DETERMINISTIC BRIDGE REGRESSION FOR COMPRESSIVE CLASSIFICATION

KAR-ANN TOH[1], GIUSEPPE MOLTENI[2], AND ZHIPING LIN[3]

Abstract. Pattern classification using a compact representation is a crucial component of machine intelligence. Specifically, it is essential to learn a model with well-regulated parameters to achieve good generalization. Bridge regression provides a mechanism for regulating parameters through a penalized ℓ_p-norm. However, due to the nonlinear nature of the formulation, an iterative numerical search is typically used to solve the optimization problem. In this work, we propose an analytic solution for bridge regression based on solving a penalized error formulation using an approximated ℓ_p-norm. The solution is presented in primal form for over-determined systems and in dual form for under-determined systems. The primal form is suitable for low-dimensional problems with a large number of data samples, while the dual form is suitable for high-dimensional problems with a small number of data samples. We also extend the solution to problems with multiple classification outputs. Numerical studies using simulated and real-world data demonstrate the effectiveness of our proposed solution.

Information Sciences **648**, n. 119505 pp. 22 (2023).
electronically published on August 14, 2023.
DOI: https://doi.org/10.1016/j.ins.2023.119505

1. Introduction

Pattern classification is an important component for decision making in signal, image and information processing. Apart from the classifier model and the data concerned, an accurate classification prediction is also hinged upon factors such as feature representation and model complexity. In particular, an appropriate selection or weighting between informative features and non-informative ones can be crucial towards accurate prediction [1]. Moreover, a compact representation of features not only serves towards resource savage, but also possible feature discovery [2]. This leads to the attention of compressed sensing [3] that seeks a sparse solution of systems concerned. To enforce sparseness of estimation, a penalized ℓ_0-norm can be incorporated for learning optimization. In view of the NP hard nature of the ℓ_0-norm formulation [4] for feature selection, alternative solutions for sparse estimation considering ℓ_1 [5] and ℓ_p norms [6] have been investigated. Due to the nonlinear nature of the formulation, an iterative search is often adopted for solution seeking, where convergence becomes a concern.

Besides the features compression aspect, there are also situations when data samples are scarce. For examples, scarce images of rare disease [7], archaeological samples [8] and other objects may cast difficulties in effective learning. Several approaches are available to deal with this situation. These approaches include reduction of model complexity [9, 10], data augmentation [11], transfer learning [12] and attention mechanism [13]. Depending on the availability of supplementary information and the requirement of applications, each approach has its strengths and limitations. The approaches by data augmentation, transfer learning and attention mechanism might provide accurate prediction. However, their successful adoption is highly hinged upon matching of the distribution of augmented, pretrained or focused data with respect to that of the unknown target data. The approach by model complexity reduction offers a simplified model but might face the bottleneck when the data for learning is not representative.

In view of the lack of a deterministic solution for analysis and the outlook for an exactly converged recursive form for online applications, we seek a proximal solution in analytic form to bridge regression for compressive classification. To deal with problems of small sample size, we utilize only the given data and work on the approach of model complexity reduction to

2020 *Mathematics Subject Classification.* Primary 68T05, Secondary 62H30.

Key words and phrases. Pattern Classification, Least Squares Regression, Ridge Regression, Bridge Regression, Compressive Estimation.

suppress non-informative variables. Different from the iterative approach to solve the ℓ_0, ℓ_1, elastic-net and ℓ_p formulations (see e.g., [9, 10, 14]), we formulate a novel deterministic solution and algorithm for solving the bridge regression which utilizes the ℓ_p norm penalty on weight coefficients with the error cost function. The main contributions of this work are enumerated as follows:

- Based on an approximation to the ℓ_p-norm, an analytic solution for the bridge regression has been derived independently in primal form for over-determined systems, and in dual form for under-determined systems. The solution in primal form is found to have a similar expression as that obtained based on a different cost function utilizing a local quadratic approximation as the regularization term. The solution in dual form does not find any precedent in the literature as there has been no attempt from this perspective to the best of our knowledge. This formulation can be useful for few-shots learning when data is scarce.
- The analytic solution in primal and dual forms are extended to solve for problems with multiple outputs. In pattern classification, this formulation is useful for multiple category prediction. Although one could stack multiple predictors of single-output for multi-outputs prediction, the proposed solution for multi-outputs has the advantage of having a common covariance for output alignment.
- An algorithm that packs the two analytic solutions for multiple outputs under a single estimation framework has been constructed. The embodied under-determined and multiple output solutions are not seen in the existing bridge regression implementation in R. An extensive study has been performed based on both simulated data and real-world data sets. Both the solution of the primal form and the dual form show stretchable weight coefficients estimation for different penalty settings. The solution in dual form shows higher trade-off between prediction accuracy and coefficient sparseness than that of solution in primal form.

The significance of this research outcome lies in the establishment of a set of solutions for bridge regression in deterministic form that is useful for analysis and learning recognition applications. Such a deterministic form not only guarantees the convergence of solution but also is computationally more efficient since it does not need an iterative search for solution. This opens up the feasibility of a future development for a convergent bridge solution for online compressive learning of data that arrives sequentially.

The remainder of this paper is organized as follows. Section 2 provides a brief account of mathematical preliminaries and related works. Section 3 presents the proposed methodology for solving bridge regression in primal form and in dual form. The solutions for binary classification problems are subsequently extended to multi-class problems. Section 4 shows some numerical case studies for observing the profile of coefficient shrinkage. Section 5 contains extensive experiments to demonstrate the effectiveness of the proposed solution. Section 6 provides an observation of results and discussion. Section 7 summarizes the paper with some concluding remarks.

2. Related Works

Given a data set $\{\boldsymbol{x}_i, y_i\}_i^M$ of M samples, the *ordinary linear least squares* [15, 16] regression utilizes the model

$$y_i = \boldsymbol{x}_i^T \boldsymbol{\alpha} + \epsilon_i, \quad \boldsymbol{\alpha} \in \mathbb{R}^D, \tag{1}$$

to minimize the sum of squared errors $\sum_{i=1}^M \epsilon_i^2 = \sum_{i=1}^M (y_i - \boldsymbol{x}_i^T\boldsymbol{\alpha})^2$ with an optimal estimation given by $\hat{\boldsymbol{\alpha}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ where $\mathbf{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M]^T$ and $\mathbf{y} = [y_1, \ldots, y_M]^T$ for $M \geq D$ when $\mathbf{X}^T\mathbf{X} \in \mathbb{R}^{D\times D}$ has full rank. For the situation when $M < D$ and when $\mathbf{X}\mathbf{X}^T \in \mathbb{R}^{M\times M}$ has full rank, the solution given by $\hat{\boldsymbol{\alpha}} = \mathbf{X}^T \left(\mathbf{X}\mathbf{X}^T\right)^{-1} \mathbf{y}$ is known as the *least norm* solution, which is exact. Here, we pack them together and call it the *ols* solution as follows:

$$\hat{\boldsymbol{\alpha}} = \begin{cases} \left(\mathbf{X}^T\mathbf{X}\right)^{-1} \mathbf{X}^T\mathbf{y}, & M \geq D \quad \text{(primal form)} \\ \mathbf{X}^T \left(\mathbf{X}\mathbf{X}^T\right)^{-1} \mathbf{y}, & M < D \quad \text{(dual form)} \end{cases}. \tag{2}$$

The ols, which minimizes the residual sum of squared errors, provides an unbiased estimation. However, the estimation comes with large variance when the input features have collinearity. The *ridge* regression [17] regularizes the ols learning by inclusion of a penalty to the weight coefficients $\boldsymbol{\alpha}$ (also known as learning parameters) based on the ℓ_2-norm. For $\lambda > 0$, the resulted solution for minimizing $\sum_{i=1}^{M}(y_i - \boldsymbol{x}_i^T\boldsymbol{\alpha})^2 + \lambda\sum_{j=1}^{D}\alpha_j^2$ can be written as

$$\hat{\boldsymbol{\alpha}} = \begin{cases} \left(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}\right)^{-1}\mathbf{X}^T\mathbf{y}, & M \geq D \quad \text{(primal form)} \\ \mathbf{X}^T\left(\mathbf{X}\mathbf{X}^T + \lambda\mathbf{I}\right)^{-1}\mathbf{y}, & M < D \quad \text{(dual form)} \end{cases}, \tag{3}$$

where $\mathbf{I}$ is an identity matrix congruence to its summing term. Effectively, the ridge regression provides an estimation with a set of shrunk weight coefficients relative to that of the ols. Due to the utilization of the ℓ_2-norm penalty, the shrinkage is uniform in each of the D dimensions. The ridge regression circumvents the problem of singularity by shrinking the estimator via penalizing the weight coefficients in ℓ_2-norm during minimization. Since then, the penalized models have evolved beyond the ℓ_2-norm dealing with challenges related to features weighting and selection.

The *bridge* regression [6] generalizes the ridge regression by replacing the ℓ_2-norm penalty with an ℓ_p-norm penalty of the weight coefficients where the range of p is commonly taken within $0 < p \leq 2$ for compression reason. In other words, the bridge regression uses the following cost function for minimization:

$$\sum_{i=1}^{M}(y_i - \boldsymbol{x}_i^T\boldsymbol{\alpha})^2 + \lambda\sum_{j=1}^{D}|\alpha_j|^p\ . \tag{4}$$

Due to the difficulty in dealing with the absolute operator and the nonlinear formulation, an analytic or closed-form solution is yet to be available. Moreover, a consolidated treatment according to the under-determined and the over-determined scenarios are not available in the literature. The formulation for the under-determined scenario is particularly useful when the data is scarce. This was termed *small sample size* (SSS) problem [18, 19] before the deep learning era and is also known as *few-shot learning* [20, 21] in the current literature.

The bridge regression was first seen in [6] where several statistical tools for chemometrics regression were studied. According to the study, the parameter p can be viewed as the degree to which the prior probability is concentrated along the favored directions. A value of $p \to 0$ places the prior mass towards the directions of the coordinate axes, expressing the prior belief that only a few of the predictor variables are likely to have high relative influence on the response. When $p = 0$, the penalized learning is known as *variable subset selection* [16]. When $p = 1$, the *least absolute shrinkage and selection operator* (lasso) [5] shrinks the estimator with some parameters being zero based on the ℓ_1-norm penalty. In order to encompass both capabilities of variable selection and variable shrinkage, the *elastic-net* [22] leverages amidst ridge and lasso by weighting between the ℓ_1-norm and the ℓ_2-norm penalties. Different from the elastic-net, the *bridge regression* [6] penalizes the sum of squared errors by the ℓ_p-norm of weight coefficients. It does variable selection when $0 < p \leq 1$, and shrinks the coefficients when $p > 1$. For $1 < p < 2$, the *bridge* regression shrinks the coefficients unevenly with a higher penalty towards those less relevant ones. Attributed to the general penalty form of ℓ_p-norm, the bridge regression fits well into situations when it needs variable selection or weighting.

The structure of bridge was studied in [23] where a general approach to solve the bridge regression for $p \geq 1$ was developed. The algorithm, which was based on a modified Newton-Raphson method, solved iteratively for the unique solution for bridge for $p \geq 1$. According to [9], the solution for bridge is continuous only when $p \geq 1$. In their proposal, a local quadratic approximation (LQA) has been adopted iteratively for the ℓ_p penalized likelihood. It turned out that the minimization problem can be reduced to a quadratic minimization problem where the Newton-Raphson algorithm can be adopted to search for a solution [9]. In [10], a local linear approximation has been proposed to replace the local quadratic approximation for solving the penalized likelihood with better computational efficiency. In [24], both the local linear and local quadratic approximations have been studied. They showed that the bridge estimator is a robust choice under various circumstances comparing with ridge, lasso, and elastic net. In [25], the author introduced an interesting discussion regarding the metric relationship among several penalized norms.

From the perspective of balancing between computation and compression in applications, the work of [26] used an iterative alternating direction method of multipliers (ADMM) for computing the Elastic net penalized quantile regression. In another instance, the work of [27] considered incorporation of uncertainty penalty into Bayesian bridge quantile regression. For evaluation of bridge regression models, the work of [28] proposed a Bayesian selection criterion where the LQA approximation has been adopted for bridge penalty. To expedite the multi-task lasso, the work of [29] proposed a feature elimination rule. To gain control of the interpretability of decision trees, the work of [30] integrated the lasso regularization in the tree induction to find the best set of attributes that built a regression model.

Collectively, the bridge regression fits well into situations when it needs variable selection or weighting involving penalized norm values lower than two. An iterative search for the solution has been the mainstream approach where convergence becomes a concern. Moreover, a consolidated treatment between the over-determined systems and the under-determined systems is lacking. In particular, the under-determined systems can lead to the small sample size problem where learning generalization is a concern. We address these issues by proposing a proximal bridge regression paying particular attention to over-determined and under-determined systems.

3. Method: Proximal Bridge Regression

In this section, we present an analytic solution for an approximated bridge regression called *proximal bridge* regression. The solution comes in primal form for over-determined systems and in dual form for under-determined systems. We shall introduce an approximation to the ℓ_p-norm in section 3.1 before presenting the two solution forms in sections 3.2-3.3. Sections 3.4-3.6 present the multiple output extension, variance analysis and the algorithm construction.

3.1. A k-measure for ℓ_p-norm approximation.

Consider a positive valued penalty term that is an approximation of the ℓ_p-norm, in which the absolute value operator is replaced by a differentiable function f_ϵ:

$$|\wr\boldsymbol{\alpha}\wr|_k := \left(\sum_{j=1}^{D} f_\epsilon(\alpha_j)^k\right)^{1/k}, \tag{5}$$

where $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_D]^T$ is a parameter vector. Here, the power term k replaces p in the ℓ_p-norm to indicate the approximation. A convenient choice for approximating the absolute operator, which can be efficiently computed, is $f_\epsilon(\alpha_j) = \sqrt{\alpha_j^2 + \epsilon} \approx |\alpha_j|$, $\epsilon > 0$ (see [31, 32]) and Fig. 1). Note that $\lim_{\epsilon \to 0} f_\epsilon(\cdot) = |\cdot|$ for arbitrary $\epsilon > 0$. For finite ϵ values, the function $|\wr\boldsymbol{\alpha}\wr|_k$ is not a norm because it does not have the absolute homogeneity property. We shall call $|\wr\cdot\wr|_k$ (5) a *k-measure* operator for convenience hereon.

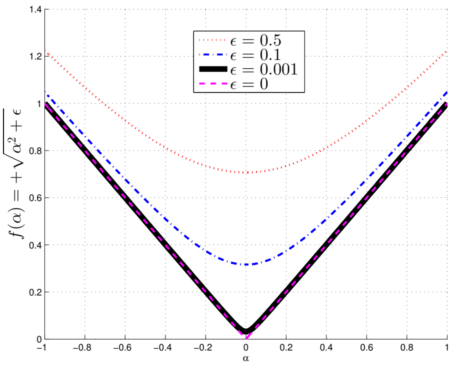


Figure 1. Plot of $f(\alpha) = \sqrt{\alpha^2 + \epsilon}$ at several ϵ values.

In the following, the raised power form of k-measure is shown to be convex when the approximation function $f_\epsilon(\cdot)$ is convex.

