Kernel Information Augmentation: Unfolding the Hidden Structure

Anonymous Author(s)
Affiliation
Address
email

Abstract

We propose a kernel information augmentation method to improve the current dimension reduction and regression methods, two fundamental machine learning devices. The method directly uses mean and variance structures of the feature vector to enhance the estimation efficiency, and in turn improves the prediction accuracy. The efficiency improvements are justified theoretically and numerically through extensive simulations and real data analysis. We also implement the proposed method for improving the accuracies of deep learning algorithms. The KIA is widely applicable to improve various dimension reduction and regression procedures.

1 Introduction

In supervised learning, when the dimension p of feature $\mathbf{X}$ is high, we wish to have a lower d dimensional features $\mathbf{Z}$ so that the relation between outcome Y and $\mathbf{X}$ is sufficiently captured by the relation between Y and $\mathbf{Z}$. Such dimension reduction not only eases the computation burden but also allows for data visualization and understanding the important features that associated with the outcome. To facilitate the estimation and theoretic derivations, $\mathbf{Z}$ is often assumed to be a linear projection of $\mathbf{X}$ in the form of $\boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}$, where $\boldsymbol{\beta}$ is a $p \times d$ dimensional matrix. When a lower dimensional feature is of the main interest, the exact relation between Y and $\mathbf{Z}$ can be unspecified. This falls into the sufficient dimension reduction framework [1, 2, 3, 4, 10, 12]. Once a lower dimensional $\mathbf{Z}$ is obtained, a parametric regression between Y and $\mathbf{Z}$ is often good enough for prediction. One typical regression model is the generalized linear model (GLM) where Y given $\mathbf{Z}$ follows a distribution in the exponential family [5].

The most representative sufficient dimension reduction methods are the sliced inverse regression (SIR) [3] and the sliced average variance estimation (SAVE) [1]. SIR and SAVE use the eigenvectors associated with the top d eigenvalues of $\boldsymbol{\Lambda} = \mathrm{cov}(\mathbf{X})^{-1}\mathrm{cov}\{E(\mathbf{X}|Y)\}$ and $\boldsymbol{\Lambda} = \mathrm{cov}(\mathbf{X})^{-1}E[\{\mathrm{cov}(\mathbf{X}) - \mathrm{cov}(\mathbf{X}|Y)\}^2]$ to obtain a version of $\mathbf{Z}$, respectively. Here d is selected as the largest d eigenvalues of $\boldsymbol{\Lambda}$ that together explain a large proportion, for example over 80%, of the total variations represented by the total sum of the eigenvalues. These procedures are particularly useful in text mining problems [7, 9]. In the analysis of the 20 news group data, we show that a 20 dimensional $\mathbf{Z}$ is sufficient to capture the relation between Y and a 3000 dimensional $\mathbf{X}$. The corresponding top 1 prediction accuracy is 72% when using 20 dimension $\mathbf{Z}$ as the covariate, while the accuracy is 74% when using the entire 3000 dimensional $\mathbf{X}$. It can be seen that the sufficient dimension reduction reduces the dimension of the covariate without sacrificing accuracy.

SIR and SAVE have been successfully implemented when the linearity condition and constant variance conditions are satisfied. Here the linearity condition means that $E(\mathbf{X} \mid \boldsymbol{\beta}^{\mathrm{T}}\mathbf{X})$ is a linear function of $\boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}$ and the constant variance condition means that $\mathrm{cov}(\mathbf{X} \mid \boldsymbol{\beta}^{\mathrm{T}}\mathbf{X})$ is a constant matrix. The linearity and constant variance conditions are not stringent, they are automatically satisfied when

Submitted to 32nd Conference on Neural Information Processing Systems (NIPS 2018). Do not distribute.

$\mathbf{X}$ has an elliptical distribution, such as the most widely used Gaussian, t and Laplace distributions. Further, the two conditions hold independently of the generating mechanism of the outcome Y. So a question is: *Would it be useful to incorporate these distribution conditions to improve the estimation efficiency?* We think the answer is positive. Based on this structure, we develop a kernel information augmentation (KIA) method, which updates the orignial sufficient dimension reduction estimator to achieve better estimation efficiency and prediction accuracy.

Given a low dimension projection $\mathbf{Z}$, we can form a generalized linear model between Y and $\mathbf{Z}$ for prediction. The estimations are often performed through minimizing the negative logarithm of the likelihood. The resulting GLM estimators are optimal with the smallest estimation variations provide the conditional distribution of Y given $\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{Z}$ is correctly specified. Because $\mathbf{Z}$ is a linear transformation of $\mathbf{X}$, the linearity and constant variance conditions of $\mathbf{Z}$ given $\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{Z}$ readily hold. This motivates us to develop a KIA method that directly augments the loss function under the GLM framework to achieve better efficiency.

