An Approach to Identifying and Quantifying Bias in Biomedical Data

M. Clara De Paolis Kaluza, Shantanu Jain, Predrag Radivojac
Northeastern University, Boston, MA 02115, U.S.A.

Data biases are a known impediment to the development of trustworthy machine learning models and their application to many biomedical problems. When biased data is suspected, the assumption that the labeled data is representative of the population must be relaxed and methods that exploit a typically representative unlabeled data must be developed. To mitigate the adverse effects of unrepresentative data, we consider a binary semi-supervised setting and focus on identifying whether the labeled data is biased and to what extent. We assume that the class-conditional distributions were generated by a family of component distributions represented at different proportions in labeled and unlabeled data. We also assume that the training data can be transformed to and subsequently modeled by a nested mixture of multivariate Gaussian distributions. We then develop a multi-sample expectation-maximization algorithm that learns all individual and shared parameters of the model from the combined data. Using these parameters, we develop a statistical test for the presence of the general form of bias in labeled data and estimate the level of this bias by computing the distance between corresponding class-conditional distributions in labeled and unlabeled data. We first study the new methods on synthetic data to understand their behavior and then apply them to real-world biomedical data to provide evidence that the bias estimation procedure is both possible and effective.

Keywords: Bias detection, bias estimation, semi-supervised learning

1. Introduction

The development and application of machine learning methods have become commonplace in biomedical sciences and have the potential to transform clinical care.[1,2] Many of those predictive modeling approaches take place in a binary semi-supervised setting; that is, where the prediction outcome is dichotomized and the available data for training and evaluation contains samples of labeled and unlabeled examples. One such scenario is the prediction of the effect of genomic variants as pathogenic or benign, where labeled data contains pathogenic (positive) and benign (negative) variants from databases such as ClinVar[3] and the unlabeled data is often a large reference set of observed variants such as gnomAD.[4]

A traditional approach in semi-supervised learning is to assume that the labeled data is representative of unlabeled data, thus requiring little sophistication during model development, model selection, and performance evaluation. However, a distinguishing feature of real biomedical data is that the labeled examples may not be representative of the unlabeled data; that is, the labeled data may be biased.[5] Data biases can have adverse effects on the ability of models to be optimized for the unlabeled data at hand and can also lead to poor estimation

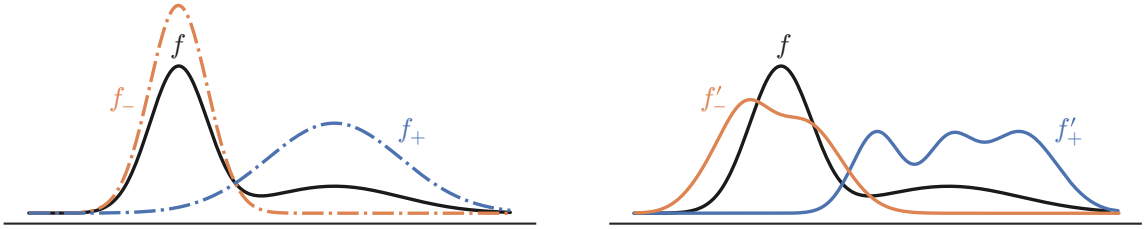


Fig. 1: An illustration of bias in labeled data. Left: unbiased (unobserved, dash-dotted lines) distributions of positive (f_+) and negative (f_-) classes that comprise the (observed, solid line) unbiased mixture distribution $f = \alpha f_+ + (1-\alpha) f_-$, drawn here with $\alpha = 0.3$. Right: the same unbiased observed mixture f together with biased observed distributions of positive (f'_+) and negative (f'_-) classes. The objective of this work is to use datasets from (f, f'_+, f'_-) to estimate the existence and extent of the differences between f_+ and f'_+ and between f_- and f'_-.

of a classifier's performance on a reference distribution.[6] More generally, biased data presents an obstacle to the development of trustworthy methods that are necessary for the societal acceptance of machine learning-based predictive technologies.[7,8]

Learning under sample selection bias is a well-known problem.[9] Early approaches relaxed the assumption of fully representative data by assuming the same class-conditional distributions in labeled and unlabeled data, thus reducing the problem of posterior estimation to estimation of class priors in unlabeled data.[10,11] Other approaches consider situations where at least one class-conditional distribution from which the labeled data is generated is representative of its unlabeled counterpart.[12–15] While such methods have advanced the treatment of sample selection bias, we are not aware of methods that can identify whether and to what extent labeled data differs from unlabeled data for a general form of bias.

The objective of this work is to develop a statistical test for identifying biased labeled data while simultaneously quantifying the level of bias. We assume that the real-world data can be transformed and subsequently modeled using nested mixtures of multivariate Gaussian distributions; that is, with both positive and negative samples being Gaussian mixtures themselves. We then model these class-conditional distributions in both labeled and unlabeled data by the shared underlying component distributions, but permit the proportions at which the data is sampled from those component distributions to differ between labeled and unlabeled data. We finally develop an expectation-maximization (EM) algorithm that learns both individual and shared parameters from the combined data which allows us to identify and quantify bias. Our experiments on synthetic and real-world data demonstrate the ability of this procedure to detect bias and provide useful information to data scientists in their workflows.

