**	**93.32**	**94.22**	**94.57**	94.32	**94.50**	**94.89**
SVM	90.42	93.13	93.75	94.24	93.50	93.09	93.38
FDA	86.12	90.04	91.10	93.18	92.71	93.18	93.62
Twonorm							
MPM	97.59	97.65	97.68	97.68	97.74	97.76	97.80
HP-MPM	**97.67**	**97.69**	**97.71**	97.70	97.75	97.76	97.82
SVM	97.53	97.60	97.67	**97.72**	**97.75**	**97.77**	**97.84**
FDA	97.53	97.63	97.67	97.67	97.73	97.74	97.81
Vote							
MPM	92.86	95.28	95.75	95.71	95.74	95.95	96.03
HP-MPM	**94.95**	95.42	95.58	95.53	95.37	95.51	95.59
SVM	94.47	95.20	94.99	95.35	95.26	95.41	95.81
FDA	93.37	**95.51**	**95.97**	**96.00**	**96.17**	**96.40**	**96.19**

The best performing results for each dataset and training proportion are reported in bold typeface

From Table 2 we can observe that when using minimal amounts of training data i.e. 10% of the full dataset, the HP-MPM method is nearly always the top performing algorithm. As the size of the training sample increases, the advantage of the HP-MPM begins to erode and its performance comes in line with the original MPM. This is to be expected in the case of large amounts of available data since we know that the HP-MPM will eventually converge towards the original MPM solution as moment uncertainty decreases to zero.

In Table 3 we present the performance of the kernelised version of the algorithms. Here we used the popular Gaussian kernel $k(\mathbf{x}, \mathbf{x}') = \exp(-||\mathbf{x} - \mathbf{x}'||^2/\sigma)$, where the width of the kernel $\sigma \in \{10^{-3}, \ldots, 10^3\}$ was chosen using the same validation scheme outlined

Table 3 Kernel experiments: we show how the performance of the classification algorithms on the datasets vary as the amount of data used during training changes

	Series training proportion						
	0.1	0.2	0.3	0.4	0.5	0.6	0.7
Australian							
MPM	83.71	85.67	85.71	85.93	86.06	86.12	86.23
HP-MPM	**85.06**	**85.91**	**86.16**	**86.01**	**86.15**	**86.25**	**86.49**
SVM	84.83	85.84	85.53	85.29	85.30	85.53	85.90
FDA	79.55	81.68	82.22	82.22	81.99	82.21	83.28
BCI							
MPM	59.35	62.71	**69.88**	**73.33**	**75.32**	**76.10**	**77.79**
HP-MPM	59.14	**65.00**	69.01	72.23	73.90	74.21	74.89
SVM	**59.73**	64.65	69.40	72.56	74.07	74.36	75.79
FDA	52.43	53.50	54.41	56.09	58.47	58.62	61.05
Breast							
MPM	96.31	97.08	**97.19**	**97.23**	**97.20**	**97.17**	97.09
HP-MPM	**97.07**	**97.18**	97.03	97.12	97.11	97.16	**97.09**
SVM	96.52	96.76	96.89	96.84	96.91	96.84	96.67
FDA	95.92	96.01	95.97	95.69	96.00	95.72	95.64
Diabetes							
MPM	72.50	74.25	74.95	75.08	75.09	74.88	74.98
HP-MPM	73.15	74.43	74.68	74.81	74.44	74.55	74.32
SVM	**74.39**	**75.99**	**76.21**	**76.52**	**76.47**	**76.76**	**76.79**
FDA	69.32	71.23	72.05	72.70	72.75	72.93	73.56
Digit1							
MPM	90.23	92.57	94.52	95.66	96.21	96.19	96.68
HP-MPM	**93.73**	**96.01**	96.84	**97.23**	97.46	97.43	97.58
SVM	92.96	95.85	**96.88**	97.22	**97.51**	**97.62**	97.66
FDA	91.84	94.65	95.94	96.72	97.21	97.34	**97.68**
German							
MPM	69.99	71.24	71.75	72.06	72.37	72.35	72.49
HP-MPM	70.63	71.47	72.07	72.37	72.43	72.65	72.99
SVM	**71.62**	**73.80**	**74.51**	**74.90**	**75.25**	**75.68**	**76.43**
FDA	69.97	70.15	69.97	69.87	69.89	70.17	70.15
Heart							
MPM	78.07	79.66	**80.32**	80.78	81.14	**81.75**	**82.32**
HP-MPM	**79.19**	**80.08**	80.14	80.79	80.89	81.58	81.71
SVM	78.84	79.68	80.23	**80.82**	**81.15**	81.46	81.57
FDA	76.82	77.91	78.01	78.45	78.90	79.04	78.99
Ionosphere							
MPM	80.93	83.14	87.18	89.10	90.25	90.78	91.38
HP-MPM	**91.04**	**93.38**	**93.89**	**94.15**	94.44	94.61	94.58
SVM	86.76	92.88	93.82	94.00	**94.53**	**95.07**	**95.62**
FDA	86.92	86.47	89.31	91.12	91.98	92.67	93.62