As we will show in the empirical study, the KIA approach works well in practice, comparing favorably to the standard inverse regression and generalized linear model methods. In addition, the KIA method is widely applicable to improve general regression procedures. To demonstrate this, we assess the performance of the KIA method through simulations under various settings. Further, we illustrate its applications in the internet of things data analysis, text analysis, and under the deep learning framework. We also provide insights and rigorous justification of the estimation efficiency improvement.

Notations and the identifiability condition Random variables are denoted with upper-case characters. $\|\cdot\|_2$ denote the L_2 norm. The matrices are denoted by bold face letters. Let $\boldsymbol{\Sigma} = \mathrm{cov}(\mathbf{X})$, $\mathbf{P}_{\boldsymbol{\beta}} \equiv \boldsymbol{\Sigma}\boldsymbol{\beta}(\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\beta})^{-1}\boldsymbol{\beta}^{\mathrm{T}}$, and $\mathbf{Q}_{\boldsymbol{\beta}} \equiv \boldsymbol{\Sigma} - \mathbf{P}_{\boldsymbol{\beta}}\boldsymbol{\Sigma}\mathbf{P}_{\boldsymbol{\beta}}^{\mathrm{T}}$. For the identification of $\boldsymbol{\beta}$, we impose the condition that the upper $d \times d$ matrix of $\boldsymbol{\beta}$ is identity. Let $\boldsymbol{\beta}_L$ be the lower $(p-d)\times d$ submatrix of $\boldsymbol{\beta}$. $\mathrm{vecl}(\boldsymbol{\beta}) = \mathrm{vec}(\boldsymbol{\beta}_L)$. Further, let $\mathbf{P}_{\boldsymbol{\beta},L} \equiv (\mathbf{0}_{(p-d)\times d}, \mathbf{I}_{p-d})\mathbf{P}_{\boldsymbol{\beta}}$ be the lower $(p-d)\times p$ submatrix of $\mathbf{P}_{\boldsymbol{\beta}}$ and $\mathbf{Q}_{\boldsymbol{\beta},L} \equiv (\mathbf{0}_{(p-d)\times d}, \mathbf{I}_{p-d})\mathbf{Q}_{\boldsymbol{\beta}}$ be the lower $(p-d)\times p$ submatrix of $\mathbf{Q}_{\boldsymbol{\beta}}$.

2 Kernel Information Augmentation on the Sufficient Dimension Reduction

2.1 KIA algorithms and properties

In dimension reduction problems, we have independent identically distributed data $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)$, and additional $\mathbf{X}_{n+1}, \ldots, \mathbf{X}_N$ without corresponding response variables. Our goal is to make the use of the linearly and constant variance properties on $\mathbf{X}$ to improve the performance of the sufficient dimension reduction. Follow the linearity condition, we have $E(\mathbf{X} \mid \boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}) = \mathbf{P}_{\boldsymbol{\beta}}\mathbf{X}$ [3]. And with the constant variance condition, it is easy to show that $\mathrm{var}(\mathbf{X} \mid \boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}) = \mathbf{Q}_{\boldsymbol{\beta}}$ [3].

To incorporate the linearity condition, note that $E(\mathbf{X} \mid \boldsymbol{\beta}^{\mathrm{T}}\mathbf{X})$ can be estimated by the Nadaraya–Watson estimator [6, 11] in the form of

$$\boldsymbol{\psi}(\boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}_j, \boldsymbol{\beta}) \equiv \frac{\sum_{i=1}^{N} \mathbf{X}_i K_{\mathbf{h}}(\boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}_i - \boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}_j)}{\sum_{i=1}^{N} K_{\mathbf{h}}(\boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}_i - \boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}_j)},$$

where $K_{\mathbf{h}}(\mathbf{x}) = \prod_{j=1}^{d} 1/h_j K(\mathbf{h}^{-1}\mathbf{x})$ is a symmetric kernel function with bandwidth $\mathbf{h} = \mathrm{diag}(h_1, \ldots, h_d)$. A simple way of making use of the additional covariates information is to directly update the original SIR estimators $\widehat{\boldsymbol{\beta}}_{\mathrm{sir}}$ via minimizing

$$\sum_{j=1}^{N} \left\| \boldsymbol{\psi}(\widehat{\boldsymbol{\beta}}_{\mathrm{sir}}^{\mathrm{T}}\mathbf{X}_j, \widehat{\boldsymbol{\beta}}_{\mathrm{sir}}) - \mathbf{P}_{\boldsymbol{\beta}}\mathbf{X}_j \right\|_2^2 \tag{1}$$

with repect to $\boldsymbol{\beta}$, which gives the minimier $\widehat{\boldsymbol{\beta}}$.