2. Problem Formulation

We consider the binary classification problem where input features $x \in \mathbb{R}^D$ are used to predict class label $y \in \mathcal{Y} = \{-, +\}$, where $+$ and $-$ represent the positive and negative class, respectively. Let $p(x, y)$ be the unknown joint distribution that governs how x appears in nature or in a target population of interest and its relationship with y. We refer to $p(x, y)$ as the unbiased distribution, where we expect a classifier to perform optimally. Let $f_+(x) = p(x|y = +)$ and $f_-(x) = p(x|y = -)$ denote the positive and negative class-conditional distributions, re-

spectively. Let $f(x) = p(x)$ denote the marginal distribution over x and $\alpha = p(y = +)$ be the probability that a random point from $p(x, y)$ is positive, the class prior for the positive class. It can be shown that f is a mixture distribution with components f_+ and f_- and mixing proportions α and $1 - \alpha$, respectively; i.e.,

$$f(x) = \alpha f_+(x) + (1 - \alpha) f_-(x). \tag{1}$$

Let L^+ and L^- represent sets of positive and negative labeled examples, respectively and U represent a set of unlabeled examples, available for training. Though we observe examples drawn randomly from $f(x)$ in U, unlike the standard classification setting, we might not observe labeled examples drawn randomly from $f_+(x)$ and $f_-(x)$. Instead L^+ and L^- are drawn from potentially biased class-conditional distributions $f'_+(x)$ and $f'_-(x)$, respectively (Fig. 1). We use the term bias here in a purely statistical sense; the labeled positives and negatives in the observed data are systemically different from those in the unlabeled data such that they cannot be interpreted to be drawn i.i.d. from the same distribution. In this work, we are interested in detecting and quantifying the extent to which the examples in L^+ and L^- differ from the positives and negatives in U, without the knowledge of the class labels in U.

2.1. *Assumptions*

If $f'_+(x)$ and $f'_-(x)$ are arbitrarily different from $f_+(x)$ and $f_-(x)$, respectively, detecting and quantifying the bias is an intractable problem. Fortunately, for most practical settings the biased and unbiased distributions are related. In this work, we employ a (G)aussian (c)omponent-based "(m)ixing (b)ias" assumption (MB-GC),[16] relating the biased and unbiased distributions. Formally, we assume both $f_+(x)$ and $f'_+(x)$ can be expressed as mixtures with the same K^+ shared Gaussian component distributions, but with differing mixing proportions. $f_-(x)$ and $f'_-(x)$ are assumed to be related in the same manner with K^- shared Gaussian components. Mathematically,

$$f_*(x) = \sum_{k \in \mathcal{K}^*} w_k^* \phi_k^*(x) \quad \text{and} \quad f'_*(x) = \sum_{k \in \mathcal{K}^*} v_k^* \phi_k^*(x), \tag{MB-GC}$$

where $*$ is a placeholder for $+$ or $-$; $\mathcal{K}^* = \{1, 2, \ldots, K^*\}$; $\boldsymbol{w}^* = [w_k^*]_{k \in \mathcal{K}^*}$ and $\boldsymbol{v}^* = [v_k^*]_{k \in \mathcal{K}^*}$ are probability vectors; i.e., $w_k^*, v_k^* \geq 0$, $\sum_{j \in \mathcal{K}^*} w_j^* = 1$ and $\sum_{j \in \mathcal{K}^*} v_j^* = 1$; and $\phi_k^*(x) = \phi(x; \mu_k^*, \Sigma_k^*)$ is the D-dimensional Gaussian density function with mean μ_k^* and covariance Σ_k^*. We use the shorthand $\boldsymbol{\mu}^* = \{\mu_k^*\}_{k \in \mathcal{K}^*}$ and $\boldsymbol{\Sigma}^* = \{\Sigma_k^*\}_{k \in \mathcal{K}^*}$ to group the parameters.

It is important to mention that a parametric approximation of the distributions becomes a universal nonparametric approximator as $K^+, K^- \to \infty$.[17] However, picking a large number of components may lead to a complex model prone to overfitting and identifiability issues. We therefore restrict ourselves to a relatively small number of components, up to eight, in each class-conditional representation, as in the parametric paradigm.

Since Gaussian mixture models are effective up to a moderate number dimensions, for high-dimensional data, we employ the MB-GC assumption after dimensionality reduction. Conceptually, we interpret the input feature $x \in \mathbb{R}^D$ as a low-dimensional representation of D_r-dimensional raw features ($D_r > D$) in such cases. It is conceivable that neither the raw features nor the dimensionality-reduced features appear exactly as Gaussian mixtures,

especially with a small number of components. In spite of this limitation, we argue that the modern representation learning approaches[18,19] can be used to learn embeddings that do satisfy that property, potentially making our assumptions and methods even more effective.

2.2. *Quantifying Bias*

Although various distance measures can be used,[20] we quantify the bias between f_+ and f'_+ as the area under the ROC curve (AUC) of an optimal binary classifer, or a score function $s : \mathbb{R}^D \to \mathbb{R}$, between them. Based on the probabilistic interpretation of AUC,[21] it is the probability that a randomly drawn example from f_+ achieves a higher score than a randomly drawn example from f'_+, as per an optimal score function. Mathematically, for $\mathcal{S}$ being the family of all real-valued score functions defined on $\mathbb{R}^D$,

$$\mathrm{AUC}(f_+, f'_+) = \max_{s\in\mathcal{S}} \mathrm{AUC}_s(f_+, f'_+),$$

where, correcting for ties, $\mathrm{AUC}_s(f_+, f'_+) = p(s(X_{f_+}) > s(X_{f'_+})) + \frac{1}{2}p(s(X_{f_+}) = s(X_{f'_+}))$; X_{f_+} and $X_{f'_+}$ are random variables distributed according to f_+ and f'_+, respectively. Note that AUC is symmetric; i.e., $\mathrm{AUC}(f_+, f'_+) = \mathrm{AUC}(f'_+, f_+)$. It ranges from 0.5 to 1, with a higher value indicating a larger difference between the two distributions and consequently a larger bias. Typically, values between 0.5 and 0.6 are considered to be small enough that the distributions can be interpreted to be practically indistinguishable. A value of 1 corresponds to a perfect classifier; that is, a situation when the supports between f'_+ and f_+ are distinct. Thus, in this work, a value of 0.5 indicates no bias and a value of 1 indicates maximum bias (Fig. 2).

If samples from f_+ and f'_+ were available, $\mathrm{AUC}(f_+, f'_+)$ could be estimated by first training a classifier to separate the samples, and then computing AUC in the standard manner as the area under the ROC curve. Though a sample from f_+ is not readily available, such a sample is procured using the approach presented in Methods. The bias between f_- and f'_- can be quantified as $\mathrm{AUC}(f_-, f'_-)$ and estimated similarly.