Lemma 1. *For all $k \geq 1$, $|\wr\boldsymbol{\alpha}\wr|_k^k$ is convex on $\boldsymbol{\alpha}$ when f_ϵ is convex.*

Proof: Based on the convexity of f_ϵ on each element α_i, $i = 1, \ldots, D$ (of the parameter vector $\boldsymbol{\alpha}$), we have

$$f_\epsilon(\mu\alpha_{i1} + (1-\mu)\alpha_{i2}) \leq \mu f_\epsilon(\alpha_{i1}) + (1-\mu) f_\epsilon(\alpha_{i2}), \quad 0 \leq \mu \leq 1. \tag{6}$$

Suppose $h(\theta) := \theta^k$ with $k, \theta > 0$ where we know that h is nondecreasing and convex on θ since $dh/d\theta = k\theta^{k-1} \geq 0$ and $d^2h/d\theta^2 = k(k-1)\theta^{k-2} \geq 0$, $\forall k \geq 1$ (see also [33]). Using (6) plus the fact that h is nondecreasing and convex, we have for each $i = 1, \dots, D$,

$$\begin{aligned} h(f_\epsilon(\mu\alpha_{i1} + (1-\mu)\alpha_{i2})) &\leq h(\mu f_\epsilon(\alpha_{i1}) + (1-\mu)f_\epsilon(\alpha_{i2})) \\ &\leq \mu h(f_\epsilon(\alpha_{i1})) + (1-\mu)h(f_\epsilon(\alpha_{i2})), \quad 0 \leq \mu \leq 1. \end{aligned} \tag{7}$$

Since summation of convex functions preserves the convexity, we have

$$\sum_{i=1}^{D} h(f_\epsilon(\mu\alpha_{i1} + (1-\mu)\alpha_{i2})) \leq \sum_{i=1}^{D} \mu h(f_\epsilon(\alpha_{i1})) + (1-\mu)h(f_\epsilon(\alpha_{i2})), \quad 0 \leq \mu \leq 1, \tag{8}$$

which means convexity of $\sum_{i=1}^{D} h(\alpha_i) = \sum_{i=1}^{D} f_\epsilon(\alpha_i)^k = \wr\boldsymbol{\alpha}\wr_k^k$ on $\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_D]^T$. ■

Fig. 2 shows the contours of the ℓ_p-norm metric for $1 < p \leq 2$ and the corresponding k-measure ($\wr\boldsymbol{\alpha}\wr_k$, (5)) together with its k-powered form ($\wr\boldsymbol{\alpha}\wr_k^k$) within the same interval. From the third row of plots in Fig. 2, except for the difference in curvature, we see that the entire k-measure and its k-powered form approximate well to the solution p-norm for the plotted range of $1 < \{p, k\} \leq 2$. This suggests vertices of the vector space being feasible solutions for the desired constrained solution search. Such an observation shall be exploited in the following development for compressive solution when $1 < k < 2$.

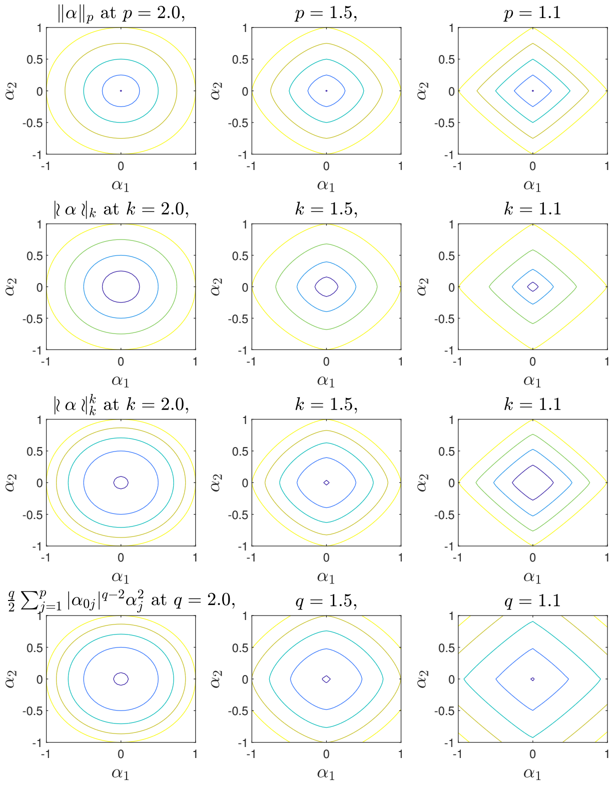


FIGURE 2. Contour plots at levels [0.1, 0.4, 0.7, 1]. Top row: $\|\boldsymbol{\alpha}\|_p$ for $p \in \{2.0, 1.5, 1.1\}$. Second row: $\wr\boldsymbol{\alpha}\wr_k$ for $k \in \{2.0, 1.5, 1.1\}$ at $\epsilon = 0.0001$. Third row: $\wr\boldsymbol{\alpha}\wr_k^k$ for $k \in \{2.0, 1.5, 1.1\}$ at $\epsilon = 0.0001$. Bottom row: $\frac{q}{2}\sum_{j=1}^{p}|\alpha_{0j}|^{q-2}\alpha_j^2$ for $q \in \{2.0, 1.5, 1.1\}$ with $\alpha_{0j} = \alpha_j - 0.01$.

The bottom row of Fig. 2 shows another approximation of the ℓ_p-norm by a Local Quadratic Approximation (LQA, [9]) given by

$$|\alpha_j|_q^q \approx |\alpha_{0j}|^q + \frac{q}{2}\frac{|\alpha_{0j}|^{q-1}}{|\alpha_{0j}|}(\alpha_j^2 - \alpha_{0j}^2), \tag{9}$$

where the minimization problem of an approximated bridge regression can be expressed as

$$\arg\min_{\boldsymbol{\alpha}} \left\{ (\mathbf{y}-\mathbf{X}\boldsymbol{\alpha})^T(\mathbf{y}-\mathbf{X}\boldsymbol{\alpha}) + \frac{q}{2}\sum_{j=1}^{D} |\alpha_{0j}|^{q-2}\,\alpha_j^2 \right\}. \tag{10}$$

The plots of LQA in this row have been generated based on $\frac{q}{2}\sum_{j=1}^{p} \mid \alpha_{0j} \mid^{q-2}\ \alpha_j^2$ for $q \in \{2.0, 1.5, 1.1\}$ at $\alpha_{0j} = \alpha_j - 0.01$. These plots show much difference between the adopted k-measure and the existing LQA [9].

3.2. **Proximal bridge regression in primal form.** For over-determined systems, we minimize the sum of squared errors with a k-measure penalty as shown in Theorem 1 below. We call such minimization *proximal bridge regression in primal form* (or *primal p-bridge* in brief). In this formulation, $\circ$ denotes an element-wise operator. For example, $\mathbf{A}^{\circ k}$ indicates raising each element of $\mathbf{A}$ to the power k. Also, $\operatorname{diag}(\boldsymbol{a})$ denotes a diagonal matrix with its diagonal elements given by vector $\boldsymbol{a}$, and $\operatorname{eig}_j(\mathbf{A})$ denotes the jth eigenvalue of matrix $\mathbf{A}$.

Theorem 1. *Given the data $\{\boldsymbol{x}_i, y_i\}$, $i = 1, \ldots, M$ where $\boldsymbol{x}_i = [x_{i,1}, \cdots, x_{i,D}]^T$ and y_i are respectively the regressors and the response for the ith observation. Consider the linear regression model $\mathbf{X}\boldsymbol{\alpha}$ with parameter vector $\boldsymbol{\alpha} \in \mathbb{R}^D$ and regressor matrix $\mathbf{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_M]^T$. Suppose $\mathbf{X}^T\mathbf{X}$ is of full rank. Then, under the limiting case of $\boldsymbol{\epsilon} \to \mathbf{0}$ and for $k \geq 1$, $\hat{\boldsymbol{\alpha}}$ that satisfies*

$$\boldsymbol{\alpha} \;=\; \left(\frac{\lambda k}{2}\operatorname{diag}\{|\boldsymbol{\alpha}|^{\circ(k\,-\,2)}\} + \mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y} \tag{11}$$

minimizes

$$(\mathbf{y}-\mathbf{X}\boldsymbol{\alpha})^T(\mathbf{y}-\mathbf{X}\boldsymbol{\alpha}) + \lambda|\wr\boldsymbol{\alpha}\wr|_k^k, \tag{12}$$

when the matrix $(\frac{\lambda k}{2}\operatorname{diag}\{|\boldsymbol{\alpha}|^{\circ(k\,-\,2)}\} + \mathbf{X}^T\mathbf{X})$ is invertible. This happens for sure as soon as

$$\frac{\lambda k}{2}\max_{j\in\{1,\ldots,D\}}(|\alpha_j|^{(k-2)}) < \min_{j\in\{1,\ldots,D\}}(\operatorname{eig}_j(\mathbf{X}^T\mathbf{X})). \tag{13}$$

Proof: According to the definition of k-measure in (5), let $\bar{\boldsymbol{\alpha}} := [(\alpha_0^2+\epsilon)^{k/4}, \cdots, (\alpha_{D-1}^2+\epsilon)^{k/4}]^T$ where we can write $|\wr\boldsymbol{\alpha}\wr|_k = (\bar{\boldsymbol{\alpha}}^T\bar{\boldsymbol{\alpha}})^{1/k}$ and $|\wr\boldsymbol{\alpha}\wr|_k^k = (\bar{\boldsymbol{\alpha}}^T\bar{\boldsymbol{\alpha}})$. Next, take the first derivative of (12) with respect to $\boldsymbol{\alpha}$ and set it to zero:

$$\begin{aligned}
\frac{\partial}{\partial\boldsymbol{\alpha}}\left((\mathbf{y}-\mathbf{X}\boldsymbol{\alpha})^T(\mathbf{y}-\mathbf{X}\boldsymbol{\alpha}) + \lambda\bar{\boldsymbol{\alpha}}^T\bar{\boldsymbol{\alpha}}\right) &= \mathbf{0}\\
-2\mathbf{X}^T(\mathbf{y}-\mathbf{X}\boldsymbol{\alpha}) + \lambda\frac{k}{4}\cdot 2\boldsymbol{\alpha}\circ\left(\boldsymbol{\alpha}^{\circ 2}+\boldsymbol{\epsilon}\right)^{\circ(\frac{k}{4}\,-\,1)}\circ 2\bar{\boldsymbol{\alpha}} &= \mathbf{0}\\
-2\mathbf{X}^T(\mathbf{y}-\mathbf{X}\boldsymbol{\alpha}) + \lambda k\,\boldsymbol{\alpha}\circ\left(\boldsymbol{\alpha}^{\circ 2}+\boldsymbol{\epsilon}\right)^{\circ(\frac{k}{4}\,-\,1)}\circ\left(\boldsymbol{\alpha}^{\circ 2}+\boldsymbol{\epsilon}\right)^{\circ\frac{k}{4}} &= \mathbf{0}\\
\Rightarrow\quad \lambda k\,\boldsymbol{\alpha}\circ\left(\boldsymbol{\alpha}^{\circ 2}+\boldsymbol{\epsilon}\right)^{\circ(\frac{k}{2}\,-\,1)} &= 2\mathbf{X}^T(\mathbf{y}-\mathbf{X}\boldsymbol{\alpha}).
\end{aligned} \tag{14}$$

For the limiting case of $\boldsymbol{\epsilon}$, we have

$$\begin{aligned}
\lim_{\boldsymbol{\epsilon}\to\mathbf{0}} k\,\boldsymbol{\alpha}\circ\left(\boldsymbol{\alpha}^{\circ 2}+\boldsymbol{\epsilon}\right)^{\circ(\frac{k}{2}\,-\,1)} &= k\,\boldsymbol{\alpha}\circ\left(\boldsymbol{\alpha}^{\circ 2}\right)^{\circ(\frac{k}{2}\,-\,1)}\\
&= k\operatorname{sgn}(\boldsymbol{\alpha})\circ(\boldsymbol{\alpha}^{\circ 2})^{\frac{1}{2}}\circ\left(\boldsymbol{\alpha}^{\circ 2}\right)^{\circ\frac{k-2}{2}}\\
&= k\operatorname{sgn}(\boldsymbol{\alpha})\circ\left(\boldsymbol{\alpha}^{\circ 2}\right)^{\circ\frac{k-1}{2}}.
\end{aligned} \tag{15}$$

Equation (14) can then be written as

$$\begin{aligned}
\lambda k \,\mathrm{sgn}(\boldsymbol{\alpha}) \circ \left(\boldsymbol{\alpha}^{\circ 2}\right)^{\circ \frac{k-1}{2}} &= 2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\boldsymbol{\alpha}) \\
\Rightarrow \frac{\lambda k}{2} \mathrm{sgn}(\boldsymbol{\alpha}) \circ |\boldsymbol{\alpha}|^{\circ(k\,-\,1)} &= \mathbf{X}^T(\mathbf{y} - \mathbf{X}\boldsymbol{\alpha}). \\
\frac{\lambda k}{2} \mathrm{sgn}(\boldsymbol{\alpha}) \circ |\boldsymbol{\alpha}|^{\circ(k\,-\,1)} + \mathbf{X}^T\mathbf{X}\boldsymbol{\alpha} &= \mathbf{X}^T\mathbf{y}, \\
\frac{\lambda k}{2} \boldsymbol{\alpha} \circ |\boldsymbol{\alpha}|^{\circ(k\,-\,2)} + \mathbf{X}^T\mathbf{X}\boldsymbol{\alpha} &= \mathbf{X}^T\mathbf{y}, \\
\left(\frac{\lambda k}{2} \mathrm{diag}\{|\boldsymbol{\alpha}|^{\circ(k\,-\,2)}\} + \mathbf{X}^T\mathbf{X}\right) \boldsymbol{\alpha} &= \mathbf{X}^T\mathbf{y}, \qquad (16)
\end{aligned}$$

which leads to (11) when $\left(\frac{\lambda k}{2} \mathrm{diag}\{|\boldsymbol{\alpha}|^{\circ(k\,-\,2)}\} + \mathbf{X}^T\mathbf{X}\right)$ is nonsingular.

Let $\mathbf{A} = \frac{\lambda k}{2} \mathrm{diag}\{|\boldsymbol{\alpha}|^{\circ(k\,-\,2)}\}$ and $\mathbf{B} = \mathbf{X}^T\mathbf{X}$ which is given to be of full rank, then $\mathbf{A} + \mathbf{B}$ can be written as $\mathbf{B}(\mathbf{I} + \mathbf{B}^{-1}\mathbf{A})$. Based on the power series expansion, we have $(\mathbf{I} + \mathbf{B}^{-1}\mathbf{A})^{-1} = \mathbf{I} + (-\mathbf{B}^{-1}\mathbf{A}) + (-\mathbf{B}^{-1}\mathbf{A})^2 + (-\mathbf{B}^{-1}\mathbf{A})^3 + \cdots$, that actually converges to the inverse of $\mathbf{I} + \mathbf{B}^{-1}\mathbf{A}$ whenever $\| - \mathbf{B}^{-1}\mathbf{A}\| < 1$ for any sub-multiplicative norm. When $\mathbf{A}$ is symmetric, we have $\|\mathbf{A}\| = \max_j(|\,\mathrm{eig}_j(\mathbf{A})\,|)$ for matrix norm given by $||\mathbf{A}|| := \sup_{\mathbf{v}\neq 0} \frac{||\mathbf{A}\mathbf{v}||}{||\mathbf{v}||}$. Since this norm is sub-multiplicative, we also have $\| - \mathbf{B}^{-1}\mathbf{A}\| \leq \| - \mathbf{B}^{-1}\|\cdot\|\mathbf{A}\|$. Hence, $\| - \mathbf{B}^{-1}\mathbf{A}\| < 1$ is implied by

$$\max_j(|\mathrm{eig}_j((\mathbf{X}^T\mathbf{X})^{-1})|) \cdot \frac{\lambda k}{2} \max_j(|\alpha_j|^{(k-2)}) < 1, \qquad (17)$$

or

$$\frac{1}{\min_j(|\mathrm{eig}_j(\mathbf{X}^T\mathbf{X})|)} \cdot \frac{\lambda k}{2} \max_j(|\alpha_j|^{(k-2)}) < 1. \qquad (18)$$

This leads to (13) where the absolute values are not necessary as eigenvalues for $\mathbf{X}^T\mathbf{X}$ are positive.

