Divide and Imitate: Multi-cluster Identification and Mitigation of Selection Bias

Katharina Dost[1(✉)], Hamish Duncanson[1], Ioannis Ziogas[2], Patricia Riddle[1], and Jörg Wicker[1]

[1] University of Auckland, Auckland, New Zealand
{kdos481,hdun603}@aucklanduni.ac.nz, {p.riddle,j.wicker}@auckland.ac.nz
[2] University of Mississippi, Oxford, USA
ziogas@olemiss.edu

Abstract. Machine Learning can help overcome human biases in decision making by focussing on purely logical conclusions based on the training data. If the training data is biased, however, that bias will be transferred to the model and remains undetected as the performance is validated on a test set drawn from the same biased distribution. Existing strategies for selection bias identification and mitigation generally rely on some sort of knowledge of the bias or the ground-truth. An exception is the Imitate algorithm that assumes no knowledge but comes with a strong limitation: It can only model datasets with one normally distributed cluster per class. In this paper, we introduce a novel algorithm, Mimic, which uses Imitate as a building block but relaxes this limitation. By allowing mixtures of multivariate Gaussians, our technique is able to model multi-cluster datasets and provide solutions for a substantially wider set of problems. Experiments confirm that Mimic not only identifies potential biases in multi-cluster datasets which can be corrected early on but also improves classifier performance.

1 Introduction

Throughout the years, Machine Learning and Data Mining have gained influence into a wide variety of applications, typically under the assumption that they ideally overcome conscious and unconscious human biases, prejudices, and emotions in decision making. To overcome limitations of our own knowledge and experience, Machine Learning learns concepts from – hopefully unbiased – data and thereby discovers latent knowledge. As such, it has been applied to domains with large amounts of data that are no longer humanly processable and require us to rely, up to a certain degree, on the models trained in automated settings, e.g., credit scoring [9], medical diagnoses [15], or crime risk assessment [7].

In reality, although these models improve in accuracy, the data is often flawed and induces biases in the models that are largely overlooked since the performance is evaluated against equally biased test data. Existing bias mitigation strategies not only require the user to be aware of the bias but also to have a

J. Gama et al. (Eds.): PAKDD 2022, LNAI 13281, pp. 149–160, 2022.
https://doi.org/10.1007/978-3-031-05936-0_12

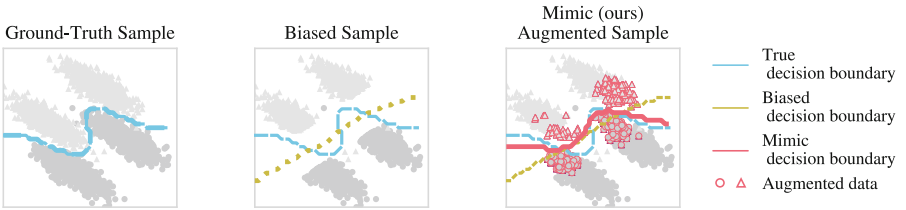

Fig. 1. Decision Boundaries of Support Vector Machines trained on three different datasets: a sample representative for the ground-truth (left), a biased subset (center), and the biased subset augmented with our algorithm, MIMIC (right).

certain knowledge of the ground-truth. But what if the user does not suspect any bias? In this case they will use the data and train a biased model delivering poor performance when applied to previously unseen or underrepresented cases trusting in the quality of its predictions.

Biases are easily induced during the data gathering phase, for example in clinical trials [15] where the data is collected from local volunteers that might not represent the entire population. However, the resulting model will be used to predict the reactions to treatments or drugs for the entire population. Knowledge of the bias early in the development process cannot only help improve the data quality, but can also mitigate its effect on the learned model.

In order to identify and mitigate selection biases where no additional information is available, Dost *et al.* [5] proposed Imitate, a technique that, given a biased dataset, aims to estimate the ground-truth distribution and generate data points to augment the dataset accordingly. While the authors demonstrate Imitate's ability to improve model performance through pre-augmentation on several examples, it is limited by a major assumption: the underlying ground-truth is expected to be normally distributed. In practice, this strongly limits the applicability of Imitate as it is neither flexible enough to model non-Gaussian distributions nor can it capture datasets consisting of several clusters.

