Original Research Article

SMMR
STATISTICAL METHODS IN MEDICAL RESEARCH

Statistical Methods in Medical Research
1–21

DOI: 10.1177/09622802221084596
journals.sagepub.com/home/smm
SAGE

Integrative nearest neighbor classifier for block-missing multi-modality data

Guan Yu[1] **and Surui Hou**[1]

Abstract

In modern biomedical classification applications, data are often collected from multiple modalities, ranging from various omics technologies to brain scans. As different modalities provide complementary information, classifiers using multi-modality data usually have good classification performance. However, in many studies, due to the high cost of measures, in a lot of samples, some modalities are missing and therefore all data from those modalities are missing completely. In this case, the training data set is a block-missing multi-modality data set. In this paper, considering such classification problems, we develop a new weighted nearest neighbors classifier, called the integrative nearest neighbor (INN) classifier. INN harnesses all available information in the training data set and the feature vector of the test data point effectively to predict the class label of the test data point without deleting or imputing any missing data. Given a test data point, INN determines the weights on the training samples adaptively by minimizing the worst-case upper bound on the estimation error of the regression function over a convex class of functions. Our simulation study shows that INN outperforms common weighted nearest neighbors classifiers that only use complete training samples or modalities that are available in each sample. It performs better than methods that impute the missing data as well, even for the case where some modalities are missing not at random. The effectiveness of INN has been also demonstrated by our theoretical studies and a real application from the Alzheimer's disease neuroimaging initiative.

Keywords

Block-missing, classification, multi-modality data, nearest neighbor, prediction

1 Introduction

Binary classification is a fundamental problem in statistics. Given a training data set, the goal of binary classification is to build a classifier that can predict the class label $y \in \{0, 1\}$ of a new data point using its feature vector X. It has many important applications in biomedical research such as cancer diagnosis and medical image classification. With the advance of science and technology, features in modern biomedical classification applications are often collected from multiple modalities (sources or types), ranging from microarray gene expression to single nucleotide polymorphism (SNP) chip array, microRNA expression, DNA methylation, and biomedical images. Since different modalities could provide complementary information, integrative classifiers using multi-modality data usually deliver better classification performance than methods that only use features from a single modality. The advantages of integrative classifiers have been demonstrated empirically and theoretically. For example, as shown in studies about the diagnosis of Alzheimer's disease,[1,2] integrative classifiers using features from structural magnetic resonance imaging (MRI) for brain atrophy measurement, functional imaging (e.g. positron emission tomography (PET)) for hypometabolism quantification, and cerebrospinal fluid (CSF) for quantification of specific proteins, deliver much better classification performance than methods that only use features from one of those modalities. In a recent theoretical study, Li and Li[3] demonstrated that an integrative linear discriminant

[1]Department of Biostatistics, The State University of New York at Buffalo, NY, USA

Corresponding author:
Guan Yu, Department of Biostatistics, The State University of New York at Buffalo, NY, USA.
Email: guanyu@buffalo.edu

analysis using features from multiple modalities is guaranteed to asymptotically reduce classification error compared with running linear discriminant analysis on features from each modality individually.

Although the integrative analysis of multi-modality data provides an effective way of pooling complementary information, one special challenge for using multi-modality data is to handle missing data, which is prevalent due to some reasons such as the high cost of measures, patient non-compliance and drop-outs.[3–7] In many cases, all features from a certain modality are missing completely, that is, a complete block of the data is missing. For example, in some genome-wide association studies where we often combine data from multiple studies, subjects in certain studies may have both gene expression and DNA methylation measurements while subjects in other studies may only have gene expression measurements.[5] In the Alzheimer's disease neuroimaging initiative (ADNI) study (http://adni.loni.ucla.edu/), half of the subjects have both structural and functional imaging scans while others only have structural imaging scans. In such situations, data are missing by blocks. They are called block-missing multi-modality data in this paper.

One example of block-missing multi-modality data is shown in the left panel of Figure 1. In this example, features are collected from three modalities. For $i = 1, 2, 3$, we have p_i features from the ith modality. The blank regions with a question mark indicate missing data and the colored regions represent the observed data. For some training data points, features collected from Modality 2 and/or Modality 3 are missing completely. The training data points are divided into four groups according to four missing patterns. The number of training data points with complete features, denoted by n_1, can be much smaller than the total number of the training data points $n = n_1 + n_2 + n_3 + n_4$. As we consider the supervised classification problem in this paper, we assume that the class label of each training data point is observed. The n-dimensional class label vector, which is available, is not shown in Figure 1.

In order to use a block-missing multi-modality training data set to build a classifier that can predict the class label of a test data point with complete features, a common strategy is to use the complete training samples only and classification methods such as k-NN, logistic regression, and linear discriminant analysis. This strategy can waste a lot of useful information, especially when we have a significant amount of incomplete training samples. Moreover, it ignores the sampling bias when some modalities are missing not at random. Another strategy is to impute the missing features first and then build classifiers using the imputed training data set. We can impute the missing features with the mean values of the available observations, the weighted mean values,[8] or the conditional expectations based on the expectation–maximization algorithm.[9] Some other imputation methods include the matrix completion method incorporating the soft-thresholded singular value decomposition[10,11] and methods utilizing the random forest algorithms.[12–14] These imputation methods are generally effective when the positions of the missing data are random, but they can be unstable when data are missing in blocks. In the past several years, motivated by applications in genomic data integration, Cai et al.[5] proposed a framework of structured matrix completion to impute block-missing multi-modality data. Their proposed method can be used in the case where features are collected from two modalities. Xue and Qu[7] proposed a multiple block-wise imputation (MBI) approach. The effectiveness of MBI has been demonstrated in the linear regression setting. It is not clear whether a two-step method, which uses MBI to impute the block-missing multi-modality data first and then build a classifier using the imputed data, is effective for classification problems.

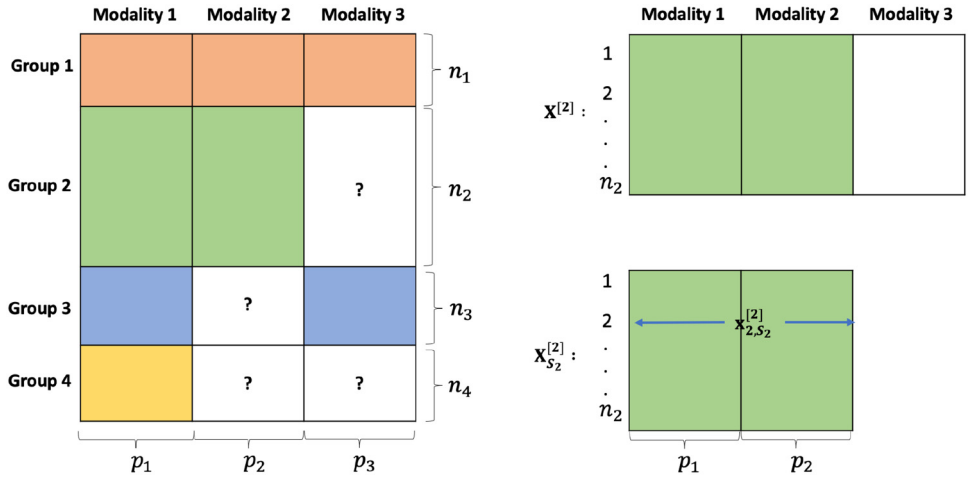

Figure 1. An example of block-missing multi-modality data with three modalities.

Besides the above methods, there are a few integrative classification methods that utilize all available information in the block-missing multi-modality data without deleting or imputing missing data. For example, Yuan et al.[2] proposed the incomplete multi-source feature learning (IMSF) method. IMSF formulates the prediction problem as a multi-task learning problem by first decomposing the prediction problem into a set of classification tasks (one for each missing pattern), and then building linear classifiers for all tasks simultaneously. Xiang et al.[15] introduced an incomplete source-feature selection (ISFS) method which performs simultaneous feature-level and modality-level analysis. ISFS can be considered as a constrained version of IMSF. Recently, Li and Li[3] proposed an integrative linear discriminant analysis method. To the best of our knowledge, most existing integrative classification methods focus on parametric models for block-missing multi-modality data. They assume either certain parametric distribution of the data or certain forms of decision boundaries. These parametric models are of limited use in applications where little knowledge of the data generation process is available a priori. For multi-modality data, as different modalities and the class label are often associated in different ways, it is difficult to assume a reasonable parametric model. In comparison, non-parametric models are usually more flexible in accommodating different data structures. There is a pressing need for the development of flexible non-parametric classification methods for block-missing multi-modality data.

For classification problems with a complete training data set, there is an extensive literature on non-parametric methods (see, e.g. the book[16] and the references therein). With the flexibility and computational efficiency, k-nearest neighbor (k-NN) is one of the most popular non-parametric classifiers. Stone[17] and Devroye et al.[18] have shown that k-NN is universally consistent if we let k grow with the sample size n and k/n converge to zero. Some recent studies on weighted nearest neighbors classifiers[19–25] indicate that the classification performance of k-NN can be further improved by using decreasing weights on the successively more distant neighbors and/or choosing the number of nearest neighbors adaptively according to the density of the feature vector. Among those weighted nearest neighbors classifiers, the k^*-nearest neighbor (k^*-NN) classifier proposed by Anava and Levy[21] is perhaps the most flexible one. It adaptively determines both the number of nearest neighbors k and the weights on those nearest neighbors. For each test data point $\mathbf{x}_0$, the weight vector in the k^*-NN is calculated by minimizing an upper bound of the estimation error of the regression function $\eta(\mathbf{x}) = \mathbb{P}(y = 1 \mid X = \mathbf{x})$ at $\mathbf{x}_0$. The effectiveness of k^*-NN has been demonstrated empirically on many benchmark data sets, showing superior performance over the standard k-NN classifier.

In this paper, motivated by the idea of k^*-NN and the minimax affine estimator introduced by Dohono[26] for the estimation of the regression function, we develop a new weighted nearest neighbor classifier, called the integrative nearest neighbor (INN) classifier, for binary classification problems with a block-missing multi-modality training data set. INN does not delete or impute any missing data. It is a plug-in classifier that uses all available information to estimate the regression function $\eta(\mathbf{x})$. For a test data point $\mathbf{x}_0$, the proposed INN estimate of $\eta(\mathbf{x}_0)$ is a weighted average of the class labels (0 or 1) of training data points, where the weights on the class labels are determined by minimizing the worst-case upper bound on the estimation error of $\eta(\mathbf{x}_0)$ over a convex class of functions. The corresponding optimization problem is a convex minimization problem which can be solved efficiently by an iteration algorithm. Our proposed INN classifier harnesses all available information in the block-missing multi-modality training data and the feature vector of the test data point effectively to estimate the regression function and predict the class label. It can deliver better classification performance than methods that only use complete training samples or modalities that are available in each sample. In addition, for different test data points, INN uses adaptive number of nearest neighbors and weights. It can outperform many existing weighted nearest neighbor classifiers that use the same weight vector for all test data points. The effectiveness of INN has been demonstrated by our theoretical studies, simulated examples, and a real application from the Alzheimer's Disease Neuroimaging Initiative. The comparison between INN and methods that impute the missing data also indicates the advantages of INN.

The rest of this paper is organized as follows. In Section 2, we introduce the statistical setting and our proposed INN classifier. In Section 3, we show some theoretical properties of INN. In Sections 4 and 5, we demonstrate the effectiveness of INN using simulated examples and the ADNI data set, respectively. We conclude this paper in Section 6 and provide all technical proofs in the Appendix. Some additional simulation studies are shown in the Supplemental Material. The notations in this paper are defined as follows. We use bold upper case letters to denote matrices. For a $n \times p$ matrix $\mathbf{X}$, we use $\mathbf{X}^T$ to denote its transpose, and $\mathbf{x}_i^T$ and X_j to denote its i-th row and jth column, respectively. For a set $\mathcal{S} = \{j_1, j_2, , j_s\}$, we use $|\mathcal{S}|$ to denote the number of elements in the set $\mathcal{S}$ and $\mathcal{S}^c$ to denote the complement of $\mathcal{S}$. For a vector $\mathbf{x}_i = (x_{i1}, x_{i2}, , x_{ip})^T$, we use $\mathbf{x}_{i,\mathcal{S}}$ to denote the sub-vector $(x_{ij_1}, x_{ij_2}, , x_{ij_s})^T$. We write $\mathbf{x}_i \geq 0$ if $x_{ij} \geq 0$ for each j. In addition, we denote the ℓ_2-norm of a vector $\mathbf{x}$ as $\|\mathbf{x}\|_2$. For a random sample $\theta_1, \theta_2, , \theta_n$, let $\theta_{(1)}, \theta_{(2)}, , \theta_{(n)}$ denote a permutation of $\theta_1, \theta_2, , \theta_n$ such that $\theta_{(1)} \leq \theta_{(2)} \leq \ldots \leq \theta_{(n)}$. We denote $\mathcal{I}_{\{\cdot\}}$ as the indicator function, and $\mathbf{1}$ as the vector of 1's.

2 Statistical setting and methodology

2.1 Statistical setting

Consider the binary classification problem with the class label $y \in \{0, 1\}$ and the p-dimensional feature vector X collected from K modalities. Assume that there are p_k features from the kth modality. The feature vector X is assumed to be sampled from a multivariate distribution $\mathbb{P}_X$ with support Ω, and its label y is 1 with probability $\eta(X)$ and 0 with probability $1 - \eta(X)$. Furthermore, we assume that each modality can be missing. The mechanism of missingness can be missing completely at random (MCAR), missing at random (MAR), or missing not at random (MNAR).[27] If the kth modality is missing, all p_k features from that modality are missing completely.