Finally, as both $(\mathbf{y} - \mathbf{X}\boldsymbol{\alpha})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\alpha})$ and $|\wr\boldsymbol{\alpha}\wr|_k^k$ (Lemma 1) are convex on $\boldsymbol{\alpha}$ for $k \geq 1$, the summation of two convex functions in the objective function (12) is convex. Hence the minimizer. ■

Remark 1: In practice, when the samples are uncorrelated, it is frequent to have $\mathbf{X}^T\mathbf{X}$ invertible or have condition (13) satisfied.

It is interesting to observe that the solution given by (11) appears to have a similar form as that in [9, 24] (see (19)) where a different ℓ_p-norm approximation, based on the LQA (i.e., minimization of (10)) instead of the k-measure (i.e., minimization of (12)), had been utilized for the penalty term.

$$\hat{\boldsymbol{\alpha}}_j = \left(\frac{\lambda q}{2} \mathrm{diag}\{|\boldsymbol{\alpha}_{0j}|^{\circ(q\,-\,2)}\} + \mathbf{X}^T\mathbf{X}\right)^{-1} \mathbf{X}^T\mathbf{y}. \qquad (19)$$

□

3.3. Proximal bridge regression in dual form.

For under-determined systems, we minimize $|\wr\boldsymbol{\alpha}\wr|_k^k$ subject to $\mathbf{y} = \mathbf{X}\boldsymbol{\alpha}$. We call such minimization *proximal bridge regression in dual form* (or *dual p-bridge* in brief). Similar to the primal proximal bridge regression, our goal here is to have a compressive estimate for $1 < k < 2$.

Theorem 2. *Consider an under-determined system* $\mathbf{y} = \mathbf{X}\boldsymbol{\alpha}$*, where* $\mathbf{y} \in \mathbb{R}^M$ *is the given target vector,* $\mathbf{X} \in \mathbb{R}^{M\times D}$ *is the regressor matrix and* $\boldsymbol{\alpha} \in \mathbb{R}^D$ *is the parameter vector, with number of samples* $M < D$ *regressor dimensions. Assume* $\mathbf{X}\mathbf{X}^T$ *and* $\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}$ *are of full rank for certain* $k > 1$*. Then for that* $k > 1$ *and under the limiting case of* $\boldsymbol{\epsilon} \to \mathbf{0}$*, the stationary point given by*

$$\hat{\boldsymbol{\alpha}} \;=\; \mathrm{sgn}\,(\boldsymbol{\theta}) \circ \left|\mathbf{X}^T \left(\mathbf{X}\mathbf{X}^T\right)^{-1} \mathbf{X}\boldsymbol{\theta}^{\circ(k\,-\,1)}\right|^{\circ\frac{1}{k-1}}, \qquad (20)$$

where

$$\boldsymbol{\theta} \;=\; |\mathbf{X}^T|^{\circ\frac{1}{k-1}} \left[\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\right]^{-1} \mathbf{y}, \qquad (21)$$

minimizes

$$|\!\wr\boldsymbol{\alpha}\wr\!|_k^k \text{ subject to } \mathbf{y} = \mathbf{X}\boldsymbol{\alpha}. \tag{22}$$

Proof: According to the definition of k-measure in (5), let $\bar{\boldsymbol{\alpha}} := [(\alpha_0^2+\epsilon)^{k/4}, \cdots, (\alpha_{D-1}^2+\epsilon)^{k/4}]^T$ where we can write $|\!\wr\boldsymbol{\alpha}\wr\!|_k = (\bar{\boldsymbol{\alpha}}^T\bar{\boldsymbol{\alpha}})^{1/k}$ and $|\!\wr\boldsymbol{\alpha}\wr\!|_k^k = (\bar{\boldsymbol{\alpha}}^T\bar{\boldsymbol{\alpha}})$. Then, taking the first derivative of the Lagrange function of (22) and setting it to zero gives:

$$\begin{aligned}
\frac{\partial}{\partial\boldsymbol{\alpha}}\left(\bar{\boldsymbol{\alpha}}^T\bar{\boldsymbol{\alpha}} + \boldsymbol{\beta}^T(\mathbf{y} - \mathbf{X}\boldsymbol{\alpha})\right) &= \mathbf{0}\\
\frac{k}{4}\cdot 2\boldsymbol{\alpha}\circ\left(\boldsymbol{\alpha}^{\circ 2}+\boldsymbol{\epsilon}\right)^{\circ(\frac{k}{4}\,-\,1)}\circ 2\bar{\boldsymbol{\alpha}} - \mathbf{X}^T\boldsymbol{\beta} &= \mathbf{0}\\
k\,\boldsymbol{\alpha}\circ\left(\boldsymbol{\alpha}^{\circ 2}+\boldsymbol{\epsilon}\right)^{\circ(\frac{k}{4}\,-\,1)}\circ\left(\boldsymbol{\alpha}^{\circ 2}+\boldsymbol{\epsilon}\right)^{\circ\frac{k}{4}} - \mathbf{X}^T\boldsymbol{\beta} &= \mathbf{0}\\
\Rightarrow\quad k\,\boldsymbol{\alpha}\circ\left(\boldsymbol{\alpha}^{\circ 2}+\boldsymbol{\epsilon}\right)^{\circ(\frac{k}{2}\,-\,1)} &= \mathbf{X}^T\boldsymbol{\beta}. \qquad (23)
\end{aligned}$$

For the limiting case of $\boldsymbol{\epsilon}$, we have

$$\lim_{\boldsymbol{\epsilon}\to\mathbf{0}} k\,\boldsymbol{\alpha}\circ\left(\boldsymbol{\alpha}^{\circ 2}+\boldsymbol{\epsilon}\right)^{\circ(\frac{k}{2}\,-\,1)} = k\,\boldsymbol{\alpha}\circ\left(\boldsymbol{\alpha}^{\circ 2}\right)^{\circ(\frac{k}{2}\,-\,1)}, \tag{24}$$

which implies

$$\begin{aligned}
k\,\boldsymbol{\alpha}\circ\left(\boldsymbol{\alpha}^{\circ 2}\right)^{\circ\frac{k-2}{2}} &= \mathbf{X}^T\boldsymbol{\beta}\\
k\,\mathrm{sgn}(\boldsymbol{\alpha})\circ(\boldsymbol{\alpha}^{\circ 2})^{\frac{1}{2}}\circ\left(\boldsymbol{\alpha}^{\circ 2}\right)^{\circ\frac{k-2}{2}} &= \mathbf{X}^T\boldsymbol{\beta}\\
k\,\mathrm{sgn}(\boldsymbol{\alpha})\circ\left(\boldsymbol{\alpha}^{\circ 2}\right)^{\circ\frac{k-1}{2}} &= \mathbf{X}^T\boldsymbol{\beta}\\
\Rightarrow\quad \left(\boldsymbol{\alpha}^{\circ 2}\right)^{\circ\frac{k-1}{2}} &= \mathrm{sgn}(\boldsymbol{\alpha})\circ\left(\frac{1}{k}\mathbf{X}^T\boldsymbol{\beta}\right). \qquad (25)
\end{aligned}$$

Taking square elementwise for both sides of (25), we have

$$\left(\boldsymbol{\alpha}^{\circ 2}\right)^{\circ(k\,-\,1)} = \left(\frac{1}{k}\mathbf{X}^T\boldsymbol{\beta}\right)^{\circ 2}. \tag{26}$$

We know that the vector $\lim_{\boldsymbol{\epsilon}\to\mathbf{0}}(\boldsymbol{\alpha}^{\circ 2}+\boldsymbol{\epsilon})$ has nonnegative elements and thus $\lim_{\boldsymbol{\epsilon}\to\mathbf{0}}\left(\boldsymbol{\alpha}^{\circ 2}+\boldsymbol{\epsilon}\right)^{\circ(\frac{k}{2}\,-\,1)}$ has nonnegative elements. Hence, we deduce from (23) that $\mathrm{sgn}(\boldsymbol{\alpha}) = \mathrm{sgn}(\mathbf{X}^T\boldsymbol{\beta})$, and

$$\boldsymbol{\alpha} = \mathrm{sgn}(\mathbf{X}^T\boldsymbol{\beta})\circ\left|\mathbf{X}^T\left\{\frac{1}{k}\boldsymbol{\beta}\right\}\right|^{\circ\frac{1}{k-1}}. \tag{27}$$

Next, suppose that

$$\mathrm{sgn}(\mathbf{X}^T\boldsymbol{\beta})\circ\left|\mathbf{X}^T\left\{\frac{1}{k}\boldsymbol{\beta}\right\}\right|^{\circ\frac{1}{k-1}} = |\mathbf{X}^T|^{\circ\frac{1}{k-1}}\boldsymbol{\gamma} \tag{28}$$

for some $\boldsymbol{\gamma}$, then premultiply $\mathbf{X}$ to both sides of (27) gives

$$\begin{aligned}
\mathbf{X}\boldsymbol{\alpha} &= \mathbf{X}\,\mathrm{sgn}(\mathbf{X}^T\boldsymbol{\beta})\circ\left|\mathbf{X}^T\left\{\frac{1}{k}\boldsymbol{\beta}\right\}\right|^{\circ\frac{1}{k-1}}\\
\Rightarrow\mathbf{X}\boldsymbol{\alpha} &= \mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\boldsymbol{\gamma}, \quad\text{according to (28)}\\
\Rightarrow\mathbf{y} &= \mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\boldsymbol{\gamma}, \quad\text{since } \mathbf{y} = \mathbf{X}\boldsymbol{\alpha}\\
\Rightarrow\boldsymbol{\gamma} &= \left[\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\right]^{-1}\mathbf{y}, \quad\text{since } \left[\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\right]^{-1} \text{ is invertible.} \qquad (29)
\end{aligned}$$

Knowing also that $\mathbf{X}\mathbf{X}^T$ is invertible, we substitute (29) into (28) and get

$$\begin{aligned}
\mathrm{sgn}(\mathbf{X}^T\boldsymbol{\beta}) \circ \left|\mathbf{X}^T\left\{\frac{1}{k}\boldsymbol{\beta}\right\}\right|^{\circ\frac{1}{k-1}} &= |\mathbf{X}^T|^{\circ\frac{1}{k-1}}\left[\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\right]^{-1}\mathbf{y} \\
\mathbf{X}^T\left\{\frac{1}{k}\boldsymbol{\beta}\right\} &= \mathrm{sgn}(\mathbf{X}^T\boldsymbol{\beta}) \circ \left||\mathbf{X}^T|^{\circ\frac{1}{k-1}}\left[\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\right]^{-1}\mathbf{y}\right|^{\circ(k\,-\,1)} \\
\mathbf{X}\mathbf{X}^T\left\{\frac{1}{k}\boldsymbol{\beta}\right\} &= \mathrm{sgn}(\mathbf{X}^T\boldsymbol{\beta}) \circ \mathbf{X}\left||\mathbf{X}^T|^{\circ\frac{1}{k-1}}\left[\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\right]^{-1}\mathbf{y}\right|^{\circ(k\,-\,1)} \\
\left\{\frac{1}{k}\boldsymbol{\beta}\right\} = \mathrm{sgn}(\mathbf{X}^T\boldsymbol{\beta}) &\circ \left(\mathbf{X}\mathbf{X}^T\right)^{-1}\mathbf{X}\left||\mathbf{X}^T|^{\circ\frac{1}{k-1}}\left[\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\right]^{-1}\mathbf{y}\right|^{\circ(k\,-\,1)}.
\end{aligned} \tag{30}$$

Subsequently, substitute (30) into (27) and we have

$$\begin{aligned}
\hat{\boldsymbol{\alpha}} &= \mathrm{sgn}(\mathbf{X}^T\boldsymbol{\beta}) \circ |\mathbf{X}^T\left(\mathbf{X}\mathbf{X}^T\right)^{-1}\mathbf{X}\left(|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\left[\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\right]^{-1}\mathbf{y}\right)^{\circ(k\,-\,1)}|^{\circ\frac{1}{k-1}} \\
&= \mathrm{sgn}\left(\boldsymbol{\theta}\right) \circ |\mathbf{X}^T\left(\mathbf{X}\mathbf{X}^T\right)^{-1}\mathbf{X}\boldsymbol{\theta}^{\circ(k\,-\,1)}|^{\circ\frac{1}{k-1}},
\end{aligned} \tag{31}$$

where

$$\boldsymbol{\theta} = |\mathbf{X}^T|^{\circ\frac{1}{k-1}}\left[\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\right]^{-1}\mathbf{y}. \tag{32}$$

The sign of $\mathrm{sgn}(\mathbf{X}^T\boldsymbol{\beta}) = \mathrm{sgn}\left(\boldsymbol{\theta}\right)$ has been deduced from the top row of (30). Equations (31)-(32) hold well without singularity for all $\boldsymbol{\theta} \in \mathbb{R}^D$ and all $\mathbf{X} \in \mathbb{R}^{M\times D}$ for $k > 1$. Finally, since $|\wr\boldsymbol{\alpha}\wr|_k^k$ is convex according to Lemma 1 and the linear constraint function ($\mathbf{y} = \mathbf{X}\boldsymbol{\alpha}$) is also convex, the Lagrange function of (22) is convex. Hence the minimizer. ■

Remark 2: In [31], a simpler version of global solution was conjectured for a weighted least norm regression. However, Theorem 2 reveals that such a solution cannot be any simpler. From application viewpoint, although the validity of k stretches beyond 2 in Theorem 1 and in Theorem 2, the region of interest for parametric shrinking is $k < 2$. We shall thus focus on $k \in [1, 2]$ for over-determined systems and $k \in (1, 2]$ for under-determined systems in our development, both included the well-known non-compressive ℓ_2-norm for benchmarking purpose. □

3.4. Extension to Multiple Outputs.

The above results can be extended for regression with multiple outputs. Particularly, by utilizing the same regressor matrix ($\mathbf{X}$) with different outputs ($\mathbf{y}_l$, $l = 1, \ldots, C$), the solution can be stacked for concurrent prediction. For example, suppose $\hat{\boldsymbol{A}}$ stacks the multiple columns of estimated coefficient vectors $[\hat{\boldsymbol{\alpha}}_1, \ldots, \hat{\boldsymbol{\alpha}}_C] \in \mathbb{R}^{D\times C}$, then the prediction can be computed as $\hat{\mathbf{Y}} = \mathbf{X}\hat{\boldsymbol{A}}$. For classification applications, an one-hot encoding can be adopted for learning and the winner-take-all technique can be used to predict the outcome from the multi-category responses. As the outputs do not depend on each other, the extensions for the primal and the dual forms are straightforward.

