Multivariate Probability Calibration with Isotonic Bernstein Polynomials

Yongqiao Wang[1*] and **Xudong Liu**[2]
[1]College of Finance, Zhejiang Gongshang University, China
[2]School of Computing, University of North Florida, USA
{wangyq@zjsu.edu.cn,xudong.liu@unf.edu}

Abstract

Multivariate probability calibration is the problem of predicting class membership probabilities from classification scores of multiple classifiers. To achieve better performance, the calibrating function is often required to be coordinate-wise non-decreasing; that is, for every classifier, the higher the score, the higher the probability of the class labeling being positive. To this end, we propose a multivariate regression method based on shape-restricted Bernstein polynomials. This method is universally flexible: it can approximate any continuous calibrating function with any specified error, as the polynomial degree increases to infinite. Moreover, it is universally consistent: the estimated calibrating function converges to any continuous calibrating function, as the training size increases to infinity. Our empirical study shows that the proposed method achieves better calibrating performance than benchmark methods.

1 Introduction

To predict class membership probability for a given sample is a crucial component in various machine learning-based decision making systems [He *et al.*, 2015; Ustun and Rudin, 2019; Schwarz and Heider, 2019]. One approach to obtain such predictions is to acquire a probability function by estimating the joint distribution function of the class label and the sample vector. However, since the sample vector usually includes both categorical and continuous random variables, estimating this joint distribution function is a challenging task.

An alternative is *probability calibration* that consists of two steps as follows. First, it independently trains M classifiers, whose underlying scoring functions map a feature vector to a dimension-M score vector. Second, it trains a probability calibrating function that maps the dimension-M score vector to a membership probability. We see two advantages of this alternative over direct estimation. One, it incorporates well-established classification algorithms of choice to come up with sample scores. The other, it decreases the dimension of the joint distribution to M, assuming M is much smaller than the dimension of the feature vector.

When $M = 1$, i.e., there is only one classifier, many parametric and non-parametric methods have been proposed for probability calibration. The related work section of Wang *et al*. [2019] provides a survey of such methods for binary classification. In general, parametric methods suffer from the lack of flexibility and the vulnerability to mis-specification, while non-parametric calibration methods, including histogram binning and isotonic (or monotonic) regression, usually require a large amount of data to perform reasonably well.

Recently, researchers have shown that imposing isotonicity and smoothness on the calibrating function can greatly improve out-of-sample prediction performance, when the training size is small [Zadrozny and Elkan, 2002; Jiang *et al.*, 2011; Naeini and Cooper, 2016]. However, to require monotonicity is a rather difficult task, for it is imposed on every point in the domain along every dimension, which contributes infinitely many inequality constraints.

To this end, we propose to solve the *multivariate* probability calibration problem for binary classified samples, by shape-restricted regression with multivariate *Bernstein polynomials* (BernPolyFusion, for short). In statistics, univariate Bernstein polynomials have found many successful applications in nonlinear regression under shape-restrictions, such as monotonicity and convexity [Wang and Ghosh, 2012].

The key contributions of this work are as follows. Firstly, this non-parametric method has asymptotic *universal flexibility*; that is, as the polynomial degree increases, the proposed function family can approximate any continuous multivariate calibrating function with any specified error. This is a great advantage over Ozdemir *et al.* [2017], in which the specified parametric copula family limits flexibility. Secondly, the fitting calibrating function estimated by this method is coordinate-wise *isotonic* over the entire domain. This is an advantage over Zhong and Kwok [2013].

In this paper, all deterministic variables are in lower-case letters, while random variables are written in capital letters. Scalars are written in normal letters, and vectors are written in boldfaced letters. For example, the i-th element of constant vector $\mathbf{s}$ is s_i, and the i-th element of random vector $\mathbf{X}$ is X_i. We write $\mathbb{P}[A]$ the probability of event A, and $\mathbb{E}[\xi]$ the mathematical expectation of variable ξ. We denote by $\mathbb{I}\{\cdot\}$ the indicator function, and $\triangleq$ reads "equal to by definition".

*Corresponding Author

2 Problem Formulation

In a binary classification problem, one considers a random vector $(\mathbf{X}, Y)$, where the feature vector $\mathbf{X} \in \mathscr{X} \subset \mathbb{R}^d$ and the class label $Y \in \{0, 1\}$. The classification problem aims to predict the class label Y of each $\mathbf{x} \in \mathscr{X}$. A typical classification model first estimates a scoring function $s : \mathscr{X} \to \mathbb{R}$, then predicts the class label of $\mathbf{x}$ according to whether its score $s(\mathbf{x})$ is above or below a threshold. A good scoring function is expected to have good ranking power: for any $\mathbf{x}_1, \mathbf{x}_2 \in \mathscr{X}$, if $s(\mathbf{x}_1) > s(\mathbf{x}_2)$, $\mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}_1] \geq \mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}_2]$. Without loss of generality, this paper assumes $s(\mathscr{X}) \subset [0, 1]$. Or else one can achieve this by any increasing transformation.

Instead of class labels, this paper studies how to predict the conditional probability function

$$g : \mathscr{X} \to [0, 1], \quad g(\mathbf{x}) \triangleq \mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}]. \tag{1}$$

Then $\mathbb{P}[Y = 0|\mathbf{X} = \mathbf{x}] = 1 - g(\mathbf{x})$. One can obtain $g(\cdot)$ by estimating the joint distribution of $(\mathbf{X}, Y)$, i.e. two conditional distributions of $\mathbf{X}|Y = 1$ and $\mathbf{X}|Y = 0$, from a training data. However, this task is tough in many applications.