Minimizing (1) is a linear programming problem, which can be solved efficiently through the following steps in **Algorithm 1**:

Algorithm 1 Update $\widehat{\boldsymbol{\beta}}_{\rm sir}$

Inputs: $\widehat{\boldsymbol{\beta}}_{\rm sir}$, $\mathbf{X}_i$, $i = 1, \ldots N$;

Step 1: Construct the Nadaraya–Watson estimator $\boldsymbol{\psi}(\widehat{\boldsymbol{\beta}}_{\rm sir}^{\rm T}\mathbf{X}_j, \widehat{\boldsymbol{\beta}}_{\rm sir})$;

Step 2: Center $\mathbf{X}_j$ and $\boldsymbol{\psi}(\widehat{\boldsymbol{\beta}}_{\rm sir}^{\rm T}\mathbf{X}_j, \widehat{\boldsymbol{\beta}}_{\rm sir})$, denoting the centered results by $\mathbf{X}_{jc}$ and $\boldsymbol{\psi}_c(\widehat{\boldsymbol{\beta}}_{\rm sir}^{\rm T}\mathbf{X}_j, \widehat{\boldsymbol{\beta}}_{\rm sir})$, respectively;

Step 3: Obtain $\widehat{\boldsymbol{\beta}}$ as the eigen-vectors of $\sum_{j=1}^{N}\{\boldsymbol{\psi}_c(\widehat{\boldsymbol{\beta}}_{\rm sir}^{\rm T}\mathbf{X}_j, \widehat{\boldsymbol{\beta}}_{\rm sir})\mathbf{X}_{jc}^{\rm T}\}(\sum_{j=1}^{N}\mathbf{X}_{jc}\mathbf{X}_{jc}^{\rm T})^{-1}$ corresponding to its d largest eigen-values.

The properties of the kernel and the bandwidth selection are crucial to ensure the consistency of the estimators. Without loss of generality, we assume $\mathbf{X}$ has zero mean and identity variance. We further require

(C1) The kernel function $K(\cdot)$ is symmetric, has compact support and is Lipschitz continuous on its support. It satisfies

$$\int K(\mathbf{t})d\mathbf{t} = 1, \ \int \mathbf{t}^{\rm T}K(\mathbf{t})d\mathbf{t} = 0, \ \int \mathbf{t}\mathbf{t}^{\rm T}K(\mathbf{t})d\mathbf{t} < \infty.$$

(C2) The bandwidth $h_j = O(n^{-\kappa})$ for $1/4 < \kappa < (2d)^{-1}$.

Theorem 1. *Under the linearity and constant variance conditions, suppose Conditions (C1) and (C2) hold, we have*

$$n[var\{vecl(\widehat{\boldsymbol{\beta}})\} - var\{vecl(\widehat{\boldsymbol{\beta}}_{\rm sir})\}] = -3n/(4N)(\boldsymbol{\beta}^{\rm T}\boldsymbol{\beta})^{-1} \otimes (\mathbf{Q}_{\boldsymbol{\beta},L}\mathbf{Q}_{\boldsymbol{\beta},L}^{\rm T})^{-1} + o_p(1).$$

The difference var$\{$vecl$(\widehat{\boldsymbol{\beta}})\}$ − var$\{$vecl$(\widehat{\boldsymbol{\beta}}_{\rm sir})\}$ is negative definite, which implies $\widehat{\boldsymbol{\beta}}$ has smaller estimation variation. Hence, the KIA-SIR generally achieves better accuracy through improving the estimation efficiency.

Using the same algorithm, while replacing $\widehat{\boldsymbol{\beta}}_{\rm sir}$ by the SAVE estimator $\widehat{\boldsymbol{\beta}}_{\rm save}$, we can also show that

Theorem 2. *Under the linearity and constant variance conditions, suppose Conditions (C1) and (C2) hold, we have*

$$n[var\{vecl(\widehat{\boldsymbol{\beta}})\} - var\{vecl(\widehat{\boldsymbol{\beta}}_{\rm save})\}] = -3n/(4N)(\boldsymbol{\beta}^{\rm T}\boldsymbol{\beta})^{-1} \otimes (\mathbf{Q}_{\boldsymbol{\beta},L}\mathbf{Q}_{\boldsymbol{\beta},L}^{\rm T})^{-1} + o_p(1).$$

Theorem 2 further fortifies the strength of the KIA estimator, which, compared with the standard SAVE methods, has smaller variation. It is worth mentioning that the KIA method can be easily generalized to improve other commonly used sufficient dimension reduction estimators.