Our goal is to build a classifier that can predict the class label $y \in \{0, 1\}$ when all p features collected from K modalities are observed. If the regression function $\eta(X)$ is known, we can use the Bayes classifier

$$\Phi^*(X) = \mathcal{I}_{\{\eta(X) \geq 1/2\}},$$

which is the theoretically optimal classifier that minimizes the misclassification error rate. In practice, $\eta(X)$ is often unknown, but we have access to n training samples $\{(\mathbf{x}_i, y_i) : i = 1, 2, , n\}$ generated from the above model. As each modality of each training sample can be missing, we use γ_{ik} to denote the missingness indicator for the kth modality of the ith training sample, where $\gamma_{ik} = 1$ when the modality is observed, and 0 otherwise. Let $\boldsymbol{\gamma}_i = (\gamma_{i1}, \gamma_{i2}, , \gamma_{iK})^T \in \mathbb{R}^K$ and $\mathbf{\Gamma} = (\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, , \boldsymbol{\gamma}_n)^T \in \mathbb{R}^{n \times K}$. The feature matrix of the training data, denoted by $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, , \mathbf{x}_n)^T \in \mathbb{R}^{n \times p}$, is a block-missing multi-modality data set as shown in Figure 1. Since we consider the supervised classification problem here, we assume the class labels of all training samples are available. Denote the class label vector of the training data as $\mathbf{y} = (y_1, y_2, , y_n)^T$.

As there are K modalities in total and each modality can be missing, there are $2^K - 1$ missing patterns with at least one observed modality theoretically. For $m = 1, 2, , 2^K - 1$, we use $\mathcal{S}_m$ to denote the index set of the observed features in the mth missing pattern. Without loss of generality, we assume that only the first M missing patterns appear in the training data set. In practice, M can be much smaller than $2^K - 1$. According to the missing pattern of the feature vector, we can divide the n training samples into M groups. For each $m \in \{1, 2, , M\}$, let $s_m = |\mathcal{S}_m|$, and n_m denote the number of training samples in the mth group. For $1 \leq i \leq n_m$, let $\mathbf{x}_i^{[m]} \in \mathbb{R}^p$, $\mathbf{x}_{i,\mathcal{S}_m}^{[m]} \in \mathbb{R}^{|\mathcal{S}_m|}$, and $y_i^{[m]}$ denote the vector of all features, the vector of the observed features, and the class label of the ith training sample in the mth group, respectively. In addition, let $\mathbf{y}^{[m]} = (y_1^{[m]}, y_2^{[m]}, , y_{n_m}^{[m]})^T$, $\mathbf{X}^{[m]} = (\mathbf{x}_1^{[m]}, \mathbf{x}_2^{[m]}, , \mathbf{x}_{n_m}^{[m]})^T$, and $\mathbf{X}_{\mathcal{S}_m}^{[m]} = (\mathbf{x}_{1,\mathcal{S}_m}^{[m]}, \mathbf{x}_{2,\mathcal{S}_m}^{[m]}, , \mathbf{x}_{n_m,\mathcal{S}_m}^{[m]})^T$. We can use the example shown in Figure 1 to understand these notations. The feature matrix $\mathbf{X}$ shown in the left panel of Figure 1 are partitioned into four groups according to four different missing patterns ($M = 4$). We have $\mathcal{S}_1 = \{1, 2, ..., p_1 + p_2 + p_3\}$, $\mathcal{S}_2 = \{1, 2, ..., p_1 + p_2\}$, $\mathcal{S}_3 = \{1, 2, ..., p_1, p_1 + p_2 + 1, ..., p_1 + p_2 + p_3\}$, and $\mathcal{S}_4 = \{1, 2, ..., p_1\}$. As shown in the right panel of Figure 1, the $n_2 \times (p_1 + p_2 + p_3)$ matrix $\mathbf{X}^{[2]}$ denotes the matrix of all features of the training samples in Group 2. The $n_2 \times (p_1 + p_2)$ matrix $\mathbf{X}_{\mathcal{S}_2}^{[2]}$ denotes the matrix of the observed features of the training samples in Group 2, and $\mathbf{x}_{2,\mathcal{S}_2}^{[2]}$ denotes the vector of the observed features of the second training sample in Group 2.

2.2 Method

Given a block-missing multi-modality training data set $\{(\mathbf{X}_{\mathcal{S}_m}^{[m]}, \mathbf{y}^{[m]})\}_{m=1}^M$ and a test data point $\mathbf{x}_0 \in \mathbb{R}^p$, we will develop a plug-in classifier, $\Phi_n(\mathbf{x}_0) = \mathcal{I}_{\{\hat{\eta}(\mathbf{x}_0) \geq 1/2\}}$, which mimics the Bayes classifier by using an estimator of the regression function $\eta(\mathbf{x}_0) = \mathbb{P}(y = 1 \mid X = \mathbf{x}_0)$. Note that if the features in the test data point $\mathbf{x}_0$ are collected from K' modalities only where $K' < K$, we can still use our following proposed method and the block-missing multi-modality training data collected from those K' modalities to build an applicable classifier. Therefore, without loss of generality, we assume that all p features in the test data point $\mathbf{x}_0$ are available.

We consider an estimator of $\eta(\mathbf{x}_0)$ as a weighted average of the training labels with the following format

$$\hat{\eta}(\mathbf{x}_0) = \sum_{m=1}^{M} \sum_{i=1}^{n_m} \omega_i^{[m]} y_i^{[m]}, \tag{1}$$

where $\sum_{m=1}^{M} \sum_{i=1}^{n_m} \omega_i^{[m]} = 1$ and $\omega_i^{[m]} \geqslant 0$ for each $1 \leq m \leq M$ and $1 \leq i \leq n_m$. To obtain a good weight vector $\boldsymbol{\omega} = (\omega_1^{[1]}, \omega_2^{[1]}, ..., \omega_{n_1}^{[1]}, \omega_1^{[2]}, \omega_2^{[2]}, ..., \omega_{n_M}^{[M]})^T$, we consider minimizing the estimation error $|\hat{\eta}(\mathbf{x}_0) - \eta(\mathbf{x}_0)|$, that is, solving the following minimization problem

$$\min_{\boldsymbol{\omega}} |\hat{\eta}(\mathbf{x}_0) - \eta(\mathbf{x}_0)| \text{ s.t. } \boldsymbol{\omega} \geq 0 \text{and} \sum_{\mathrm{m}=1}^{\mathrm{M}} \sum_{\mathrm{i}=1}^{\mathrm{n_m}} \omega_{\mathrm{i}}^{[\mathrm{m}]} = 1. \tag{2}$$

However, due to the unknown $\eta(\mathbf{x}_0)$, we cannot solve the above optimization problem directly. Motivated by the idea of k^*-NN and the minimax affine estimator introduced by Donoho[26] for the estimation of the regression function, we assume that the regression function $\eta(\mathbf{x}_0)$ is known to lie in a convex function class. Then, for each $\mathbf{x}_0$, we propose to learn the weight vector $\boldsymbol{\omega}$ by minimizing the worst-case upper bound of $|\hat{\eta}(\mathbf{x}_0) - \eta(\mathbf{x}_0)|$ over the convex function class.

To that end, for each $1 \le m \le M$, we define $\eta_m(\mathbf{x}) = \mathbb{P}(y = 1 \mid X_{\mathcal{S}_m} = \mathbf{x}_{\mathcal{S}_m})$. We can show that $\eta_m(\mathbf{x}) = \mathbb{E}(\eta(X) \mid X_{\mathcal{S}_m} = \mathbf{x}_{\mathcal{S}_m})$. For any $\tau = (\tau_1, \tau_2, , \tau_M) \ge 0$ and $L > 0$, let $\mathcal{F}(\tau, L)$ denote the set of functions $f : \Omega \to \mathbb{R}$ which satisfies $|\mathbb{E}(f(X)|X_{\mathcal{S}_m} = \mathbf{x}_{\mathcal{S}_m}) - f(\mathbf{x}_0)| \le \tau_m + L\|\mathbf{x}_{\mathcal{S}_m} - \mathbf{x}_{0,\mathcal{S}_m}\|_2$ for each $\mathbf{x}, \mathbf{x}_0 \in \Omega$ and $m \in \{1, 2, , M\}$. We assume that $\eta(\mathbf{x}) \in \mathcal{F}(\tau^*, L^*)$, where $\tau^* = (\tau_1^*, \tau_2^*, , \tau_M^*) \ge 0$ and $L^* > 0$. We can check that $\mathcal{F}(\tau^*, L^*)$ is a convex function class. In addition, we can show that the assumption $\eta(\mathbf{x}) \in \mathcal{F}(\tau^*, L^*)$ is satisfied when the regression function $\eta(\mathbf{x})$ is Lipschitz continuous and there is no missing data (that is, $M = 1$ and $\mathcal{S}_1 = \{1, 2, ..., p\}$). Therefore, the condition $\eta(\mathbf{x}) \in \mathcal{F}(\tau^*, L^*)$ can be considered as a generalized Lipschitz condition for the block-missing multi-modality data.

Next, we derive the worst-case upper bound of $|\hat{\eta}(\mathbf{x}_0) - \eta(\mathbf{x}_0)|$ over the convex function class $\mathcal{F}(\tau^*, L^*)$. To obtain that, we firstly decompose (2) as follows

$$\begin{aligned}
\left|\hat{\eta}(\mathbf{x}_0) - \eta(\mathbf{x}_0)\right| &= \left|\sum_{m=1}^{M}\sum_{i=1}^{n_m} \omega_i^{[m]} y_i^{[m]} - \eta(\mathbf{x}_0)\right| \\
&\leqslant \underbrace{\left|\sum_{m=1}^{M}\sum_{i=1}^{n_m} \omega_i^{[m]}\left(y_i^{[m]} - \eta_m(\mathbf{x}_i)\right)\right|}_{:=P_1} \\
&\quad + \underbrace{\left|\sum_{m=1}^{M}\sum_{i=1}^{n_m} \omega_i^{[m]}\left(\eta_m(\mathbf{x}_i) - \eta(\mathbf{x}_0)\right)\right|}_{:=P_2}.
\end{aligned}$$

We consider the case where the weight vector $\boldsymbol{\omega}$ only depends on the observed features $\{\mathbf{X}_{\mathcal{S}_m}^{[m]}\}_{m=1}^M$ of the training data and the test data point $\mathbf{x}_0$. In addition, we assume that $y_1, y_2, , y_n$ are conditionally independent given $\{\mathbf{X}_{\mathcal{S}_m}^{[m]}\}_{m=1}^M$ and the missing data indicator matrix $\boldsymbol{\Gamma}$. As $|y_i^{[m]} - \eta_m(\mathbf{x}_i)| \le 1$, by the Hoeffding's inequality, we can show that for any $t_0 > 0$,

$$\mathbb{P}\left(P_1 \ge t_0\|\boldsymbol{\omega}\|_2 \mid \mathbf{x}_0, \{\mathbf{X}_{\mathcal{S}_m}^{[m]}\}_{m=1}^M, \boldsymbol{\Gamma}\right) \le 2\exp(-t_0^2/2).$$

In addition, according to the assumption that $\eta(\mathbf{x}) \in \mathcal{F}(\tau^*, L^*)$, we have

$$P_2 \le \sum_{m=1}^{M}\sum_{i=1}^{n_m} \omega_i^{[m]}\left(\tau_m^* + L^*\|\mathbf{x}_{i,\mathcal{S}_m}^{[m]} - \mathbf{x}_{0,\mathcal{S}_m}\|_2\right).$$

Hence, with probability at least $1 - 2\exp(-t_0^2/2)$, the worst-case estimator error $\sup_{\eta \in \mathcal{F}(\tau^*, L^*)} |\hat{\eta}(\mathbf{x}_0) - \eta(\mathbf{x}_0)|$ can be bounded by

$$t_0\|\boldsymbol{\omega}\|_2 + \sum_{m=1}^{M}\sum_{i=1}^{n_m} \omega_i^{[m]}\left(\tau_m^* + L^*\|\mathbf{x}_{i,\mathcal{S}_m}^{[m]} - \mathbf{x}_{0,\mathcal{S}_m}\|_2\right). \tag{3}$$

Therefore, we propose to learn the weight vector $\boldsymbol{\omega}$ by minimizing the above worst-case upper bound, equivalently, solving the following minimization problem

$$\begin{aligned}
&\min_{\boldsymbol{\omega}} \left\{\|\boldsymbol{\omega}\| + \sum_{m=1}^{M}\sum_{i=1}^{n_m} \omega_i^{[m]} \cdot \left(\tilde{L}\|\mathbf{x}_{i,\mathcal{S}_m}^{[m]} - \mathbf{x}_{0,\mathcal{S}_m}\|_2 + \tilde{\tau}_m\right)\right\} \\
&\text{s.t. } \boldsymbol{\omega} \ge 0 \text{ and } \sum_{m=1}^{M}\sum_{i=1}^{n_m} \omega_i^{[\mathrm{m}]} = 1,
\end{aligned} \tag{4}$$

where $\tilde{L} = L^*/t_0$ and $\tilde{\tau}_m = \tau_m^*/t_0$. In practice, as we do not know $\tilde{L}$ or $\tilde{\tau}_m$'s, we can consider them as tuning parameters. An independent validation data set or cross-validation techniques can be used to choose the values of these parameters. We only need to tune t_0 if we know τ^* and L^*.

The minimization problem (4) is a convex optimization problem. We adopt the algorithm shown in Anava and Levy[21] to obtain the exact solution of (4). Let $\hat{\omega} = (\hat{\omega}_1^{[1]}, ..., \hat{\omega}_{n_1}^{[1]}, ..., \hat{\omega}_1^{[M]}, ..., \hat{\omega}_{n_M}^{[M]})^T$ denote the solution of (4). Our proposed INN classifier is

$$\Phi_n(\mathbf{x}_0) = \begin{cases} 1 & \text{if } \hat{\eta}(\mathbf{x}_0) = \sum_{m=1}^{M} \sum_{i=1}^{n_m} \hat{\omega}_i^{[m]} y_i^{[m]} \geq \frac{1}{2}, \\ 0 & \text{otherwise.} \end{cases}$$

The details of INN classifier are shown as follows.