An alternative is post-processing that relies on classification models. Assume that M classification models have been independently trained, $\mathbf{s}(\mathbf{x}) \triangleq (s_1(\mathbf{x}), \cdots, s_M(\mathbf{x}))^\top$. If the calibrating function is

$$f : [0, 1]^M \to [0, 1], \quad f(\mathbf{s}) \triangleq \mathbb{P}[Y = 1|\mathbf{S} = \mathbf{s}], \tag{2}$$

the membership probability for $\mathbf{x}$ is

$$\mathbb{P}[Y = 1|\mathbf{X} = \mathbf{x}] = f(\mathbf{s}(\mathbf{x})). \tag{3}$$

$f(\cdot)$ should be estimated by calibration models from the training data $\{(\mathbf{s}(\mathbf{X}_i), Y_i)\}_{i=1}^n$. Please note that this calibration is necessary for probability prediction, since many classification methods produce scoring functions that have no direct relationship with membership probability [Guo *et al.*, 2017; Ott *et al.*, 2018; Kuleshov *et al.*, 2018; Kuleshov and Ermon, 2017]. For simplifying notation, we let $\mathbf{s} \triangleq \mathbf{s}(\mathbf{x})$, $\mathbf{S} \triangleq \mathbf{s}(\mathbf{X})$, $\mathbf{S}_i \triangleq \mathbf{s}(\mathbf{X}_i)$, $\mathcal{D}_n \triangleq \{(\mathbf{S}_i, Y_i)\}_{i=1}^n$.

3 Related Work

3.1 CopulaFusion

CopulaFusion, proposed by Ozdemir *et al.* [2017], estimates $f(\cdot)$ by the Bayesian theorem

$$f(\mathbf{s}) = \frac{\mathbb{P}[\mathbf{S} = \mathbf{s}|Y = 1] \times \mathbb{P}[Y = 1]}{\sum_{\ell=\{0,1\}} \mathbb{P}[\mathbf{S} = \mathbf{s}|Y = \ell] \times \mathbb{P}[Y = \ell]}. \tag{4}$$

Each $\mathbf{S}|Y = \ell$ has the following copula representation

$$\begin{aligned}&\mathbb{P}[S_1 \leq s_1, \cdots, S_M \leq s_M|Y = \ell]\\ =&C_\ell(\mathbb{P}[S_1 \leq s_1|Y = \ell], \cdots, \mathbb{P}[S_M \leq s_M|Y = \ell])\end{aligned} \tag{5}$$

where $C_\ell : [0, 1]^M \to [0, 1]$ is the copula function. In this representation, each multivariate distribution consists of M marginal distribution $\mathbb{P}[S_m \leq s_m|Y = \ell]$ and one copula function $C_\ell(\cdot)$.

To estimate the calibrating function $f(\cdot)$ from the training data $\mathcal{D}_n$, one should estimate two priors $\mathbb{P}[Y = \ell]$, $2M$ marginal conditional distributions $\mathbb{P}[S_m \leq s_m|Y = \ell]$, and two copulas C_ℓ. $\mathbb{P}[Y = \ell]$ can be replaced with its plug-in estimator $\sum_{i=1}^n \mathbb{I}_{\{Y_i=\ell\}}/n$. Each $\mathbb{P}[S_m \leq s_m|Y = \ell]$ can be obtained independently with kernel density estimation. Each C_ℓ can be obtained by maximum likelihood estimation.

3.2 MR-MIC

MR-MIC proposed by Zhong and Kwok [2013] estimates the membership probabilities $\{p_i\}_{i=1}^n$ for $\{\mathbf{X}_i\}_{i=1}^n$ with the following optimization

$$\min_{\mathbf{p}} \quad \sum_{i=1}^n (Y_i - p_i)^2 + \frac{\lambda}{2}\mathbf{p}^\top \mathbf{\Omega} \mathbf{p} \tag{6a}$$

$$s.t. \quad p_i \geq p_j, \text{ if } (i, j) \in \mathcal{E} \tag{6b}$$

where $\sum_{i=1}^n [Y_i - p_i]^2$ is the empirical error that is a plug-in estimator of the L_2 risk $\mathbb{E}[f(\mathbf{S}) - Y]^2$, and $\mathbf{p}^\top \mathbf{\Omega} \mathbf{p}$ is the manifold regularization for smoothness, and $\lambda > 0$ is a hyper-parameter for the trade-off between the empirical error and the regularization. Eq. (6b) is the isotonic constraint. $\mathcal{E}$ is the transitive reduction of the relation $\{(i, j)|\mathbf{S}_i > \mathbf{S}_j\}$.

However, the requirement of monotonicity only on ordered pairs of sample points cannot guarantee the monotonicity over the entire domain $[0, 1]^M$. It has the possibility of failing to keep monotone, especially when the training size is small.

4 Methodology

To keep the presentation simple, we limit our discussion to the case $M = 2$. The extension to $M \geq 3$ is straightforward. This paper estimates the calibrating function $f(\cdot)$ with a shape-restricted non-parametric regression

$$\min_{f \in \mathcal{F}} \sum_{i=1}^n [f(\mathbf{S}_i) - Y_i]^2 \tag{7}$$

where $\mathcal{F}$ is the set

$$\mathcal{F} \triangleq \left\{ f \in \mathcal{C}_{[0,1]^2} \,\middle|\, \begin{array}{l} f(0, 0) \geq 0, f(1, 1) \leq 1 \\ f \text{ is coordinate-wise non-decreasing} \end{array} \right\}$$

where $f \in \mathcal{C}_{[0,1]^2}$ means that f is continuous over $[0, 1]^2$. The requirement of coordinate-wise non-decrease arises from the ranking power of each scoring function.

The requirement of coordinate-wise non-decrease is very strong. Since it is imposed on every point of $[0, 1]^2$, the calibrating function is continuously constrained and involves an uncountable number of inequality constraints. Generally, this kind of optimization is a semi-infinite program [Reemtsen and Rückmann, 1998], because it involves a finite number of decision variables and an infinite number of inequality constraints. Over-restricted models specify simple functional families that are easy to impose shape restrictions, but these models achieve computational convenience at the expense of flexibility. Under-restricted models simplify these shape restrictions with finitely many inequality constraints on all samples or finite grid points, thus fail to guarantee full adherence.