3. Methods

In order to detect and quantify the bias, we derive an expectation-maximization (EM) algorithm from multi-sample Gaussian mixtures. Under the MB-GC assumptions each of L^+, L^- and U contain examples drawn i.i.d. from a Gaussian mixture. Formally,

$$\forall x \in L^*,\ x \sim \sum_{k\in\mathcal{K}^*} v_k^* \phi_k^*(x) \qquad \forall x \in U,\ x \sim \sum_{k\in\mathcal{K}^+} \alpha w_k^+ \phi_k^+(x) + \sum_{k\in\mathcal{K}^-} (1-\alpha) w_k^- \phi_k^-(x),$$

where the second equation for the distribution of U is obtained by combining MB-GC assumptions with Eq. 1 and $*$ is a placeholder for $+$ or $-$. Note that the resultant distribution is a mixture of $K^+ + K^-$ components. The combined data log-likelihood is given by

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}; L^+, L^-, U) = &\sum_{x\in L^+} \log\left(\sum_{k\in\mathcal{K}^+} v_k^+ \phi_k^+(x)\right) + \sum_{x\in L^-} \log\left(\sum_{k\in\mathcal{K}^-} v_k^- \phi_k^-(x)\right) \\ &+ \sum_{x\in U} \log\left(\sum_{k\in\mathcal{K}^+} \alpha w_k^+ \phi_k^+(x) + \sum_{k\in\mathcal{K}^-} (1-\alpha) w_k^- \phi_k^-(x)\right),\end{aligned}$$

where $\boldsymbol{\theta} = \{\alpha, \boldsymbol{w}^+, \boldsymbol{w}^-, \boldsymbol{v}^+, \boldsymbol{v}^-, \boldsymbol{\mu}^+, \boldsymbol{\mu}^-, \boldsymbol{\Sigma}^+, \boldsymbol{\Sigma}^-\}$ represent all unknown parameters. To obtain the maximum likelihood estimates of the parameters, we derive the following update equations, under the EM framework.

$$\hat{\alpha} = \frac{1}{|U|}\sum_{x\in U}\sum_{k\in\mathcal{K}^+}\omega_k^+(x;\check{\boldsymbol{\theta}}), \qquad \hat{w}_k^* = \frac{1}{\check{\alpha}^*|U|}\sum_{x\in U}\omega_k^*(x;\check{\boldsymbol{\theta}}), \qquad \hat{v}_k^* = \frac{1}{|L^*|}\sum_{x\in L^*}\nu_k^*(x;\check{\boldsymbol{\theta}})$$

$$\hat{\mu}_k^* = \frac{\sum_{x\in U}\omega_k^*(x;\check{\boldsymbol{\theta}})x + \sum_{x\in L^*}\nu_k^*(x;\check{\boldsymbol{\theta}})x}{\sum_{x\in U}\omega_k^*(x;\check{\boldsymbol{\theta}}) + \sum_{x\in L^*}\nu_k^*(x;\check{\boldsymbol{\theta}})} \qquad \text{(EM-update)}$$

$$\hat{\Sigma}_k^* = \frac{\sum_{x\in U}\omega_k^*(x;\check{\boldsymbol{\theta}})(x-\check{\mu}_k^*)(x-\check{\mu}_k^*)^T + \sum_{x\in L^*}\nu_k^*(x;\check{\boldsymbol{\theta}})(x-\check{\mu}_k^*)(x-\check{\mu}_k^*)^T}{\sum_{x\in U}\omega_k^*(x;\check{\boldsymbol{\theta}}) + \sum_{x\in L^*}\nu_k^*(x;\check{\boldsymbol{\theta}})},$$

where $\check{}$ and $\hat{}$ are used to represent the current and updated parameters, respectively, during an EM iteration; $\alpha^+ = \alpha$ and $\alpha^- = 1-\alpha$; $\omega_k^*(x;\boldsymbol{\theta})$ is the probability that a given $x \in U$ comes from ϕ_k^*; similarly, $\nu_k^*(x;\boldsymbol{\theta})$ is the probability that a given $x \in L^*$ comes from ϕ_k^*; i.e.,

$$\omega_k^*(x;\boldsymbol{\theta}) = \frac{\alpha^* w_k^* \phi(x;\mu_k^*,\Sigma_k^*)}{\sum_{k\in\mathcal{K}^+}\alpha w_k^+\phi\big(x;\mu_k^+,\Sigma_k^+\big) + \sum_{k\in\mathcal{K}^-}(1-\alpha)w_k^-\phi\big(x;\mu_k^-,\Sigma_k^-\big)}$$

$$\nu_k^*(x;\boldsymbol{\theta}) = \frac{v_k^*\phi(x;\mu_k^*,\Sigma_k^*)}{\sum_{k\in\mathcal{K}^*}v_k^*\phi\big(x;\mu_k^*,\Sigma_k^*\big)}.$$

Starting with an initial value, as discussed in Section 3.3, the parameters in $\boldsymbol{\theta}$ are iteratively updated using Eq. EM-update until convergence, when the relative change in the log-likelihood, $(\mathcal{L}(\hat{\boldsymbol{\theta}};L^+,L^-,U) - \mathcal{L}(\check{\boldsymbol{\theta}};L^+,L^-,U))/\mathcal{L}(\check{\boldsymbol{\theta}};L^+,L^-,U)$, is less than a small predefined threshold (δ) or until the number of iterations reaches a predefined maximum (I).