Since different modalities and the class label are often associated in different ways, it is difficult to assume a reasonable parametric model for the multi-modality data. As a non-parametric classifier, INN does not assume any specific structure of the regression function or shape of the decision boundary. It is appropriate for classification problems with features collected from multiple modalities. In addition, as shown in step 4, since both $\alpha_i^{[m]}$ and λ_k depend on $\mathbf{x}_0$, INN uses adaptive numbers of nearest neighbors as well as weights for different test data points. It is more flexible than some existing weighted nearest neighbor classifiers such as k-NN, optimal weighted nearest neighbor classifier,[19] and the sliced k-NN[20] that use either a fixed number of nearest neighbors or a fixed weight vector. The proposed INN classifier includes the k^*-NN classifier[21] for complete data sets as a special case. INN incorporates all available information in the block-missing multi-modality training data and the feature vector of the test data point effectively to estimate the regression function and predict the class label. It does not delete or impute any missing data. Next, we will demonstrate the effectiveness of INN by both theoretical and numerical studies.

3 Theoretical study

In this section, we establish the theoretical guarantee of INN. We assume that the underlying regression function $\eta(\mathbf{x}) \in \mathcal{F}(\tau^*, L^*)$, where $\tau^* = (\tau_1^*, \tau_2^*, , \tau_M^*) \geq 0$ and $L^* > 0$. In order to evaluate whether INN can utilize incomplete samples, we assume the missing data indicator matrix $\mathbf{\Gamma}$ is given and therefore the number of samples with different missing patterns, denoted by $n_1, n_2, , n_M$, are known.

We will first study how the parameter t_0 determines the number of nearest neighbors used in the INN classifier. Then, we show that the proposed INN classifier agrees with the optimal Bayes classifier with high probability for each test data point $\mathbf{x}_0 \in \Omega_\epsilon = \{\mathbf{x} : |\eta(\mathbf{x}) - 1/2| \geq \epsilon\}$, where $\epsilon \in (0, 1/2)$. For $m = 1, 2, ..., M$ and $i = 1, 2, ..., n_m$, define

$$\beta_i^{[m]} = L^* \|\mathbf{x}_{i,\mathcal{S}_m}^{[m]} - \mathbf{x}_{0,\mathcal{S}_m}\|_2 + \tau_m^*.$$

Let $\beta_{(1)}, \beta_{(2)}, , \beta_{(n)}$ be a permutation of $\beta_i^{[m]}$'s such that $\beta_{(1)} \leq \beta_{(2)} \leq ... \leq \beta_{(n)}$. For each $\mathbf{x}_0 \in \Omega_\epsilon$ and $m = 1, 2, ..., M$, define $F_{\mathbf{x}_0}^{[m]}(t) = \mathbb{P}(\|\mathbf{x}_{i,\mathcal{S}_m} - \mathbf{x}_{0,\mathcal{S}_m}\| \leqslant t \mid \mathbf{x}_0, \mathbf{\Gamma})$. In addition, define $N(\mathbf{x}_0) = \sum_{m=1}^{M} n_m F_{\mathbf{x}_0}^{[m]}((\epsilon - \tau_m^*)/L^*)$.

Our following Proposition 1 shows how the parameter t_0 determines the number of nearest neighbors $k_{\mathbf{x}_0}$ used in the INN classifier for the test data point $\mathbf{x}_0$.

Proposition 1 Let $k_{\mathbf{x}_0}$ denote the number of positive $\hat{\omega}_i^{[m]}$'s for the test data point $\mathbf{x}_0$. We have

(a) $k_{\mathbf{x}_0} = 1$ if and only if $t_0 \leq \beta_{(2)} - \beta_{(1)}$.
(b) $k_{\mathbf{x}_0} = k + 1$ if and only if

$$\sqrt{\sum_{i=1}^{k} (\beta_{(k+1)} - \beta_{(i)})^2} < t_0 \leq \sqrt{\sum_{i=1}^{k+1} (\beta_{(k+2)} - \beta_{(i)})^2},$$

where $1 \leq k \leq n - 2$.
(c) $k_{\mathbf{x}_0} = n$ if and only if $t_0 > \sqrt{\sum_{i=1}^{n-1} (\beta_{(n)} - \beta_{(i)})^2}$.

Proposition 1 indicates that the number of nearest neighbors used in INN for the test data point $\mathbf{x}_0$ increases from 1 to n as t_0 increases from 0 to the infinity. The proposed INN classifier is equivalent to the 1 nearest neighbor classifier when $t_0 \leq \beta_{(2)} - \beta_{(1)}$. When t_0 is sufficiently large, INN is similar to the n nearest neighbor classifier where the weight of each training data point is about $1/n$. In our proposed INN classifier, we assume that t_0 does not depend on $\mathbf{x}_0$. As all $\beta_i^{[m]}$'s depend on the test data point $\mathbf{x}_0$, the numbers of nearest neighbors used in INN are still different for different test data points although we use a common t_0.

Similar to many existing nearest neighbor based classifiers,[19,20,23] the property of INN depends on the bound on the distance between $\mathbf{x}_0$ and its nearest neighbors. For the block-missing multi-modality data considered in this paper, $\beta_{(k)}$ can be viewed as the distance between the test data point $\mathbf{x}_0$ and its kth nearest neighbor in the block-missing multi-modality training data set. Our following Proposition 2 provides a high probability bound on $\beta_{(k)}$.

Proposition 2 Suppose that $\beta_i^{[m]}$'s are conditionally independent given the test data point $\mathbf{x}_0$ and the missing data indicator matrix $\mathbf{\Gamma}$. For any $\epsilon > 0$, $\mathbf{x}_0 \in \Omega_\epsilon$ and positive integer $k \leq N(\mathbf{x}_0)/2$, we have

$$\mathbb{P}(\beta_{(k)} > \epsilon \mid \mathbf{x}_0, \mathbf{\Gamma}) \leq \exp(-3k/14).$$

Next, we establish the theoretical guarantee of INN by showing its point-wise agreement with the Bayes classifier.

Theorem 1 Consider a fixed test data point $\mathbf{x}_0 \in \Omega_\epsilon = \{\mathbf{x} : |\eta(\mathbf{x}) - 1/2| \geq \epsilon\}$, where $\epsilon \in (0, 1/2)$. Let $\delta \in (0, 1)$ be a probability tolerance and choose $t_0 = \sqrt{2\log(4/\delta)}$, $\tilde{L} = L^*/t_0$, $\tilde{\tau} = \tau^*/t_0$ in INN. Suppose the following assumptions hold:

A1. The regression function $\eta(\mathbf{x}) \in \mathcal{F}(\tau^*, L^*)$, where $\tau^* = (\tau_1^*, \tau_2^*, , \tau_M^*) \geq 0$ and $L^* > 0$.

A2. The class labels $y_1, y_2, , y_n$ are conditionally independent given $\{\mathbf{X}_{S_m}^{[m]}\}_{m=1}^M$ and the missing data indicator matrix $\mathbf{\Gamma}$. The distances $\beta_i^{[m]}$'s are conditionally independent given the test data point $\mathbf{x}_0$ and the missing data indicator matrix $\mathbf{\Gamma}$.

A3. For any positive integer $k \geq 5\log(4/\delta)$, we have

$$\mathbb{E}\left(\exp\left(-\sum_{i=1}^{k-1}\frac{(\beta_{(k)} - \beta_{(i)})^2}{2}\right) \mid \mathbf{x}_0, \mathbf{\Gamma}\right) \leq \frac{\delta^2}{16}.$$

If the number of training data points with different missing patterns satisfy

$$N(\mathbf{x}_0) = \sum_{m=1}^{M} n_m F_{\mathbf{x}_0}^{[m]}\left(\frac{\epsilon - \tau_m^*}{L^*}\right) \geq 10\log\left(\frac{4}{\delta}\right),$$

then we have

$$\mathbb{P}(\Phi_n(\mathbf{x}_0) = \Phi^*(\mathbf{x}_0) \mid \mathbf{x}_0, \mathbf{\Gamma}) \geq 1 - \delta.$$

Note that Assumption A1 can be considered as a generalized Lipschitz condition on the regression function for the block-missing multi-modality data. When this assumption holds, we can estimate $\eta(\mathbf{x}_0)$ from those of the neighbors of $\mathbf{x}_0$ where we can use $\beta_i^{[m]}$'s to measure the distances between $\mathbf{x}_0$ and all training data points in the block-missing multi-modality data set. Similar smoothness conditions[28,20,23] on the regression function have been widely used for non-parametric regression and classification problems with a complete training data set.

Assumption A2 is a condition on the missing data mechanism. It is a weak condition which can hold when modalities are missing completely at random, missing at random, or missing not at random. For example, consider a binary classification problem where features are collected from two modalities and there is only one feature from each modality. Suppose these two features are independent, each follows the uniform distribution $U[0, 1]$, and $\mathbb{P}(y_i = 1 \mid \mathbf{x} = \mathbf{x}_i) = (x_{i1} + x_{i2})/2$. In addition, assume that the first feature is always observed and the second feature is MCAR, that is, $\mathbb{P}(\gamma_{i2} = 1 \mid y_i, \mathbf{x}_i) = \rho$, where $\rho \in (0, 1)$ is a constant. Suppose that there are n training samples generated independently by the above model. Without loss of generality, assume that the first n_1 samples are complete. Then, we have $M = 2$, $\mathcal{S}_1 = \{1, 2\}$, $\mathcal{S}_2 = \{1\}$, $\eta_1(\mathbf{x}_i) = (x_{i1} + x_{i2})/2$, and $\eta_2(\mathbf{x}_i) = \mathbb{E}((x_{i1} + x_{i2})/2 \mid x_{i1}) = (2x_{i1} + 1)/4$. We can check that both Assumptions $A1$ and $A2$ hold. These two assumptions also hold when the second feature is MAR (e.g. $\mathbb{P}(\gamma_{i2} = 1 \mid y_i, \mathbf{x}_i) = \mathcal{I}_{\{x_{i1} < 0.5\}}$) or MNAR (e.g. $\mathbb{P}(\gamma_{i2} = 1 \mid y_i, \mathbf{x}_i) = \mathcal{I}_{\{x_{i2} < 0.5\}}$).

Assumption A3 is a technical condition that guarantees that INN will not use too many nearest neighbors for $\mathbf{x}_0$ with high probability. When A3 holds, we can check that with probability at least $1 - \delta/4$, the number of nearest neighbors used in INN for the test data point $\mathbf{x}_0$ is less than $5\log(4/\delta)$.

Theorem 1 indicates that under some assumptions, our proposed INN classifier agrees with the optimal Bayes classifier with high probability as long as $N(\mathbf{x}_0)$ is large enough. For the classification problem with a complete training data set,

similar theoretical results for some non-parametric classifiers such as k-NN and the fixed-radius nearest neighbor classifier have been shown in Chen and Shah.[29] As far as we know, there is no existing similar result for non-parametric classifiers learned from a block-missing multi-modality training data set.

The quantity $N(\mathbf{x}_0)$ in Theorem 1 can be viewed as the number of samples in the block-missing multi-modality training data set that are effectively utilized by INN. The contribution of the data with the mth missing pattern is $n_m F_{\mathbf{x}_0}^{[m]}((\epsilon - \tau_m^*)/L^*)$, which depends on the sample size n_m, the difference between $\eta_m(X)$ and $\eta(X)$, and the distribution of the feature vector. If we only use complete samples (suppose they are in Group 1) to build the INN classifier, only $n_1 F_{\mathbf{x}_0}^{[1]}((\epsilon - \tau_1^*)/L^*)$ samples are effectively utilized, which can be much smaller than $N(\mathbf{x}_0)$ in many cases, especially when the percentage of complete samples is low. Therefore, Theorem 1 not only provides the theoretical guarantee of INN but also shows that INN can effectively incorporate incomplete samples in the block-missing multi-modality training data set.

In Theorem 1, we can also choose $\tilde{L} = L/\sqrt{2\log(4/\delta)}$ and $\tilde{\tau} = \tau/\sqrt{2\log(4/\delta)}$ where $L \geq L^*$ and $\tau \geq \tau^*$. However, we need to guarantee that

$$\tilde{N}(\mathbf{x}_0) = \sum_{m=1}^{M} n_m F_{\mathbf{x}_0}^{[m]}\left(\frac{\epsilon - \tau_m}{L}\right) \geq 10\log\left(\frac{4}{\delta}\right).$$

If L is much greater than L^* or τ_m is much greater than τ_m^* for each m, then $\tilde{N}(\mathbf{x}_0)$ will be a relatively small number and thus we can only consider a relatively large probability tolerance δ such that the above condition is satisfied. Therefore, in this case, we can only guarantee that the proposed INN classifier agrees with the Bayes classifier with a relatively small probability. Our simulation results shown in Section 4 in the Supplemental Material also indicate that INN can perform well when we use relatively large values for $\tilde{L}$ and $\tilde{\tau}$. However, INN can lose performance if we choose too large values for $\tilde{L}$ and $\tilde{\tau}$.

4 Simulation study

In this section, we evaluate the effectiveness of INN using simulated examples. We study nine examples, including seven examples with features collected from two modalities and two examples with features collected from three modalities. In each example, to guarantee that at least one modality is available, we assume the first modality is always observed while the other modalities can be missing. Two missing mechanisms, MCAR and MNAR, are considered in each example.