This paper solves the probability calibration problem with shape-restricted multivariate Bernstein polynomials. Univariate Bernstein polynomials are widely regarded as the best shape-preserving functions and have been successfully applied to univariate shape-restricted non-parametric regression [Wang and Ghosh, 2012]. Let $f(\cdot)$ be approximated with a degree-(K_1, K_2) Bernstein polynomial

$$B_{K_1K_2}(\mathbf{s}) \triangleq \sum_{k_1=0}^{K_1} \sum_{k_2=0}^{K_2} \beta_{k_1k_2} b_{k_1,K_1}(s_1) b_{k_2,K_2}(s_2) \tag{8}$$

where $b_{k_m,K_m}(\cdot)$ is a univariate degree-K_m basis Bernstein polynomial

$$b_{k_m,K_m}(s_m) \triangleq \binom{K_m}{k_m} s_m^{k_m}(1-s_m)^{K_m-k_m}. \quad (9)$$

$B_{K_1K_2}$ is a linear combination of $(K_1+1)(K_2+1)$ bivariate basis Bernstein polynomials.

Theorem 1. *A sufficient condition for $B_{K_1K_2} \in \mathcal{F}$ is*

$$\beta_{00} \geq 0, \quad \beta_{K_1K_2} \leq 1 \quad (10)$$
$$\beta_{0k_2} \leq \beta_{1k_2} \leq \cdots \leq \beta_{K_1k_2}, \quad \forall k_2 \in \{0,\cdots,K_2\} \quad (11)$$
$$\beta_{k_10} \leq \beta_{k_11} \leq \cdots \leq \beta_{k_1K_2}, \quad \forall k_1 \in \{0,\cdots,K_1\}. \quad (12)$$

Proof. This proof is based on three properties of univariate basis Bernstein polynomials

$$b_{k,K}(0) = \begin{cases} 1 & k=0 \\ 0 & k=1,\cdots,K \end{cases}$$

$$b_{k,K}(1) = \begin{cases} 0 & k=0,\cdots,K-1 \\ 1 & k=K \end{cases}$$

$$\frac{\mathrm{d}b_{k,K}(s)}{\mathrm{d}s} = \begin{cases} -Kb_{0,K-1}(s) & \text{if } k=0 \\ Kb_{K-1,K-1}(s) & \text{if } k=K \\ K[b_{k-1,K-1}(s) - b_{k,K-1}(s)] & \text{otherwise.} \end{cases}$$

Since $B_{K_1K_2}(0,0) = \beta_{00}$ and $B_{K_1K_2}(1,1) = \beta_{K_1K_2}$, we have $B_{K_1K_2}(0,0) \geq 0 \Leftrightarrow \beta_{00} \geq 0$, and $B_{K_1K_2}(1,1) \leq 1 \Leftrightarrow \beta_{K_1K_2} \leq 1$. For all $\mathbf{s} \in [0,1]^2$, we have

$$\frac{\partial B_{K_1K_2}(\mathbf{s})}{K_1 \partial s_1} = \sum_{k_1=0}^{K_1-1}\sum_{k_2=0}^{K_2}\left[\beta_{(k_1+1)k_2} - \beta_{k_1k_2}\right] b_{k_2,K_2}(s_2).$$

Because $\forall s \in [0,1]$, $b_{k,K}(s) \geq 0$, Eq. (11) implies that $B_{K_1K_2}(s_1,s_2)$ is increasing with respect to s_1. The sufficiency of Eq. (12) for s_2 is similar. □

The probability calibration problem with two fore-going classification models can be solved by fitting a bivariate degree-(K_1,K_2) Bernstein polynomial with training samples $\{(S_{i1},S_{i2},Y_i)\}_{i=1}^n$. The corresponding quadratic program is

$$\min_{\boldsymbol{\beta}} \quad \sum_{i=1}^n \left[\sum_{k_1=0}^{K_1}\sum_{k_2=0}^{K_2} \beta_{k_1k_2} b_{k_1,K_1}(S_{i1}) b_{k_2,K_2}(S_{i2}) - Y_i\right]^2$$
$$s.t. \ \text{Eq.}(10)-(12). \quad (13)$$

This quadratic program is convex and tractable, which can be solved efficiently by many off-the-shelf optimization software. If the optimal solution for optimization (13) is $\boldsymbol{\beta}^*$, the estimated calibrating function is

$$\hat{f}_n(\mathbf{s}) = \sum_{k_1=0}^{K_1}\sum_{k_2=0}^{K_2} \beta^*_{k_1k_2} b_{k_1,K_1}(s_1) b_{k_2,K_2}(s_2). \quad (14)$$

let $\mathcal{F}_K$ be the family of shape-restricted degree-K bivariate Bernstein polynomials

$$\mathcal{F}_K \triangleq \Big\{B_K(\mathbf{s}) = \sum_{k_1=0}^K\sum_{k_2=0}^K \beta_{k_1k_2} b_{k_2,K}(s_2) b_{k_2,K}(s_2) : \boldsymbol{\beta} \text{ satisfies Eq. (10)-(12)}\Big\}. \quad (15)$$

4.1 Universal Flexibility

This subsection proves that $\cup_{K=1}^\infty \mathcal{F}_K$ is dense in $\mathcal{F}$ with respect to sup-norm, which means that shape-restricted Bernstein polynomials can be used to approximate any continuous coordinate-wise non-decreasing calibrating function with any arbitrary accuracy as the degree K increases to infinite.

Theorem 2. *The sequence of function families $\mathcal{F}_K$ is nested in $\mathcal{F}$, i.e. $\mathcal{F}_1 \subset \mathcal{F}_2 \subset \cdots \subset \mathcal{F}_K \subset \cdots \subset \mathcal{F} \subset L_2[0,1]^2$, where $f \in L_2[0,1]^2$ means $\int_{[0,1]^2} |f(\mathbf{s})|^2 \mathrm{d}\mathbf{s} < +\infty$.*

Proof. (I) $\mathcal{F}_K \subset \mathcal{F}_{K+1}, \forall K$. For any $B_K \in \mathcal{F}_K$, $B_K(\mathbf{s}) = \sum_{k_1=0}^K\sum_{k_2=0}^K \beta_{k_1k_2} b_{k_1,K}(s_1) b_{k_2,K}(s_2)$, we can rewritten it as a degree-$(K+1)$ Bernstein polynomial

$$B_K(\mathbf{s}) = \sum_{k_1=0}^{K+1}\sum_{k_2=0}^{K+1} \tilde{\beta}_{k_1k_2} b_{k_2,K+1}(s_2) b_{k_2,K+1}(s_2) \quad (16)$$

where

$$\begin{aligned}\tilde{\beta}_{k_1k_2} =& \frac{(K+1-k_1)(K+1-k_2)}{(K+1)^2}\beta_{k_1k_2} \\ &+ \frac{(K+1-k_1)k_2}{(K+1)^2}\beta_{k_1(k_2-1)} \\ &+ \frac{k_1(K+1-k_2)}{(K+1)^2}\beta_{(k_1-1)k_2} \\ &+ \frac{k_1k_2}{(K+1)^2}\beta_{(k_1-1)(k_2-1)}. \end{aligned} \quad (17)$$