Theorem 3. *Given the data $\{\boldsymbol{x}_i, y_{i,l}\}$, $i = 1, \ldots, M$, $l = 1, \ldots, C$ where $\boldsymbol{x}_i = [x_{i,1}, \cdots, x_{i,D}]^T$ and $y_{i,l}$ are respectively the regressors and the response for the ith observation of the lth output. Consider the linear regression model $\mathbf{X}\boldsymbol{A}$ with parameter matrix $\boldsymbol{A} = [\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_C] \in \mathbb{R}^{D\times C}$ and regressor matrix $\mathbf{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_M]^T$. Suppose $\mathbf{X}^T\mathbf{X}$ is of full rank. Then, under the limiting case of $\boldsymbol{\epsilon}_l \to \mathbf{0}$ and for $k \geq 1$, $\hat{\boldsymbol{\alpha}}_1, \ldots, \hat{\boldsymbol{\alpha}}_C$ that satisfies*

$$\boldsymbol{\alpha}_l = \left(\frac{\lambda k}{2}\,\mathrm{diag}\{|\boldsymbol{\alpha}_l|^{\circ(k\,-\,2)}\} + \mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}_l, \quad l = 1, \ldots, C, \tag{33}$$

minimizes

$$(\mathbf{y}_l - \mathbf{X}\boldsymbol{\alpha}_l)^T(\mathbf{y}_l - \mathbf{X}\boldsymbol{\alpha}_l) + \lambda|\wr\boldsymbol{\alpha}_l\wr|_k^k, \tag{34}$$

for each target-parameter pair $\{\mathbf{y}_l, \boldsymbol{\alpha}_l\}$*,* $l = 1, \ldots, C$*,* $\mathbf{y}_l \in \mathbb{R}^M$*,* $\boldsymbol{\alpha}_l \in \mathbb{R}^D$ *when the matrix* $(\frac{\lambda k}{2}\,\mathrm{diag}\{|\boldsymbol{\alpha}_l|^{\circ(k-2)}\} + \mathbf{X}^T\mathbf{X})$ *is invertible. This happens for sure as soon as*

$$\frac{\lambda k}{2} \max_{j\in\{1,\ldots,D\}} (|\alpha_{j,l}|^{(k-2)}) < \min_{j\in\{1,\ldots,D\}} (\mathrm{eig}_j(\mathbf{X}^T\mathbf{X})). \tag{35}$$

Proof: Since the $l = 1, \ldots, C$ outputs are independent of each other, the regression for estimating each $\boldsymbol{\alpha}_l$, $l = 1, \ldots, C$ can be performed independently. Hence the result. ■

Theorem 4. *Consider the under-determined systems* $\mathbf{y}_l = \mathbf{X}\boldsymbol{\alpha}_l$*,* $l = 1, \ldots, C$*, where* $\mathbf{y}_l \in \mathbb{R}^M$ *is the given target vector for each output,* $\mathbf{X} \in \mathbb{R}^{M\times D}$ *is the regressor matrix and* $\boldsymbol{\alpha} \in \mathbb{R}^D$ *is the parameter vector, with number of samples* $M < D$ *regressor dimensions. Assume* $\mathbf{X}\mathbf{X}^T$ *and* $\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}$ *are of full rank for certain* $k > 1$*. Then for that* $k > 1$ *and under the limiting case of* $\boldsymbol{\epsilon}_l \to \mathbf{0}$*, the stationary point given by*

$$\hat{\boldsymbol{\alpha}}_l \;=\; \mathrm{sgn}\,(\boldsymbol{\theta}_l) \circ \left|\mathbf{X}^T\left(\mathbf{X}\mathbf{X}^T\right)^{-1}\mathbf{X}\boldsymbol{\theta}_l^{\circ(k-1)}\right|^{\circ\frac{1}{k-1}}, \tag{36}$$

where

$$\boldsymbol{\theta}_l \;=\; |\mathbf{X}^T|^{\circ\frac{1}{k-1}}\left[\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\right]^{-1}\mathbf{y}_l, \tag{37}$$

minimizes

$$|\!\wr\boldsymbol{\alpha}_l\wr\!|_k^k \text{ subject to } \mathbf{y}_l = \mathbf{X}\boldsymbol{\alpha}_l, \quad l = 1, \ldots, C. \tag{38}$$

Proof: Since the $l = 1, \ldots, C$ outputs do not depend on each other, the regression for estimating each $\boldsymbol{\alpha}_l$, $l = 1, \ldots, C$ can be performed independently. Hence the result. ■

3.5. **Variance Analysis.** In this subsection, we analyze the variance of the estimate and observe the essential properties. We shall work on the single output case only since the multiple outputs case is a direct stacking of the single output case. Assume the data is generated according to $\mathbf{y} = \mathbf{X}\boldsymbol{\alpha} + \boldsymbol{\epsilon}$ with $\mathbf{X} \in \mathbb{R}^{M\times D}$, $\boldsymbol{\alpha} \in \mathbb{R}^D$ where $\boldsymbol{\epsilon}$ is a zero mean noise with covariance matric $\mathbf{C}$. For the over-determined case, we have $M \geq D$. Suppose the estimation is initialized by $\boldsymbol{\alpha}_0$, then the expectation of $\hat{\boldsymbol{\alpha}}$ is

$$\begin{aligned} E[\hat{\boldsymbol{\alpha}}] &= E\left[\left(\frac{\lambda k}{2}\,\mathrm{diag}\{|\boldsymbol{\alpha}_0|^{\circ(k-2)}\} + \mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T(\mathbf{X}\boldsymbol{\alpha} + \boldsymbol{\epsilon})\right] \\ &= \left(\frac{\lambda k}{2}\,\mathrm{diag}\{|\boldsymbol{\alpha}_0|^{\circ(k-2)}\} + \mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{X}\boldsymbol{\alpha} \\ &\neq \boldsymbol{\alpha}, \quad \forall k > 1, \lambda > 0. \end{aligned} \tag{39}$$

This shows that the estimation is biased for the ranges of our working k values and λ values. For the special case when $\lambda = 0$, we have an unbiased ols estimation since $E[\hat{\boldsymbol{\alpha}}] = E\left[\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T(\mathbf{X}\boldsymbol{\alpha} + \boldsymbol{\epsilon})\right] = E\left[\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{X}\boldsymbol{\alpha}\right] = \boldsymbol{\alpha}$.

For the under-determined system where $M < D$, the expectation of estimate is

$$\begin{aligned} E[\hat{\boldsymbol{\alpha}}] &= E\left[\mathrm{sgn}\,(\boldsymbol{\theta}) \circ \left|\mathbf{X}^T\left(\mathbf{X}\mathbf{X}^T\right)^{-1}\mathbf{X}\boldsymbol{\theta}^{\circ(k-1)}\right|^{\circ\frac{1}{k-1}}\right], \\ &\qquad \boldsymbol{\theta} = |\mathbf{X}^T|^{\circ\frac{1}{k-1}}\left[\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\right]^{-1}(\mathbf{X}\boldsymbol{\alpha} + \boldsymbol{\epsilon}), \\ &\neq \boldsymbol{\alpha}. \end{aligned} \tag{40}$$

The inequality holds because the rank of

$$\left|\mathbf{X}^T\left(\mathbf{X}\mathbf{X}^T\right)^{-1}\mathbf{X}\left\{|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\left[\mathbf{X}|\mathbf{X}^T|^{\circ\frac{1}{k-1}}\right]^{-1}\mathbf{X}\boldsymbol{\alpha}\right\}^{\circ(k-1)}\right|^{\circ\frac{1}{k-1}}$$

is at most M (due to $\left(\mathbf{X}\mathbf{X}^T\right)^{-1}$) which is smaller than D. The above analyses for the over- and the under-determined cases show that both the estimates are biased, and this is consistent with the compressed estimation where some ideal parameters have been suppressed.

For variance of the over-determined case, we have

$$
\begin{aligned}
&E[(\hat{\boldsymbol{\alpha}} - E[\hat{\boldsymbol{\alpha}}])(\hat{\boldsymbol{\alpha}} - E[\hat{\boldsymbol{\alpha}}])^T] \\
&= E\left\{\left[\left(\frac{\lambda k}{2}\,\mathrm{diag}\{|\boldsymbol{\alpha}_0|^{\circ(k\,-\,2)}\}{+}\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\boldsymbol{\epsilon}\right]\left[\left(\frac{\lambda k}{2}\,\mathrm{diag}\{|\boldsymbol{\alpha}_0|^{\circ(k\,-\,2)}\}{+}\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\boldsymbol{\epsilon}\right]^T\right\} \\
&= \left[\left(\frac{\lambda k}{2}\,\mathrm{diag}\{|\boldsymbol{\alpha}_0|^{\circ(k\,-\,2)}\}{+}\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\right] E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T]\left[\left(\frac{\lambda k}{2}\,\mathrm{diag}\{|\boldsymbol{\alpha}_0|^{\circ(k\,-\,2)}\}{+}\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\right]^T \\
&= \left[\left(\frac{\lambda k}{2}\,\mathrm{diag}\{|\boldsymbol{\alpha}_0|^{\circ(k\,-\,2)}\}{+}\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\right]\mathbf{C}\left[\left(\frac{\lambda k}{2}\,\mathrm{diag}\{|\boldsymbol{\alpha}_0|^{\circ(k\,-\,2)}\}{+}\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\right]^T. \qquad (41)
\end{aligned}
$$

When $k = 2$, it reduces to ridge regression where standard analysis applies. For other k values, the situation becomes complicated.

For the under-determined case, it is difficult to simplify $E[(\hat{\boldsymbol{\alpha}} - E[\hat{\boldsymbol{\alpha}}])(\hat{\boldsymbol{\alpha}} - E[\hat{\boldsymbol{\alpha}}])^T]$ due to the nonlinearity incurred by the absolute exponent. However, when $k = 2$, it also reduces to the minimum norm solution case where standard analysis applies. Again, for other k values, the situation becomes complicated.

3.6. **Algorithm.** The algorithm for the proposed proximal bridge regression can be readily implemented following the pseudo-code below. The pipeline of the algorithm is shown in Fig. 3. Here, both the over- and under-determined cases have been included according to the shape of the regression matrix. In the algorithm, $\mathbf{Y}$ denotes the packed target matrix given by $[\mathbf{y}_1, \ldots, \mathbf{y}_C]$ and the estimated parameters are packed as $\hat{\mathbf{A}} = [\hat{\boldsymbol{\alpha}}_1, \ldots, \hat{\boldsymbol{\alpha}}_D]$. The initialization for $\hat{\mathbf{A}}$ in the primal solution has been based on the ordinary least squares (ols) solution. For numerical stability under practical considerations, all the inverse terms have included regularization.

Algorithm 1: *p*-bridge

```
Inputs: input samples $\mathbf{X} \in \mathbb{R}^{M\times D}$ from the training set, label matrix $\mathbf{Y} \in \mathbb{R}^{M\times C}$,
        proximal-norm value $k$ and regularization factor $\lambda$.
if $M < D$ then
    if $k = 2$ then
        $\mathbf{P} = \mathbf{X}^T$;
    else
        $\mathbf{P} = |\mathbf{X}^T|^{\circ\frac{1}{k-1}}$;
    end
    $\boldsymbol{\Theta} = \mathbf{P}[\mathbf{X}\mathbf{P} + \lambda\mathbf{I}]^{-1}\mathbf{Y}$;
    $\hat{\mathbf{A}} = \mathrm{sgn}(\boldsymbol{\Theta}) \circ \left|\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \lambda\mathbf{I})^{-1}\mathbf{X}\boldsymbol{\Theta}^{\circ(k-1)}\right|^{\circ\frac{1}{k-1}}$;
else
    $\hat{\mathbf{A}}_{(0)} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{Y}$;
    if $k < 2$ then
        for $j \leftarrow 0$ to $4$ do
            $\hat{\mathbf{A}}_{(j+1)} = \left(\mathbf{X}^T\mathbf{X} + \frac{\lambda k}{2}\mathrm{diag}\{|\hat{\mathbf{A}}_{(j)}|^{\circ(k-2)}\}\right)^{-1}\mathbf{X}^T\mathbf{Y}$;
        end
        $\hat{\mathbf{A}} = \hat{\mathbf{A}}_{(5)}$
    else
        $\hat{\mathbf{A}} = \hat{\mathbf{A}}_{(0)}$
    end
end
Output: $\hat{\mathbf{A}}$.
```

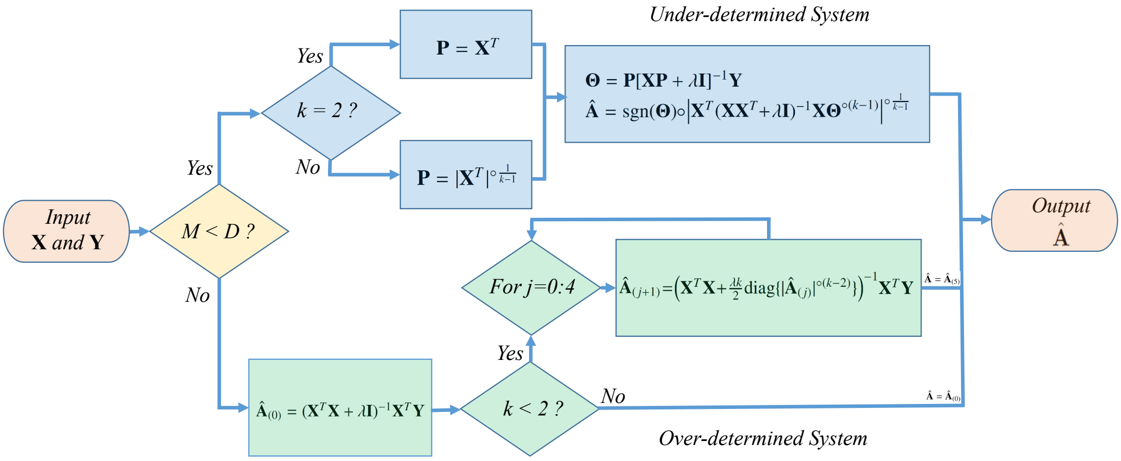

Under-determined System
Yes
$\mathbf{P} = \mathbf{X}^T$
$\mathbf{\Theta} = \mathbf{P}[\mathbf{XP} + \lambda\mathbf{I}]^{-1}\mathbf{Y}$
$\hat{\mathbf{A}} = \mathrm{sgn}(\mathbf{\Theta})\circ\left|\mathbf{X}^T(\mathbf{XX}^T+\lambda\mathbf{I})^{-1}\mathbf{X}\mathbf{\Theta}^{\circ(k-1)}\right|^{\circ\frac{1}{k-1}}$
$k = 2$?
Yes
No
$\mathbf{P} = |\mathbf{X}^T|^{\circ\frac{1}{k-1}}$
Input
X and Y
$M < D$?
Output
$\hat{\mathbf{A}}$
No
For j=0:4
$\hat{\mathbf{A}}_{(j+1)} = \left(\mathbf{X}^T\mathbf{X} + \frac{\lambda k}{2}\mathrm{diag}\{|\hat{\mathbf{A}}_{(j)}|^{\circ(k-2)}\}\right)^{-1}\mathbf{X}^T\mathbf{Y}$
$\hat{\mathbf{A}} = \hat{\mathbf{A}}_{(5)}$
Yes
$\hat{\mathbf{A}}_{(0)} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{Y}$
$k < 2$?
No
$\hat{\mathbf{A}} = \hat{\mathbf{A}}_{(0)}$
Over-determined System

FIGURE 3. Pipeline of the algorithm

4. CASE STUDIES

In this section, we perform some empirical studies under the Matlab platform on a real-world data set and a simulated data set namely, the prostate cancer data, the Exclusive-OR (XOR) problem. Our goal here is to observe the behavior of the proposed method comparing with state-of-the-arts. For the prostate cancer data and the XOR problem, both the coefficient profiles and the estimation results will be observed.