Table 3 continued

	Series training proportion						
	0.1	0.2	0.3	0.4	0.5	0.6	0.7
Ringnorm							
MPM	97.76	97.82	97.84	97.85	97.83	97.84	97.83
HP-MPM	**98.51**	98.52	98.53	98.51	98.50	98.51	98.49
SVM	98.47	**98.56**	**98.56**	**98.57**	**98.55**	**98.54**	**98.55**
FDA	96.89	95.68	95.65	95.64	95.61	95.59	95.59
Sonar							
MPM	66.37	71.85	76.00	78.52	80.96	82.83	84.60
HP-MPM	**70.09**	**76.37**	78.61	**81.34**	**84.00**	**84.68**	**86.87**
SVM	68.90	75.88	**78.94**	81.02	83.52	84.54	86.53
FDA	65.66	72.90	76.81	80.16	82.84	84.63	86.80
Splice							
MPM	82.90	85.00	85.90	86.51	86.81	86.81	86.81
HP-MPM	**85.19**	**87.87**	**89.23**	**89.95**	**90.40**	**90.40**	**90.40**
SVM	84.70	87.54	88.87	89.82	90.26	90.26	90.26
FDA	71.25	72.49	79.97	82.46	83.87	83.87	83.87
Toy							
MPM	87.92	92.52	93.45	94.24	94.00	94.45	94.50
HP-MPM	**91.00**	**93.57**	94.05	**94.53**	**94.71**	**94.55**	**95.25**
SVM	90.77	93.35	**94.15**	94.18	94.29	94.00	94.75
FDA	88.08	90.39	91.70	92.29	92.57	92.45	92.62
Twonorm							
MPM	97.57	97.64	97.68	97.69	97.69	97.70	97.70
HP-MPM	**97.72**	**97.74**	**97.76**	**97.77**	**97.77**	**97.77**	**97.76**
SVM	97.69	97.72	97.73	97.71	97.69	97.72	97.70
FDA	97.61	97.64	97.67	97.68	97.67	97.67	97.63
Vote							
MPM	92.40	94.95	95.70	95.72	95.76	**96.07**	**96.09**
HP-MPM	**94.91**	**95.51**	**95.72**	**95.84**	**95.80**	95.88	96.00
SVM	94.42	95.19	95.14	95.64	95.63	95.81	95.66
FDA	92.68	94.01	94.60	94.26	94.31	94.14	93.53

The best performing results for each dataset and training proportion are reported in bold typeface

earlier. We see that in general each algorithm's performance is similar to its performance in the linear setting, however there are noticeable improvements on the ionosphere, ringnorm and sonar datasets when using the Gaussian kernel. This suggests that these input spaces are better separated with a non-linear decision boundary, whereas for the others a simple linear decision boundary will suffice. In the kernelised form we see that the MPM approach to classification , MPM or HP-MPM, is extremely competitive with the SVM, being the top performing algorithm for a large proportion of the dataset/training set size combinations.