The above statement is obtained by the iterative property of univariate Bernstein polynomials

$$b_{k,K}(s) = \frac{K+1-k}{K+1} b_{k,K+1}(s) + \frac{k+1}{K+1} b_{k+1,K+1}(s).$$

To verify $B_K \in \mathcal{F}_{K+1}$, we should further prove that $\tilde{\boldsymbol{\beta}} \in \mathbb{R}^{(K+2)\times(K+2)}$ satisfies

$$\tilde{\beta}_{00} \geq 0, \tilde{\beta}_{(K+1)(K+1)} \leq 1$$
$$\tilde{\beta}_{0k_2} \leq \tilde{\beta}_{1k_2} \leq \cdots \leq \tilde{\beta}_{(K+1)k_2}, \forall k_2 \in \{0,\cdots,K+1\}$$
$$\tilde{\beta}_{k_10} \leq \tilde{\beta}_{k_11} \leq \cdots \leq \tilde{\beta}_{k_1(K+1)}, \forall k_1 \in \{0,\cdots,K+1\}.$$

The first requirement can be verified by $\tilde{\beta}_{00} = \beta_{00}$, $\tilde{\beta}_{(K+1)(K+1)} = \beta_{KK}$ and Eq. (10). The second requirement can be obtained by Eq. (11) and

$$\begin{aligned}&\tilde{\beta}_{(k_1+1)k_2} - \tilde{\beta}_{k_1k_2} \\ =& \frac{(K-k_1)(K+1-k_2)}{(K+1)^2}\left[\beta_{(k_1+1)k_2} - \beta_{k_1k_2}\right] \\ &+ \frac{k_1(K+1-k_2)}{(K+1)^2}\left[\beta_{k_1k_2} - \beta_{(k_1-1)k_2}\right] \\ &+ \frac{(K-k_1)k_2}{(K+1)^2}\left[\beta_{(k_1+1)(k_2-1)} - \beta_{k_1(k_2-1)}\right] \\ &+ \frac{k_1k_2}{(K+1)^2}\left[\beta_{k_1(k_2-1)} - \beta_{(k_1-1)(k_2-1)}\right]. \end{aligned} \quad (18)$$

The proof for the third requirement is similar.

(II) Since all basis Bernstein polynomials and their linear combinations belong to $L_2[0,1]^2$, $\mathcal{F}_K \subset L_2[0,1]^2, \forall K$. □

Theorem 3. *$\cup_{K=1}^{\infty}\mathcal{F}_K$ is dense in $\mathcal{F}$ with respect to sup-norm, i.e. for every $f \in \mathcal{F}$, we have*

$$\lim_{K\to\infty} \min_{B_K\in\mathcal{F}_K} \sup_{\mathbf{s}\in[0,1]^2} |f(\mathbf{s}) - B_K(\mathbf{s})| = 0. \tag{19}$$

Proof. According to DeVore and Lorentz [1993, p.10], for any multivariate continuous function f, there are multivariate Bernstein polynomials that converge uniformly to f as the degrees increase to infinite. So that the set of unconstrained bivariate Bernstein polynomials is dense in $\mathcal{C}_{[0,1]^2}$ with respect to super-norm. $\mathcal{F} \subset \mathcal{C}_{[0,1]^2}$ implies that, for each $f \in \mathcal{F}$, for any approximation error $\epsilon > 0$, there exists a $K \in \mathbb{N}$ such that the following function

$$B_K(\mathbf{s}) = \sum_{k_1=0}^{K}\sum_{k_2=0}^{K} f\left(k_1/K, k_2/K\right) b_{k_1,K}(s_1) b_{k_2,K}(s_2)$$

satisfies $\sup_{\mathbf{s}\in[0,1]^2} |B_K(\mathbf{s}) - f(\mathbf{s})| < \epsilon$. Therefore, if we let

$$\boldsymbol{\beta} \in \mathbb{R}^{(K+1)(K+1)}, \beta_{k_1k_2} = f\left(k_1/K, k_2/K\right), \tag{20}$$

$\boldsymbol{\beta}$ satisfies Eq. (10) - (12) by the properties of f. Thus, $B_K(\cdot) \in \mathcal{F}_K$. Combining this with Theorem 2, we obtain Eq. (19). □

4.2 Consistency

The problem studied in this paper is called random-design regression. Each $(\mathbf{S}_i, Y_i)$ is random, and $(\mathbf{S}, Y)$, $(\mathbf{S}_1, Y_1)$, $\cdots$, $(\mathbf{S}_n, Y_n)$ are independent and identically distributed. To measure the error of a regression estimate, we use the L_2 error: $\int_{\mathbf{s}\in[0,1]^2} |\hat{f}_n(\mathbf{s}) - f(\mathbf{s})|^2 \mathrm{d}(\mathbf{s})$.

Since the estimate $\hat{f}_n$ depends on the data $\mathcal{D}_n$, this L_2 error is a random variable. The following theorem on the consistency of the estimator relies on Lemma 1 [Györfi *et al.*, 2006, Lemma 10.1].