We compare the empirical misclassification error rate (multiplied by 100) of INN with those of (1) kNN-C: the standard k-NN classifier only using the complete data; (2) k^*NN-C: the k^*-NN classifier only using the complete data; (3) kNN-1: the standard k-NN classifier only using the data from Modality 1; (4) k^*NN-1: the k^*-NN classifier only using the data from Modality 1; (5) kNN-Mean: the standard k-NN classifier using the imputed data set in which the missing features are imputed by the mean imputation method; (6) k^*NN-Mean: the k^*-NN classifier using the imputed data set in which the missing features are imputed by the mean imputation method; (7) kNN-RF: the standard k-NN classifier using the imputed data set in which the missing features are imputed by the random forest algorithm in the randomForest R package[30]; (8) k^*NN-RF: the k^*-NN classifier using the imputed data set in which the missing features are imputed by the random forest algorithm.

We first consider seven examples with features collected from two modalities. In each example, we set $p_1 = p_2 = 20$. Features from Modality 1 are always observed while features from Modality 2 can be missing. Therefore, we have $M = 2$, $\mathcal{S}_1 = \{1, 2, ..., 40\}$, and $\mathcal{S}_2 = \{1, 2, ..., 20\}$. To generate a block-missing multi-modality training data set, we obtained a complete data set first and then used a MCAR or MNAR model to generate missing data. In each example, when we used the MCAR model, we set $\mathbb{P}(\gamma_{i2} = 1) = \rho$. When we used the MNAR model, we set $\mathbb{P}(\gamma_{i2} = 1 \mid \mathbf{x}_i, y_i) = 1 - \Phi(a \cdot (\mathbf{x}_{i,\mathcal{A}_2} - \mathbf{c})^T(\mathbf{x}_{i,\mathcal{A}_2} - \mathbf{c}) - b)$, where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution and the set $\mathcal{A}_2 = \{p_1 + 1, p_1 + 2, ..., p_1 + p_2\}$. The parameters a, b, and the vector $\mathbf{c}$ were chosen such that the percentage of complete samples in the MNAR case was around ρ for comparison purpose. The details of these seven examples are shown as follows.

Example 1: All features of samples in Class 1 and Class 0 are generated independently from $N(-0.3, 1.5^2)$ and $N(0.3, 1.5^2)$, respectively. For the MCAR case, we set $\rho = 0.4$. For the MNAR case, we set $a = -1$, $b = -50$, and $\mathbf{c} = \mathbf{0}$ such that the percentage of complete samples is about 40%.

Example 2: Feature vectors of samples in Class 1 and Class 0 are generated from *Multivariate Normal* $(\boldsymbol{\mu_1}, \sigma^2\boldsymbol{\Sigma})$ and *Multivariate Normal* $(\boldsymbol{\mu_2}, \sigma^2\boldsymbol{\Sigma})$, respectively, where $\boldsymbol{\mu_1} = (-0.3, \ldots, -0.3)^T$, $\boldsymbol{\mu_2} = (0.3, \ldots, 0.3)^T$, and $\sigma = 2$.

The matrix $\mathbf{\Sigma} = \boldsymbol{A} \otimes \boldsymbol{B}$, where $\boldsymbol{A}$ is a 2×2 matrix with off-diagonal elements as 0.2 and the diagonal elements as 1, and $\boldsymbol{B}$ is a matrix with the off-diagonal elements as 0.5 and the diagonal elements as 1. For the MCAR case, we set $\rho = 0.4$. For the MNAR case, we set $a = 0.01$, $b = 1.3$, and $\mathbf{c} = \mathbf{0}$.

Example 3: This example is almost the same as Example 2. We only change the matrix $\boldsymbol{B}$ to be a matrix where the element in the ith row and jth column is $0.5^{|i-j|}$.

Example 4: All features of samples in Class 1 are generated independently from $N(-0.3, 1.5^2)$. All features of samples in Class 0 are generated independently from the t-distribution with 6 degrees of freedom and the noncentral parameter as 0.4. For the MCAR case, we set $\rho = 0.4$. For the MNAR case, we set $a = 0.01$, $b = 0.2$, and $\mathbf{c} = (0.161, 0.161, ..., 0.161)^T$.

Example 5: Feature vectors of samples in Class 1 and Class 0 are generated from *Multivariate Normal* $(\boldsymbol{\mu_1}, \sigma^2\mathbf{\Sigma}_1)$ and *Multivariate Normal* $(\boldsymbol{\mu_2}, \sigma^2\mathbf{\Sigma}_2)$, respectively, where $\boldsymbol{\mu_1} = (-0.3, \ldots, -0.3)^T$, $\boldsymbol{\mu_2} = (0.3, \ldots, 0.3)^T$, and $\sigma = 2$. The matrices $\mathbf{\Sigma}_1 = \boldsymbol{A} \otimes \boldsymbol{B}_1$ and $\mathbf{\Sigma}_2 = \boldsymbol{A} \otimes \boldsymbol{B}_2$, where $\boldsymbol{A}$ is a 2×2 matrix with off-diagonal elements as 0.2 and the diagonal elements as 1, $\boldsymbol{B}_1$ is a matrix with the off-diagonal elements as 0.5 and diagonal elements as 1, and $\boldsymbol{B}_2$ is a matrix with the element in the ith row and jth column as $0.5^{|i-j|}$. For the MCAR case, we set $\rho = 0.4$. For the MNAR case, we set $a = 0.01$, $b = 1.3$, and $\mathbf{c} = \mathbf{0}$.

Example 6: Feature vectors of samples in Class 1 and Class 0 are generated from *Multivariate Laplace* $(\boldsymbol{\mu_1}, \sigma^2\mathbf{\Sigma}_1)$ and *Multivariate Laplace* $(\boldsymbol{\mu_2}, \sigma^2\mathbf{\Sigma}_2)$, respectively, where $\boldsymbol{\mu_1} = (-0.3, \ldots, -0.3)^T$, $\boldsymbol{\mu_2} = (0.3, \ldots, 0.3)^T$, and $\sigma = 2$. We set $\mathbf{\Sigma}_1 = \boldsymbol{A} \otimes \boldsymbol{B}_1$, $\mathbf{\Sigma}_2 = \boldsymbol{A} \otimes \boldsymbol{B}_2$, where $\boldsymbol{A}$ is a 2×2 matrix with off-diagonal elements as 0.2 and diagonal elements as 1, $\boldsymbol{B}_1$ is a matrix with the off-diagonal elements as 0.5 and diagonal elements as 1, and $\boldsymbol{B}_2$ is a matrix with the element in the ith row and jth column as $0.5^{|i-j|}$. For the MCAR case, we set $\rho = 0.4$. For the MNAR case, we set $a = 0.01$, $b = 1.3$, and $\mathbf{c} = \mathbf{0}$.

Example 7: All features of samples in Class 1 and Class 0 are generated independently from $N(-0.3, 1.5^2)$ and $N(0.3, 1.5^2)$, respectively. In this example, we consider different scenarios with different percentages of complete samples. For the MCAR case, we consider different values of ρ in the set $\{0.1, 0.3, 0.5, 0.7, 0.9\}$. For the MNAR case, we set $a = 0.01$, $\mathbf{c} = \mathbf{0}$, and different values of b in the set $\{1.6, 1.1, 0.5, 0, -0.8\}$.

In each experiment, we generated a training data set with 100 samples and an independent validation data set with 200 complete samples. The validation data set was used to choose all tuning parameters. As samples in Group 1 were complete, we fixed $\tilde{\tau}_1 = 0$ in our proposed method. We considered 10 values of $\tilde{L}$ in $\{0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 5, 10\}$, and 20 values of $\tilde{\tau}_2$ in $[0.01, 1)$. For all k-NN methods (kNN-C, kNN-1, kNN-Mean, and kNN-RF), we considered 10 values of the number of nearest neighbors in $\{1, 2, \ldots, 10\}$. For all k^*-NN methods (k^*NN-C, k^*NN-1, k^*NN-Mean, and k^*NN-RF), similar to Anava and Levy's tuning method,[21] we considered different values of the tuning parameter in $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 5, 10\}$. After choosing those tuning parameters, we used an independent test data set with 500 complete samples to compute the empirical misclassification error rates (multiplied by 100) of different methods.

Note that Example 1 is a simple setting with independent normally distributed features. Examples 2 and 3 explore the cases with correlated features. Two different correlation structures are considered. Example 4 is used to evaluate whether our proposed INN classifier can perform well when the distributions of some features have heavy tails. Examples 5 and 6 together explore the performance of different methods when the feature vectors of samples in different classes follow different multivariate distributions. For these six examples, we generated the box plots of the empirical misclassification error rates (multiplied by 100) of different methods using our simulation results from 50 experiments. The box plots of Examples 1 to 6 are shown in Figures 2 to 7, respectively.

As shown in Figures 2 to 7, INN outperforms all competitors in both the MCAR and MNAR cases of Examples 1 to 6. Theoretically, our proposed INN classifier is the same as k^*NN-C when we only use complete training samples, and it is the same as k^*NN-1 when we only use the data from Modality 1. The comparison between INN and k^*NN-C, k^*NN-1 in the above six examples indicates that our proposed INN classifier can utilize all available information in the block-missing multi-modality training data set effectively and therefore could deliver a lower missclassification error rate than methods that only use complete training samples or modalities that are available in each sample. As k^*-NN is more flexible than the standard k-NN by using different weight vectors for different test data points, in most cases, k^*NN-C, k^*NN-1, k^*NN-Mean, and k^*NN-RF perform better than kNN-C, kNN-1, kNN-Mean, and kNN-RF, respectively. We can also observe that except for Example 4, kNN-Mean and kNN-RF that use an imputed training data set do not perform better than kNN-C which only uses complete samples. However, in many cases, k^*NN-Mean and k^*NN-RF perform better than k^*NN-C. Although it is generally difficult to handle missing not at random data, the above numerical results indicate that INN can make use of the block-missing multi-modality data effectively to build a better classifier in some cases where

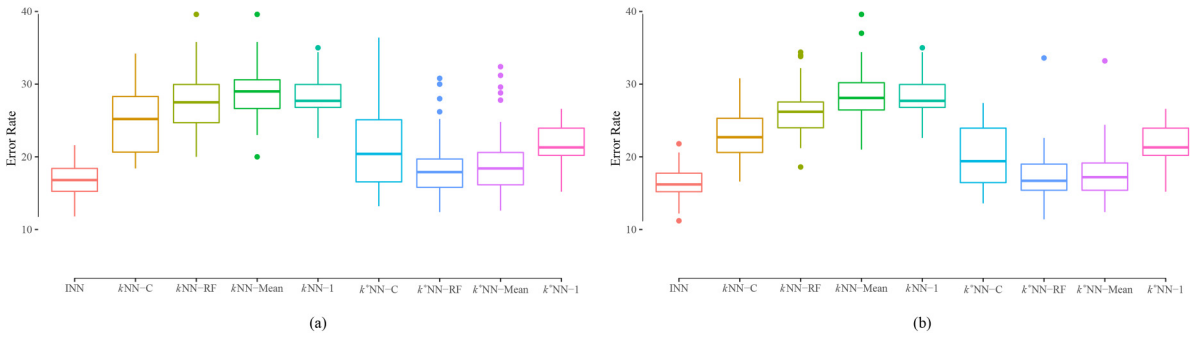

Figure 2. Box plots of the empirical misclassification error rates (multiplied by 100) of four classifiers based on *k*-NN, four classifiers based on *k**-NN, and the proposed INN classifier: (a) MCAR case of Example 1 and (b) MNAR case of Example 1. *k*-NN: *k*-nearest neighbor; INN: integrative nearest neighbor; MCAR: missing completely at random; MNAR: missing not at random.

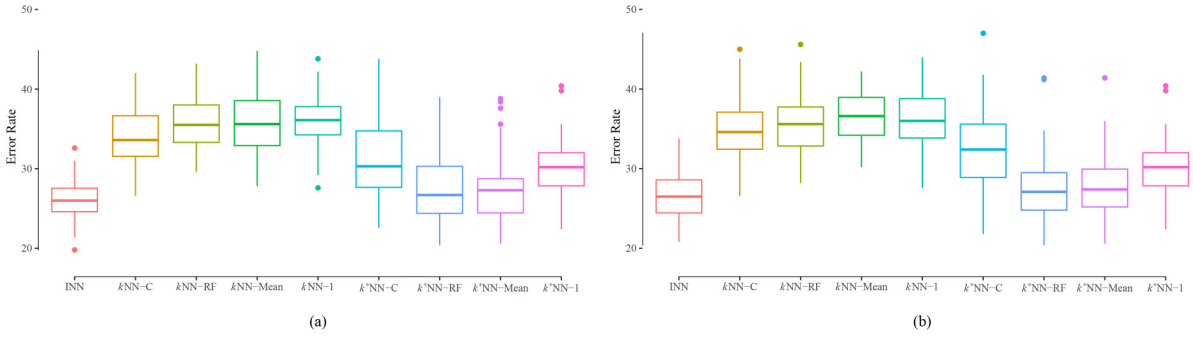

Figure 3. Box plots of the empirical misclassification error rates (multiplied by 100) of four classifiers based on *k*-NN, four classifiers based on *k**-NN, and the proposed INN classifier: (a) MCAR case of Example 2 and (b) MNAR case of Example 2. *k*-NN: *k*-nearest neighbor; INN: integrative nearest neighbor; MCAR: missing completely at random; MNAR: missing not at random.