In this paper, we introduce MIMIC *(Multi-IMItate Bias Correction)*, a multi-cluster solution for identification and mitigation of selection biases that exploits Imitate as a building block. Modeling data as a mixture of possibly biased and overlapping multivariate Gaussians, MIMIC overcomes Imitate's limitations and greatly increases its applicability. The parameters of these Gaussians bridge between the estimated and the present distribution and can indicate underrepresented regions in the data that are likely to correspond to a selection bias. Generating points in these regions helps mitigate the effect of the bias and pushes the decision boundary towards the ground-truth (see Fig. 1). Our contributions are as follows:

- We propose MIMIC, a novel selection bias identification and mitigation strategy that does not require any knowledge of the bias or the ground-truth.
- In contrast to existing approaches, MIMIC is able to function in a multi-cluster setting and hence drastically increases the range of datasets and distributions that can be modeled.

– In a set of experiments, we demonstrate the shortcomings of existing techniques and highlight the potential of MIMIC in these scenarios. We made our Python+sklearn [16] implementation publicly available[1].

The remainder of this paper is organized as follows: Sections 2 and 3 review the problem statement including the notation and the related research fields, respectively. We introduce our proposed method in Sect. 4 before evaluating it in Sect. 5. Section 6 concludes the paper.

2 Problem Statement

Assuming that we are facing a biased dataset, we aim to generate additional data points that are able to mitigate the bias. Following the notation in [5], this key idea can be formalized as the following problem statement:

Reconstruction Problem. *Let $D \subset \mathbb{R}^n$ be an n-dimensional dataset (potentially with class labels) that is representative of an underlying distribution which we consider to be the ground-truth. Given only a biased subset $B \subset D$, the task is to approximate $I \coloneqq D \setminus B$ with a generated dataset $\hat{I}$ such that a model trained on the augmented dataset $B \cup \hat{I}$ is minimally different from one trained on D.*

The problem was first introduced by Dost *et al.* [5] where D is required to be normally distributed (when split into classes). This assumption is well motivated due to two factors: First, Bareinboim *et al.* [2] prove theoretically that the true class label distribution cannot be recovered from the biased dataset alone without utilising additional data or assumptions, so some assumption is necessary. Second, following the Central Limit Theorem[2], numerical real-world observations frequently are approximately Gaussian which makes normal distributions very common [13]. In this paper, however, we relax the requirement of normal distributions and assume each class of D consists of a mixture of Gaussians. In other words, we assume that each class of the dataset can be represented by a set of possibly overlapping Gaussian clusters.

3 Related Work

Apart from Imitate, to the best of our knowledge, there does not exist any research attempting to solve the problem defined in the previous section. However, methods have been proposed that solve the problem under additional assumptions. This section provides an overview of related research areas.

Bias Mitigation Using Additional Information. If only a subset of the variables is affected by a selection bias, *Missing Value Imputation* techniques [17] can impute these values. For a dataset X with labels Y, however, they

[1] Implementation and Supplementary Material: https://github.com/KatDost/Mimic.

[2] The Central Limit Theorem states that a sequence of independent and identically distributed (i.i.d.) random variables converges almost surely to a Gaussian [10]. Since we can typically assume that real-world measurements are not perfectly i.i.d. but rather combinations of different effects, we will often observe this effect.

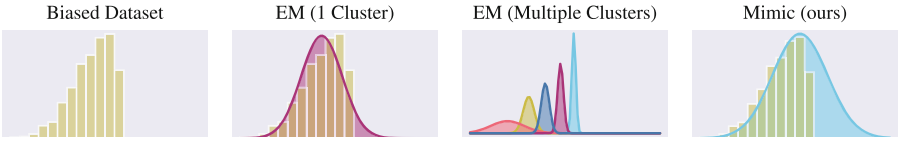

Fig. 2. When facing a biased sample (1st plot from left), the EM algorithm will fit one (2nd) or multiple (3rd; here controlled by BIC) Gaussians to minimize the error on the presented data. Imitate and MIMIC (4th) instead use the histogram bin heights as weights for the fitting procedure and capture the underlying ground-truth more closely.