Lemma 1. *Let $\mathcal{F}_n = \mathcal{F}_n(\mathcal{D}_n)$ be a class of functions $\tilde{f} : \mathbb{R}^d \to \mathbb{R}$ depending on the data $\{(\mathbf{S}_i, Y_i)\}_{i=1}^n$. If $\hat{f}_n$ is obtained by*

$$\hat{f}_n = \arg\min_{\tilde{f}\in\mathcal{F}_n} \frac{1}{n}\sum_{i=1}^{n} \left|\tilde{f}(\mathbf{S}_i) - Y_i\right|^2 \tag{21}$$

then

$$\begin{aligned}
&\int \left|\hat{f}_n(\mathbf{s}) - f(\mathbf{s})\right|^2 \mu(\mathrm{d}\mathbf{s}) \\
\le& 2 \sup_{\tilde{f}\in\mathcal{F}_n} \left| \frac{1}{n}\sum_{i=1}^{n} \left[\tilde{f}(\mathbf{S}_i) - Y_i\right]^2 - \mathbb{E}\left[\tilde{f}(\mathbf{S}) - Y\right]^2 \right| \\
&+ \inf_{\tilde{f}\in\mathcal{F}_n} \int \left|\tilde{f}(\mathbf{s}) - f(\mathbf{s})\right|^2 \mu(\mathrm{d}\mathbf{s})
\end{aligned} \tag{22}$$

where μ denotes the distribution of $\mathbf{S}$.

Theorem 4. *Provided that $\hat{f}_n$ is obtained by Eq.* (13) - (14)*, if the degree K satisfies*

$$K \uparrow \infty, \quad K^2/n \to 0 \quad (n \to \infty), \tag{23}$$

then,

(I) $\hat{f}_n$ is strongly universally consistent, i.e. for any $f \in \mathcal{F}$, as $n \to \infty$

$$\int_{[0,1]^2} \left|\hat{f}_n(\mathbf{s}) - f(\mathbf{s})\right|^2 \mu(\mathrm{d}\mathbf{s}) \to 0, \qquad a.s. \tag{24}$$

(II) $\hat{f}_n$ is weakly universally consistent, i.e. for any $f \in \mathcal{F}$, as $n \to \infty$,

$$\mathbb{E}\int_{[0,1]^2} \left|\hat{f}_n(\mathbf{s}) - f(\mathbf{s})\right|^2 \mu(\mathrm{d}\mathbf{s}) \to 0. \tag{25}$$

Proof. (I) STRONG CONSISTENCY. Due to the Lemma 1, to prove the strong universal consistency of $\hat{f}_n$, it suffices to verify that, as $n \to \infty$,

$$\sup_{\tilde{f}\in\mathcal{F}_n} \left| \frac{1}{n}\sum_{i=1}^{n} \left[\tilde{f}(\mathbf{S}_i) - Y_i\right]^2 - \mathbb{E}\left[\tilde{f}(\mathbf{S}) - Y\right]^2 \right| \to 0, a.s. \tag{26}$$

$$\inf_{\tilde{f}\in\mathcal{F}_n} \int_{[0,1]^2} \left|\tilde{f}(\mathbf{s}) - f(\mathbf{s})\right|^2 \mu(\mathrm{d}\mathbf{s}) \to 0, \quad a.s. \tag{27}$$

Let us first prove Eq. (27). According to Theorem 3, $\cup_{K=1}^{\infty}\mathcal{F}_K$ is dense in $\mathcal{F}$ with respect to sup norm, which follows that $\cup_{K=1}^{\infty}\mathcal{F}_K$ is dense in $\mathcal{F}$ with respect to $L_2(\mu)$ for any distribution μ. It implies that, for every $f \in \mathcal{F}$, for any arbitrarily small error $\epsilon > 0$, there exists a $B_{K_0(\epsilon)} \in \cup_{K=1}^{\infty}\mathcal{F}_K$ that satisfies $\int_{[0,1]^2} |B_{K_0(\epsilon)}(\mathbf{s}) - f(\mathbf{s})|^2 \mu(\mathrm{d}\mathbf{s}) < \epsilon$. Because $K \uparrow \infty$ as $n \to \infty$, there is a $n_0(\epsilon) \in \mathbb{N}$ such that $K > K_0(\epsilon)$ for all $n \ge n_0(\epsilon)$. Combining this with the nested property of $\mathcal{F}_n$s, one concludes that

$$\inf_{\tilde{f}\in\mathcal{F}_n} \int_{[0,1]^2} \left|\tilde{f}(\mathbf{s}) - f(\mathbf{s})\right|^2 \mu(\mathrm{d}\mathbf{s}) < \epsilon, \quad \forall n \ge n_0(\epsilon). \tag{28}$$

Since $\epsilon > 0$ is arbitrary, this implies Eq. (27).

To verify Eq. (26), let $\mathcal{F}_n^+$ be the class of all subgraphs of functions of $\mathcal{F}_n$

$$\mathcal{F}_n^+ \triangleq \left\{ \left\{ (\mathbf{s}, t) \in [0,1]^{M+1} : t \le \tilde{f}(\mathbf{s}) \right\} : \tilde{f} \in \mathcal{F}_n \right\}, \tag{29}$$

and $V_{\mathcal{F}_n^+}$ be the Vapnik-Chervonenkis (VC) dimension of $\mathcal{F}_n^+$. Since each $\tilde{f} \in \mathcal{F}_n$ is a linear combination of $(K+1)^2$ basis Bernstein polynomials, and $\mathcal{F}_n^+$ is a subset of

$$\left\{ \left\{ (\mathbf{s}, t) \in [0,1]^{M+1} : \tilde{f}(\mathbf{s}) + \alpha \cdot t \ge 0 \right\} : \tilde{f} \in \mathcal{F}_n, \alpha \in \mathbb{R} \right\}$$

which is a vector space with dimension $(K+1)^2 + 1$ of real functions on $\mathbb{R}^3$, by Theorem 9.5 of [Györfi *et al.*, 2006], we have

$$V_{\mathcal{F}_n^+} \le (K+1)^2 + 1. \tag{30}$$

Then we have

$$\begin{aligned}
&\mathbb{P}\left\{ \sup_{\tilde{f}\in\mathcal{F}_n} \left| \frac{1}{n}\sum_{i=1}^{n} \left[\tilde{f}(\mathbf{S}_i) - Y_i\right]^2 - \mathbb{E}\left[\tilde{f}(\mathbf{S}) - Y\right]^2 \right| > \epsilon \right\} \\
\le& 24 \left(\frac{32e}{\epsilon}\log\frac{48e}{\epsilon}\right)^{(K+1)^2+1} \exp\left\{-\frac{n\epsilon^2}{128}\right\} \\
\le& 24 \left(\frac{48e}{\epsilon}\right)^{2(K+1)^2+2} \exp\left\{-\frac{n\epsilon^2}{128}\right\} \\
=& 24 \exp\left\{ \left(2(K+1)^2 + 2\right)\log\frac{48e}{\epsilon} - \frac{n\epsilon^2}{128} \right\}.
\end{aligned} \tag{31}$$