As a representative example for over-determined systems, the prostate cancer data from [34], which has been used by [16, 5, 22] as a benchmark example, is adopted to learn a linear regression model based on 67 training samples. This data set has a continuous response with 8 input variables. These inputs together with an intercept term give rise to 9 estimation coefficients for the linear regression model and this forms an over-determined regression system. On the other hand, as a representative example for under-determined systems, a 3rd-order polynomial model given by

$$p(x_1,x_2) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_1^2 + \alpha_4 x_2^2 + \alpha_5 x_1 x_2 + \alpha_6 x_1^3 + \alpha_7 x_2^3 + \alpha_8 x_1^2 x_2 + \alpha_9 x_1 x_2^2, \tag{42}$$

is deployed to learn the well-known XOR problem with four training data samples. The inputs to the XOR problem are $(x_1, x_2) \in \{(0,1),(2,1),(1,0),(1,2)\}$ with their corresponding target outputs given by $y \in \{0,0,1,1\}$, respectively. The system formed by learning the XOR data using the polynomial model constitutes an under-determined system since the number of parametric coefficients $(\alpha_0,\dots,\alpha_9)$ is larger than the four learning samples. A total of 200 test samples for this XOR problem has been generated for mapping evaluation. These test samples have been generated by a bivariate Gaussian random number generator with centers located at the four training points, each center corresponds to 50 samples with an identity covariance matrix scaled by 0.3. The responses of these test data follow the labels of the four centers respectively.

4.1. **Over-determined system: prostate cancer example. Coefficient Profile:** The profile of each coefficient estimate is observed with respect to variation of shrinkage settings following [16, 5, 22]. We shall first show the coefficient profile for the well-known ridge regression according to [16] for immediate reference. Fig. 4 shows the ridge coefficient estimates plotted as function of $df(\lambda) = \mathrm{tr}[\mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T]$, the *effective degrees of freedom* implied by the penalty λ (see [16], Section 3.4).

As one of our interests here is to check the compression capability of the primal p-bridge regression in Theorem 1, we shall observe the coefficient estimation at $k = 1$ and compare it with that of the well-known the least absolute shrinkage and selection operator (lasso) [5] (see

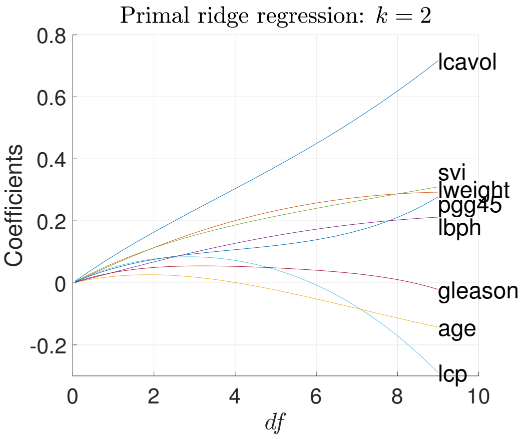
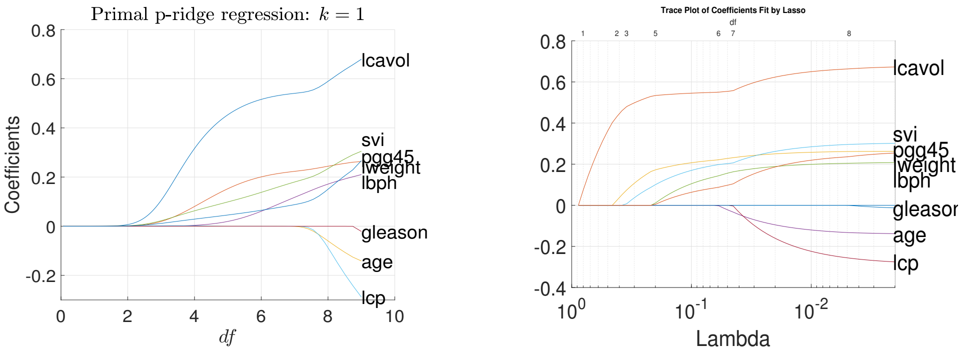


FIGURE 4. Profiles of ridge coefficients for the over-determined system: prostate cancer example.

[35] for the library codes in Matlab). Fig. 5 shows the coefficient profile of primal p-bridge regression at $k = 1$ and that of lasso. This plot shows much resemblance of shrinkage behaviors between p-bridge and lasso for several coefficients, particularly for that of 'lcp', 'age' and 'gleason' which shrunk to zero with a similar order of sequence when lowering the df (raising the λ) value. However, for p-bridge regression, the coefficients of 'pgg45', 'lweight', 'svi', and 'lcavol' do not appear to reach zero sequentially as that of lasso. Moreover, for p-bridge regression, the coefficient of 'pgg45' does not terminate together with that of 'lbph' as in the case of lasso.



(a) primal p-bridge at $k = 1$

(b) lasso

FIGURE 5. Profiles of primal p-bridge (at $k = 1$) and lasso coefficients for the over-determined system: prostate cancer example.

Prediction results: The prediction results of p-bridge is compared with several state-of-art methods namely, the ordinary least squares regression (ols) [15, 16], the ridge regression (ridge) [15, 16], the lasso [5], the elastic-net [22], and the bridge [36]. Table 1 shows the results of the best chosen models obtained from tenfold cross-validation based on the 67 training observations. The mean-squared errors (MSE) between the estimated output and the measured output of the compared methods are reported based on the 30 test samples. These results show comparable performance of the primal p-bridge with those of the well-known state-of-the-arts. The estimated coefficients are shown in Table 2 for each method. Here we note that the chosen models are not sparse in favor of the cross-validated MSE.

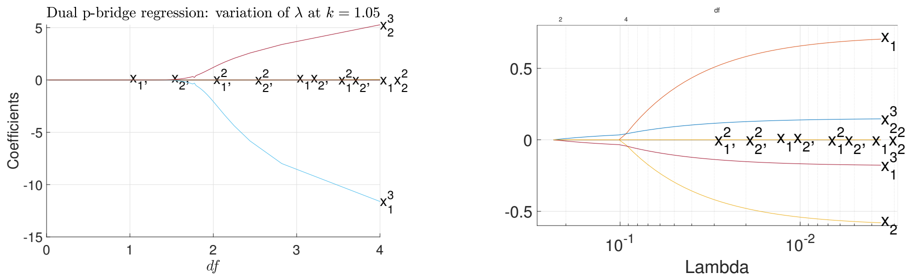
Table 1. Prostate example: test mean squared error (MSE) and variables (excluding the intercept) selected

Method	Tuned Parameter(s)	Test MSE	Variables selected
ols	–	0.520 (0.174)	All
ridge regression	$\lambda = 1$	0.516 (0.175)	All
lasso at (`Alpha` = 1)	`Lambda` = 0.02	0.483 (0.160)	(1,2,3,4,5,6,8)
elastic-net	`Lambda` = 0.06, `Alpha` = 0.11	0.492 (0.164)	(1,2,3,4,5,6,8)
bridge	$\lambda = 2$, $k = 1$	0.491 (0.164)	(1,2,3,4,5,6,8)
p-bridge	$\lambda = 2$, $k = 1$	0.494 (0.167)	(1,2,3,4,5,6,8)

Table 2. Prostate example: estimated coefficients for the chosen setting based cross-validation on the training set

Predictor	ols	ridge	lasso	elastic-net	p-bridge at $k = 1$	p-bridge
0. intcpt	2.452	2.416	2.467	2.466	2.452	2.452
1. lcavol	0.716	0.690	0.624	0.590	0.637	0.637
2. lweight	0.293	0.292	0.250	0.254	0.256	0.256
3. age	-0.143	-0.135	-0.095	-0.102	-0.106	-0.106
4. lbph	0.212	0.210	0.189	0.197	0.193	0.193
5. svi	0.310	0.304	0.262	0.274	0.274	0.274
6. lcp	-0.289	-0.256	-0.161	-0.157	-0.196	-0.196
7. gleason	-0.021	-0.011	0	0	-0.000	-0.000
8. pgg45	0.277	0.258	0.186	0.199	0.206	0.206

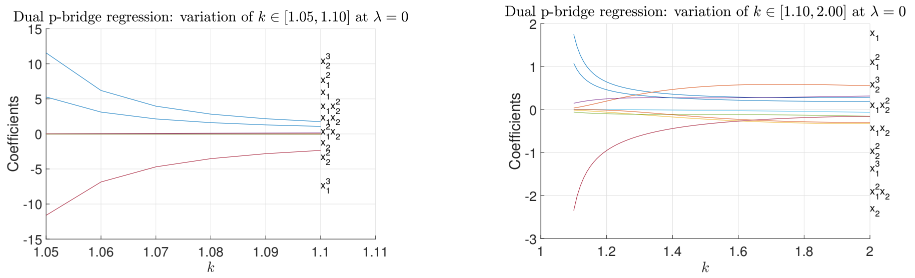
4.2. **Under-determined system: the XOR problem. Coefficient Profile:** The under-determined formulation in Theorem 2 (p-bridge regression in dual form) is applied to learn the XOR problem for coefficient profiling since there are more coefficients than training samples. Fig. 6 shows the coefficient profiles of dual p-bridge and lasso. This plot shows a highly sparse estimation for p-bridge at low k-value ($k = 1.05$) comparing with lasso. Particularly, when the shrinkage penalty is low (at small λ value), p-bridge suppresses all coefficients except those of x_1^3 and x_2^3 whereas lasso emphasizes the coefficients of x_1 and x_2 more than that of x_1^3 and x_2^3 while suppressing all other coefficients. Fig. 7(a) shows that for the range $1.05 \leq k \leq 1.1$, p-bridge suppresses most of the coefficients except for the coefficients of x_1^3 and x_2^3 even at $\lambda = 0$. The effect of sparseness begins to vanish after $k > 1.1$ as seen from Fig. 7(b). The decision boundaries for both the dual p-bridge regression and the lasso in Fig. 8 show much resemblance in view of the high contribution of the coefficients of x_1^3 and x_2^3.



(a) dual p-bridge (b) lasso

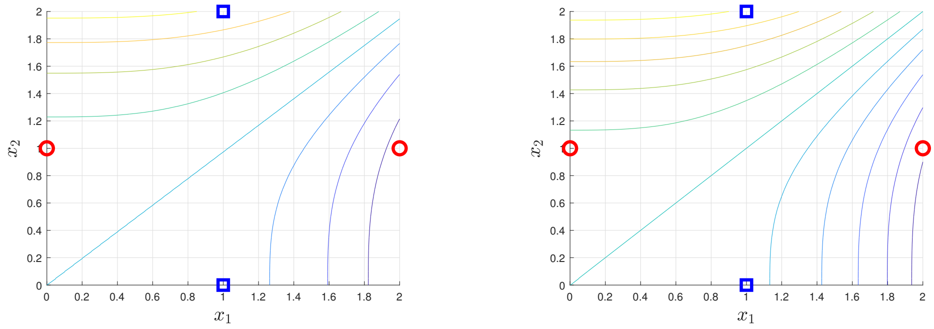
Figure 6. Profiles of dual p-bridge and lasso coefficients for the under-determined XOR problem.

Prediction results: Table 3 shows the results of chosen models based on training the four observations. The mean-squared errors between the estimated and the generated outputs of the compared methods are reported based on the 200 test observations. These results show comparable mapping performance of the dual p-bridge with that of the state-of-the-arts. Here we note that the bridge [36] did not consider the under-determined case and encountered the problem of matrix inverse. The estimated coefficients as seen from Table 4 show sparseness for lasso at `Lambda` = 0.1 (`Alpha` = 1) and dual p-bridge at $k = 1.05$ and $\lambda = 30$.



(a) (b)

FIGURE 7. Profiles of dual p-bridge coefficients (by variation of k values at $\lambda = 0$) for the under-determined XOR problem.



(a) dual p-bridge at $k = 1.05$ (b) lasso

FIGURE 8. Decision contours of dual p-bridge and lasso.

TABLE 3. XOR: test mean squared error (MSE) and variables $(\alpha_0, \ldots, \alpha_9)$ selected based on the four training points

Method	Tuned Parameter(s)	Test MSE	Variables selected
ols	–	0.513 (0.089)	All
ridge regression	$\lambda = 6$	0.503 (0.047)	(0,1,2,3,4,6,7,8,9)
lasso*	–	0.225 (0.011)	(0,6,7)
lasso at ($\texttt{Alpha} = 1$)	$\texttt{Lambda} = 0$	0.799 (0.189)	(0,1,2,3,4,6,7,8,9)
elastic-net	$\texttt{Lambda} = 0$, $\texttt{Alpha} = 0.01$	0.799 (0.189)	(0,1,2,3,4,6,7,8,9)
bridge	(problem with matrix inverse)	–	–
p-bridge at ($k = 1.05$)	$\lambda = 30$	0.504 (0.040)	(6,7)
p-bridge	$\lambda = 0$, $k = 2$	0.513 (0.089)	All

lasso*: lasso at $\texttt{Alpha} = 1$ and $\texttt{Lambda} = 0.1$.

TABLE 4. XOR: estimated coefficients for the chosen setting based on the four training points (parameters within parenthesis have been prefixed, parameters without parenthesis have been determined based on training MSE)

	State-of-arts				Proposed p-bridge regression	
Predictor	ols	ridge	lasso	elastic-net	($k = 1.05$) $\lambda = 30$	$k = 2$, $\lambda = 0$
0. intcpt	0.288	0.200	0.500	0.644	0.000	0.288
1. x_1	0.554	0.040	0	0.002	0.000	0.554
2. x_2	-0.329	-0.040	0	-0.002	-0.000	-0.329
3. x_1^2	0.316	-0.038	0	0.760	0.000	0.316
4. x_2^2	-0.154	0.038	0	-0.914	-0.000	-0.154
5. x_1x_2	-0.063	-0.000	0	0.000	0.000	-0.063
6. x_1^3	-0.159	-0.071	-0.034	0.406	-0.050	-0.159
7. x_2^3	0.195	0.071	0.034	0.272	0.054	0.195
8. $x_1^2x_2$	-0.301	-0.051	0	-0.179	-0.000	-0.301
9. $x_1x_2^2$	0.111	0.051	0	0.460	0.000	0.111

lasso: at $\texttt{A1}$ and $\texttt{L0.1}$.