3.1. *Estimating Bias*

Once $\boldsymbol{\theta}$ is estimated, we use the estimated value of $\boldsymbol{w}^+$ to infer the distribution of the unbiased positives, f_+, as per Eq. MB-GC. In order to estimate the bias in the labeled positive sample, we first subsample from U, to procure a set, $\hat{L}^+$, representing estimated f_+. To this end, we use the responsibility, $r^+(x;\theta) = \sum_{k\in\mathcal{K}^+}\omega_k^+(x;\theta)$, giving the probability that a given $x \in U$ is a positive. Precisely, $\forall x \in U$, if

$$\text{Bernoulli}\big(r^+(x;\theta)\big) = \begin{cases} 1 & \text{add } x \text{ to } \hat{L}^+, \\ 0 & \text{discard } x, \end{cases}$$

where $r^+(x;\theta)$ is used as the success probability of the Bernoulli distribution. Once $\hat{L}^+$ is procured, we estimate the bias, $\text{AUC}\big(f_+, f'_+\big)$, by training a classifier between L^+ and $\hat{L}^+$ treated as positives and negatives, respectively, and compute the AUC using the classifier's score function. The bias in L^- can be similarly estimated using the responsibility $r^-(x;\theta) = \sum_{k\in\mathcal{K}^-}\omega_k^-(x;\theta)$ to subsample $\hat{L}^-$ from U and then computing the AUC for a classifier trained to separate $\hat{L}^-$ and L^-. For a dataset $S = (L^+, L^-, U)$, we denote the estimated bias as $\text{Bias}_{\text{est}}(S)$.

3.2. *Detecting Bias*

We focus the subsequent presentation on bias detection in L^+ only; the detection of bias in L^- can be approached similarly. Due to model misspecification and errors in the parameter

and bias estimation, a bias higher than 0.5 is likely to be estimated, when, in fact, the data is unbiased. To mitigate this issue, we introduce a bias threshold, $\tau \in [0.5, 1]$, and interpret a dataset to contain bias only if its estimated bias is above τ. A higher value of τ would decrease the probability that an unbiased dataset is detected to have bias (type-1 error), $e(\tau)$. However, it will also decrease the probability that a biased dataset is detected to have bias (power), $q(\tau)$. To achieve a low type-1 error and a high power, we determine an appropriate value of τ by controlling for type-1 error on synthetic datasets; see Synthetic Data.

Let $\mathcal{S}^{\text{ub}}_{\text{syn}}$ and $\mathcal{S}^{\text{b}}_{\text{syn}}$ be two families of unbiased and biased synthetic datasets, respectively, where each dataset is of the form (L^+, L^-, U) and bias is defined as per the current context. Let $e_{\text{syn}}(\tau) = |\{\text{Bias}_{\text{est}}(S) \geq \tau,\ S \in \mathcal{S}^{\text{ub}}_{\text{syn}}\}|/|\mathcal{S}^{\text{ub}}_{\text{syn}}|$ and $q_{\text{syn}}(\tau) = |\{\text{Bias}_{\text{est}}(S) \geq \tau,\ S \in \mathcal{S}^{\text{b}}_{\text{syn}}\}|/|\mathcal{S}^{\text{b}}_{\text{syn}}|$ be the fraction of unbiased and biased synthetic datasets with estimated bias above τ, respectively. We define $\tau_\eta = \min_\tau e_{\text{syn}}(\tau) \leq \eta$ as a suitable threshold for which type-1 error computed w.r.t $\mathcal{S}^{\text{ub}}_{\text{syn}}$ is η (typically, $\eta \in [0, 0.1]$); i.e., $e_{\text{syn}}(\tau_\eta) = \eta$. The power computed w.r.t. $\mathcal{S}^{\text{b}}_{\text{syn}}$ at τ_η is $q_{\text{syn}}(\tau_\eta)$. Using this framework, for any real-world dataset $S = (L^+, L^-, U)$, we enable computing a p-value for bias detection as p-value$(S) = e_{\text{syn}}(\text{Bias}_{\text{est}}(S))$, the proportion of unbiased synthetic datasets estimated to have a bias above $\text{Bias}_{\text{est}}(S)$.

Note that estimates of type-1 error, power and p-value computed w.r.t. synthetic datasets are representative of their true values to the extent that they capture the diversity of the real-world datasets. In addition to explicitly diversifying the synthetic datasets to a feasible extent, we address this issue by also estimating type-1 error and power w.r.t. selected unbiased and biased real-world datasets, still using the synthetic data threshold; see Data and Results.

3.3. *Implementation Details*

Initialization Parameter estimates of our algorithm are likely sensitive to the initial parameters; it is known to be the case for the standard EM algorithm (GMM) for a single Gaussian mixture sample.[22,23] Because we have access to labeled data, we leverage it for parameter initialization. However, in order to introduce more diversity to initialization across multiple restarts, we do not use parameters estimates on only labeled data as our initial parameters; e.g., by using parameters from GMM on each L^*. Instead, we initialize parameters in the following steps. (1) Run GMM with K^* components on L^* to obtain initial estimates of $\boldsymbol{v}^*$, for $* \in \{+, -\}$ and save the location parameter estimates $\boldsymbol{u}^* = \{u^*_k\}_{k \in \mathcal{K}^*}$ (2) Run k-means++[24] on unlabeled data U with $K^+ + K^-$ centers. Sort the centers based on the minimum distance to any location in $\boldsymbol{u}^+$. Pick the top K^+ centers to initialize $\boldsymbol{\mu}^+$ and the remaining centers as $\boldsymbol{\mu}^-$. (3) Compute the distance from unlabeled points $x \in U$ to each of the $K^+ + K^-$ centers and assign them to the closest one. This gives an assignment for all points to a cluster which has already been assigned as positive or negative. (4) Use the assignments to compute $\alpha = \frac{\sum_{k \in \mathcal{K}^+} |A^+_k|}{|U|}$, $w^*_k = \frac{|A^*_k|}{\sum_{k \in \mathcal{K}^*} |A^*_k|}$ and $\Sigma^*_k = \frac{1}{|A^*_k|} \sum_{x \in A^*_k} (x_i - \mu^*_k)(x_i - \mu^*_k)^T$, where A^+_k (A^-_k) indicate points assigned to the k-th positive (negative) cluster.

Model Selection Parameter estimation with EM algorithms when the number of components is unknown is not trivial and many methods exist for model selection.[25,26] We employ the one-fold cross-validation-based information criterion (CVIC)[25] for model selection by running

our EM optimization for various values of K^+, K^- and selecting the model that achieves the highest log-likelihood on a validation set.