operate under the assumption that $\mathbb{P}[X|Y]$ and $\mathbb{P}[Y|X]$ are unchanged between the training and the test set. The *Selection Bias* literature [14] widely assumes that all data points in D are known (or at least their distribution), but only a biased subset of the labels is available. More general is the field of *Covariate Shift Correction* [18] where $\mathbb{P}[Y|X]$ is assumed to be shared whereas $\mathbb{P}[X|Y]$ can differ between the training and the test set and will be "shifted". Methods in both fields typically operate model-free and require an unbiased sample to estimate the bias and assign more weight to data points in underrepresented regions during training [20,21]. In the field of *Fairness in Machine Learning*, different techniques to test for biases in models have been proposed, e.g., using the AI Fairness 360 toolkit [3]. These methods require the user to decide which attributes in the dataset might be critical and need to be protected, e.g., gender, and the detected biases can be validated using additional data if possible [19].

If a researcher does not suspect a concrete bias or deals with a numerical tabular dataset without ground-truth information, none of the above mentioned approaches are feasible. Dataset visualization [12] can be considered here, but it is either limited to simple biases or requires inherent bias detection mechanisms to decide upon the kind of visualization, and it detects biases rather than mitigates them. Hence, in the situation of the Reconstruction Problem (see Sect. 2), the Imitate algorithm is, to the best of our knowledge, the only option if neither the ground-truth nor the bias are known.

Imitate. When facing a biased dataset B, Imitate [5] splits it into classes $c \in C$ and treats each resulting subset B_c separately. The dataset B_c is transformed using *Independent Component Analysis (ICA)* [11] to obtain statistically independent components that reveal non-Gaussian densities and allow individual analysis. For each of these components d, the data is represented as a histogram h_d, or using kernel density estimators, and the bin heights are exploited as weights for a least squares optimizer fitting a Gaussian g_d to the histogram. Note that this design puts more emphasis on the existing data points than potentially missing ones and therefore yields fundamentally different results than typical Expectation-Maximization fitting if a selection bias is present (see Fig. 2 for an example). Once all components have been processed, additional data points are generated such that the gaps between g_d and h_d are filled and the distributions g_d are preserved. Then the new points are back-transformed into the original data space. These data points not only indicate a potential selection bias if focussed

on certain areas in the space, but can also be used to augment B_c and mitigate the effect the bias has on subsequent modeling tasks (see [5] for details). Due to the particular design of Imitate that uses ICA for component-wise fitting of one Gaussian, the algorithm is restricted to one normally distributed cluster per class only. In this paper, we relax this restriction.

4 Proposed Method

Aiming to provide a bias mitigation strategy for a wide range of problems, in this paper, we assume that ground-truth data consists of a mixture of multivariate Gaussians. Although this is still a limiting assumption, it substantially widens the range of datasets that can be modeled when compared to existing techniques, i.e., the Imitate algorithm (see Sect. 2 for a discussion of that assumption). Before analyzing each Gaussian for potential biases, we need to find a suitable mixture model for the ground-truth based solely on the biased dataset.

If no bias is present in the dataset, *Gaussian Mixture Models (GMMs)* can fulfill the task as they are able to identify the optimal Gaussians to describe a presented dataset given suitable initial cluster centers. These centers (and the number of clusters) could be found using, for example, the *Bayesian information criterion (BIC)* [8]. In the case of a selection bias, however, one biased cluster might be split into several Gaussian clusters as that mixture fits the presented dataset better, as shown in Fig. 2. Assume a clinical study testing the impact of a new drug on test and control groups. While GMM breaks the group of participants into many small clusters as it models the presented datasets, we need to find clusters that give an indication of where some data might be missing and thereby indicating that, e.g., women below a certain age did not participate due to safety concerns. Therefore, we need to develop a novel strategy to cluster biased datasets into separate potentially overlapping Gaussians that capture the ground-truth rather than the biased presented data.

The central idea for MIMICis simple: We start with a large number of clusters and let Imitate indicate where data might be missing. In contrast to Agglomerative Clustering, we operate on a point-basis rather than by subsequently merging clusters. If data is available in another cluster to fill in the gap, we let the cluster grow by assigning these data points until it is approximately normally distributed or no suitable data points can be found. In this case, we found a potential selection bias and generate data points to mitigate it. Once all initial clusters have been fully grown, a merging procedure purges duplicates and combines suitable clusters to overcome locally optimal solutions. This process is carried out for every class of the initial dataset (if any) separately, but we describe it for only one class in the following in order to simplify. See Algorithm 1 for an overview and the following for a detailed discussion of the components.