It would appear as though the validation procedure used to determine the parameters for the kernelised form of FDA failed to ensure that increased training data resulted in an improvement in the performance. This could be due to inappropriate values of λ used during regularisation, however there is very little guidance in the literature on a suitable degree of regularisation, whereas the HP-MPM has a simple range $\nu \in [0, 1]$ from which to choose. Furthermore, it is straightforward to work out what maximum value of ν will result in $\kappa > 0$ i.e. the conditions for non-zero κ require $\nu \in [0, 1]$ to satisfy $||\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0|| \geq \sqrt{2\nu A_1} + \sqrt{2\nu A_0}$.

The MPM schemes are generally competitive with the other approaches, however they seem to perform comparatively poorly, some 3% worse than the SVM, on the *german* dataset. This weakness of the MPM was previously identified in Huang et al. (2004), and is due to the MPMs prior assumption that the prior probability of each class is the same. We know from Table 1 that this is not the case for the *german* dataset, and that the probability of belonging to class 1 is much higher than class 0. One could foresee the HP-MPM making this situation potentially even worse given the nature in which in constructs its solution, and its natural bias towards placing the decision boundary closer to the mean of the class where moment uncertainty is lower i.e. the one with more observations. This is illustrated in Fig. 1, where we see the hyperplane is positioned nearer to the mean of class 0 because the training sample consists of many more observations from this class. Fortunately this is not the case and we see that the HP-MPM's performance is similar to that of the MPM. This is largely a result of the low levels of confidence in future performance i.e. small κ_*, which results in large levels of implicit regularisation for both classes as seen in the expression for the ellipsoids (11). This results in a decision boundary that is not overly biased towards predicting that new observations belong to the minority class. A simpler explanation of its similar performance can be given by the validation procedure that was used to determine what degree ν of regularisation to choose i.e. more likely that a smaller value of ν was used since it would place the decision boundary less close to the mean of the more common class, and therefore not be overly biased towards predicting that a new observation belongs to the less probable class.

To improve the performance of the HP-MPM on unbalanced training samples, we propose a simple solution that adjusts the bias term used in the construction of the decision boundary. During the training step we use the optimal weight vector $\mathbf{w}_*$ found using the standard HP-MPM algorithm, and then select the bias term to be the one that maximises the accuracy on the training set. These weight vectors and biases are then evaluated on the validation sample. Alternatively one could use the validation set to set the bias term, however this could be thought of as given this a glimpse of additional training samples and therefore an unfair advantage. Geometrically, this adjustment corresponds to a translational movement of the hyperplane where its direction $\mathbf{w}$ remains the same. In the *german* dataset, this corresponds to shifting the decision boundary in the direction of the mean of class 0, since we want to increase the probability that a new observation is predicted to belong to class 1. In Table 4 we show the results obtained on the *german* dataset by selecting the value of the bias term, when using the Gaussian kernel. We see that this simple approach to selecting the value of the bias term b, represented by column bHP-MPM in Table 4, improves the performance of the HP-MPM, correcting its implicit assumption that classes are equally likely, and brings its performance in line with the SVM. This would suggest that the direction $\mathbf{w}$ found by the HP-MPM is a useful method for discriminating between classes, and the bias term can be selected to take into consideration the relative class probabilities. However in doing so, the worst-case error rates that are found using the HP-MPM are no longer valid as we have repositioned the location of the separating hyperplane.