The proof for the first inequality is same as [Wang *et al.*, 2019, Theorem 4]. The second inequality follows from $\log(x) \leq x - 1 \leq x$ for $x > 0$. Therefore,

$$\begin{aligned}
&\sum_{n=1}^{\infty} \mathbb{P}\left[\sup_{\tilde{f}\in\mathcal{F}_n}\left|\frac{1}{n}\sum_{i=1}^{n}\left[\tilde{f}(\mathbf{S}_i)-Y_i\right]^2-\mathbb{E}\left[\tilde{f}(\mathbf{S})-Y\right]^2\right|>\epsilon\right] \\
&\leq \sum_{n=1}^{\infty} 24\exp\left\{\left(2(K+1)^2+2\right)\log\frac{48e}{\epsilon}-\frac{n\epsilon^2}{128}\right\} \\
&= \sum_{n=1}^{\infty} 24\exp\left\{-n\left(\frac{\epsilon^2}{128}-2\frac{(K+1)^2+1}{n}\log\frac{48e}{\epsilon}\right)\right\}.
\end{aligned}$$

Hence, provided that $\lim_{n\to\infty} K^2/n \to 0$, we have

$$\sum_{n=1}^{\infty} \mathbb{P}\left[\sup_{\tilde{f}\in\mathcal{F}_n}\left|\frac{1}{n}\sum_{i=1}^{n}\left[\tilde{f}(\mathbf{S}_i)-Y_i\right]^2-\mathbb{E}\left[\tilde{f}(\mathbf{S})-Y\right]^2\right|>\epsilon\right] < \infty. \quad (32)$$

By the Borel-Cantelli lemma, we obtain Eq. (26). Therefore, Eq. (23) is a sufficient condition for the strong universal convergence of $\hat{f}_n(\cdot)$.

(II) WEAK CONSISTENCY. To prove the weak universal consistency of $\hat{f}_n(\cdot)$, it suffices to verify, as $n \to \infty$,

$$\mathbb{E}\left\{\sup_{\tilde{f}\in\mathcal{F}_n}\left|\frac{1}{n}\sum_{i=1}^{n}\left[\tilde{f}(\mathbf{S}_i)-P_i\right]^2-\mathbb{E}\left[\tilde{f}(\mathbf{S})-Y\right]^2\right|\right\}\to 0 \quad (33)$$

$$\mathbb{E}\left\{\inf_{\tilde{f}\in\mathcal{F}_n}\int_{[0,1]^2}\left[\tilde{f}(\mathbf{s})-f(\mathbf{s})\right]^2\mu(\mathrm{d}\mathbf{s})\right\}\to 0. \quad (34)$$

Because Eq. (34) directly follows from Eq. (28), it is sufficient to verify Eq. (33). If ξ is a nonnegative random variable, for an arbitrary $\epsilon > 0$, we have

$$\mathbb{E}[\xi] = \int_0^{\infty}\mathbb{P}[\xi > t]\mathrm{d}t \leq \epsilon + \int_{\epsilon}^{\infty}\mathbb{P}[\xi > t]\mathrm{d}t. \quad (35)$$

Following this,

$$\begin{aligned}
&\mathbb{E}\left[\sup_{\tilde{f}\in\mathcal{F}_n}\left|\frac{1}{n}\sum_{i=1}^{n}\left[\tilde{f}(\mathbf{S}_i)-Y_i\right]^2-\mathbb{E}\left[\tilde{f}(\mathbf{S})-Y\right]^2\right|\right] \\
&\leq \epsilon + \int_{\epsilon}^{\infty} 24\left(\frac{48e}{t}\right)^{2(K+1)^2+2}\exp\left\{-\frac{nt^2}{128}\right\}\mathrm{d}t \\
&\leq \epsilon + 24\left(\frac{48e}{\epsilon}\right)^{2(K+1)^2+2}\int_{\epsilon}^{\infty}\exp\left\{-\frac{n\epsilon t}{128}\right\}\mathrm{d}t \\
&= \epsilon + 24\left(\frac{48e}{\epsilon}\right)^{2(K+1)^2+2}\frac{128}{n\epsilon}\exp\left\{-\frac{n\epsilon^2}{128}\right\} \\
&= \epsilon + \frac{3072}{n\epsilon}\exp\left\{-n\left(\frac{\epsilon^2}{128}-2\frac{(K+1)^2+1}{n}\log\frac{48e}{\epsilon}\right)\right\}.
\end{aligned}$$

The first inequality is obtained by Eq. (35) and (31). Eq. (33) holds if $K^2/n \to 0$ as $n \to \infty$. Hence, the condition (23) is sufficient for the weak universal consistency of $\hat{f}_n(\cdot)$. □

5 Experiments

In this experiment, to make each $s_m(\mathscr{X}) \in [0,1]$, every score $s_m(\mathbf{X}_i)$ is transformed by the empirical cumulative distribution function using $\{s_m(\mathbf{X}_i)\}_{i=1}^n$. In CopulaFusion, the copula function for each class is chosen according to the criterion of maximum likelihood. Candidate copula families are Gaussian, Clayton, rotated Clayton, Plackett, Frank, Gumbel, rotated Gumbel and Student's t. In MR-MIC, $\mathbf{\Omega}$ is the similarity matrix with $\omega_{ij} = 1/\|\mathbf{S}_i - \mathbf{S}_j\|$. Because model (Eq.6) only obtains calibrated probabilities $\{\hat{p}_i\}_{i=1}^n$ for training samples $\{\mathbf{S}_i\}_{i=1}^n$, we estimate the calibrating function by linear interpolation with the scattered data $\{(\mathbf{S}_i, \hat{p}_i)\}_{i=1}^n$. Quadratic optimizations for MR-MIC and BernPolyFusion are solved with CVX [Grant and Boyd, 2014].