2.2 Simulation Study

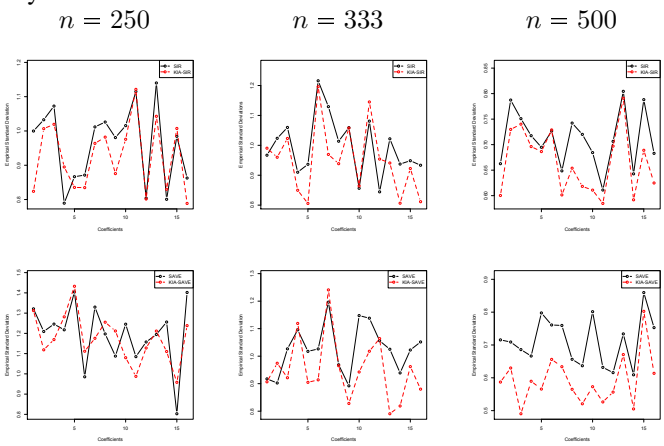
We conduct simulations to explore the efficiency improvement for the proposed procedures. We generate $N = 1000$ copies of $\mathbf{X}$ from Gaussian distribution with mean 0, var$(X_k) = 1$ and cov$(X_k, X_l) = 0.5$ for $k \neq l$, where X_k is the kth covariate, $k, l = 1, \ldots, 10$. We define $\boldsymbol{\beta}_1$ to be a vector of 1's with length 10, and $\boldsymbol{\beta}_2 = (1, -1, 1, -1, 1, -1, 1, -1, 1, -1)^{\rm T}$. Let $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2)$. Further, we select the first n samples as the supervised data, and generate the corresponding response Y from the model $Y_i = (\boldsymbol{\beta}_1^{\rm T}\mathbf{X}_i)/\{1+(\boldsymbol{\beta}_2^{\rm T}\mathbf{X}_i+1)^2\}+\epsilon_i$, where ϵ_i is a standard Gaussian random error. We implement the SIR and SAVE estimation procedures based on the supervised data. Furthermore, we implement the KIA-SIR and KIA-SAVE methods to enhance estimation efficiency through **Algorithm 1** described above. We use the trace correlation trace$(\mathbf{P}_1, \mathbf{P}_2)$ to measure the closeness between $\mathbf{P}_1, \mathbf{P}_2$. The matrices with larger trace correlation have smaller distance in terms of the Frobenius norm, and are in turn closer to each other.

Table 1 shows the empirical mean of the trace correlation between the estimators and truth based on 100 simulations. Clearly, the KIA estimators outperform the original SIR and SAVE estimators with larger trace correlations with the truth. Further, we plot the empirical standard deviation of the vectorized estimators in Figure 1. It can be seen that when n/N increases, the variance reduction becomes more clear. This implies KIA-SIR and KIA-SAVE outperforms their counterparts when n/N is sufficiently large.

Table 1: Comparisons between KIA-SIR, SIR and KIA-SAVE, SAVE on trace correlations over 100 simulations with $N = 1000$. Larger value indicates better performance.

n	KIA-SIR	SIR	KIA-SAVE	SAVE
250	0.631	0.594	0.382	0.322
333	0.678	0.649	0.534	0.480
500	0.772	0.751	0.737	0.696

Figure 1: The empirical standard derivations from different estimators over 100 simulations. $N = 1000$. The red line and black line represent the empirical standard deviations for KIA-SIR and SIR methods, respectively.



2.3 Internet of Things Application

We apply the proposed method to analyze the continuously monitored blood pressure data, a typical internet of things data. The dataset is from a nation-wise stroke study where $N = 297$ observations with complete blood pressure trajectory ($p = 96$ dimensional $\mathbf{X}$) enter the analysis, within which $n = 174$ observations have time to stroke recurrence (Y). To construct the training set, we randomly pick $n = 160$ supervised observations into the training set. The rest 14 supervised data serve as the testing data.