Table 4 *German* dataset: we evaluate the performance (classification accuracy) of selecting the bias term for the HP-MPM according to its performance on the validation set

Fraction	MPM (%)	HP-MPM (%)	bHP-MPM (%)	SVM (%)	FDA (%)
0.1	69.99	70.63	**73.04**	71.62	69.97
0.2	71.24	71.47	**74.33**	73.80	70.15
0.3	71.75	72.07	**75.22**	74.51	69.97
0.4	72.06	72.37	**75.42**	74.90	69.87
0.5	72.37	72.43	**76.08**	75.25	69.89
0.6	72.35	72.65	**76.11**	75.68	70.17
0.7	72.49	72.99	**76.72**	76.43	70.15

We see that this simple approach to adjusting the decision boundary, represented in column bHP-MPM, improves the performance of the HP-MPM, correcting for its implicit assumption that classes are equally likely, and brings its performance inline with the SVM

The best performing results for each training sample size are reported in bold

Table 5 *Adult* dataset lines experiment: we evaluate the performance (classification accuracy) of the proposed algorithm on a large scale dataset

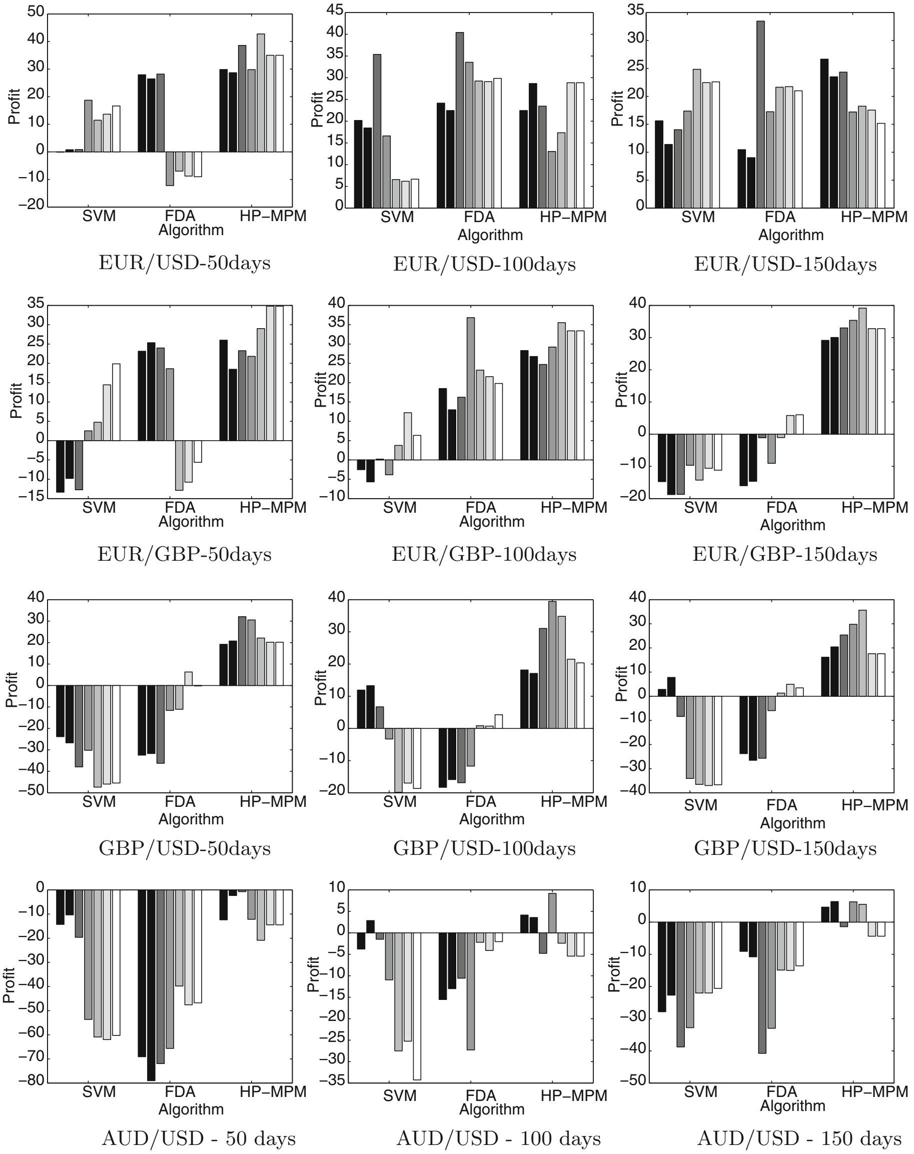
m	MPM (%)	HP-MPM (%)	bHP-MPM (%)	SVM (%)	FDA (%)
50	60.63	78.13	**79.35**	78.41	73.01
100	74.63	78.30	**81.37**	79.87	76.42
200	77.90	79.03	**82.37**	81.41	78.52
500	79.52	79.63	**83.30**	82.83	81.08
1000	80.08	80.07	**83.79**	83.53	82.61
5000	80.47	80.44	84.34	**84.46**	84.23
10,000	80.56	80.52	84.49	**84.65**	84.43