5. EXPERIMENTS

In this section, we conduct experiments on physical data to observe the applicability of the proposed solution on data of significant size. Firstly, the NIPS 2003 challenge data sets [37] are experimented to observe the estimation and prediction behaviors of the proposed p-bridge solution on data sets of relatively high dimension. Subsequently, we visualize the compression behavior of p-bridge along with handwritten digits recognition based on two benchmark data sets namely, the Optdigit [39, 40] data set and the MNIST data set [41, 42].

5.1. **NIPS Data sets.** The NIPS 2003 challenge data sets consist of both under- and over-determined scenarios for binary classification. According to [37], the task of the Arcene data set is to distinguish between cancer and normal patterns based on continuous input mass-spectrometric data. The task of Dexter is to filter texts about "corporate acquisitions" based on sparse continuous input variables. The task of Dorothea is to predict which compounds bind to Thrombin based on sparse binary input variables. The task of Gisette is to discriminate between two confusable handwritten digits, namely the digit four and the digit nine, based on sparse continuous input variables. The task of Madelon is to classify random data based on sparse binary input variables. Table 5 summarizes these data sets in terms of their data dimensions and sample sizes for training, validation and testing.

Table 5. NIPS Feature Selection Challenge Data Sets

Dataset	Domain	Type	# Feat.	# Train	# Valid.	# Test
Arcene	Mass spec.	Dense	10000	100	100	700
Dexter	Text categ.	Sparse	20000	300	300	2000
Dorothea	Drug discov.	Sp. bin.	100000	800	350	800
Gisette	Digit recog.	Dense	5000	6000	1000	6500
Madelon	Artifical data	Dense	500	2000	600	1800

(i) Comparison setup and protocol. Similar to that in the section of case study, the state-of-the-art methods included in this experimental study are the ordinary least squares regression (ols, [15, 16]), the ridge regression (ridge, [15, 16]), the lasso, [5], and the elastic-net ([22]). As the bridge [36] did not consider the under-determined case and encountered the problem of matrix inverse in most datasets, it is not included for comparison. The platform for this evaluation has been based on Python 3.9.7 running on an Intel i7 CPU of 2.8GHz with 16GB of RAM. In view of the stability in handling both the over- and the under-determined systems, the ols has been implemented based on numpy's pseudoinverse function (`numpy.linalg.pinv`) for computation of the weight coefficients estimate (i.e., using $\boldsymbol{\alpha} = \texttt{pinv}(\mathbf{X})@\mathbf{y}$). The ridge regression has utilized `sklearn.linear_model.Ridge` function. The elastic-net has been implemented using `sklearn.linear_model.MultiTaskElasticNet`. In this function, the lasso corresponds to setting `l1_ratio` $= 1$ (where `l1_ratio` is the mixing value that controls the relative balance between $\ell_2$and ℓ_1 penalties), and the elastic-net corresponds to setting $0 <$ `l1_ratio` < 1. The parameter `alpha` in `MultiTaskElasticNet` controls the overall strength of the penalty term which is composed of the ℓ_1 and ℓ_2 penalties mixture. Hence, for lasso the only tuning parameter is the strength of constraint `alpha` ≥ 0 (with `l1_ratio` fixed at 1) while for elastic-net the tuning parameters are `alpha` ≥ 0 and `l1_ratio` $\in (0, 1)$. Finally, parallel to lasso and elastic-net, we have included two versions of our proposed p-bridge in this study. The respective versions are p-bridge at $k = 1.05$ which tunes only λ and p-bridge which tunes both λ and k value. Here we note that there is fundamental difference between the k value of p-bridge, which corresponds to the norm value itself, and `l1_ratio` of lasso which mixes between ℓ_1 and ℓ_2 norms.

As the test labels of these data sets are not released to the public, we shall use the validation set to test the classification prediction. Except for ols, the hyper-parameters λ, k, `alpha` and `l1_ratio` for the above methods have been determined based on a twofold cross-validation utilizing only the training set. These cross-validated hyper-parameters are subsequently utilized to retrain each method based on the entire training set for test prediction utilizing the unseen validation set. For tuning the hyper-parameters, the utilized search ranges are `l1_ratio` $\in$ [0.01, 0.1:0.1:1], `alpha`, $\lambda \in$ [0:0.1:1, 2:1:10, 20:10:100, 200:100:1000] and $k \in$ [1:0.1:2].

(ii) Results and observation. Accuracy: The results in terms of the average classification prediction accuracy obtained from 20 runs of evaluation using different training-test partitions are shown in Fig. 9. The error bars in the plot indicate the maximum and the minimum values. The average value with standard deviation in brackets are also marked above each bar. The classification prediction accuracy is the fraction of samples with their category being correctly predicted. These results show that p-bridge with full tuning capability (i.e., with both λ and k adjustable) has good prediction accuracy relative to that of the state-of-arts in all data sets while p-bridge at $k = 1.05$ shows inferior performance for the Arcene and the Dexter data sets. The lasso also shows under-performed predictions for these two data sets of under-determined systems even though these results are better than that of p-bridge at $k = 1.05$. The elastic-net shows relatively under-performed prediction only for the Dexter data set. Attributed to the utilization of the stable sklearn library and the pseudoinverse implementation, the ridge and the ols are observed to have good prediction accuracy over all data sets. It is observed that for p-bridge, the chosen k-values for these five data sets based on training validation are respectively 1.7, 1.9, 1.7, 1.9 and 1.05. These results show that the Gaussian prior (at $k = 2$) might not give the best fit in each case and p-bridge provides the alternatives. The additional degree of freedom provided by tuning the k values on top of the penalty λ term plays a part in determining a suitable model in face of data diversity. In terms of statistical significance concerning multiple algorithms on multiple data sets, Fig. 10 shows the Nemenyi test plot [38] in terms of the ranking of the six compared classifiers. This result shows that p-bridge ranks among the top two performers. However, there is no statistical significance among the compared algorithms since all the rankings fall within the critical value. While the performance is about the same as the top performer in average ranking, the advantage of our proposed method is the compression beyond the ℓ_2-norm penalty offered by these compared algorithms.

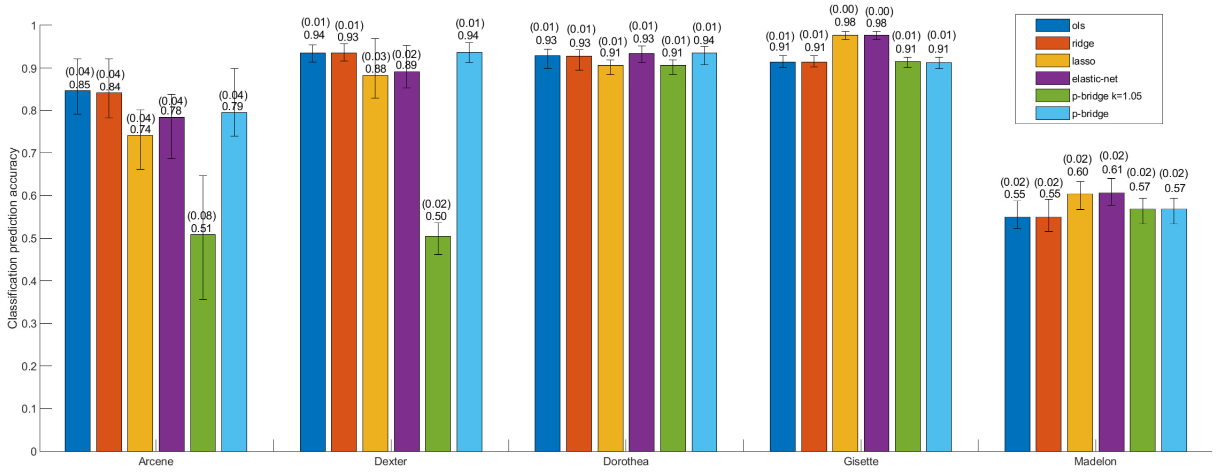


FIGURE 9. Average classification test accuracies with error bars indicating the maximum and the minimum values. The numbers on top of each bar indicate the average value and the standard deviation in brackets.

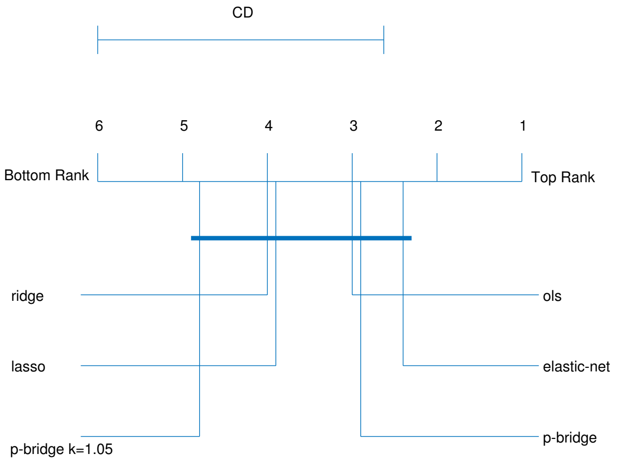
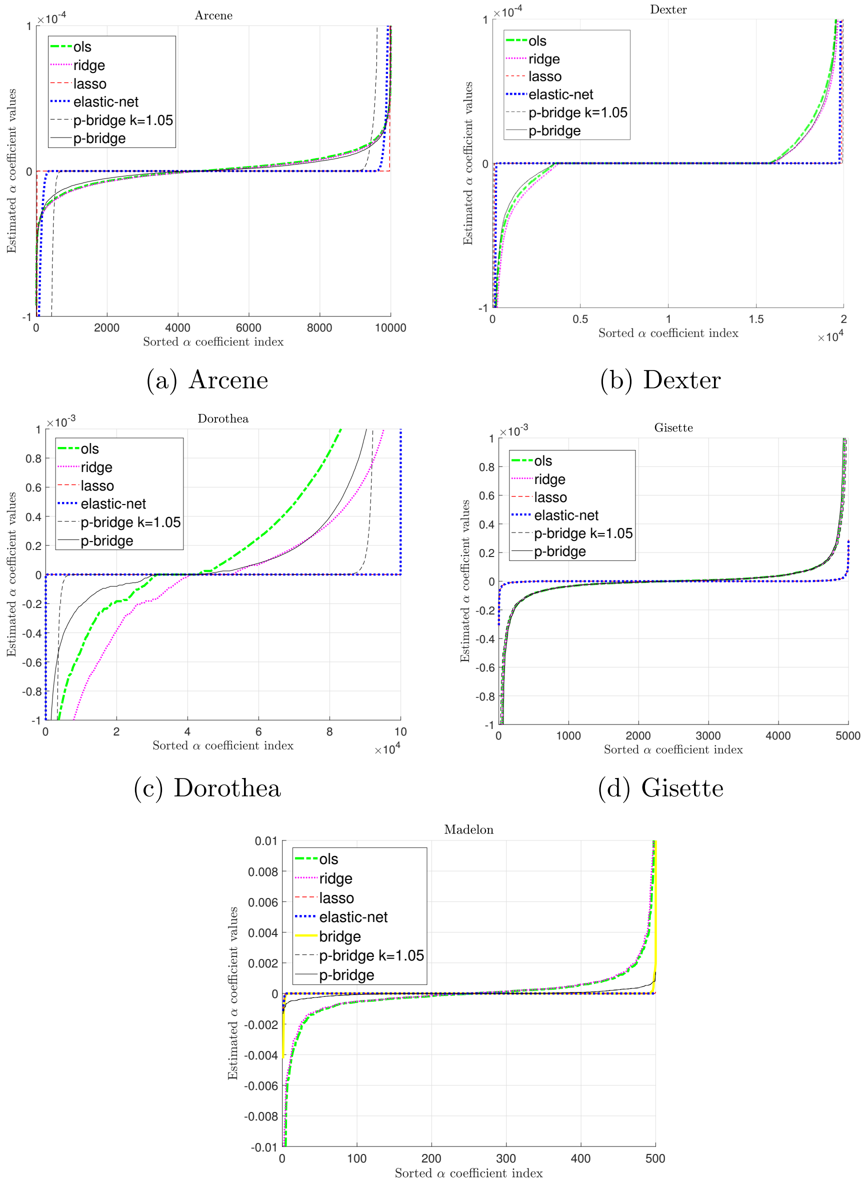


FIGURE 10. Statistical Nemenyi average rank plot where CD indicates the critical distance value. A lower value indicates a higher/better rank.

Sparseness of estimated coefficients: The estimated regression weight coefficients (sorted in ascending order) for each method on each data set are plotted in Fig. 11. These results show that the ols and the ridge have the densest estimation while the lasso and the elastic-net have the sparest estimation among the compared methods. The p-bridge at $k = 1.05$ shows sparser estimation than that of the p-bridge with tuned k and λ values for all data sets. The p-bridge at $k = 1.05$ show similar sparseness of estimation with lasso only for the Dexter data set, while lasso show the sparest estimation for the remaining data sets.



(a) Arcene

(b) Dexter

(c) Dorothea

(d) Gisette

(e) Madelon

Figure 11. Sorted weight coefficients for each data set

Training processing time: The training CPU processing time for each method is shown in Table 6. Due to the utilization of the computational intensive (but more stable) pseudoinverse in the ols, the fastest training CPU processing time goes to ridge as it has four data sets clocking the lowest processing time. Comparing p-bridge and elastic-net, it shows faster processing time in three data sets (Arcene, Dexter and Gisette) but slower processing time in two data sets (Dorthea and Madelon). The trend is similar for comparing between p-bridge at $k = 1.05$ and lasso. Here, we note that our implementation of the p-bridge has been based on the algorithm shown in section 3.6 without optimization of codes.

Table 6. Training CPU time in seconds

Method	Arcene	Dexter	Dorothea	Gisette	Madelon
ols	0.125	1.000	24.297	157.578	0.313
ridge	**0.063**	**0.109**	3.391	**10.922**	**0.016**
lasso	3.500	1.250	**1.297**	106.234	0.063
elastic-net	2.266	3.625	2.531	108.594	0.125
p-bridge at $k = 1.05$	0.125	0.969	14.984	100.875	0.313
p-bridge	0.172	1.047	16.547	97.375	0.313

Summarizing the experiments on the NIPS data sets, in terms of the prediction accuracy, the good performance relative to state-of-arts on all data sets shows the capability of the p-bridge to balance between the constraint weightage (λ) and the k-value which is related to the underlying ℓ_p-norm prior. The sparseness of estimation for p-bridge is seen to be controlled by the λ and k values. For under-determined systems, the accuracy of prediction appears to be much affected by the k values close to 1. In terms of training processing time, the current implementation of p-bridge shows faster processing speed than that of elastic-net in three data sets but slower processing speed in two data sets. These results show competing prediction accuracy and processing time with the state-of-the-arts.

5.2. **Recognition of Handwritten Digits.** The goal of this experiment is to observe the stretching behavior of p-bridge for prediction between the ℓ_1-norm and the ℓ_2-norm minimization of the parameter vector, particularly for data sets with large regions of empty background as a form of multicollinearity.