Initialization [Algorithm 1; Lines 1–2]**.** Starting with only the biased dataset B, the Initialization step divides it into a large number of initial clusters that MIMIC uses to search each of them for non-normality. It then uses this information to "steal" data points from other clusters into this one and grow it. If the initial

Algorithm 1. MIMIC

Input: biased dataset B
Output: parameters $\theta_i = (\mu_i, \Sigma_i)$ for each cluster i; a set P of generated points to mitigate the bias

▷ remove outliers using LOF
1: $B' \leftarrow$ `removeOutliers`(B)
▷ initialize clustering using KMeans with large K
2: $l \leftarrow$ `initializeClustering`(B')
3: $\theta \leftarrow \emptyset$
4: $L \leftarrow$ `largestValidCluster`(l)
▷ Grow every valid cluster. A cluster is valid if it is large and dense enough and has neither been processed before nor subsumed by a previous iteration
5: **while** L exists **do**
6: $l, \theta_L \leftarrow$ `growCluster`(L, B', l)
7: $\theta \leftarrow \theta \cup \theta_L$
▷ select the largest valid cluster based on the updated labels l (if possible)
8: $L \leftarrow$ `largestValidCluster`(l)
▷ merge clusters if it improves normality
9: $\theta \leftarrow$ `merge`(θ, B')
▷ generate data to mitigate the bias
10: $P \leftarrow$ `augment`(θ, B)
11: **return** θ, P

Algorithm 2. `growCluster`

Input: label L to be grown, outlier-free dataset B' with labels l
Output: updated labels l, parameters θ_L for cluster L

1: **repeat**
2: $B'_L \leftarrow B'|_{l=L}$ ▷ cluster L
▷ run Imitate on L to obtain G_L (grid representing where data might be missing), n_L (number of missing points), θ_L (parameters of the fitted Gaussian)
3: $G_L, n_L, \theta_L \leftarrow$ `Imitate`(B'_L)
▷ score all remaining data points based on if they are likely to help improve the fit of the Gaussian
4: $s \leftarrow$ `score`$(B' \setminus B'_L, G_L, \theta_L)$
▷ identify n_L suitable candidates in batches b_i; sample based on s
5: **for** batches b_i with $\sum_i b_i = n_L$ **do**
6: $C_i \leftarrow$ `sample`$(B' \setminus B'_L, b_i, s)$
▷ assign a batch of candidates to the cluster if it improves the likelihood of the model fitting the data
7: **if** $\mathbb{P}[\theta_L \mid B'_L \cup C_i] > \mathbb{P}[\theta_L \mid B'_L]$ **then**
8: $l(C_i) \leftarrow L$ ▷ update l for accepted C_i
9: **until** l did not change
10: **return** l, θ_L

clusters are already sufficiently normal, no direction for growth can be identified. Therefore, after pre-processing the data with *Local Outlier Factor (LOF)* [4] for higher cluster quality, MIMIC starts off with non-Gaussian initial clusters like those obtained from KMeans. A high number of initial clusters increases the probability that for each true cluster, a less overlapping part is captured in an initial cluster that can later be grown, even if overlaps exist. In order to use a sufficient number of initial clusters, we use twice the number that maximizes the Silhouette score [8], and split further if we detect two density peaks in a histogram instead of one. From here on, the outlier-free dataset is denoted as B' and is passed on to the next step together with the initial labels l.

Identifying Valid Clusters [Algorithm 1; Lines 4, 8]**.** Once a large number of initial clusters has been found, MIMIC grows them into Gaussian clusters where possible using points from B. Aiming to secure reliable performance during the subsequent fitting of a multivariate normal distribution, we filter out all clusters that are either (i) too small (fewer than 10 data points in our implementation) or (ii) too widespread with low density (that is, if the cluster's LOF lies below the 3σ-interval of the average cluster LOF). Note that the latter is a necessary measure as we can expect to obtain unreliable results when fitting a normal distribution to a set of singletons. Additionally, we reduce the computational burden by ensuring that no cluster is grown more than once and no cluster that has been fully subsumed in previous iterations is processed. Thereby, we reduce the number of duplicate clusters we obtain and focus on the most promising ones. Each iteration selects the largest valid cluster and grows it as described below, until no valid clusters remain.