The number of training samples m is varied and we observe the changes in classifier performance. We see that with a small number of training examples the bHP-MPM tends to outperform the other approaches, with its relative advantage deteriorating as m increases

The best performing results for each training sample size are reported in bold

We evaluate the performance of the proposed method on the relatively large *adult* dataset with the results presented in Table 5. This table reports the performance using the linear version of all proposed methods. We see that the bHP-MPM approach is the top performing approach up until 5,000 training examples are provided to the learning algorithm, after which the SVM becomes the top performing predictor. This supports our argument that in the case of limited data availability the incorporation of moment uncertainty can improve the performance of predictors. As the number of data points increases and our information of the class-conditional distribution improves, the worst-case assumptions and the regularisation imposed by the HP-MPM, may hinder the construction of predictions, whereas the SVM is able to take advantage of better knowledge of the true periphery of the class-conditional distributions.

Currency movement prediction We conclude our experiments by testing the performance of the different classification algorithms on predicting the daily price movement of four common currency pairs. The daily foreign exchange (FX) data was freely downloaded from http://www.dukascopy.com, and ranges from October 2008 to October 2014. The currency pairs that we investigated were; EUR-GBP, EUR-USD, EUR-GBP and AUD-USD. We now



EUR/USD-50days EUR/USD-100days EUR/USD-150days

EUR/GBP-50days EUR/GBP-100days EUR/GBP-150days

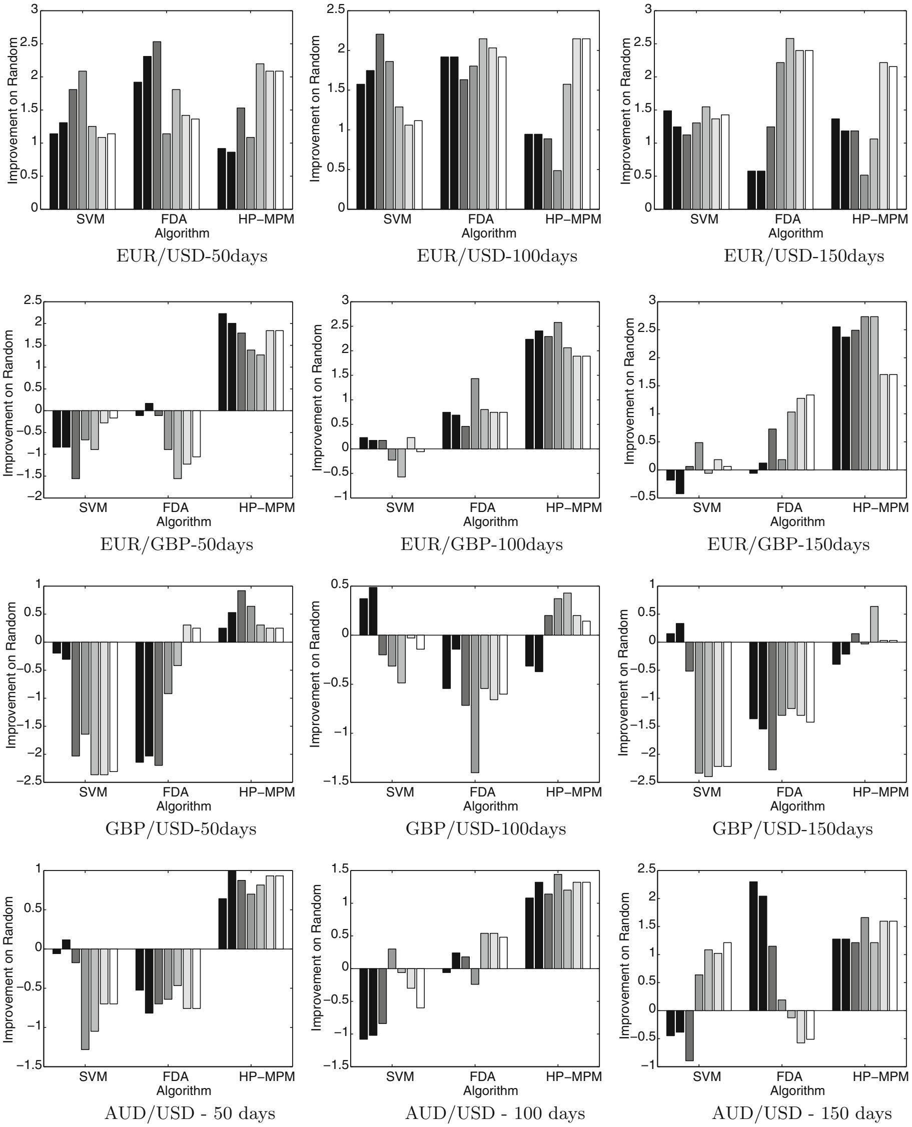
GBP/USD-50days GBP/USD-100days GBP/USD-150days

AUD/USD - 50 days AUD/USD - 100 days AUD/USD - 150 days