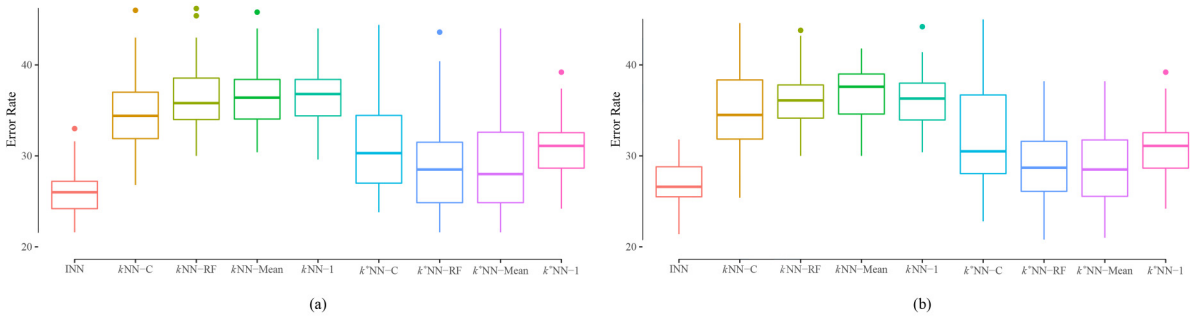

Figure 4. Box plots of the empirical misclassification error rates (multiplied by 100) of four classifiers based on *k*-NN, four classifiers based on *k**-NN, and the proposed INN classifier: (a) MCAR case of Example 3 and (b) MNAR case of Example 3. *k*-NN: *k*-nearest neighbor; INN: integrative nearest neighbor; MCAR: missing completely at random; MNAR: missing not at random.

modalities are missing not at random. The results in the MCAR case and the MNAR case of these simulated examples look similar to some extent. One possible reason is that we control the proportion of missing data carefully such that we have similar amount of missing data in the MCAR case and the MNAR case.

Example 7 is similar to Example 1. We consider different scenarios with different percentages of complete samples. This example is used to investigate the effect of the percentage of missing data. The average empirical misclassification error rates (multiplied by 100) of different classifiers in different scenarios based on 50 experiments are shown in Figure 8. We can observe that except for *k*NN-1 and *k**NN-1 that only use all the data from Modality 1, all other classifiers

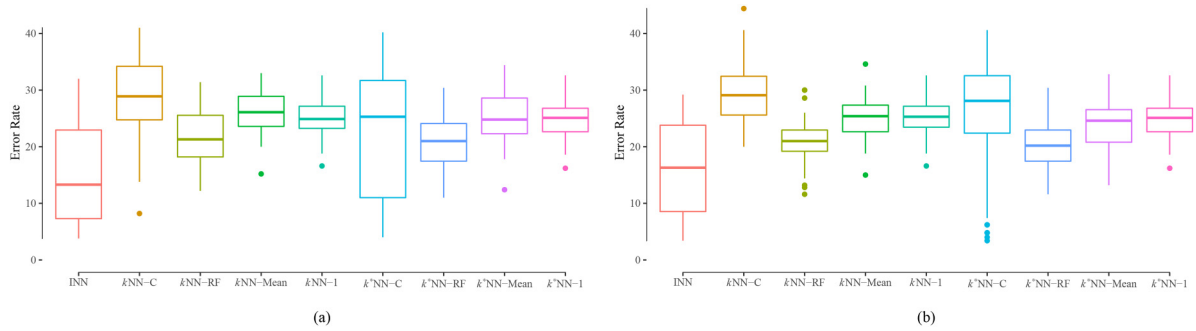

Figure 5. Box plots of the empirical misclassification error rates (multiplied by 100) of four classifiers based on *k*-NN, four classifiers based on *k**-NN, and the proposed INN classifier: (a) MCAR case of Example 4 and (b) MNAR case of Example 4. *k*-NN: *k*-nearest neighbor; INN: integrative nearest neighbor; MCAR: missing completely at random; MNAR: missing not at random.

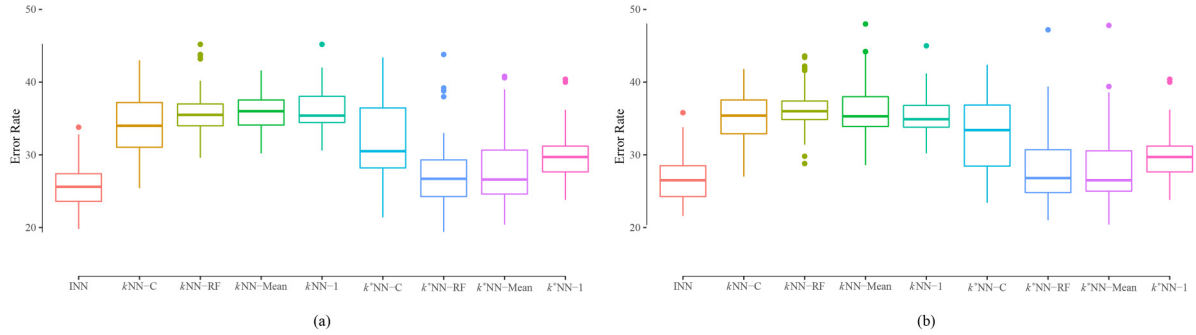

Figure 6. Box plots of the empirical misclassification error rates (multiplied by 100) of four classifiers based on *k*-NN, four classifiers based on *k**-NN, and the proposed INN classifier: (a) MCAR case of Example 5 and (b) MNAR case of Example 5. *k*-NN: *k*-nearest neighbor; INN: integrative nearest neighbor; MCAR: missing completely at random; MNAR: missing not at random.

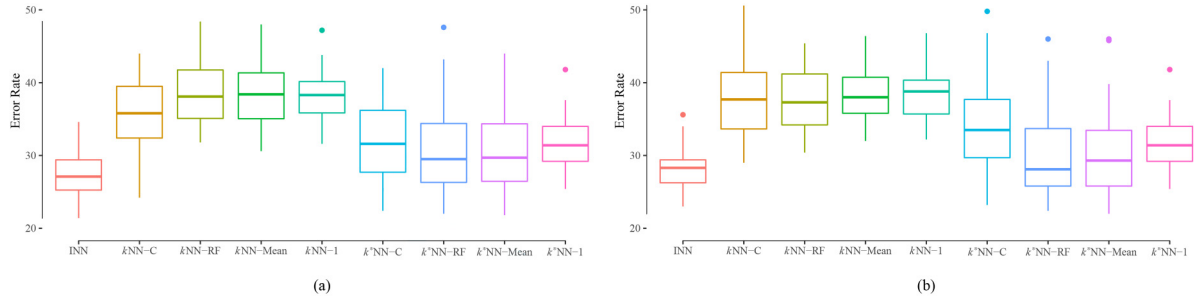

Figure 7. Box plots of the empirical misclassification error rates (multiplied by 100) of four classifiers based on *k*-NN, four classifiers based on *k**-NN, and the proposed INN classifier: (a) MCAR case of Example 6 and (b) MNAR case of Example 6. *k*-NN: *k*-nearest neighbor; INN: integrative nearest neighbor; MCAR: missing completely at random; MNAR: missing not at random.

deliver decreasing misclassification errors as the percentage of complete samples increases. In all scenarios, our proposed INN classifier performs best. Compared with the methods only using complete samples (*k*NN-C and *k**NN-C), INN has greater improvement in the classification performance when we have fewer complete samples. Although they use the same imputed data set, *k**NN-Mean and *k**NN-RF seem to utilize the imputed data more effectively than *k*NN-Mean and *k*NN-RF. One possible reason is that the missing data imputation makes the distribution of the feature vector be more non-uniform (different regions have very different densities), and therefore it is more important to use flexible number of nearest neighbors for test data points in different regions. For all these simulated examples with two modalities, the average computation time of our proposed INN method is about 2s.

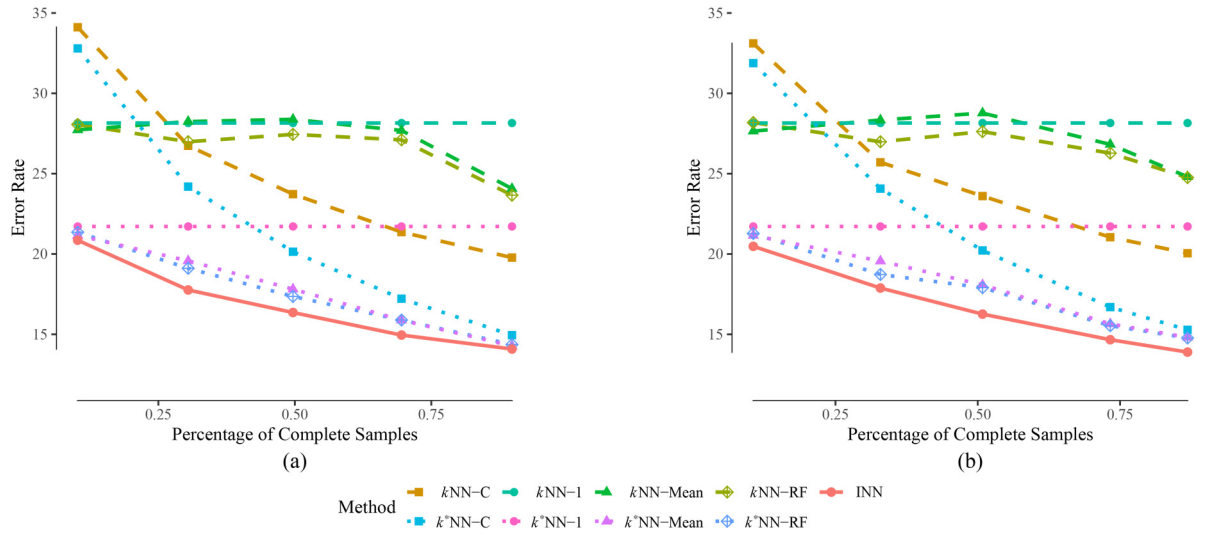

Figure 8. The average empirical misclassification error rates (multiplied by 100) of different classifiers: (a) MCAR case of Example 7 and (b) MNAR case of Example 7. MCAR: missing completely at random; MNAR: missing not at random.

Next, we compare the classification performance of different classifiers in some complicated cases where features are collected from three modalities. As we often have fewer complete samples when data are collected from more modalities, integrative classifiers for block-missing multi-modality data without deleting or imputing missing data are more needed here. In the following Examples 8 and 9, we assume that features from Modality 1 are always observed while features from the other two modalities can be missing. Therefore, we have $M = 4$ in these two examples. We set $p_1 = p_2 = p_3 = 60$. When we used the MCAR model, we set $\mathbb{P}(\gamma_{i2} = 1) = \mathbb{P}(\gamma_{i3} = 1) = \rho$. When we used the MNAR model, we set $\mathbb{P}(\gamma_{i2} = 1 \mid \mathbf{x}_i, y_i) = 1 - \Phi(a \cdot (\mathbf{x}_{i,\mathcal{A}_2} - \mathbf{c})^T(\mathbf{x}_{i,\mathcal{A}_2} - \mathbf{c}) - b)$ and $\mathbb{P}(\gamma_{i3} = 1 \mid \mathbf{x}_i, y_i) = 1 - \Phi(a \cdot (\mathbf{x}_{i,\mathcal{A}_3} - \mathbf{c})^T(\mathbf{x}_{i,\mathcal{A}_3} - \mathbf{c}) - b)$, where the sets $\mathcal{A}_2 = \{p_1 + 1, p_1 + 2, ..., p_1 + p_2\}$ and $\mathcal{A}_3 = \{p_1 + p_2 + 1, p_1 + p_2 + 2, ..., p_1 + p_2 + p_3\}$. The other details about Examples 8 and 9 are shown as follows.

Example 8: All features of samples in Class 0 and Class 1 are generated independently from $N(-0.5, 3^2)$ and $N(0.5, 3^2)$, respectively. For the MCAR case, we set $\rho = 0.4$. For the MNAR case, we set $a = 1$, $b = 490$, and $\mathbf{c} = \mathbf{0}$.
Example 9: Feature vectors of samples in Class 0 and Class 1 are generated from *Multivariate Normal* $(\boldsymbol{\mu_1}, \sigma^2\boldsymbol{\Sigma})$ and *Multivariate Normal* $(\boldsymbol{\mu_2}, \sigma^2\boldsymbol{\Sigma})$, respectively, where $\boldsymbol{\mu_1} = (-0.3, \ldots, -0.3)^T$, $\boldsymbol{\mu_2} = (0.3, \ldots, 0.3)^T$ and $\sigma^2 = 4$. The matrix $\boldsymbol{\Sigma} = \boldsymbol{A} \otimes \boldsymbol{B}$, where $\boldsymbol{A}$ is a 3×3 matrix with the off-diagonal elements as 0.2 and the diagonal elements as 1, and $\boldsymbol{B}$ is a compound symmetric matrix with off-diagonal elements as 0.5. For the MCAR case, we set $\rho = 0.4$. For the MNAR case, we set $a = 0.01$, $b = 6.9$, and $\mathbf{c} = \mathbf{0}$.

In the above two examples, we used 160 training samples. As we set $\rho = 0.4$ and the parameters in the MNAR case accordingly, there were about 15% complete samples in each case. Similar to the previous examples, we used a validation data set with 200 complete samples to choose all tuning parameters, and a test data set with 500 complete samples to calculate the misclassification errors. The results of Examples 8 and 9 are shown in Figures 9 and 10, respectively. Similar to the two modalities cases, INN also outperforms all competitors in both the MCAR and MNAR cases. For these two examples with three modalities, the average computation time of INN is about 17s. Overall, our proposed INN classifier is computationally efficient.

Besides the above simulation studies, we have conducted some numerical studies to investigate the influence of the dimensionality of the data set, the noise intensity and outliers. Those results are shown in Sections 1–3 in the Supplemental Material. We also used the Wilcoxon signed-rank test to compare the misclassification errors of INN and its competitors for all examples except Example 7 which contains multiple cases. The p-values are shown in Section 5 in the Supplemental Material. In most cases, the p-value is less than 0.001. These results indicate that INN performs significantly better than all competitors.