Since we do not know the values of the underlying true parameters, to evaluate the methods, we test the dependency between Y and $\boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}$ through the distance correlation measure proposed in [8]. We select $d = 6$ as the sufficient dimension, because the first six sufficient directions comprise over 80% of the total variation. We repeat this process 100 times to obtain the average performances for the KIA-SIR, SIR, KIA-SAVE, and SAVE in Table 2. Clearly, the KIA methods outperform their orignial counterparts with larger correlations between the sufficient directions and the outcomes. We further use boxplot to display the efficiency gain of the KIA method over 540 unknown parameters. Figure 2 shows that the KIA-SIR and KIA-SAVE have smaller estimation variations compare with their counterparts.

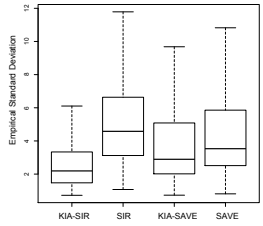
Table 2: The empirical distance correlation over 100 simulation by using KIA-SIR, SIR, KIA-SAVE and SAVE algorithms.

KIA-SIR	SIR	KIA-SAVE	SAVE
0.143	0.132	0.140	0.130

2.4 20 news group data analysis

Through reducing the redundant dimensions, the sufficient dimension reductions retain the most representative features to describe the data. To see this, we implement the SIR on analyzing the 20

Figure 2: The boxplot of the empirical standard derivations of 540 coefficients



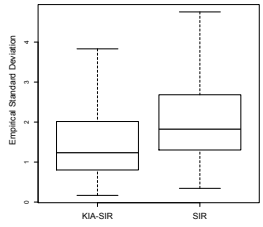
news group data. In the dataset, Y is 20 news categories, each corresponding to a different topic, and $\mathbf{X}$ comprises the counts for top 3000 relevant phrases. The total sample size is $N = 18774$, from which we chose 7505 to form the test data. In the training data, we randomly select $n = 10000$ as the labeled data, which is roughly 50% of the total sample size. We utilize the SIR method to obtain an estimator for $\boldsymbol{\beta}$, denoted by $\widehat{\boldsymbol{\beta}}_{\text{sir}}$ and model the relation between Y and $\widehat{\boldsymbol{\beta}}_{\text{sir}}^{\text{T}}\mathbf{X}$ through softmax regression. We select $d = 20$ to be the column dimension for $\boldsymbol{\beta}$, corresponding to the number of the categories in the data. For comparison, we implement the softmax ridge regression between Y and $\mathbf{X}$ with 0.01 weight on the L_2 penalty. It can be seen from Table 3 that SIR uses only 20 sufficient directions to achieve 68% prediction accuracy, which significantly outperforms ridge regression.

We further use the **Algorithm 1** to obtain KIA-SIR estimator. It can be seen from Table 3 that the KIA-SIR improves the prediction accuracy upon the orignal SIR method. This improvement is not unexpected, as we can see from Figure 3, the estimation variation for $\boldsymbol{\beta}$ is largely reduced compared to the SIR method, which undoubtedly leads to a better prediction.

Table 3: The mean top 1 accuracies over the testing data for the KIA-SIR, SIR, Softmax-Ridge regression over 100 random samples. Here the ridge regression penalty weight is 0.01.

KIA-SIR	SIR	Softmax-Ridge
0.717	0.678	0.644

Figure 3: The boxplot of the empirical standard derivations of 59600 coefficients



3 Kernel Information Augmentation on the Generalized Linear Model

3.1 KIA algorithms and properties

After obtaining the low dimensional projection $\mathbf{Z}$, we further develop an algorithm to integrate the information from the covariates through KIA to enhance the estimation efficiency in the generalized linear model.

We name the proposed algorithm KIA-GLM which targets at minimizing the weighted combination of the average negative log likelihood $-l(\boldsymbol{\alpha})$, and the L_2 loss (1) to improve the estimation, i.e. we

minimize

$$-l(\boldsymbol{\alpha}) + \lambda/2 L(\boldsymbol{\alpha}),$$

where

$$L(\boldsymbol{\alpha}) = N^{-1} \sum_{j=1}^{N} \left\| \boldsymbol{\psi}(\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{Z}_j, \boldsymbol{\alpha}) - \mathbf{P}_{\boldsymbol{\alpha}}\mathbf{Z}_j \right\|^2$$

with $\lambda > 0$. Since the likelihood is known, $\boldsymbol{\alpha}$ is identifiable without additional constraints.