Adapting Imitate to Our Needs [Algorithm 2; Line 3]**.** Given a cluster L, Imitate estimates a multivariate Gaussian (see Sect. 3) and indicates based on a grid where (and how many) points need to be generated in order to smooth out the cluster's density and have it resemble the fitted Gaussian. Note that the Imitate algorithm as described in the original paper continues to operate on the grid representation which would result in a high complexity given our repeated Imitate calls and does not allow for precise probability assignments, hence we adjust: Assume we fitted one Gaussian (μ_i, σ_i^2) per component i' in the ICA-transformed space. Since the components are independent, this results in a multivariate Gaussian with mean $\mu = (\mu_1, \ldots, \mu_d)$ for d dimensions and the covariance matrix $\Sigma \in \mathbb{R}^{d \times d}$ with diagonal $(\sigma_1^2, \ldots, \sigma_d^2)$ and 0 elsewhere. Let $I \in \mathbb{R}^{d \times d}$ be the ICA transformation matrix. The multivariate Gaussian (μ, Σ) can then be back-transformed into the original space and yields the Gaussian $(I^{-1}\mu, I^{-1}\Sigma(I^{-1})^T)$. We refer to Suppl. A for the proofs of both claims.

Additionally, we adjusted Imitate's method of selecting the grid granularity: Instead of repeating the entire modeling and augmentation process and using the results with the highest confidence score (see the original paper), we use the *corrected Akaike Information Criterion (AICc)* [8] (see Suppl. C for additional experiments justifying this choice) to select, for each dimension, the grid over which a histogram represents the data best. This adjustment is necessary since Mimic uses repeated calls of the Imitate fitting procedure and the inflicted computational expense of the confidence-based strategy would be infeasible.

Growing Clusters [Algorithm 2]**.** For a cluster L, Imitate provides us with a multivariate Gaussian θ_L, a grid G_L indicating where and how much (n_L) data might be missing. As outlined in Algorithm 2, both are passed on to a scoring function that estimates for each point p outside L how well it contributes to filling in the gap between the present (h) and fitted (f) density (first term), and how likely it belongs to that distribution (second term):

$$s(p) = d \log[\max\{f(p) - h(p), 0\} + 1] + \log[f(p) + 1]$$

where d denotes the number of features and puts more emphasis on filling the gap for higher dimensions. Using the score, Mimic then searches for n_L fitting candidates in batches b_i to overcome locally optimal solutions. A batch of candidates C_i is drawn randomly with probabilities based on the score values s and added to the cluster if it fulfills $\mathbb{P}[\theta_L \mid B'|_{l=L} \cup C_i] > \mathbb{P}[\theta_L \mid B'|_{l=L}]$, that is, if adding the candidates to the cluster improves the likelihood of the fitted Gaussian given the assigned data points (see Suppl. for the calculations). In our implementation, we restart the sampling (with replacement) of a rejected batch twice in order to avoid "unlucky" choices. If points have been added, Mimic fits another multivariate Gaussian and repeats the process until no further points are added. The parameters of the last fitted Gaussian represent this cluster.

Merging [Algorithm 1; Line 9]**.** Once the parameters for all clusters have been obtained, we make sure not to have duplicate clusters or those that are locally optimally normal but can be combined into a better fit. Additionally, Mimic

risks overgrowing clusters if the initial clustering was particularly poor, e.g., if it captures the overlapping area of two clusters. Here, the point density is higher and the Imitate procedure will demand to grow the cluster in all directions simultaneously such that it never reaches a Gaussian-like shape and continues to grow, absorbing more and more data. Such a cluster L is typically characterized by a very wide probability distribution reaching low density values for all points, such that the points p with $L = \arg\max_i \mathbb{P}[p \mid \theta_i]$ exhibit a substantially larger distance to each other than average and can be detected as such. We identify and remove these overgrown clusters as a first step of the merging procedure.

The overlap of two clusters can be quantified by counting the points in the dataset for which the cluster membership is not entirely clear and weighting them using their probabilities. MIMIC calculates the overlap between each combination of two clusters and merges greedily until no further merge improves the fitting of the Gaussians (see Suppl. A for details).