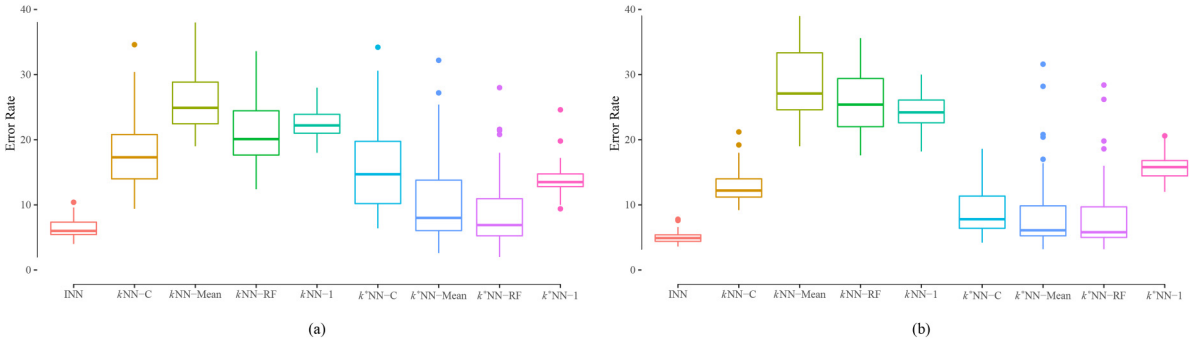

Figure 9. Box plots of the empirical misclassification error rates (multiplied by 100) of four classifiers based on *k*-NN, four classifiers based on *k**-NN, and the proposed INN classifier: (a) MCAR case of Example 8 and (b) MNAR case of Example 8. *k*-NN: *k*-nearest neighbor; INN: integrative nearest neighbor; MCAR: missing completely at random; MNAR: missing not at random.

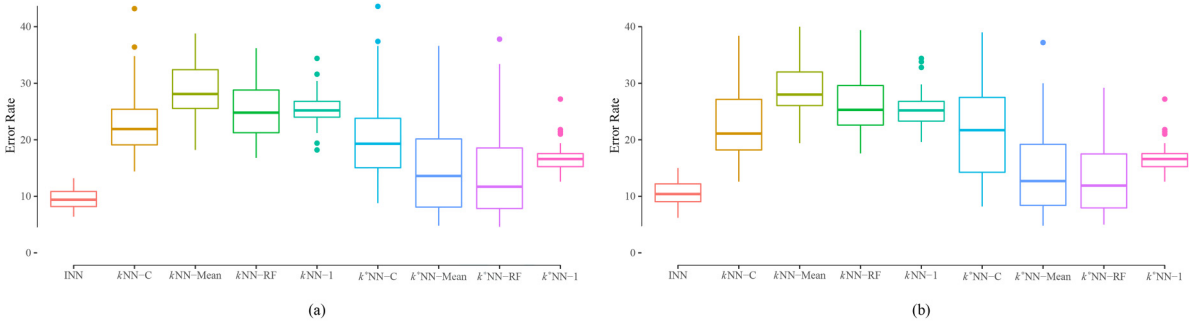

Figure 10. Box plots of the empirical misclassification error rates (multiplied by 100) of four classifiers based on *k*-NN, four classifiers based on *k**-NN, and the proposed INN classifier: (a) MCAR case of Example 9 and (b) MNAR case of Example 9. *k*-NN: *k*-nearest neighbor; INN: integrative nearest neighbor; MCAR: missing completely at random; MNAR: missing not at random.

5 Real data analysis

In this section, we compare the performance of INN and the competitors shown in Section 4 on the diagnosis of Alzheimer's disease (AD).

AD is a neurodegenerative disorder characterized by cognitive decline with loss of memory. As reported by Matthews et al.,[31] there were 5 millions Americans aged over 65 years suffering from AD and related dementias in 2014. By 2050, the burden of AD could rise to 11 millions in the United States and 100 millions worldwide.[32] Symptoms of AD typically begin with Mild Cognitive Impairment (MCI) which is a transitional state between the cognitive decline of normal aging and the more serious decline of AD. However, patients with MCI often do not automatically convert to AD. As shown in Misra et al.,[33] only approximately 10% to 15% patients with MCI will progress to AD while many MCI patients remain stable or even revert to the normal cognition. Therefore, MCI patients can be further divided into two categories: progressive MCI (pMCI) patients who will progress to AD and stable MCI (sMCI) patients who will not.

Despite all scientific efforts, there is no effective treatment for AD currently.[34]It is essential to develop sensitive biomarkers and accurate classifiers for the early detection of AD and MCI so that timely treatments can be used to slow down the progression of the disease. As one of famous AD studies, the Alzheimer's Disease Neuroimaging Initiative (ADNI) (http://adni.loni.usc.edu) is a longitudinal multicenter study designed to develop clinical, imaging, genetic, and biochemical biomarkers for the early detection and tracking of AD. All data generated by the ADNI study investigators are available in the ADNI data repository. As a real data example to demonstrate the effectiveness of our proposed approach, in this study, we focused on the baseline data from ADNI1 which is the first phase of the ADNI study. Features from three important modalities, including the structural MRI, fluorodeoxyglucose PET, and CerebroSpinal Fluid (CSF), were used to build classifiers for the detection of AD, MCI, and pMCI.

As MRI and PET are medical imaging modalities, imaging pre-processing is needed to extract features from MRI and PET images. Similar to the image analysis performed by Zhang and Shen,[35] for MRI images, after some correction, spatial

segmentation, and registration steps, we obtained the subject labeled image based on the Jacob template[36] with 93 manually labeled regions of interest (ROI). For each of the 93 ROIs, we computed the volume of the gray matter as a feature. For each PET image, we first aligned the PET image to its corresponding MRI using affine registration. Then, we calculated the average intensity of every ROI in the PET image as a feature. Therefore, we obtained one MRI feature and one PET feature from each ROI. Note that three features from the CSF modality, including the amyloid β ($A\beta 42$), CSF total tau (t-tau), and tau hyperphosphorylated at threonine 181 (p-tau), were used in our study.

After the above data processing steps, we got 93 features from each MRI image, 93 features from each PET image, and three features from each CSF sample. Although the ADNI team has been actively collecting MRI, PET, and CSF data from each subject, unfortunately, not all subjects in the study have data from all three modalities. While all subjects have baseline MRI data, only about half of them have PET data and another different half have CSF data. As shown in Thung et al.,[37] PET and CSF modalities are missing due to several reasons such as the high cost of the PET scan and the unwillingness of the patients to receive invasive tests for the collection of CSF samples through the lumber puncture. As we will have no PET feature (and/or CSF feature) from a subject if the PET (and/or CSF) modality is missing, the ADNI data set used in our study is a block-missing multi-modality data set. The sample size information about this data set is shown in Table 1. There are 805 subjects in total, including 186 subjects with AD, 226 normal controls (NC), 226 subjects with sMCI, and 167 subjects with pMCI. These 805 subjects can be also divided into four groups according to four missing patterns. There are 202 subjects in group 1 (MRI, PET, CSF) with complete features, 194 subjects in group 2 (MRI, PET) with MRI and PET features only, 204 subjects in group 3 (MRI, CSF) with MRI and CSF features only, and 205 subjects in group 4 (MRI) with MRI features only. Figure 11 shows the scatter plots of the first two principle components of the complete data. We can observe that AD subjects are relatively easier to be distinguished from normal controls comparing with MCI subjects, while pMCI and sMCI subjects are difficult to be distinguished.

In this study, we considered three binary classification problems (AD vs. NC, MCI vs. NC, and pMCI vs. sMCI) that are of great interest. For each problem, we used both our proposed approach and the competitors to build classifiers. In each analysis, the ADNI block-missing multi-modality data set is randomly divided into a training data set, a validation data set, and a test data set 50 times. The training data set consists of 25% randomly selected complete samples along with all incomplete samples. The validation data set consists of another 25% randomly selected complete samples. Similar to the

Table 1. Sample size information about the ADNI data set used in this study.

	(MRI, PET, CSF)	(MRI, PET)	(MRI, CSF)	(MRI)	Total
Number of AD subjects	51	42	51	42	186
Number of normal controls	52	49	60	65	226
Number of sMCI subjects	56	70	50	50	226
Number of pMCI subjects	43	33	43	48	167
Total	202	194	204	205	805

ADNI: Alzheimers Disease Neuroimaging Initiative; MRI: magnetic resonance imaging; PET: positron emission tomography; CSF: cerebrospinal fluid; Alzheimers disease: AD; pMCI: progressive Mild Cognitive Impairment; sMCI: stable Mild Cognitive Impairment.

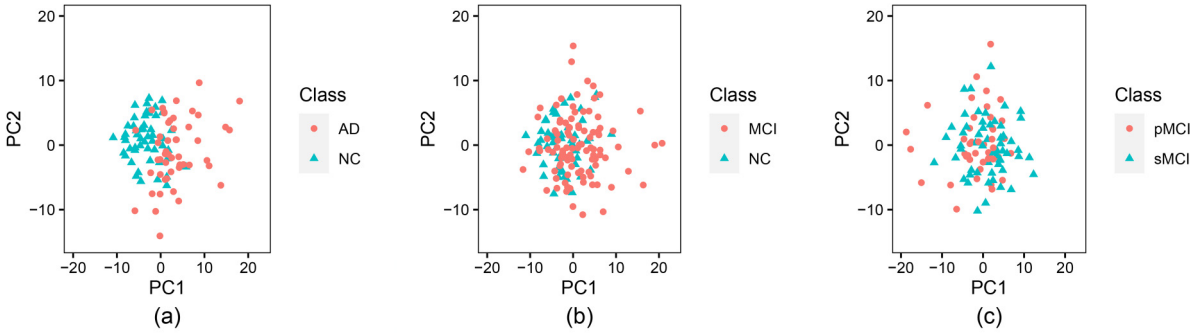

Figure 11. The first two principle components (PC) of the complete data: (a) AD vs. NC; (b) MCI vs. NC; and (c) pMCI vs. sMCI. Alzheimers disease: AD; NC: normal control; MCI: Mild Cognitive Impairment: pMCI: progressive Mild Cognitive Impairment; sMCI: stable Mild Cognitive Impairment.

simulation study, this validation data set was used to choose all the tuning parameters. The test data set consists of the rest 50% complete samples.

The average misclassification errors of different classifiers for the AD versus NC, MCI versus NC, and pMCI versus sMCI classifications are shown in Table 2. The results show that each classifier based on the k^*-NN performs better than the corresponding classifier based on the k-NN for both the AD versus NC and MCI versus NC classifications. However, for the pMCI versus sMCI classification, as the classification task becomes more difficult, classifiers based on the k^*-NN and kNN perform similarly. Compared with the methods only using the complete data or the MRI data, methods based on the missing data imputation have better performance in most cases. However, we can observe that for the MCI versus NC classification, kNN-mean and k^*NN-mean perform worse than kNN-1 and k^*NN-1, resepectively. This indicates that classification methods based on the missing data imputation may not be able to utilize the information in the block-missing multi-modality training data set effectively.

As shown in Table 2, our proposed INN classifier outperforms all competitors from the easiest AD versus NC classification to the most difficult pMCI versus sMCI classification. For the AD versus NC classification and MCI versus NC classification, comparing with the best competitor (k^*NN-RF in this example), INN improves the misclassification error by 13.77% and 4.69%, respectively. For the difficult pMCI versus sMCI classification, while the improvement of INN over methods based on the missing data imputation is not significant, INN performs much better than methods only using the complete data or the MRI data.

Similar to many weighted nearest neighbors classifiers, INN suffers from the curse of dimensionality. It is important to reduce the data dimension first by applying dimension reduction techniques such as the principle component analysis (PCA) if we plan to use INN for a relatively high-dimensional data set. For block-missing multi-modality data such as the ADNI data set considered here, since we cannot calculate the principle components of incomplete samples, we cannot apply PCA on the whole data set directly. Therefore, we attempted to reduce the dimensionality by applying PCA on each modality separately. For each modality, we selected top principle components that cumulatively explained 70% variance and then repeated our analysis. The results are shown in Table 3. Our proposed INN classifier still has the best performance. If we reduce the data dimension by PCA first, kNN-1 and k^*NN-1 that only use MRI data have significantly improved performance on all three classification tasks. The k^*NN-RF method also performs slightly better. However, all other classifiers perform worse on some classification tasks. Our proposed INN has a subtle improvement in the classification of MCI versus NC while performs slightly worse on the other two classification tasks. As shown in this real data analysis example, applying PCA on different modalities separately may not be an effective way to reduce the dimension for block-missing multi-modality data. One possible reason is that we could not apply PCA on the whole data set and thus had to ignore the correlation among different modalities to some extent. This raises an interesting and important future work about the development of effective dimension reduction techniques for high-dimensional block-missing multi-modality data.

Overall, this real data analysis example indicates that our proposed method could incorporate all available information in the block-missing multi-modality training data effectively. It can serve as a promising tool for modern real classification applications where features are often collected from multiple modalities and some modalities can be missing.

Table 2. Misclassification errors of different classifiers for the AD vs. NC, MCI vs. NC, and pMCI vs. sMCI classifications.