Fig. 2 Currency experiments: profit on trading decisions advised by the different algorithms. We see that the HP-MPM performs consistently well across the majority of settings (training window and regularisation). However, all of the algorithm seem to struggle with the AUD/USD currency pair

describe the classification setting used in the experiments. Let the opening price of the currency at day t be given by p_t, we represent the input space using a range of n-past log returns. For example, if $n = 3$, then the input representation $\mathbf{x}_t$ at time t is given by

$$\mathbf{x}_t = \left[\log\left(p_t/p_{t-1}\right), \log\left(p_{t-1}/p_{t-2}\right), \log\left(p_{t-2}/p_{t-3}\right)\right].$$



EUR/USD-50days EUR/USD-100days EUR/USD-150days

EUR/GBP-50days EUR/GBP-100days EUR/GBP-150days

GBP/USD-50days GBP/USD-100days GBP/USD-150days

AUD/USD - 50 days AUD/USD - 100 days AUD/USD - 150 days

Fig. 3 Currency experiments: improvement of accuracy on random guessing i.e. improvement over 50% correct. The HP-MPM appears to be the most consistently performing algorithm and only does worse than random guessing on two particular parameterisations. The other algorithms appear to perform quite considerably worse in terms of accuracy, with the SVM only consistently better than random for the EUR/USD currency pair

Given $\mathbf{x}_t$, our goal is to make a prediction whether we believe the price at the next time step will be higher than the current i.e. $y_t = 1$ if $p_{t+1} \geq p_t$, and $y_t = 0$ otherwise. In evaluating the performance of the algorithms we recorded not only the accuracy of the predictions, but also the hypothetical profit that would be made had we made a decision according to the advice of the predictor i.e. if we predicted the price to increase from t to $t+1$, then our return r_t would be the change in price over this time $r_t = (p_{t+1} - p_t)/p_t$. Similarly if we predicted the price would fall over this time horizon $r_t = (p_t - p_{t+1})/p_t$.