The first database is for optical recognition of handwritten digits (abbreviated as Optdigit) where it was collected based on a total of 43 people [39, 40]. The original 32×32 bitmaps were divided into non-overlapping blocks of 4×4 where the number of on pixels were counted within each block. This generated an input matrix of 8×8 where each element was an integer within the range [0, 16]. The dimensionality is thus reduced from 32×32 to 8×8. Each of the 10 numerical digits constitutes a category for recognition. The total number of 5620 samples are divided equally into two sets for training and testing in our experiment. The left panel of Fig. 12 shows some samples of the reduced resolution image taken from the training set (upper two rows) and the testing set (bottom two rows). The second database is the MNIST data set of handwritten digits [41, 42] which is a popular benchmark for algorithmic study and experimental comparison. This data set contains a training set of 60,000 samples and a testing set of 10,000 samples where each sample image is of 28 × 28 pixels resolution. Similar to the Optdigit, the MNIST data set has an output of 10 class labels. The right panel of Fig. 12 shows some training (upper two rows) and testing (lower two rows) sample images from the MNIST data set.

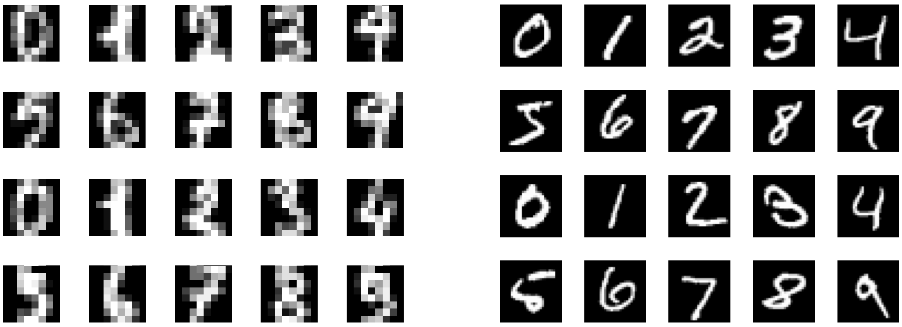


Figure 12. Samples of Optdigit (left panel) and MNIST (right panel) data sets. Samples shown in the upper two rows are taken from the training set and samples shown in the bottom two rows are taken from the test set.

(i) Experimental Setup. The input images of the two data sets of handwritten digits are mapped to the polynomial space for discrimination beyond linear decision. For the Optdigit data set, each image (8×8 pixels) is pooled at various sizes and reshaped into a row vector before expanded by a full polynomial of second order to generate the input features (1×2145 including the bias term) for regression. This data set for Optdigit is divided equally with each of the training and test matrices of 2810×2145 size. For the MNIST data set, a reduced polynomial [43] of third order has been applied to the pooled vector to generate the feature vector (1×22661 including the bias term). The training matrix is of 60,000×22661 size and the test matrix is of 10,000×22661 size for MNIST.

Both the over-determined and the under-determined settings of p-bridge will be evaluated. For the over-determined setting, the entire training matrix is utilized for training. For the under-determined setting, only 10 samples (1 sample for each of the 10 digits) are utilized for training for both the databases. This is known as *one-shot learning* in the community of computer vision. For both the over- and under-determined cases, the entire test set is utilized for evaluation.

(ii) Results and Observation. Over-determined case: Fig. 13 shows the test accuracies of the over-determined p-bridge for various k-values in $\{1.1, 1.2, \ldots, 2.0\}$ for both the databases. Alongside, the test accuracies for `MultiTaskElasticNet` is plotted at different mixing values of `l1_ratio` in $\{1.0, 0.9 \ldots, 0.3, 0.2, 0.001\}$ for Optdigit and MNIST, both at `alpha`=0.1. Here, `l1_ratio`=1 indicates the lasso setting and `l1_ratio`<1 indicates an elastic-net setting with `l1_ratioa`$\to 0$ approaches the ℓ_2-norm estimation. For both databases, the results show deviation from that of ℓ_2-norm estimation when the k-values move away from 2.0 for p-bridge and when `l1_ratio` values move away from 0. The deviation behaviors for p-bridge and elastic-net are apparently different. The p-bridge shows relatively stable prediction with a marginally uptrend whereas the elastic-net shows a deterioration trend when moving away from the ℓ_2-norm estimation. The peak performance for p-bridge shows a 99.15% accuracy at $k = 1.3$ for Optdigit and a 97.01% accuracy at $k = 1.7, 1.9$ for MNIST. For elastic-net, the mixing of ℓ_2-norm estimation with ℓ_1-norm estimation appears to cause significant deterioration of prediction accuracy for these two data sets.

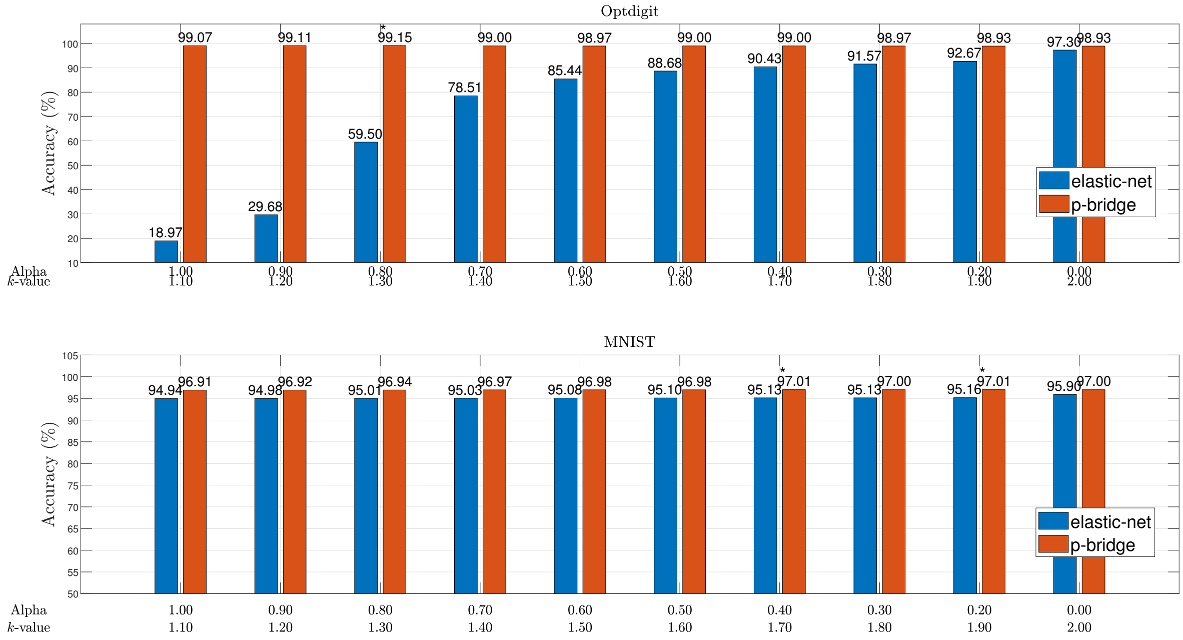


Figure 13. Accuracy results for the over-determined estimation. The asterisk (*) marks the highest accuracy achieved.

Under-determined case: Fig. 14 shows the accuracies of prediction by the p-bridge and elastic-net under the under-determined (single-shot learning) scenario. These accuracies are plotted over variation of the k-values and the `l1_ratio`-values respectively. Here, the prediction accuracy appears to degrade for both the p-bridge and the lasso for most cases when the estimations are moved away from the ℓ_2-norm formulation ($k = 2$ and `l1_ratio` $\to 0$). However, the prediction accuracy of p-bridge peaks at $k = 1.90$ and surpasses that based on the ℓ_2-norm for both the databases.

Estimated coefficients: Here, we study the estimated coefficients of p-bridge and compare them with those of lasso using the MNIST database under the over-determined setting. In order

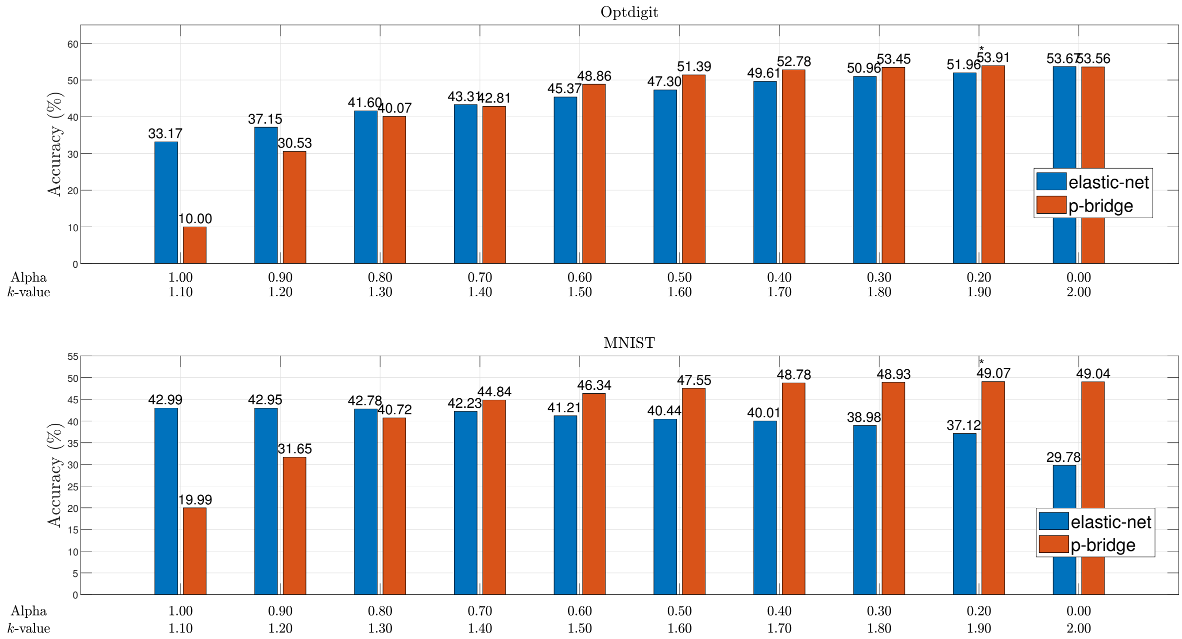


FIGURE 14. Accuracy results for the under-determined estimation (one-shot learning). The asterisk (*) marks the highest accuracy achieved.

to observe the importance of each pixel in the estimation process, a weighting coefficient is tied to each pixel for the estimation. This gives rise to a linear regression model with $28 \times 28 + 1$ parameters and 10 sets of weight coefficients corresponding to the one-hot encoded target digits. The lasso is set at (`l1_ratio`=1, `alpha`=0.01) and the p-bridge is set at ($k = 1$, $\lambda = 10$). Fig. 15 shows the heatmap of the learned coefficient values for each of the 10 digits. In general, these heatmaps show sparseness of the coefficients due to the ℓ_1 penalized learning for both lasso and p-bridge. In terms of the emphasized pixels with high absolute coefficient values, both lasso and p-bridge show similar locations, for both the positively emphasized (darker box) and the negatively emphasized (brighter box) pixels with variations. However, the value ranges of the coefficients differ for lasso and p-bridge. The lasso shows a lower prediction accuracy (82.5%) than that of the p-bridge (86.1%).

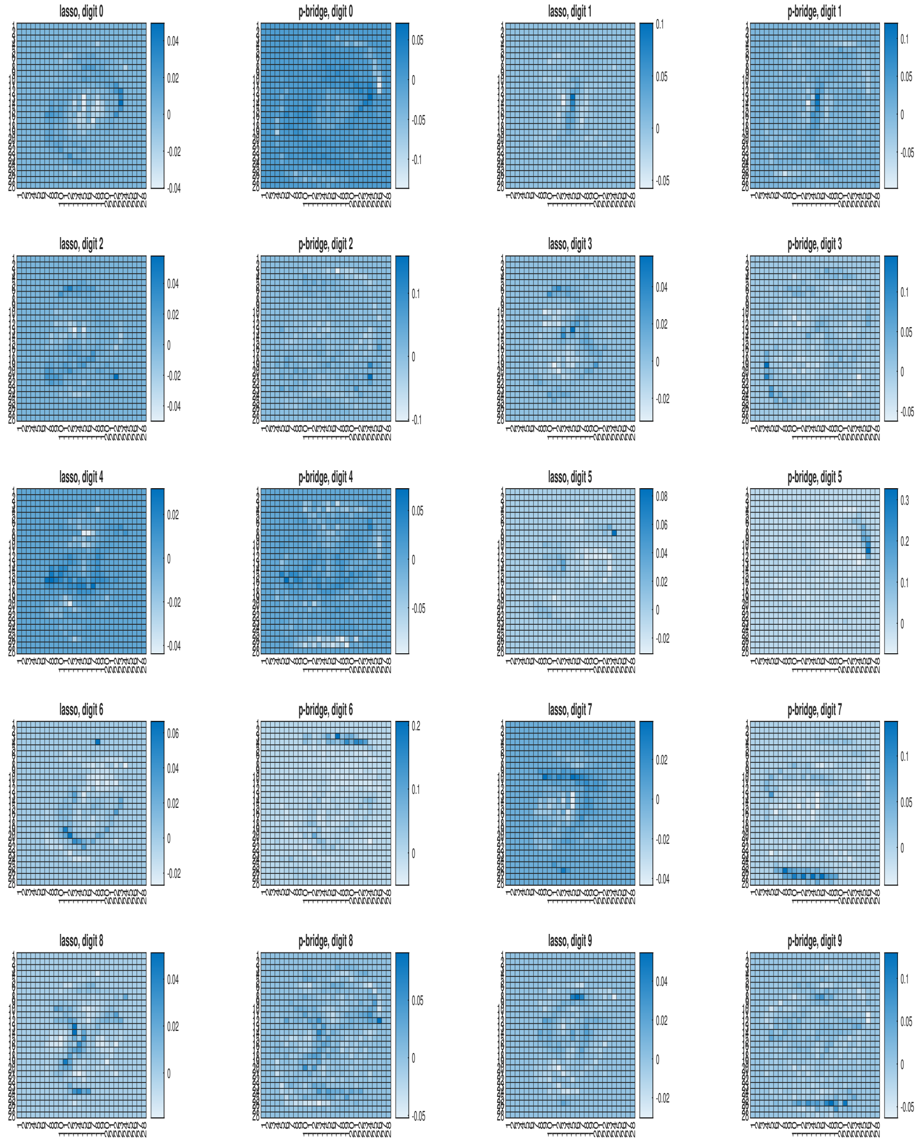


FIGURE 15. The estimated coefficients for each digit.

In order to observe which are the informative pixels responsible for discrimination, we plot the sum of training images and its logarithmic form plus the logarithm of the sum of absolute estimated coefficients corresponding to each of the pixels in Fig. 16. The main reason to plot in logarithmic form is to reveal near zero values which are not visible in the original plot. These images show close correspondence between the logarithm of sum of training images (the plot at top right) and that of the estimated coefficients (the plot at bottom right) of p-bridge. This results shows all the informative pixels of the image have been utilized by p-bridge. As for lasso, not all informative pixels have been utilized due to its crisp variable selection mechanism.

Summarizing the experiments for Optdigit and MNIST, both the over- and the under-determined cases show peaking of prediction accuracy beyond $k = 2$ (ℓ_2-norm regression) for p-bridge. This reveals the importance to move away from the Gaussian prior.