Denote the resulting estimator as $\widehat{\boldsymbol{\alpha}}$, and the regular GLM estimator by $\widehat{\boldsymbol{\alpha}}_{\mathrm{glm}}$. Further we define $\mathbf{V}_1$ to be the second derivative of the log likelihood so that $n^{-1}E(\mathbf{V}_1)^{-1}$ is the asymptotic variance of $\widehat{\boldsymbol{\alpha}}_{\mathrm{glm}}$. In addition, let

$$\mathbf{W} = \left(\frac{\mathbf{f}'(\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{Z})}{f(\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{Z})} \otimes \{\mathrm{var}(\mathbf{Z} \mid \boldsymbol{\alpha}^{\mathrm{T}}\mathbf{Z})\} + \{(\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\alpha})^{-1}\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{Z}\} \otimes \mathbf{Q}_{\boldsymbol{\alpha}} \right)^{\otimes 2},$$

where $f(\cdot)$ is the density of $\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{Z}$ and $\mathbf{f}'(\mathbf{u}) = \partial f(\mathbf{u})/\partial \mathbf{u}$. We show that

Theorem 3. *Under the linearity and constant variance conditions, suppose Conditions (C1) and (C2) hold, when $\lambda \leqslant 1$, we have*

$$\begin{aligned} & n[var\{\mathrm{vec}(\widehat{\boldsymbol{\alpha}})\} - var\{\mathrm{vec}(\widehat{\boldsymbol{\alpha}}_{\mathrm{glm}})\}] \\ = \; & \{E(\mathbf{V}_1) + \lambda E(\mathbf{W})\}^{-1} - \{E(\mathbf{V}_1)\}^{-1} + \{E(\mathbf{V}_1) + \lambda E(\mathbf{W})\}^{-1} \left[\{n/N\lambda^2 - \lambda)\} E(\mathbf{W})\right] \\ & \times \{E(\mathbf{V}_1) + \lambda E(\mathbf{W})\}^{-1} + o_p(1), \end{aligned} \tag{2}$$

which is negative definite when $\lambda \leqslant 1$.

Similar to KIA-SIR and KIA-SAVE, the KIA-GLM estimator is obviously more efficient than the standard GLM estimator.

Remark 1. *It can be show that because the score function of the orignal GLM is uncorrelated with* $\mathbf{Z}$*,* **Algorithm 1** *would not lead to efficiency improvement. Hence, unlike the KIA-SIR and KIA-SAVE, KIA-GLM improves the estimation efficiency through augmenting the loss function directly. Further, for a given λ, the variance reduction is bounded by*

$$\begin{aligned} & |\{E(\mathbf{V}_1) + \lambda E(\mathbf{W})\}^{-1} - \{E(\mathbf{V}_1)\}^{-1} - \{E(\mathbf{V}_1) + \lambda E(\mathbf{W})\}^{-1} \{\lambda E(\mathbf{W})\} \\ & \times \{E(\mathbf{V}_1) + \lambda E(\mathbf{W})\}^{-1}|. \end{aligned}$$

3.2 Simulation Study

To access the KIA-GLM procedure, we generate $N = 1000$ copies of eight dimensional $\widetilde{\mathbf{Z}}$ from standard multivariate Gaussian distribution, and let $\mathbf{Z} = (1, \widetilde{\mathbf{Z}}^{\mathrm{T}})^{\mathrm{T}}$. Suppose $\boldsymbol{\alpha} = (-1, 0, 1, -1, 0, 1, -1, 0, 1)^{\mathrm{T}}$ with β_0 be the intercept for the constant. In Setting 1, we generate responses through $Y_i = \boldsymbol{\alpha}^{\mathrm{T}}\mathbf{Z}_i + \epsilon_i$, where ϵ_i is an independent identically distributed white noise. In Setting 2, we generate binary Y_i with mean $\exp(\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{Z}_i)/\{1 + \exp(\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{Z}_i)\}$.

Table 4 illustrates the average of $\|\widehat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0\|^2$ over 100 simulations for different sample sizes. It can be seen that the KIA-GLM estimators are closer to the true values on average. Further, Table 5 shows that KIA-GLM procedure has smaller estimation variation compared with the orignal GLM. In general, the improvements are more substantial when the responses are binary in both Table 4 and 5.

Table 4: The comparison between KIA-GLM and original GLM on the average of $\|\widehat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}_0\|^2$ over 100 simulations. In all the simulations, we select $\lambda = 0.5$.

	Normal Response		Binary Response	
n	KIA-GLM	GLM	KIA-GLM	GLM
250	0.020	0.039	0.162	0.406
500	0.014	0.019	0.105	0.195

Table 5: The comparison between KIA-GLM and original GLM on the empirical standard deviation of the nine unknown parameters over 100 simulations. In all the simulations, we select $\lambda = 0.5$.