Hyper-parameters We assume $K \equiv K^+ = K^-$ for convenience in experimentation. We use the maximum number of iterations $I = 2000$ and the convergence threshold $\delta = 10^{-8}$ for termination. We run the estimation on each dataset 20 times with different random seeds.

4. Data

4.1. *Synthetic Data*

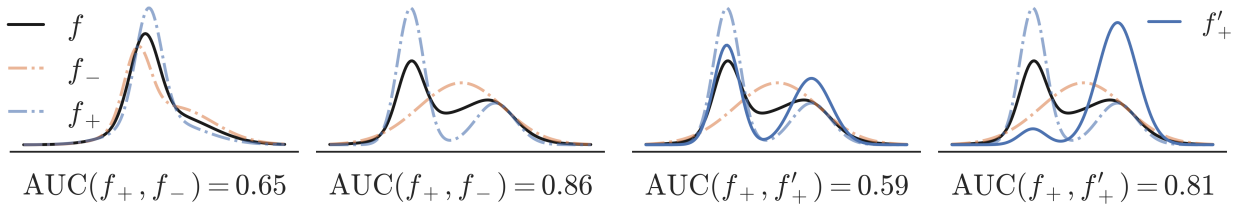
To find appropriate bias thresholds and evaluate our method, we generate synthetic Gaussian mixture datasets, following MB-GC assumptions, from known parameters. This allows us to control bias directly and evaluate performance for different levels of bias in the dataset.

Here f_+ and f_- are both K-component Gaussian mixtures. Their parameters are determined by a given AUC(f_+, f_-) range (e.g., $[0.65, 0.7]$) and mutual irreducibility parameters, support ($\sigma = 0.01$) and pairwise responsibility threshold ($\rho = 0.9$), governing the overlap between each pair of components. Let ϕ_i and ϕ_j be two of the $2K$ components and let Z_i and Z_j be samples of 1000 examples each, drawn from ϕ_i and ϕ_j, respectively. If more than σ fraction of points in Z_i have $\phi_i(\cdot) \geq \rho(\phi_i(\cdot) + \phi_j(\cdot))$ and, similarly, more than σ fraction of points in Z_j have $\phi_j(\cdot) \geq \rho(\phi_i(\cdot) + \phi_j(\cdot))$, then ϕ_i and ϕ_j are considered to be approximately mutually irreducible.[27] Starting with random values for the location and shape parameters for each component as well as the mixing proportions $\boldsymbol{w}^+$ and $\boldsymbol{w}^-$ of the two mixtures (drawn from a flat Dirichlet distribution), the parameters are perturbed until AUC(f_+, f_-), evaluated with $f_+(\cdot)/f_-(\cdot)$ as the score function (known to be optimal), lies in the desired range and all pairs of the $2K$ components are approximately mutually irreducible w.r.t. σ and ρ.

We generate 1000 unbiased datasets for each combination of dimensions $D \in \{1, 2, 8, 16\}$ and number of components $K \in \{2, 4, 8\}$. The class prior α is sampled uniformly from the range $[0.01, 0.99]$ for each dataset. Seven AUC(f_+, f_-) ranges, $[0.65, 0.7], [0.7, 0.75], \ldots, [0.95, 1]$ are approximately equally represented in the 1000 datasets for each setting. For the unbiased datasets, f'_+ and f'_- are set equal to f_+ and f_-, respectively.

To evaluate performance of bias estimation against known values of bias, we generate 1750 datasets for each dimension and number of components for varying levels of bias AUC(f_+, f'_+) between 0.5 and 1 (Fig. 2b). First α, f_+ and f_- are generated as for the unbiased data, where the seven AUC(f_+, f_-) ranges are equally represented across the 1750 datasets. A desired range of bias is achieved by drawing random mixing proportions, $\boldsymbol{v}^+$, from a flat Dirichlet distribution until AUC(f_+, f'_+) computed with the optimal score function $f_+(\cdot)/f'_+(\cdot)$ is in the target bias range. The five bias ranges $[0.5, 0.6], [0.6, 0.7], \ldots, [0.9, 1]$ are equally represented across the datasets. For simplicity, f'_- is set equal to f_-.

Each dataset has 100,000 unlabeled points from $f = \alpha f_+ + (1-\alpha) f_-$ and 5,000 labeled points from each f'_+ and f'_- with the chosen parameters. Figure 2a shows examples of 1D distributions for different values of AUC(f_+, f_-) within the range we use to sample synthetic data. These examples illustrate the complexity of synthetic datasets; even for higher AUC(f_+, f_-), the positive and negative distributions are not easily distinguished.



(a) Positive f_+ and negative f_- classes (b) Bias in positive samples

Fig. 2: Synthetic data in one dimension. Examples of (a) low and high $\mathrm{AUC}(f_+, f_-)$ and (b) low and high bias $\mathrm{AUC}(f_+, f'_+)$. Unlabeled mixtures f shown here with $\alpha = 0.5$ in all cases.

4.2. *Biomedical Data*

We selected 8 biomedical datasets from the the UCI Machine Learning Repository[28] to apply our methods. The following datasets were used, with a note that for each we give the number of examples, the fraction of examples from the positive class (α) and the number of features D in parentheses: Activity recognition with healthy older people using a wearable sensor[29] (52481, 0.29, 8), Epileptic Seizure Recognition[30] (11500, 0.18, 178), Smartphone-Based Recognition of Human Activities and Postural Transitions[31] (10929, 0.16, 561), Mushroom[28] (8124, 0.21, 126), HIV-1 protease cleavage[32] (6590, 0.20, 160), Splice-junction Gene Sequences[33] (3190, 0.24, 287), Parkinsons Telemonitoring[34] (5875, 0.48, 20), and Physicochemical Properties of Protein Tertiary Structure[28] (45730, 0.13, 9).