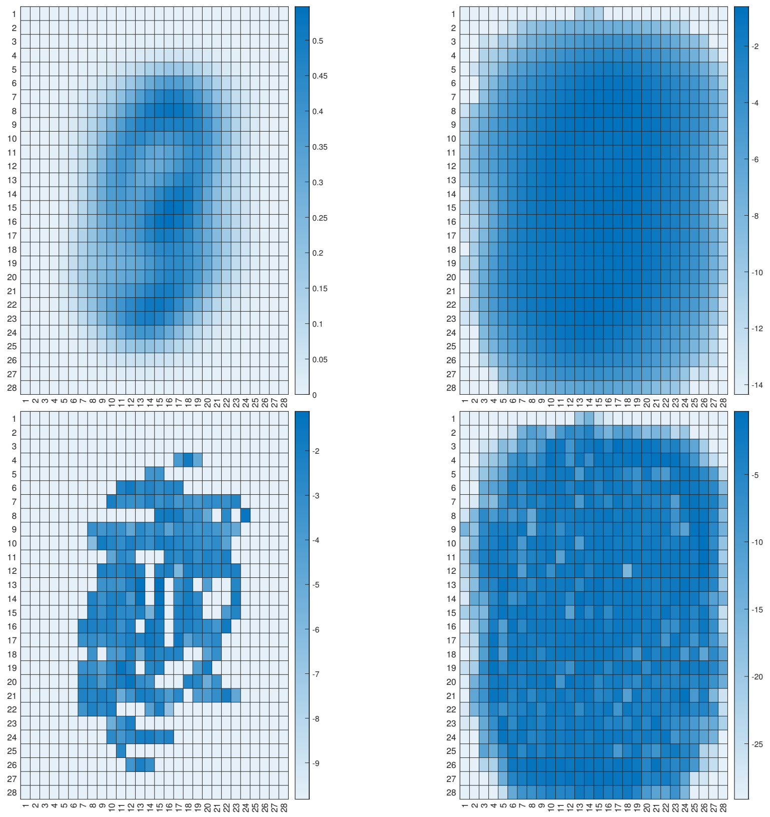


Figure 16. MNIST: The top two panels are heatmaps of the sum of training images (left) and its logarithmic (right) images. The bottom two panels are the heatmaps of the logarithm of sum of absolute coefficients for lasso (left) and p-bridge (right).

5.3. **Summary of Results and Observations. NIPS data sets:** In this set of experiments on binary classification, we have observed the estimation and prediction behaviors of p-bridge on real-world and artificial data sets of large dimension at under-determined (Arcene, Dexter, and Dorothea) and over-determined (Gisette and Madelon) settings. The results and observations are summarized as follows:

- Prediction accuracy: under various data situations, the p-bridge with adjustable λ (strength of constraints) and k-value (related to the ℓ_p-norm value) shows consistent and competing prediction accuracy relative to state-of-the-art methods namely, ols, ridge, lasso and elastic-net. For p-bridge at $k = 1.05$ (high compressive setting), the prediction accuracy shows a compromised performance for some of the under-determined cases due to the heavy bias introduced with many regressors being suppressed.
- Sparseness of coefficients: in general, the p-bridge shows comparable sparseness of estimated coefficients relative to that of elastic-net but lower sparseness comparing to that of lasso. For under-determined systems, the p-bridge at $k = 1.05$ achieves comparable sparseness of estimated coefficients by trading off the prediction accuracy.
- Training processing time: while the implemented p-bridge in Python was not code optimized, its training processing time shows competing processing speed with that of sklearn library's `MultiTaskElasticNet` except for the Dorothea and Madelon data sets.

Recognition of handwritten digits: In this set of experiments on multi-category classification based on multiple outputs formulation, the behavior of p-bridge is summarized as follows:

- Prediction accuracy: for the over-determined setting, the p-bridge shows relatively stable prediction over various k values with peak accuracy away from that at $k = 2$. This shows the inferiority of the Gaussian prior in these cases. For the under-determined setting, the p-bridge shows a decreasing prediction trend of accuracy with respect to lowering of k values. This can be interpreted as a trading off of accuracy with a higher sparseness of estimation.

- Sparseness of coefficients: an interesting observation from Fig. 16 is the high agreement of the zero-value coefficients with the background regions of the summed digits for p-bridge. Moreover, the emphasized coefficients (for both positive and negative values) are seen to fall on unique regions for each digit in Fig. 15.

6. Discussion

The proposed solution to the proximal bridge regression has been primarily motivated by the time consuming iterative search and convergence issue of those existing means. The solution has been formulated according to the two arising scenarios of over- and under-determined systems in practice. Capitalized on the better conditioning of matrix inversion among the two available options from the over- and under-determined cases, the solution for the under-determined system can be suitable for handling small sample size problems. The derivation of the solution has been based upon a smooth approximation of the absolute function by a differentiable surrogate. For the over-determined case, it turns out that the solution in primal form coincides with one derived based on a different cost function that utilized a local quadratic approximation as the regularization term. Although the solution comes in a recursion form, it turns out to affect only the regulation term. This can be utilized as a hyperparameter for generalizaton tuning. While the primal solution form works with inclusion of $k = 1$, the dual solution form works for $k > 1$. An analysis for the primal and the dual solutions shows that both estimates are biased, and this is consistent with the compressed estimation where some ideal parameters can be suppressed.

The solutions of the primal and dual forms have been extended to solve problems with multiple outputs. An algorithm that packs the two analytic solutions for multiple outputs under a single estimation framework has been constructed. Capitalized on the common covariance, estimation of the solution to multiple outputs can be packed in single matrix form. The simulated case studies verified the profiles of variables compression. The experiments on real and synthetic data sets of high dimension and large number of samples demonstrate the usefulness of the implementation. In terms of training processing times, the computation could be heavy due to the inversion of the covariance matrix. This is particular true for problems with large feature dimension and large sample size. In terms of prediction accuracy, the p-bridge shows relatively stable prediction over various k values with peak accuracy away from that at $k = 2$ under the over-determined setting. For the under-determined setting, the p-bridge shows a decreasing prediction trend of accuracy with respect to lowering of k values. This can be interpreted as a trading off of accuracy with a higher sparseness of estimation. The visualization of parameters for digits recognition shows high agreement of the zero-value coefficients with the background regions of the summed digits.

7. Conclusion

Dealing with convergence in iterative search is a challenging task in penalized learning, and the issue of small sample size learning has not been adequately addressed in bridge regression. In this work, we derive an analytic solution for bridge regression based on an approximation to the ℓ_p-norm to address the solution search problem. The solution is presented in primal form for over-determined systems and in dual form for under-determined systems. Due to better conditioning of the matrix inversion among the two available options, the dual form is particularly useful for small sample size learning. We extend these solution forms to problems with multiple outputs. Both the primal and dual estimations are shown to be biased as some coefficients are suppressed in the penalty formulation. We implement an algorithm that combines the two solution forms into one framework for recognition applications. Several numerical examples and experiments on real-world datasets demonstrate the usefulness of our algorithm for compressive applications. Specifically, our estimation allows trading off accuracy for higher sparseness of coefficients. Visualization of parameters for digit recognition reveals a linkage between the estimated coefficients and informative pixels.

CRediT authorship contribution statement

Kar-Ann Toh: Conceptualization, Methodology, implementation, Investigation, Writing – original draft. Giuseppe Molteni: Methodology, Writing – review & editing. Zhiping Lin: Methodology, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

All datasets used in the paper are publicly available.

Acknowledgment

The authors would like to thank the anonymous reviewers for their very constructive comments to improve the manuscript. This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2021R1A2C1093425).

References

[1] W. Fan, Y. Si, W. Yang, M. Sun, Class-specific weighted broad learning system for imbalanced heartbeat classification, Information Sciences 610 (2022) 525–548.

[2] S. Lee, A. Holzinger, Knowledge discovery from complex high dimensional data, in: Solving Large Scale Learning Tasks. Challenges and Algorithms, 2026, pp. 148–167.

[3] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, R. G. Baraniuk, Single-pixel imaging via compressive sampling, IEEE Signal Processing Magazine 25 (2) (2008) 83–91.

[4] W. H. Chai, S.-S. Ho, H. C. Quek, Representation recovery via l_1-norm minimization with corrupted data, Information Sciences 595 (2022) 395–426.

[5] R. Tibshirani, Regression shrinkage and selection via the lasso, J. Roy. Statist. Soc. Ser. B 58 (1996) 267–288.

[6] I. E. Frank, J. H. Friedman, A statistical view of some chemometrics regression tools, Technometrics 35 (1993) 109–148.

[7] J. Schaefer, M. Lehne, J. Schepers, F. Prasser, S. Thun, The use of machine learning in rare diseases: a scoping review, Orphanet Journal of Rare Diseases 15 (2020) 1–10.

[8] I. Hein, A. Rojas-Domínguez, M. Ornelas, G. D'Ercolec, L. Peloschek, Automated classification of archaeological ceramic materials by means of texture measures, Journal of Archaeological Science: Reports 21 (2018) 921–928.

[9] J. Fan, R. Li, Variable selection via nonconcave penalized likelihood and its oracle properties, Journal of the American Statistical Association 96 (456) (2001) 1348–1360.

[10] H. Zou, R. Li, One-step sparse estimates in nonconcave penalized likelihood models, The Annals of Statistics 36 (4) (2008) 1509–1533.

[11] H. Ding, Y. Sun, N. Huang, Z. Shen, Z. Wang, A. Iftekhar, X. Cui, RVGAN-TL: A generative adversarial networks and transfer learning-based hybrid approach for imbalanced data classification, Information Sciences 629 (2023) 184–203.

[12] W. Jiang, S. Qiu, T. Liang, F. Zhang, Cross-project clone consistent-defect prediction via transfer-learning method, Information Sciences 635 (2023) 138–150.

[13] Z. Qin, H. Wanga, C. B. Mawuli, W. Han, R. Zhang, Q. Yang, J. Shao, Multi-instance attention network for few-shot learning, Information Sciences 611 (2022) 464–475.

[14] H. Wang, J. Zhu, F. Feng, Elastic net twin support vector machine and its safe screening rules, Information Sciences 635 (2023) 99–125.

[15] R. O. Duda, P. E. Hart, D. G. Stork, Pattern Classification, 2nd Edition, John Wiley & Sons, Inc, New York, 2001.

[16] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer, New York, 2017.

[17] A. E. Hoerl, R. W. Kennard, Ridge regression: Biased estimation for nonorthogonal problems, Technometrics 12 (1) (1970) 55–67.

[18] A. Jain, D. Zongker, Feature selection: Evaluation, application, and small sample performance, IEEE Trans. on Pattern Analysis and Machine Intelligence 19 (2) (1997) 153–158.

[19] J. Lu, K. N. Plataniotis, A. N. Venetsanopoulos, Regularized discriminant analysis for the small sample size problem in face recognition, Pattern Recognition Letters 24 (2003) 3079–3087.

[20] Y. Wang, Q. Yao, J. T. Kwok, L. M. Ni, Generalizing from a few examples: A survey on few-shot learning, ACM Computing Surveys 53 (3) (2021) 1–34.

[21] L. Hu, H. Liang, L. Lu, Splicing learning: A novel few-shot learning approach, Information Sciences 552 (2021) 17–28.

[22] H. Zou, T. Hastie, Regularization and variable selection via the elastic net, Journal of the Royal Statistical Society, Series B 67 (2005) 301–320, (Part 2).

[23] W. J. Fu, Penalized regressions: The bridge versus the lasso, Journal of Computational and Graphical Statistics 7 (3) (1998) 97–416.

[24] C. Park, Y. J. Yoon, Bridge regression: Adaptivity and group selection, Journal of Statistical Planning and Inference 141 (11) (2011) 3506–3519.

[25] J. Wang, A wonderful triangle in compressed sensing, Information Sciences 611 (2022) 95–106.

[26] M. Su, W. Wang, Elastic net penalized quantile regression model, Journal of Computational and Applied Mathematics 392 (2021) 1–16.

[27] Y. Tian, X. Song, Bayesian bridge-randomized penalized quantile regression, Computational Statistics and Data Analysis 144 (2022) 1–16.

[28] S. Kawano, Selection of tuning parameters in bridge regression models via bayesian information criterion, Statistical Papers 55 (2014) 1207–1223.

[29] X. Pang, Y. Xu, A reconstructed feasible solution-based safe feature elimination rule for expediting multi-task lasso, Information Sciences 642 (2023) 1–22.

[30] M. Czajkowski, K. Jurczuk, M. Kretowski, Steering the interpretability of decision trees using lasso regression – an evolutionary perspective, Information Sciences 638 (2023) 1–15.

[31] K.-A. Toh, Z. Lin, L. Sun, Z. Li, Stretchy binary classification, Neural Networks 97 (2018) 74–91.

[32] C. Ramirez, R. Sanchez, V. Kreinovich, M. Argaez, $\sqrt{x^2+\mu}$ is the most computationally efficient smooth approximation to $|x|$: a proof, Journal of Uncertain Systems 8 (3) (2014) 205–210.

[33] G. H. Hardy, J. E. Littlewood, G. Pólya, Inequalities, Cambridge Univesity Press, London, 1934.

[34] T. Stamey, J. Kabalin, J. McNeal, I. Johnstone, F. Freiha, E. Redwine, N. Yang, Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate II. radical prostatectomy treated patients, Journal of Urology 16 (1989) 1076–1083.

[35] The MathWorks, Matlab and simulink, in: [http://www.mathworks.com/], 2023.

[36] B. Yüzbasi, M. Arashi, F. Akdeniz, Penalized regression via the restricted bridge estimator, Soft Computing 25 (2019) 8401–8416.

[37] I. Guyon, Design of experiments of the NIPS 2003 variable selection benchmark, in: `http://clopinet.com/isabelle/Projects/NIPS2003/Slides/NIPS2003-Datasets.pdf`, 2003, (NIPS Feature Selection Challenge).

[38] J. Demšar, Statistical comparisons of classifiers over multiple data sets, Journal of Machine Learning Research 7 (2006) 1–30.

[39] C. Kaynak, Methods of combining multiple classifiers and their applications to handwritten digit recognition, Master's thesis, Institute of Graduate Studies in Science and Engineering, Bogazici University (1995).

[40] M. Lichman, http://archive.ics.uci.edu/mlUCI machine learning repository (2013). `http://archive.ics.uci.edu/ml`

[41] Y. LeCun, C. Cortes, C. J. Burges, http://yann.lecun.com/exdb/mnist/The MNIST database (2018). `http://yann.lecun.com/exdb/mnist/`

[42] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 16 (11) (1998) 2278–2324.

[43] K.-A. Toh, Q.-L. Tran, D. Srinivasan, Benchmarking a reduced multivariate polynomial pattern classifier, IEEE Trans. on Pattern Analysis and Machine Intelligence 26 (6) (2004) 740–755.

[1] School of Electrical and Electronic Engineering, Yonsei University, 50 Yonsei-ro, Seodaemun-gu, Seoul, 03722, South Korea
Email address: `katoh@yonsei.ac.kr`

[2] Dipartimento di Matematica, Università di Milano, via Saldini 50, 20133 Milano, Italy
Email address: `giuseppe.molteni1@unimi.it`

[3] School of Electrical and Electronic Engineering, Nanyang Technological University, Nanyang Avenue, 639798, Singapore
Email address: `ezplin@ntu.edu.sg`