To train the model we implement a simple sliding window procedure that uses a fixed size number of examples (training window) to construct the predictor, which is then refreshed after a given number of observations (test window). By updating the predictor over time it is hoped that the predictor will be able to account for fact that the data is most likely not identically and independently distributed. Unfortunately we are unable to use the same validation technique that we used on the previous experiments, as it is likely that the most recent observations are the most important to the derivation of the predictor and we cannot make predictions based on observations in the future. Therefore in the results presented in Figs. 2 and 3 we have shown the performance of all of regularisation parameters for each classification algorithm. Given the multitude of different settings for these experiments and the limited space, we present only the results obtained when predictor is refreshed every 10 days, the input space is described using the last 5 log returns and we allow the training window to vary between 50, 100 and 200 days.

In Fig. 2 we present the hypothetical profits that would have been generated having traded on the prediction of the algorithms. We see that the HP-MPM approach performs consistently well across varying degrees of regularisation i.e. ν. It only fails to make profits on the AUD/USD currency pair, however its returns are often considerably better than those generated by the FDA or SVM. Similarly the accuracy of the HP-MPM is consistently on par with, if not exceeding that, of the other algorithms. On these datasets it would appear that the moment based algorithms, FDA and HP-MPM, perform better in terms of accuracy the SVM. We believe that this is largely due to the nature in which the solutions are constructed. The SVM will construct its solutions using points that it believes to lie on the boundary of the class-conditional distributions, whereas the moment based solutions are defined by the mean and covariances i.e. the majority of the data, rather than the outliers. Therefore when it comes to finding predictors in high-noise environments, the SVM will be constructing its solutions based on these outlying points rather than constructing it using the points that define the mass of the distribution.

7 Conclusions

In this paper we addressed an oversight of the original minimax probability machine (Lanckriet et al. 2003): that is, the worst-case future misclassification rates depend on prior knowledge of each classes mean and covariance matrix. In practice, these true moment values have to be substituted with their empirical counterparts, which are finite sample estimates of their true values. Making use of the high-probability bounds on the deviation of these estimates from their true values (Shawe-Taylor and Cristianini 2003), we derived a new optimisation scheme that takes into account the moment uncertainty and directly minimises the worst-case future misclassification rate that holds true with high-probability. We observed that in many experiments the moment uncertainty was so large that it was unable to produce meaningful results i.e. $\kappa_* = 0$. Therefore, at the expense of statistical correctness, we proposed to use fractional quantities of the true moment uncertainty as a form of regularisation. This form of regularisation, unlike most traditional schemes, implicitly takes into consideration the relative uncertainties regarding each class i.e. through different values of A_1 and A_0. During the experiments we noted that its performance was competitive with the popular SVM and FDA approaches, however its advantage was most apparent when minimal amounts of training data were used to construct the decision boundary thus providing support for this new approach to regularisation.

Earlier we briefly mentioned other learning algorithms that use the minimax formulation popularised by Lanckriet et al. (2003). Future work should investigate how best to include notion of moment uncertainty into these approaches. The minimum error MPM (MEMPM) (Huang et al. 2004) can be thought of as a more principled approach to our proposed bHP-MPM, taking into consideration the relative class probabilities in the construction of the decision hyperplane. Given its similarities to the original MPM approach, it should be straightforward to introduce the moment uncertainty into the MEMPM with an additional high-probability estimate on the prior class-probabilities. This approach could be used in place of our simple bias selection process, as a more principled approach to handling unbalanced training samples. Given that the minimax principle is used for the abundant class in Osadchy et al. (2015) it would seem unlikely that introducing moment uncertainty would be particularly beneficial. For the transductive (Huang et al. 2014) and clustering based (Huang et al. 2015) minimax approaches, the main difficulty of including moment uncertainty exist stems from the assignment of unlabelled data to classes. This would allow us to have control over the uncertainty surrounding each class and could inadvertently induce a bias that encourages equal numbers of observations for both classes. Despite these potential difficulties, the inclusion of moment uncertainty with existing minimax approaches remains an interesting area of research.