Datasets were constructed by assigning one class as positive and the remaining as negative for multi-class data or setting a threshold for regression data. For each problem, 100 unbiased datasets were generated by selecting a subset of labeled points uniformly. We generate 250 biased datasets for each biological dataset through Markov sampling. First a point x_i is selected uniformly at random from the positive class. The same point is resampled with some probability p_{stay} and a new point x_j is selected with probability $1 - p_{\text{stay}}$. The transition probability $\Pr(x_j|x_i)$ is proportional to the inverse of the squared Euclidean distance between points $||x_i - x_j||^2$. Since the true bias cannot be measured directly, we use the probability of resampling p_{stay} as a proxy for bias. Higher values of p_{stay} correspond to higher bias in labeled data since the feature space will be less uniformly sampled (Fig. 3). In each case, 20% of points are held out as a validation set used for model selection. We reduce the dimensionality with PCA for datasets with more than 8 features.

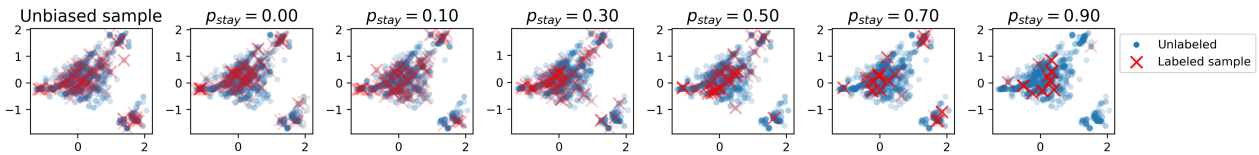


Fig. 3: Unbiased (far left) and biased samples from the dataset HIV[32] with varying probability of resampling a point p_{stay}. Features are illustrated projected onto the first two principal components.

5. Experiments

Empirical Null Distribution and Bias Threshold We use synthetic Gaussian mixture datasets to determine the bias threshold for a range of dimensions $D \in \{1, 2, 8, 16\}$ and number of components $K \in \{2, 4, 8\}$. We consider bias in positive class, but the method for estimating bias in the negative class or both would follow the same process. We run the EM optimization on each unbiased dataset to estimate all unknown parameters, $\boldsymbol{\theta}$. We use the estimated parameters $\widehat{\boldsymbol{\theta}}$ to compute the estimated bias for the positive class, $\text{AUC}(\widehat{f_+}, \widehat{f'_+})$, where $\widehat{\cdot}$ indicates the parameters estimated by the optimization procedure and the distributions parameterized by them. The true bias $\text{AUC}(f_+, f'_+)$ for these datasets is exactly 0.5 since the distributions are identical (no bias), but because there is error in the estimation $\widehat{\boldsymbol{\theta}}$, $\text{AUC}(\widehat{f_+}, \widehat{f'_+}) \geq 0.5$. For each setting of dimension D and number of components K used to generate the datasets, we determine $\tau_\eta(D, K)$ for $\eta \in \{0.05, 0.10\}$ of datasets with $\text{AUC}(\widehat{f_+}, \widehat{f'_+}) \geq \tau_\eta(D, K)$.

Model Selection To apply the appropriate bias threshold $\tau_\eta(D, K)$ to any data it is important to know the number of components that best represent the data and use the threshold found for that setting (dimension is known). However, the true or best value of K is not generally known for any dataset. We evaluate the effect of unknown K for finding the threshold τ_η by running the optimization on unbiased datasets for $K \in \{2, 4, 8\}$ on all datasets, regardless of which value was used to generate the data. For each dataset, we compute the estimated parameters log-likelihood on a validation set and choose the model that maximizes the value. The validation set is generated with the same parameters as the original dataset.

Bias Quantification and Detection To evaluate our method in detecting and estimating bias, we run our EM optimization algorithm on synthetic and biological datasets with varying amount of bias and report the estimated bias. For synthetic data where the true bias is known, we evaluate power for each level of type-1 error, $\eta \in \{0.05, 0.10\}$. Ground truth biased datasets $\mathcal{B}$ are those where the true bias $\text{AUC}(f_+, f'_+) > 0.5$, for K number of components. Predicted biased datasets $\widehat{\mathcal{B}}$ are those where $\text{AUC}(\widehat{f_+}, \widehat{f'_+}) \geq \tau_\eta(D, \widehat{K})$ for $\widehat{K}$ selected through model selection. Power is estimated as $q(\tau) = |\widehat{\mathcal{B}}|/|\mathcal{B}|$.

6. Results and Discussion

Figure 4 illustrates the thresholds found for each dimension and number of components. When the number of components, K, is smaller, parameter estimation more reliably estimates the bias lower. As the number of dimensions and number of components increases, so does the complexity of the optimization problem and the estimated value of bias. These results suggest the utility of finding dimension- and component-specific thresholds, and the empirical null distribution for ascertaining bias.

Results on quantification of bias on synthetic (Fig. 5) and biomedical (Fig. 6) data show increasing estimated bias as true bias increases. Note that for biomedical datasets the true bias is unknown and p_{stay} is not a direct measurement of bias; different data sets have different levels of compactness in their feature space. Since the sampling probability is proportional to the inverse distance between points, the bias is also dependent on the density of points. Bias will

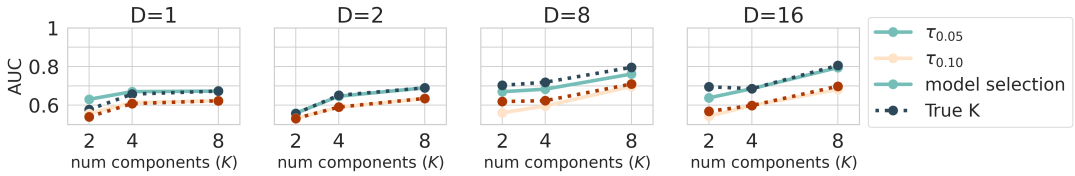


Fig. 4: Bias ($\text{AUC}(f_+, f'_+)$) thresholds found from parameter estimation on unbiased data sets.

differ across datasets for the same value of p_{stay} and estimated bias cannot be directly compared between datasets. However, for each datasets bias should increase as the sampling less uniform, *i.e.* p_{stay} increases. In synthetic data, we see excellent power (Fig. 7) for the type-1 error of 0.05 across all levels of bias, dimensionality D and the number of components (K) per class-conditional distribution. We also see for high-bias datasets ($\text{AUC}(f_+, f'_+) \geq 0.9$) on datasets with two components, that some datasets have a low estimated bias. Our investigation showed that to generate datasets with high bias and few components, the mixing proportions w_k^+ or v_k^+ must be very skewed, making the optimization difficult, sometimes unrealistically so. For one dimension, the average minimum value of the smallest w_k^+ for datasets with $\text{AUC}(f_+, f'_+) \geq 0.9$ is 0.01, 0.07 for $0.8 \leq \text{AUC}(f_+, f'_+) < 0.9$, and 0.19-0.23 for $\text{AUC}(f_+, f'_+) < 0.8$.