Methods	AD vs. NC	MCI vs. NC	pMCI vs. sMCI
kNN-C	23.93 (1.07)	36.65 (0.91)	45.57 (0.78)
k^*NN-C	22.45 (1.22)	33.17 (0.70)	45.77 (0.95)
kNN-1	24.72 (0.92)	33.71 (0.80)	39.96 (0.67)
k^*NN-1	21.06 (0.69)	32.83 (0.69)	42.59 (0.75)
kNN-Mean	18.23 (0.77)	34.65 (0.70)	38.55 (0.90)
k^*NN-Mean	17.13 (0.75)	33.46 (0.58)	38.39 (0.83)
kNN-RF	16.49 (0.77)	31.12 (0.71)	37.96 (0.74)
k^*NN-RF	16.19 (0.84)	31.12 (0.62)	38.08 (0.85)
INN	**13.96** (0.62)	**29.66** (0.67)	**37.33** (0.77)

AD: Alzheimers disease: NC: normal control; MCI: Mild Cognitive Impairment: pMCI: progressive Mild Cognitive Impairment; sMCI: stable Mild Cognitive Impairment; INN: integrative nearest neighbor.
Note: The values in the parentheses are the standard errors.

Table 3. Misclassification errors of different classifiers for the AD vs. NC, MCI vs. NC, and pMCI vs. sMCI classifications with dimension reduction via PCA.

Methods	AD vs. NC	MCI vs. NC	pMCI vs. sMCI
*k*NN-C	23.09 (1.16)	36.78 (0.92)	45.88 (0.98)
*k**NN-C	22.15 (1.15)	33.27 (0.75)	45.02 (1.01)
*k*NN-I	22.64 (0.80)	29.95 (0.73)	38.47 (0.90)
*k**NN-I	19.25 (0.74)	31.69 (0.70)	38.63 (0.90)
*k*NN-Mean	19.25 (0.73)	31.66 (0.65)	39.18 (0.75)
*k**NN-Mean	16.30 (0.54)	31.84 (0.68)	40.20 (0.73)
*k*NN-RF	17.89 (0.68)	31.45 (0.80)	36.43 (0.88)
*k**NN-RF	15.25 (0.62)	30.29 (0.67)	37.88 (0.84)
INN	**14.42** (0.55)	**29.22 (0.61)**	**37.57** (0.70)

AD: Alzheimers disease: NC: normal control; MCI: Mild Cognitive Impairment: pMCI: progressive Mild Cognitive Impairment; sMCI: stable Mild Cognitive Impairment; INN: integrative nearest neighbor.
Note: The values in the parentheses are the standard errors.

6 Concluding remarks

In this paper, we developed a new integrative nearest neighbor classifier for binary classification problems with a block-missing multi-modality training data set without deleting or imputing any missing data. For each test data point, the proposed INN classifier determines the weights on the class labels of complete and incomplete training samples adaptively by minimizing the worst-case upper bound on the estimation error of the regression function at the test data point over a convex class of functions. The corresponding optimization problem is a convex minimization problem which can be solved efficiently by an iteration algorithm. Comparing with methods that only use complete samples or modalities that are available in each sample, our proposed INN classifier has better classification performance by harnessing all available information within complete samples as well as incomplete samples. The advantage of INN has been demonstrated in both the theoretical and numerical studies. The comparison between our method with methods that impute the missing data also indicates the advantage of INN, even for the challenging case where some modalities are missing not at random.

One general shortcoming of classifiers based on nearest neighbors is the so-called "curse of dimensionality." The performance of these classifiers often deteriorates as the dimension of the considered problem increases. As INN depends on the distances between the test data point and its nearest neighbors, INN also suffers from the "curse of dimensionality." When there are a lot of features collected from some modalities (e.g. various omics-data), it is important to apply feature selection or dimension reduction techniques[38–40] to reduce the dimension first. The development of effective feature selection and dimension reduction techniques for high dimensional block-missing multi-modality data is an important future work.

Declaration of conflicting interests

The authors declared no potential conflicts of interest with respect to the research, authorship and/or publication of this article.

Funding

The authors received no financial support for the research, authorship and/or publication of this article.

ORCID iD

Guan Yu https://orcid.org/0000-0002-7833-9755

Supplemental material

Supplemental material for this article is available online.

References

1. Zhang D, Wang Y, Zhou L, et al. Multimodal classification of Alzheimer's disease and mild cognitive impairment. *Neuroimage* 2011; **55**: 856–867.
2. Yuan L, Wang Y, Thompson PM, et al. Multi-source feature learning for joint analysis of incomplete multiple heterogeneous neuroimaging data. *NeuroImage* 2012; **61**: 622–632.
3. Li Q and Li L. Integrative linear discriminant analysis with guaranteed error rate improvement. *Biometrika* 2018; **105**: 917–930.

4. Armijo-Olivo S, Warren S and Magee D. Intention to treat analysis, compliance, drop-outs and how to deal with missing data in clinical research: A review. *Phys Ther Rev* 2009; **14**: 36–49.
5. Cai T, Cai TT and Zhang A. Structured matrix completion with applications to genomic data integration. *J Am Stat Assoc* 2016; **111**: 621–633.
6. Yu G, Li Q, Shen D, et al. Optimal sparse linear prediction for block-missing multi-modality data without imputation. *J Am Stat Assoc* 2020; **115**: 1406–1419.
7. Xue F and Qu A. Integrating multisource block-wise missing data in model selection. *J Am Stat Assoc* 2020; **0**: 1–14.
8. Hastie T, Tibshirani R, Sherlock G, et al. *Imputing missing data for gene expression arrays*. Technical report, Standford: Stanford University, 1999.
9. Schneider T. Analysis of incomplete climate data: Estimation of mean values and covariance matrices and imputation of missing values. *J Clim* 2001; **14**: 853–871.
10. Hastie T, Mazumder R, Lee JD, et al. Matrix completion and low-rank svd via fast alternating least squares. *J Mach Learn Res* 2015; **16**: 3367–3402.
11. Husson F, Josse J, Narasimhan B, et al. Imputation of mixed data with multilevel singular value decomposition. *J Comput Graph Stat* 2019; **28**: 552–566.
12. Stekhoven DJ and Bühlmann P. Missforest–non-parametric missing value imputation for mixed-type data. *Bioinformatics* 2012; **28**: 112–118.
13. Tang F and Ishwaran H. Random forest missing data algorithms. *Stat Anal Data Mining: ASA Data Sci J* 2017; **10**: 363–377.
14. Ramosaj B and Pauly M. Predicting missing values: A comparative study on non-parametric approaches for imputation. *Comput Stat* 2019; **34**: 1741–1764.
15. Xiang S, Yuan L, Fan W, et al. Bi-level multi-source learning for heterogeneous block-wise missing data. *NeuroImage* 2014; **102**: 192–206.
16. Devroye L, Györfi L and Lugosi G. *A probabilistic theory of pattern recognition*, 31. New York: Springer Science & Business Media, 2013.
17. Stone CJ. Consistent nonparametric regression. *Ann Stat* 1977; **5**: 595–620.
18. Devroye L, Gyorfi L, Krzyzak A, et al. On the strong universal consistency of nearest neighbor regression function estimates. *The Annals of Statistics* 1994; **22**: 1371–1385.
19. Samworth RJ. Optimal weighted nearest neighbour classifiers. *The Annals of Statistics* 2012; **40**: 2733–2763.
20. Gadat S, Klein T and Marteau C. Classification in general finite dimensional spaces with the k-nearest neighbor rule. *The Annals of Statistics* 2016; **44**: 982–1009.
21. Anava O and Levy K. k*-nearest neighbors: From global to local. In *Advances in neural information processing systems*. pp. 4916–4924.
22. Xing Y, Song Q and Cheng G. Benefit of interpolation in nearest neighbor algorithms. *arXiv preprint arXiv:190911720* 2019;.
23. Zhao P and Lai L. Minimax rate optimal adaptive nearest neighbor classification and regression. *IEEE Trans Inf Theory* 2021; **67**: 3155–3182.
24. Balsubramani A, Dasgupta S, Freund Y, et al. An adaptive nearest neighbor rule for classification. In *Advances in Neural Information Processing Systems*. pp. 7577–7586.
25. Cannings TI, Berrett TB and Samworth RJ. Local nearest neighbour classification with applications to semi-supervised learning. *The Annals of Statistics* 2020; **48**: 1789–1814.
26. Donoho DL. Statistical estimation and optimal recovery. *The Annals of Statistics* 1994; **0**: 238–270.
27. Rubin DB. Inference and missing data. *Biometrika* 1976; **63**: 581–592.
28. Audibert JY and Tsybakov AB. Fast learning rates for plug-in classifiers. *The Annals of statistics* 2007; **35**: 608–633.
29. Chen GH and Shah D. *Explaining the success of nearest neighbor methods in prediction*. Boston: Now Publishers,, 2018.
30. Liaw A and Wiener M. Classification and regression by randomforest. *R News* 2002; **2**: 18–22. https://CRAN.R-project.org/doc/Rnews/.
31. Matthews KA, Xu W, Gaglioti AH, et al. Racial and ethnic estimates of Alzheimer's disease and related dementias in the United States (2015–2060) in adults aged≥ 65 years. *Alzheimers Dement* 2019; **15**: 17–24.
32. Mucke L. Alzheimer's disease. *Nature* 2009; **461**: 895–897.
33. Misra C, Fan Y and Davatzikos C. Baseline and longitudinal patterns of brain atrophy in MCI patients, and their use in prediction of short-term conversion to ad: results from ADNI. *NeuroImage* 2009; **44**: 1415–1422.
34. Beckman D, Chakrabarty P, Ott S, et al. A novel tau-based rhesus monkey model of Alzheimer's pathogenesis. *Alzheimers Dement* 2021; **17**: 933–945.
35. Zhang D and Shen D. Multi-modal multi-task learning for joint prediction of multiple regression and classification variables in Alzheimer's disease. *NeuroImage* 2012; **59**: 895–907.
36. Kabani N, MacDonald D, Holmes C, et al. A 3d atlas of the human brain. *NeuroImage* 1998; **7**: S717.
37. Thung KH, Wee CY, Yap PT, et al. Neurodegenerative disease diagnosis using incomplete multi-modality data via matrix shrinkage and completion. *NeuroImage* 2014; **91**: 386–400.
38. Fan J and Fan Y. High dimensional classification using features annealed independence rules. *Ann Stat* 2008; **36**: 2605–2637.
39. Fan J and Song R. Sure independence screening in generalized linear models with np-dimensionality. *The Annals of Statistics* 2010; **38**: 3567–3604.
40. Mai Q and Zou H. The kolmogorov filter for variable screening in high-dimensional binary classification. *Biometrika* 2012; **100**: 229–234.

5 Appendix

In this section, we show all the proofs.

Proof of Proposition 1: According to Theorem 3.3 in Anava and Levy,[21] Step 3 in INN halts after exactly $k_{\mathbf{x}_0}$ iterations. Therefore, $k_{\mathbf{x}_0} = 1$ if and only if $\lambda_1 \geq \alpha_{(2)}$. As $\lambda_1 = \alpha_{(1)} + 1$, we know that $k_{\mathbf{x}_0} = 1$ if and only if $t_0 \geq \beta_{(2)} - \beta_{(1)}$. Similarly, $k_{\mathbf{x}_0} = k + 1$ if and only if $\lambda_k > \alpha_{(k+1)}$ and $\lambda_{k+1} \leq \alpha_{(k+2)}$. As

$$\lambda_k = \frac{1}{k}\left(\sum_{i=1}^{k}\alpha_{(i)} + \sqrt{k + (\sum_{i=1}^{k}\alpha_{(i)})^2 - k\sum_{i=1}^{k}\alpha_{(i)}^2}\right),$$

we can show that

$$\begin{aligned}
&\lambda_k \leq \alpha_{(k+1)} \\
&\Leftrightarrow k^2\alpha_{(k+1)}^2 - 2k\alpha_{(k+1)}\sum_{i=1}^{k}\alpha_{(i)} \geq k - k\sum_{i=1}^{k}\alpha_{(i)}^2 \\
&\Leftrightarrow 1 \leq \sum_{i=1}^{k}(\alpha_{(k+1)} - \alpha_{(i)})^2 \\
&\Leftrightarrow t_0 \leq \sqrt{\sum_{i=1}^{k}(\beta_{(k+1)} - \beta_{(i)})^2}.
\end{aligned}$$

Therefore, we have $k_{\mathbf{x}_0} = k + 1$ if and only if $\sqrt{\sum_{i=1}^{k}(\beta_{(k+1)} - \beta_{(i)})^2} < t_0 \leq \sqrt{\sum_{i=1}^{k+1}(\beta_{(k+2)} - \beta_{(i)})^2}$. If $t_0 > \sqrt{\sum_{i=1}^{n-1}(\beta_{(k+2)} - \beta_{(i)})^2}$, we have $\lambda_{n-1} > \alpha_{(n)}$, and therefore, $k_{x_0} = n$ in this case.