5.1 Characteristics of Calibrating Functions

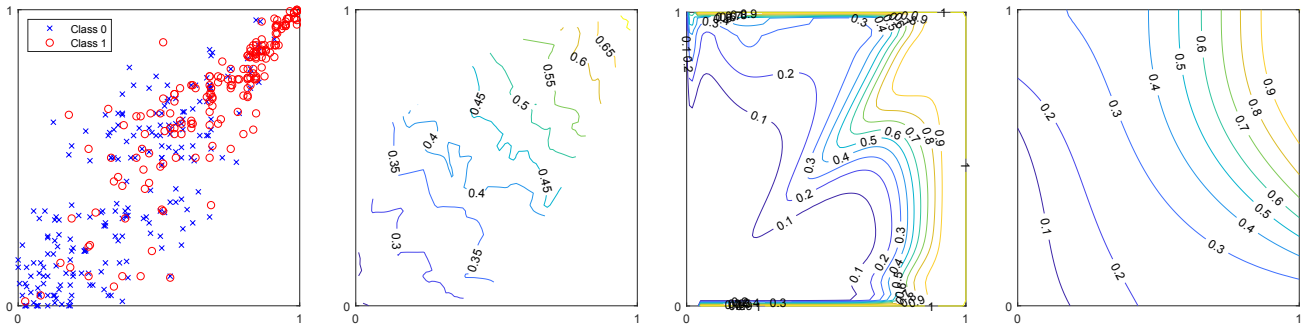
The data is SUSY from UCI Machine Learning Repository [Dua and Graff, 2017]. Two fore-going classifiers are feed-forward networks with hidden layer sizes 10 and 20. We randomly draw 200 samples for training two classifiers and 400 samples for training the calibrating model. The hyper-parameters are arbitrarily chosen: $\lambda = 10^{-5}$ in MR-MIC, and $K_1 = K_2 = 5$ in the proposed method. Figure 1(a) shows that both classifiers have good ranking power.

Figure 1(b)-1(d) shows the main characteristics of three estimated calibrating functions with contours. In all figures, by and large, the estimated calibrating probability increases as $\mathbf{S}$ moves from southwest to northeast. Among the three methods, only BernPolyFusion can adhere to the requirement of coordinate-wise non-decrease. MR-MIC imposes this requirement only on training samples $\{\mathbf{S}_i\}_{i=1}^n$, while CopulaFusion does not explicitly impose this requirement. The calibrating function from BernPolyFusion is more smooth than MR-MIC and CopulaFusion.

5.2 Model Comparison

Ideal performance measures for probability calibration should be based on the difference between the true f and the estimated $\hat{f}$, e.g. $\sup_{s\in[0,1]}|f(\mathbf{s}) - \hat{f}(\mathbf{s})|$ and $\int_{s\in[0,1]}|f(\mathbf{s}) - \hat{f}(\mathbf{s})|\mathrm{d}\mathbf{s}$. However, in practice the true f is unavailable. When the size of the test data is very large, f can be approximated in the following way: the domain $[0,1]^2$ is discretized with a uniformly-spaced grid (10×10) and the true probability at each grid cell is approximated with the percent of class-1 among samples that falls in this grid cell.

Because the majority of scores scatter around the anti-diagonal, there are few test samples scatter in grid cells around northwest and southeast corners. Thus, approximated probabilities at these grid cells are subject to large deviations. Therefore, each grid cell with the number of test samples smaller than 1000 is discarded from performance measurement. The following two measures are used: MCE $= \max_{\mathbf{s}\in\mathcal{G}}|f(\mathbf{s}) - \hat{f}(\mathbf{s})|$ and ECE $= \sum_{\mathbf{s}\in\mathcal{G}}|f(\mathbf{s}) - \hat{f}(\mathbf{s})|/|\mathcal{G}|$, where $\mathcal{G}$ is the set of centers of grid cells with the test number no less than 1000. The third performance measure is the Brier score, which is the mean squared difference between predicted probabilities and class labels.



(a) Data (b) MR-MIC (c) CopulaFusion (d) BernPolyFusion

Figure 1: Training data and contours of three calibrating functions

	Method	Adult	Census	Covertype	Dota2	SUSY	Mean
MCE	MR-MIC	6.142±0.843	7.116±0.825	5.785±0.611	**5.353**±0.576	6.185±0.479	6.116±0.667
	CopulaFusion	**5.309**±0.723	6.817±0.611	5.882±0.626	6.290±0.508	6.592±0.469	6.178±0.587
	BernPolyFusion	5.613±0.650	**6.241**±0.565	**5.367**±0.434	5.591±0.528	**5.054**±0.394	**5.573**±0.514
ECE	MR-MIC	2.459±0.441	3.787±0.468	3.444±0.359	3.124±0.393	3.182±0.258	3.199±0.384
	CopulaFusion	2.548±0.296	3.890±0.444	3.369±0.326	3.319±0.343	2.963±0.240	3.218±0.330
	BernPolyFusion	**2.226**±0.309	**3.587**±0.370	**2.847**±0.231	**2.799**±0.268	**2.432**±0.241	**2.727**±0.284
Brier	MR-MIC	15.485±0.915	19.719±1.338	19.776±0.861	18.740±0.758	19.015±0.646	18.547±0.904
	CopulaFusion	16.441±0.703	19.714±1.095	18.067±0.824	19.264±0.711	17.771±0.506	18.252±0.768
	BernPolyFusion	**14.375**±0.854	**18.966**±0.724	**16.095**±0.577	**16.811**±0.434	**15.539**±0.473	**16.357**±0.612

Table 1: Model Comparison (%. The best performance is highlighted.)

Experiments are performed on five large data from [Dua and Graff, 2017] with size larger than 40,000: Adult, Census, Covertype, Dota2 and SUSY. In Covertype, the largest class (Lodgepole Pine) is the positive class and others are the negative class. For each data, three size-500 datasets are randomly drawn. The first dataset is used for training classifier 1, and the second dataset is used for training classifier 2, and the third dataset is used for training calibration models. All other samples are used for testing calibration models. In each training step, we use 5-fold cross-validation to determine hyperparameters. Two fore-going classifiers are three-layer feed-forward networks. The above procedure is repeated 10 times, and model comparison is based on the average and standard deviation of ten rounds.