Figure 7 shows the power for bias detection on synthetic datasets for type-1 error $\eta \in \{0.05, 0.10\}$. For each setting we see generally higher power in bias detection as the true bias increases. For higher type-1 error, the detection achieves a higher power. Again there is a drop in performance for $K = 2$ in high-bias datasets due to the challenging nature of these datasets.

For real datasets we also show that our estimation of α and negative bias is not generally affected by increasingly biased samples of the positive class (Fig. 6, middle and bottom rows, respectively). Our EM algorithm is still able to detect that the set of unbiased labels from the negative class are truly unbiased (a low value of $\text{AUC}(\widehat{f_-}, \widehat{f'_-})$). The estimation for bias for negative class in UCI results is consistently better than the estimation of bias for unbiased positive samples because α is always less than or equal to 0.5. Higher estimated bias in negatives seems to be correlated with overestimation of the class prior α, particularly exemplified in the parkinsons dataset.

7. Conclusion

Despite a broad awareness that biased data may adversely impact the deployment of machine learning tools in biomedicine, there is a surprising dearth of methods built to ascertain the existence and the level of bias in available data. We set out to address this deficiency by developing and extensively evaluating a bias estimation method based on reasonable assumptions. We used synthetic and real-world biomedical data to show that technologies for bias detection and ultimately correction can be realistically implemented in future data processing pipelines.

Code

The source code for this project is available at `https://github.com/claradepaolis/bi-est`

Acknowledgements

The authors acknowledge the support by the NIH grants U01HG012022 and R01HD101246.

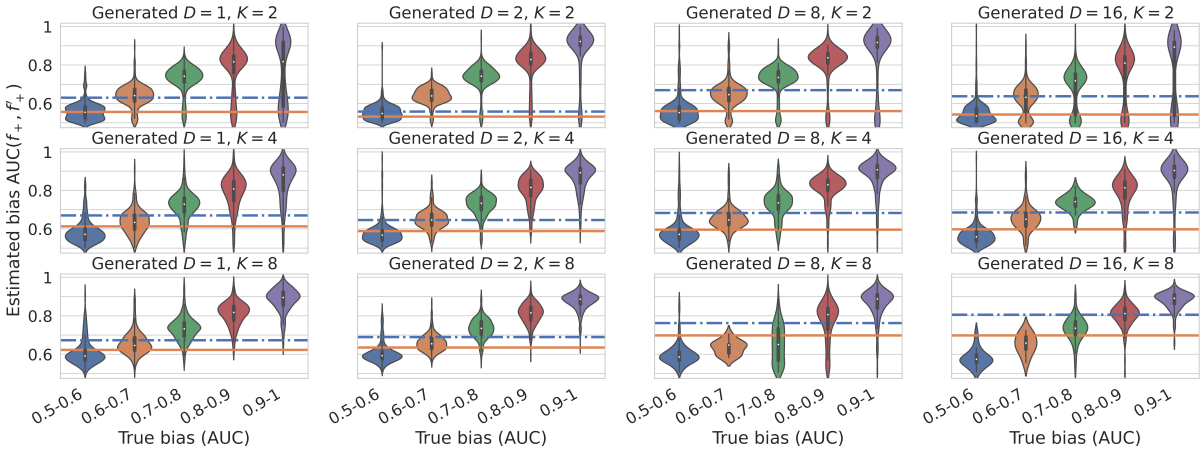


Fig. 5: Estimated bias on Gaussian mixtures with varying true bias AUC(f_+, f'_+). Bias thresholds $\tau_{0.05}, \tau_{0.10}$ shown as dash-dotted and solid lines, respectively.

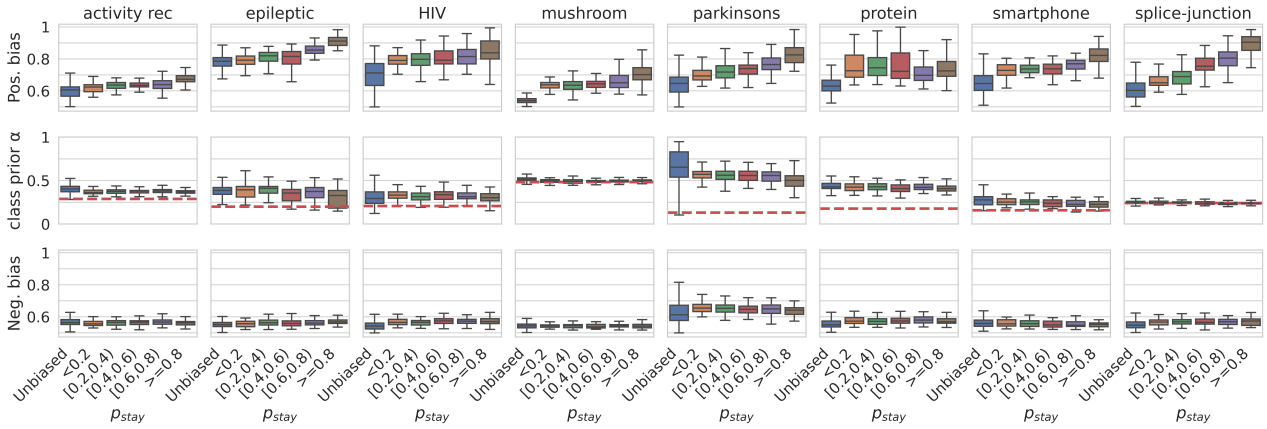


Fig. 6: Bias and parameter estimation for biomedical datasets. Each column shows results for samples from each dataset. Top: Bias estimation for positive class for unbiased (leftmost) and biased sampled datasets for increasing levels of p_{stay}, corresponding to larger bias. Middle: Estimation of the class prior α with true value shown as dashed line. Bottom: Bias estimation for negative class, which is unbiased in each case.