Methods									
	Normal Responses: $n = 250$								
KIA-GLM	0.060	0.034	0.0523	0.049	0.040	0.050	0.045	0.046	0.037
GLM	0.061	0.063	0.059	0.070	0.071	0.068	0.060	0.067	0.069
	Normal Responses: $n = 500$								
KIA-GLM	0.045	0.042	0.041	0.042	0.042	0.038	0.039	0.037	0.031
GLM	0.046	0.046	0.042	0.043	0.046	0.044	0.039	0.046	0.051
	Binary Responses: $n = 250$								
KIA-GLM	0.180	0.079	0.151	0.146	0.087	0.141	0.145	0.081	0.153
GLM	0.218	0.172	0.215	0.201	0.196	0.209	0.232	0.188	0.234
	Binary Responses: $n = 500$								
KIA-GLM	0.118	0.088	0.118	0.113	0.088	0.111	0.113	0.082	0.129
GLM	0.144	0.123	0.153	0.162	0.132	0.144	0.165	0.130	0.154

3.3 Deep Learning Application

KIA-GLM can be naturally applied to enhance the estimation efficiency in the deep learning process. To illustrate this, we implement the algorithms on the MNIST and SVHN datasets as examples. In the MNIST and SVHN data, we randomly select $n = 600$ and $n = 1000$ subsets as the supervised data into the training sets. We repeat the process and generate 100 copies of such sub-training samples. We select the negative log likelihood of the multinomial distribution as $-l(\boldsymbol{\alpha})$ for these multi-class problems.

After obtaining the random samples, we use a small deep learning model as illustrated in Diagram 1 to fit both data. The batch sizes for the supervised dataset is 200, which split the supervised to M small batches. For a given batch of the supervised sample, we randomly pick 1000 samples 20 times from the unsupervised set and combine each of them with the supervised samples to form the inputs of the deep learning neural network. We select the stochastic gradient descent method with learning rate 0.01 and momentum 0.9 as the optimization algorithm. We iterate 100 epochs until the training losses stablize. The detailed flow is described in **Algorithm 2**. The KIA-GLM is implemented after an average pooling layer, from which the outputs are sufficiently normalized.

Algorithm 2 Constructing training samples:

Inputs: $\mathbf{Z}_i, i = 1, \ldots, N$, $Y_i, i = 1, \ldots, n$, M, Algorithm
for j in 1 to maximal epoch **do**
  for l in 1 to M **do**
    Read in $n = 200$ copies $(\mathbf{Z}_{si}, Y_{si})$ from supervised training data, denote the $n \times p$ covariate matrix as $\mathbf{Z}_s$
    for k in 1 to 20 **do**
      Read in $m = 1000$ copies of $\mathbf{Z}_{ui}$ from the unsupervised training data, denote the $m \times p$ covariate matrix as $\mathbf{Z}_u$
        if Algorithm is KIA-GLM, $\mathbf{Z} = (\mathbf{Z}_s^{\mathrm{T}}, \mathbf{Z}_u^{\mathrm{T}})^{\mathrm{T}}$
        else $\mathbf{Z} = \mathbf{Z}_s$
      Process $\mathbf{Z}, Y_{si}, i = 1, \ldots, n$ to the network described in Diagram 1
    end for
  end for
end for

Table 6 illustrates the average $-l(\widehat{\boldsymbol{\alpha}})$, top 1 accuracies and their empirical standard derivations based on 100 random sampled training data. For fair comparisons, we evalate KIA-GLM and GLM on the same random subsample at each iteration. Further, we plot the average top 1 accuracies across all the samples in Figure 5. It can be seen that the KIA-GLM has consistent improvement upon orignal GLM across majority of the random samples.

Diagram 1: Network Structure: CONV standards for the convolution operation. RELU standards for rectifier activation function. The values in the parenthese are the kernel sizes.