Table 1 demonstrates that the proposed method outperforms benchmark models in all but two instances, where CopulaFusion bests on Adult and MR-MIC on Dota2, both with respect to MCE measure. Considering the results averaged over all five datasets, we see our BernPolyFusion unanimously performs the best.

5.3 Running Time

The complexity of CopulaFusion depends on the choice of the copula family. For some copula families, the optimization for the maximum likelihood estimation is non-convex. The training of MR-MIC involves a quadratic program with n decision variables. An ADMM-based algorithm for this optimization has a complexity $O(n^2)$. The training of BernPolyFusion involves a quadratic program with $(K+1)^2$ decision variables and $2K(K+1)+2$ constraints. Both numbers are independent of the training size n. According to Section 4.6.2 of [Ben-Tal, 2019], Data(P) and Size(p) are linearly increasing with n, thus the Newton complexity of ϵ-solution is $O(n)$.

For the aforementioned experiments in Subsection 5.2 where $n = 500$, the CPU time in seconds for training MR-MIC, CopulaFusion, and BernPolyFusion are 1.335 ± 0.128, 1.486 ± 0.172, and 0.501 ± 0.044, respectively. If n is 1,000, the CPU time in seconds for three methods are 7.964 ± 0.807, 1.756 ± 0.246 and 0.872 ± 0.090, respectively. Therefore, compared to MR-MIC and CopulaFusion, we see that BernPolyFusion attains less running time and better scalability.

6 Conclusions

A novel method based on shape-restricted Bernstein polynomial regression is proposed for probability calibration. This method has universal fitting flexibility and is both strongly and weakly universally consistent. Experimental results show that this method has a great advantage over benchmark methods. Future work includes the extension from binary-class to multi-class or structured classification problems [Kuleshov and Liang, 2015; Leathart *et al.*, 2019].

Acknowledgments

The work of the first author was supported by NSFC (71571163) and Zhejiang NSF (LY19G010001).

References

[Ben-Tal, 2019] Aharon Ben-Tal. Lectures on modern convex optimization. Technical report, School of Industrial & Systems Engineering, Georgia Institute of Technology, 2019.

[DeVore and Lorentz, 1993] Ronald A. DeVore and George G. Lorentz. *Constructive Approximation*. Springer-Verlag, 1993.

[Dua and Graff, 2017] Dheeru Dua and Casey Graff. UCI Machine Learning Repository, 2017.

[Grant and Boyd, 2014] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx, March 2014.

[Guo *et al.*, 2017] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proc. of ICML'17*, pages 1321–1330, 2017.

[Györfi *et al.*, 2006] László Györfi, Michael Kohler, Adam Krzyzak, and Harro Walk. *A Distribution-free Theory of Nonparametric Regression*. Springer Science & Business Media, 2006.

[He *et al.*, 2015] Jiazhen He, James Bailey, Benjamin I. P. Rubinstein, and Rui Zhang. Identifying at-risk students in massive open online courses. In *Proc. of AAAI'15*, pages 1749–1755, 2015.

[Jiang *et al.*, 2011] Xiaoqian Jiang, Melanie Osl, Jihoon Kim, and Lucila Ohno-Machado. Smooth isotonic regression: A new method to calibrate predictive models. *AMIA Summits on Translational Science Proceedings*, 2011:16, 2011.

[Kuleshov and Ermon, 2017] Volodymyr Kuleshov and Stefano Ermon. Estimating uncertainty online against an adversary. In *Proc. of AAAI'17*, pages 2110–2116, 2017.

[Kuleshov and Liang, 2015] Volodymyr Kuleshov and Percy S Liang. Calibrated structured prediction. In *Advances in Neural Information Processing Systems 28*, pages 3474–3482. MIT Press, 2015.

[Kuleshov *et al.*, 2018] Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *Proc. of ICML'18*, pages 2801–2809, 2018.

[Leathart *et al.*, 2019] Tim Leathart, Eibe Frank, Bernhard Pfahringer, and Geoffrey Holmes. On calibration of nested dichotomies. In *Proc. of PAKDD'19*, pages 69–80, 2019.

[Naeini and Cooper, 2016] Mahdi Pakdaman Naeini and Gregory F Cooper. Binary classifier calibration using an ensemble of near isotonic regression models. In *Proc. of ICDM'16*, pages 360–369, 2016.

[Ott *et al.*, 2018] Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. Analyzing uncertainty in neural machine translation. In *Proc. of ICML'18*, pages 3956–3965, 2018.

[Ozdemir *et al.*, 2017] Onur Ozdemir, Thomas G Allen, Sora Choi, Thakshila Wimalajeewa, and Pramod K Varshney. Copula based classifier fusion under statistical dependence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(11):2740–2748, 2017.

[Reemtsen and Rückmann, 1998] Rembert Reemtsen and Jan-J Rückmann. *Semi-infinite Programming*, volume 25. Springer Science & Business Media, 1998.

[Schwarz and Heider, 2019] Johanna Schwarz and Dominik Heider. GUESS: projecting machine learning scores to well-calibrated probability estimates for clinical decision-making. *Bioinformatics*, 35(14):2458–2465, 2019.

[Ustun and Rudin, 2019] Berk Ustun and Cynthia Rudin. Learning optimized risk scores. *Journal of Machine Learning Research*, 20:1–75, 2019.

[Vellanki *et al.*, 2019] Pratibha Vellanki, Santu Rana, Sunil Gupta, David Rubin de Celis Leal, Alessandra Sutti, Murray Height, and Svetha Venkatesh. Bayesian functional optimisation with shape prior. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1617–1624, 2019.

[Wang and Ghosh, 2012] J. Wang and S.K. Ghosh. Shape restricted nonparametric regression with bernstein polynomials. *Computational Statistics & Data Analysis*, 56(9):2729 – 2741, 2012.

[Wang *et al.*, 2019] Y. Wang, L. Li, and C. Dang. Calibrating classification probabilities with shape-restricted polynomial regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1813–1827, 2019.

[Zadrozny and Elkan, 2002] Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proc. of KDD'02*, pages 694–699, 2002.

[Zhong and Kwok, 2013] Leon Wenliang Zhong and James T Kwok. Accurate probability calibration for multiple classifiers. In *Proc. of IJCAI'13*, pages 1939–1945, 2013.