References

1. K. B. Johnson *et al.*, *Clin Transl Sci* **14**, 86 (2021).
2. P. Rajpurkar *et al.*, *Nat Med* **28**, 31 (2022).
3. M. J. Landrum *et al.*, *Nucleic Acids Res* **44**, D862 (2016).
4. K. J. Karczewski *et al.*, *Nature* **581**, 434 (2020).
5. T. Stoeger *et al.*, *PLoS Biol* **16**, p. e2006643 (2018).
6. B. Yu and K. Kumbier, *Proc Natl Acad Sci U S A* **117**, 3920 (2020).
7. L. Szabo, Artificial intelligence is rushing into patient care–and could raise risks, *Scientific American* **12** (2019).
8. R. Schwartz *et al.*, *Draft NIST Special Publication 1270* (2021).
9. J. Heckman, *Econometrica* **47**, 153 (1979).

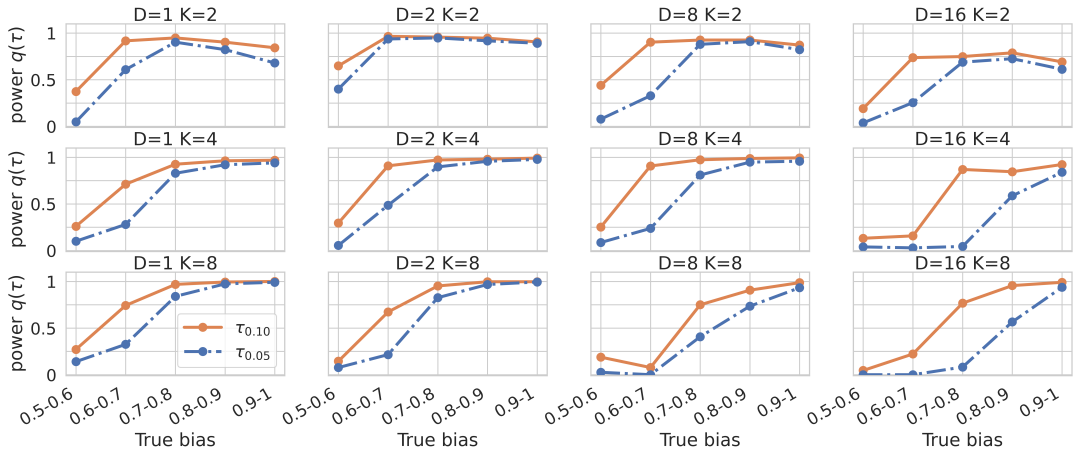


Fig. 7: Power $q(\tau)$ of bias prediction for type-1 error $\eta \in \{0.05, 0.10\}$ on unbiased datasets.

10. S. Vucetic and Z. Obradovic, Classification on data with biased class distribution, in *ECML*, 2001.
11. M. Saerens *et al.*, *Neural Comput* **14**, 21 (2002).
12. B. Zadrozny, Learning and evaluating classifiers under sample selection bias, in *ICML*, 2004.
13. J. Huang *et al.*, Correcting sample selection bias by unlabeled data, in *NeurIPS*, 2006.
14. C. Cortes *et al.*, Sample selection bias correction theory, in *ALT*, 2008.
15. Y. G. Hsieh *et al.*, Classification from positive, unlabeled and biased negative data, in *ICML*, 2019.
16. S. Jain *et al.*, Class prior estimation with biased positives and unlabeled examples, in *AAAI*, 2020.
17. W. Feller, *An introduction to probability and its applications* (Wiley, 1966).
18. M. Śmieja *et al.*, *IEEE Trans Neural Netw Learn Syst* **32**, 3930 (2020).
19. Y. Uğur *et al.*, *Entropy* **22**, p. 213 (2020).
20. M. M. Deza and E. Deza, *Encyclopedia of distances* (Springer, 2013).
21. J. Hanley and B. J. McNeil, *Radiology* **143**, 29 (1982).
22. G. J. McLachlan and T. Krishnan, *The EM algorithm and extensions* (Wiley, 2007).
23. J.-P. Baudry and G. Celeux, *Stat Comput* **25**, 713 (2015).
24. S. Vassilvitskii and D. Arthur, k-means++: The advantages of careful seeding, in *SODA*, 2006.
25. G. J. McLachlan and S. Rathnayake, *WIREs Data Mining Knowl Discov* **4**, 341 (2014).
26. T. Huang *et al.*, *Stat Sin* **27**, 147 (2017).
27. S. Jain *et al.*, Estimating the class prior and posterior from noisy positives and unlabeled data, in *NeurIPS*, 2016.
28. D. Dua and C. Graff, UCI machine learning repository (2017).
29. R. L. S. Torres *et al.*, Sensor enabled wearable RFID technology for mitigating the risk of falls near beds, in *RFID*, 2013.
30. R. G. Andrzejak *et al.*, *Phys Rev E* **64**, p. 061907 (2001).
31. J.-L. o. Reyes-Ortiz, *Neurocomputing* **171**, 754 (2016).
32. T. Rögnvaldsson *et al.*, *Bioinformatics* **31**, 1204 (2015).
33. M. Noordewier *et al.*, Training knowledge-based neural networks to recognize genes in DNA sequences, in *NeurIPS*, 1990.
34. A. Tsanas *et al.*, *IEEE Trans Biomed Eng* **57**, 884 (2010).