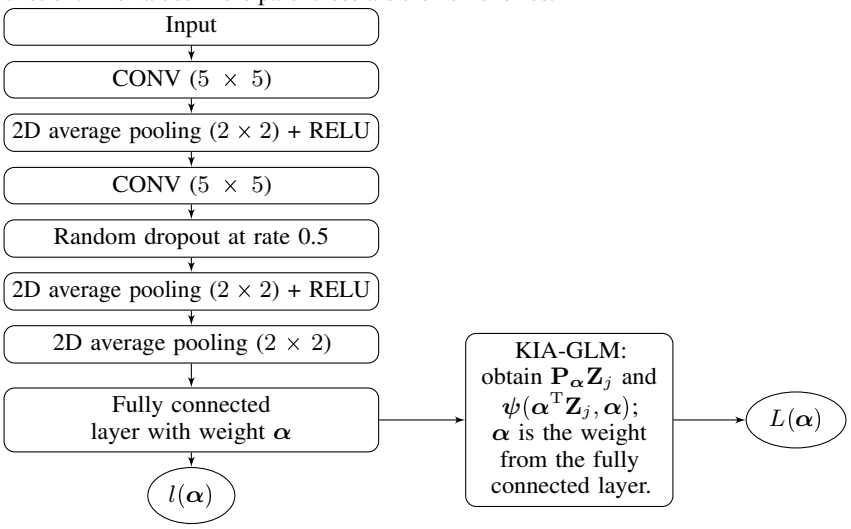
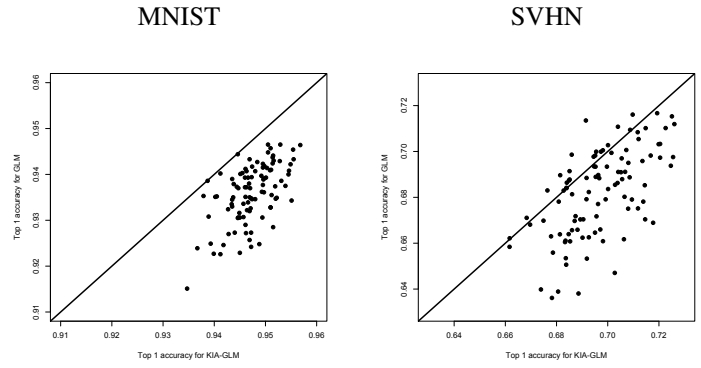


Table 6: The empirical $-l(\boldsymbol{\alpha})$ and top 1 accuracy over 100 random samples on the MNIST and SVHN datasets by using the KIA-GLM and GLM estimators.

	MNIST		SVHN	
Methods	Empirical $-l(\boldsymbol{\alpha})$	Top 1 accuracy	Empirical $-l(\boldsymbol{\alpha})$	Top 1 accuracy
KIA-GLM	0.17 (0.02)	94.73 (0.004)	1.05 (0.05)	69.67 (0.01)
GLM	0.33 (0.05)	93.59 (0.006)	1.29 (0.08)	68.19 (0.02)

Figure 5: The top 1 accuracies for evaluating the MNIST and the SVHN data (GLM v.s. KIA-GLM). Here $\lambda = 0.5$.



4 Conclusion

We propose a kernel augmentation method to make use of the linearity and constant variance properties of the covariates. We discuss the merits of the KIA algorithms in the sufficient dimension reduction and generalized linear model frameworks. We show theoretically and numerically the efficiency improvements of the KIA estimators over the standard methods. The KIA methods are generally applicable to the classification and regression problems, and can be seamlessly integrated with the deep learning algorithms to improve the prediction accuracy.

References

[1] R. Dennis Cook and Sanford Weisberg. Sliced inverse regression for dimension reduction: Comment. *Journal of the American Statistical Association*, 86(414):328–332, 1991.

[2] Kenji Fukumizu, Francis R Bach, and Michael I Jordan. Kernel dimension reduction in regression. *The Annals of Statistics*, 37(4):1871–1905, 2009.

[3] Ker-Chau Li. Sliced inverse regression for dimension reduction. *Journal of the American Statistical Association*, 86(414):316–327, 1991.

[4] Yanyuan Ma and Liping Zhu. A semiparametric approach to dimension reduction. *Journal of the American Statistical Association*, 107(497):168–179, 2012.

[5] Peter McCullagh. Generalized linear models. *European Journal of Operational Research*, 16(3):285–292, 1984.

[6] Elizbar A Nadaraya. On estimating regression. *Theory of Probability & Its Applications*, 9(1):141–142, 1964.

[7] Maxim Rabinovich and David Blei. The inverse regression topic model. In *International Conference on Machine Learning*, pages 199–207, 2014.

[8] Gábor J Székely, Maria L Rizzo, and Nail K Bakirov. Measuring and testing dependence by correlation of distances. *The annals of statistics*, pages 2769–2794, 2007.

[9] Matt Taddy. Multinomial inverse regression for text analysis. *Journal of the American Statistical Association*, 108(503):755–770, 2013.

[10] Meihong Wang, Fei Sha, and Michael I Jordan. Unsupervised kernel dimension reduction. In *Advances in Neural Information Processing Systems*, pages 2379–2387, 2010.

[11] Geoffrey S Watson. Smooth regression analysis. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 359–372, 1964.

[12] Yingcun Xia, Howell Tong, WK Li, and Li-Xing Zhu. An adaptive estimation of dimension reduction space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(3):363–410, 2002.