To improve the correctness of this approach, future work should focus on obtaining tighter bounds on the deviation of empirical moments from their true values. This would lead to statistically correct worst-case guarantees, whilst also circumventing the problem of having to use a validation set in order to choose the regularisation terms for the HP-MPM. We mentioned briefly that we have no reason to expect a sparse kernel based solution, making it difficult to handle large datasets. Future work should focus on developing specialised optimisation procedures for updating the weights of the dual variables $\boldsymbol{\gamma}$. If we are able to implement update schemes similar to those used in SVMs then we should be able to efficiently scale the HP-MPM approach to much larger datasets.

References

Alon, N., Ben-David, S., Cesa-Bianchi, N., & Haussler, D. (1997). Scale-sensitive dimensions, uniform convergence, and learnability. *Journal of the ACM (JACM)*, *44*(4), 615–631.

Bertsimas, D., & Popescu, I. (2005). Optimal inequalities in probability theory: A convex optimization approach. *SIAM Journal on Optimization*, *15*(3), 780–804.

Boser, B. E., Guyon, I. M., & Vapnik, V. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on computational learning theory* (pp. 144–152). ACM.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Human Genetics*, *7*(2), 179–188.

Ghaoui, L. E., Jordan, M. I., & Lanckriet, G. R. (2002). *Robust novelty detection with single-class MPM*. In *Advances in neural information processing systems* (pp. 905–912).

Huang, G., Song, S., Xu, Z. E., & Weinberger, K. (2014). Transductive minimax probability machine. In *Joint European conference on machine learning and knowledge discovery in databases* (pp. 579–594). Springer.

Huang, G., Zhang, J., Song, S., & Chen, Z. (2015). Maximin separation probability clustering. In *AAAI* (pp. 2680–2686).

Huang, K., Yang, H., King, I., Lyu, M. R., & Chan, L. (2004). The minimum error minimax probability machine. *The Journal of Machine Learning Research*, *5*, 1253–1286.

Lanckriet, G. R. G., Ghaoui, L. E., Bhattacharyya, C., & Jordan, M. I. (2003). A robust minimax approach to classification. *The Journal of Machine Learning Research*, *3*, 555–582.

Marchand, M., & Shawe-Taylor, J. (2002). The set covering machine. *Journal of Machine Learning Research*, *3*, 723–746.

Marshall, A. W., & Olkin, I. (1960). Multivariate chebyshev inequalities. *The Annals of Mathematical Statistics*, *31*(4), 1001–1014.

Osadchy, M., Hazan, T., & Keren, D. (2015). K-hyperplane hinge-minimax classifier. In *Proceedings of the 32nd international conference on machine learning (ICML-15)* (pp. 1558–1566).

Shawe-Taylor, J., & Cristianini, N. (2003). Estimating the moments of a random vector with applications. In *Proceedings of GRETSI 2003 conference* (pp. 47–52).

Shawe-Taylor, J., & Cristianini, N. (2004). *Kernel methods for pattern analysis*. Cambridge: Cambridge University Press.

Sokolova, M., Marchand, M., Japkowicz, N., & Shawe-Taylor, J. S. (2002). The decision list machine. In *Advances in neural information processing systems* (pp. 921–928).

Strohmann, T., & Grudic, G. Z. (2002). A formulation for minimax probability machine regression. In *Advances in neural information processing systems* (pp. 769–776).

Vapnik, V. (1995). *The nature of statistical learning theory*. New York: Springer. ISBN 0-387-94559-8.

Vapnik, V. (1998). *Statistical learning theory* (Vol. 2). New York: Wiley.

Vapnik, Vladimir, & Chervonenkis, A. Ya. (1971). On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, *16*(2), 264–280.