Proof of Proposition 2: If $k \leq N(\mathbf{x}_0)/2$, we have $k - N(\mathbf{x}_0) \leq -N(\mathbf{x}_0)/2$, and

$$\begin{aligned}
&\mathbb{P}(\beta_{(k)} > \epsilon \mid \mathbf{x}_0, \mathbf{\Gamma}) \\
&\leq \mathbb{P}(\sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathcal{I}_{\{\beta_i^{[m]} \leq \epsilon\}} \leqslant k \mid \mathbf{x}_0, \mathbf{\Gamma}) \\
&= \mathbb{P}(\sum_{m=1}^{M}\sum_{i=1}^{n_m}[\mathcal{I}_{\{\|\mathbf{x}_{i,S_m}^{[m]} - \mathbf{x}_{0,S_m}\|_2 \leqslant (\epsilon - \tau_m^*)/L^*\}} \\
&- F_{\mathbf{x}_0}^{[m]}((\epsilon - \tau_m^*)/L^*)] \leqslant k - N(\mathbf{x}_0) \mid \mathbf{x}_0, \mathbf{\Gamma}) \\
&\leq \mathbb{P}(\sum_{m=1}^{M}\sum_{i=1}^{n_m}[F_{\mathbf{x}_0}^{[m]}((\epsilon - \tau_m^*)/L) \\
&- \mathcal{I}_{\{\|\mathbf{x}_{i,S_m}^{[m]} - \mathbf{x}_{0,S_m}\|_2 \leqslant (\epsilon - \tau_m^*)/L^*\}}] \geq N(\mathbf{x}_0)/2 \mid \mathbf{x}_0, \mathbf{\Gamma}).
\end{aligned}$$

Since $\beta_i^{[m]}$'s are conditionally independent, according to the Bernstein inequality, we have

$$\begin{aligned}
&\mathbb{P}\Big(\sum_{m=1}^{M}\sum_{i=1}^{n_m}[F_{\mathbf{x}_0}^{[m]}((\epsilon-\tau_m^*)/L) \\
&-\mathcal{I}_{\{\|\mathbf{x}_{i,\mathcal{S}_m}^{[m]}-\mathbf{x}_{0,\mathcal{S}_m}\|_2\leqslant(\epsilon-\tau_m^*)/L^*\}}]\geq N(\mathbf{x}_0)/2 \mid \mathbf{x}_0,\boldsymbol{\Gamma}\Big) \\
&\leq e^{\frac{-\frac{N(\mathbf{x}_0)^2}{4}}{2\sum_{m=1}^{M}\sum_{i=1}^{n_m}F_{\mathbf{x}_0}^{[m]}(\frac{\epsilon-\tau_m^*}{L^*})(1-F_{\mathbf{x}_0}^{[m]}(\frac{\epsilon-\tau_m^*}{L^*}))+\frac{N(\mathbf{x}_0)}{3}}} \\
&\leq \exp\Big(\frac{-\frac{N(\mathbf{x}_0)^2}{4}}{2N(\mathbf{x}_0)+\frac{N(\mathbf{x}_0)}{3}}\Big)=\exp\Big(-\frac{3N(\mathbf{x}_0)}{28}\Big) \\
&\leq \exp(-3k/14).
\end{aligned}$$

Proof of Theorem 1: Let $k_{\mathbf{x}_0}$ denote the number of nonzero elements in $\hat{\omega}$. Define $\lambda_{k_{\mathbf{x}_0}}=\|\hat{\omega}\|_2+\sum_{m=1}^{M}\sum_{i=1}^{n_m}\hat{\omega}_i^{[m]}(\tilde{\tau}_m+\tilde{L}\|\mathbf{x}_{i,\mathcal{S}_m}^{[m]}-\mathbf{x}_{0,\mathcal{S}_m}\|_2)$. According to Assumptions A1 and A2, we have

$$\begin{aligned}
&\mathbb{P}(|\hat{\eta}(\mathbf{x}_0)-\eta(\mathbf{x}_0)|\geq t_0\lambda_{k_{\mathbf{x}_0}} \mid \mathbf{x}_0,\{\mathbf{X}_{\mathcal{S}_m}^{[m]}\}_{m=1}^{M},\boldsymbol{\Gamma}) \\
&\leq 2\exp(-t_0^2/2)=\frac{\delta}{2}.
\end{aligned}$$

In addition, according to Step 3 in the INN classifier, we know that $\lambda_{k_{\mathbf{x}_0}}\leq\alpha_{(k_{\mathbf{x}_0}+1)}$. Therefore, $t_0\lambda_{k_{\mathbf{x}_0}}\leq t_0\alpha_{(k_{\mathbf{x}_0}+1)}=\beta_{(k_{\mathbf{x}_0}+1)}$, and we have

$$\begin{aligned}
&\mathbb{P}\Big(|\hat{\eta}(\mathbf{x}_0)-\eta(\mathbf{x}_0)|\geq\beta_{(k_{\mathbf{x}_0}+1)} \mid \mathbf{x}_0,\{\mathbf{X}_{\mathcal{S}_m}^{[m]}\}_{m=1}^{M},\boldsymbol{\Gamma}\Big) \\
&\leq \mathbb{P}\Big(|\hat{\eta}(\mathbf{x}_0)-\eta(\mathbf{x}_0)|\geq t_0\lambda_{k_{\mathbf{x}_0}} \mid \mathbf{x}_0,\{\mathbf{X}_{\mathcal{S}_m}^{[m]}\}_{m=1}^{M},\boldsymbol{\Gamma}\Big) \\
&\leq\frac{\delta}{2}.
\end{aligned}$$

For a given $\mathbf{x}_0\in\Omega_\epsilon$, we have

$$\begin{aligned}
&\mathbb{P}\big(\boldsymbol{\Phi}_n(\mathbf{x}_0)\neq\boldsymbol{\Phi}^*(\mathbf{x}_0) \mid \mathbf{x}_0,\boldsymbol{\Gamma}\big)= \\
&\mathbb{P}\Big(\hat{\eta}(\mathbf{x}_0)\geqslant\frac{1}{2},\eta(\mathbf{x}_0)<\frac{1}{2} \mid \mathbf{x}_0,\boldsymbol{\Gamma}\Big) \\
&+\mathbb{P}\Big(\hat{\eta}(\mathbf{x}_0)<\frac{1}{2},\eta(\mathbf{x}_0)\geq\frac{1}{2} \mid \mathbf{x}_0,\boldsymbol{\Gamma}\Big).
\end{aligned}$$

Suppose that $\eta(\mathbf{x}_0)\geq 1/2+\epsilon$, then we can show that

$$\begin{aligned}
&\mathbb{P}\big(\boldsymbol{\Phi}_n(\mathbf{x}_0)\neq\boldsymbol{\Phi}^*(\mathbf{x}_0) \mid \mathbf{x}_0,\boldsymbol{\Gamma}\big) \\
=&\mathbb{P}\Big(\hat{\eta}(\mathbf{x}_0)<\frac{1}{2} \mid \mathbf{x}_0,\boldsymbol{\Gamma}\Big) \\
=&\mathbb{P}\Big(\hat{\eta}(\mathbf{x}_0)<\frac{1}{2},\big|\hat{\eta}(\mathbf{x}_0)-\eta(\mathbf{x}_0)\big|<\beta_{(k_{\mathbf{x}_0}+1)} \mid \mathbf{x}_0,\boldsymbol{\Gamma}\Big) \\
&+\mathbb{P}\Big(\hat{\eta}(\mathbf{x}_0)<\frac{1}{2},\big|\hat{\eta}(\mathbf{x}_0)-\eta(\mathbf{x}_0)\big|\geq\beta_{(k_{\mathbf{x}_0}+1)} \mid \mathbf{x}_0,\boldsymbol{\Gamma}\Big) \\
\leqslant&\mathbb{P}(\beta_{(k_{\mathbf{x}_0}+1)}>\eta(\mathbf{x}_0)-\frac{1}{2} \mid \mathbf{x}_0,\boldsymbol{\Gamma})+\frac{\delta}{2}.
\end{aligned}$$

Since we choose $t_0 = \sqrt{2\log(4/\delta)}$, according to Proposition 1 and Assumption A3, we know that for a positive integer $k_0 \in [5\log(4/\delta), N(\mathbf{x}_0)/2]$, we have

$$\begin{aligned}
&\mathbb{P}(k_{\mathbf{x}_0} \geq k_0 \mid \mathbf{x}_0, \mathbf{\Gamma}) \\
&= \mathbb{P}(t_0^2 > \sum_{i=1}^{k_0-1} (\beta_{(k_0)} - \beta_{(i)})^2 \mid \mathbf{x}_0, \mathbf{\Gamma}) \\
&= \mathbb{P}(\exp(-\frac{1}{2}\sum_{i=1}^{k_0-1} (\beta_{(k_0)} - \beta_{(i)})^2) > \frac{\delta}{4} \mid \mathbf{x}_0, \mathbf{\Gamma}) \\
&\leq \frac{4}{\delta}\mathbb{E}(\exp(-\frac{1}{2}\sum_{i=1}^{k_0-1} (\beta_{(k_0)} - \beta_{(i)})^2) \mid \mathbf{x}_0, \mathbf{\Gamma}) \leq \frac{\delta}{4}.
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
&\mathbb{P}\big(\mathbf{\Phi}_n(\mathbf{x}_0) \neq \mathbf{\Phi}^*(\mathbf{x}_0) \mid \mathbf{x}_0, \mathbf{\Gamma}\big) \\
&\leq \mathbb{P}(\beta_{(k_{\mathbf{x}_0}+1)} > \eta(\mathbf{x}_0) - \frac{1}{2} \mid \mathbf{x}_0, \mathbf{\Gamma}) + \frac{\delta}{2} \\
&\leq \mathbb{P}(\beta_{(k_0)} > \eta(\mathbf{x}_0) - \frac{1}{2} \mid \mathbf{x}_0, \mathbf{\Gamma}) + \frac{\delta}{4} + \frac{\delta}{2} \\
&\leq \mathbb{P}(\beta_{(k_0)} > \epsilon \mid \mathbf{x}_0, \mathbf{\Gamma}) + \frac{3\delta}{4}.
\end{aligned}$$

In addition, according to Proposition 2, we have

$$\begin{aligned}
\mathbb{P}(\beta_{(k_0)} > \epsilon \mid \mathbf{x}_0, \mathbf{\Gamma}) &\leq \exp(-3k_0/14) \\
&\leq \exp(-\frac{15}{14}\log(\frac{4}{\delta})) \leq \frac{\delta}{4}.
\end{aligned}$$

Hence, we conclude that

$$\mathbb{P}\big(\mathbf{\Phi}_n(\mathbf{x}_0) \neq \mathbf{\Phi}^*(\mathbf{x}_0) \mid \mathbf{x}_0, \mathbf{\Gamma}\big) \leq \delta,$$

and

$$\mathbb{P}\big(\mathbf{\Phi}_n(\mathbf{x}_0) = \mathbf{\Phi}^*(\mathbf{x}_0) \mid \mathbf{x}_0, \mathbf{\Gamma}\big) \geq 1 - \delta.$$

Similarly, suppose that $\eta(\mathbf{x}_0) \leq 1/2 - \epsilon$, then we can also show that

$$\begin{aligned}
&\mathbb{P}\big(\mathbf{\Phi}_n(\mathbf{x}_0) \neq \mathbf{\Phi}^*(\mathbf{x}_0) \mid \mathbf{x}_0, \mathbf{\Gamma}\big) = \mathbb{P}\left(\hat{\eta}(\mathbf{x}_0) \geqslant \frac{1}{2} \mid \mathbf{x}_0, \mathbf{\Gamma}\right) \\
&\leq \mathbb{P}(\beta_{(k_{\mathbf{x}_0}+1)} > \frac{1}{2} - \eta(\mathbf{x}_0) \mid \mathbf{x}_0, \mathbf{\Gamma}) + \frac{\delta}{2} \\
&\leq \mathbb{P}(\beta_{(k_0)} > \epsilon \mid \mathbf{x}_0, \mathbf{\Gamma}) + \mathbb{P}(k_{\mathbf{x}_0} \geq k_0 \mid \mathbf{x}_0, \mathbf{\Gamma}) + \frac{\delta}{2} \leq \delta,
\end{aligned}$$

and therefore

$$\mathbb{P}\big(\mathbf{\Phi}_n(\mathbf{x}_0) = \mathbf{\Phi}^*(\mathbf{x}_0) \mid \mathbf{x}_0, \mathbf{\Gamma}\big) \geq 1 - \delta.$$

Integrative nearest neighbor (INN) classifier

Input: a block-missing multi-modality training data set $\{(\mathbf{X}_{\mathcal{S}_m}^{[m]}, \mathbf{y}^{[m]})\}_{m=1}^{M}$, a test data point $\mathbf{x}_0 \in \mathbb{R}^p$, values of the parameters $\tilde{L}$ and $\tilde{\tau}_m$'s.

1. For each $1 \le m \le M$ and $1 \le i \le n_m$, calculate $\alpha_i^{[m]} = \tilde{L}\|\mathbf{x}_{i,\mathcal{S}_m}^{[m]} - \mathbf{x}_{0,\mathcal{S}_m}\|_2 + \tilde{\tau}_m$.
2. Sort $\alpha_1^{[1]}, \alpha_2^{[1]}, , \alpha_{n_1}^{[1]}, \alpha_1^{[2]}, , \alpha_{n_M}^{[M]}$ in the ascending order. Let $\alpha_{(i)}$'s denote the ordered values.
3. Set $\lambda_0 = \alpha_{(1)} + 1$ and $k = 0$. While $\lambda_k > \alpha_{(k+1)}$ and $k \le n - 1$, calculate

$$k = k + 1;$$

$$\lambda_k = \frac{1}{k}\left(\sum_{i=1}^{k}\alpha_{(i)} + \sqrt{k + \left(\sum_{i=1}^{k}\alpha_{(i)}\right)^2 - k\sum_{i=1}^{k}\alpha_{(i)}^2}\right).$$

4. For each $1 \le m \le M$ and $1 \le i \le n_m$, calculate the weight

$$\hat{\omega}_i^{[m]} = \frac{\left(\lambda_k - \alpha_i^{[m]}\right)\cdot \mathcal{I}_{\left\{\alpha_i^{[m]} < \lambda_k\right\}}}{\sum_{m=1}^{M}\sum_{i=1}^{n_m}\left(\lambda_k - \alpha_i^{[m]}\right)\cdot \mathcal{I}_{\left\{\alpha_i^{[m]} < \lambda_k\right\}}}.$$

Calculate the estimate of $\eta(\mathbf{x}_0)$

$$\hat{\eta}(\mathbf{x}_0) = \sum_{m=1}^{M}\sum_{i=1}^{n_m}\hat{\omega}_i^{[m]}\mathbf{y}_i^{[m]}.$$

Output: $\Phi_n(\mathbf{x}_0) = \mathcal{I}_{\{\hat{\eta}(\mathbf{x}_0) \ge \frac{1}{2}\}}$.