Data Augmentation [Algorithm 1; Line 10]**.** After receiving the final cluster parameter sets from the merging step, MIMIC probabilistically assigns the data points to the clusters and generates points for each cluster separately to "fill in the gap" between the found and the fitted distribution as in Imitate.

Assumptions and Expectations. Selection Biases cannot be reconstructed without making some kind of assumption regarding the ground-truth and/or the nature of the bias. Hence, MIMIC assumes a ground-truth that can be modeled by a mixture of (possibly overlapping) multivariate Gaussians which, in contrast to existing techniques, requires neither a ground-truth sample nor knowledge of the bias. This freedom, however, comes at a cost and forces some implicit requirements: (i) The data cannot contain categorical, binary, or discrete features with a very small number of values as fitting a Gaussian would not be meaningful, (ii) B itself cannot consist only of Gaussian clusters or MIMIC will not be able to identify growth directions, (iii) several strongly overlapping biased clusters might not be disentangled correctly, and (iv) the bias in each cluster is expected to have a convex shape as our component-wise analysis fails otherwise. Lastly, biases can be misleading pointing towards a different Gaussian than the true one and causing MIMIC to introduce new biases into the data. We aim to suppress that behavior by refusing to take action if the Gaussians do not fit reasonably well (see the Imitate paper for details). This, however, causes conservative results with bias reconstructions pointing towards the right locations rather than correcting entirely which is the reason for only small improvements in classification accuracy (as can be seen in the experimental results). In practice, however, this is enough to point a practitioner towards potential problems in the data that can be corrected upon confirmation.

5 Experiments and Discussion

In order to investigate MIMIC's ability to improve classifier performance, we set up all experiments similarly: we train three classifiers on a biased training set B, the augmented biased training set $B \cup \hat{I}$, and an unbiased training set D.

Fig. 3. For each classification method, we compare the impact of the dataset dimensionality and the number of clusters on the performance.

The accuracy acc_B, $\text{acc}_{B \cup \hat{I}}$, and acc_D of all three classifiers, respectively, is then evaluated on an unbiased test set with the hope that $\text{acc}_{B \cup \hat{I}} > \text{acc}_B$. After providing details on the experimental setup, we assess the impact of different characteristics of datasets on the performance.

Experimental Setup. In our experiments, we compare MIMIC not only to the biased accuracy as a baseline, but also for augmented biased datasets $B \cup \hat{I}$ where $\hat{I}$ is obtained using (i) augmentation with Imitate, (ii) clustering and augmentation with MIMIC, and (iii) clustering with GMM and augmentation with MIMIC which we denote as "GMMimic". GMM selects the number of clusters (from 1 to 20) that achieve the best BIC and initializes using KMeans. As classifiers, we use Decision Trees (DT), Support Vector Machines with RBF-kernel (SVM), and Random Forests (RF) with 100 trees. All parameters are kept at sklearn's default values. We use synthetic datasets since they allow us a high level of control, and real-world datasets to demonstrate that MIMIC is indeed applicable in practice. Real-world datasets are taken from the UCI Machine Learning Repository [1,6,22]. Semi-artificial biases are created as in [5] by splitting into B and I using a decision stump (the larger subset is taken for B). This way, the impact on the classification accuracy is guaranteed (see Suppl. B for details). All synthetic experiments are repeated 30 times to compensate for the randomness in the dataset generation, and we report the median results. Experiments on real-world datasets are repeated 10 times as there is no dataset generation step involved. Here, we report the mean together with 90% confidence intervals. We measure the performance as the *improvement over the biased accuracy* and normalize using the unbiased accuracy, i.e., $(\text{acc}_{B \cup \hat{I}} - \text{acc}_B)/(\text{acc}_D - \text{acc}_B)$.

Unbiased Datasets. Being able to mitigate a selection bias is important, however, if MIMIC is presented with an unbiased dataset, it should not "correct" it. Experiments (Suppl. C) show that substantially fewer data points (none after purging the noise) are generated for the unbiased datasets.

Dimensionality. The dimensionality of synthetic datasets is closely related to their difficulty as higher dimensions naturally increase the distance between clusters even while under the same cluster-to-center distances. Figure 3 demonstrates this, as lower dimensionalities typically exhibit poorer performance than higher ones, but this effect vanishes with larger numbers of clusters. GMMimic and

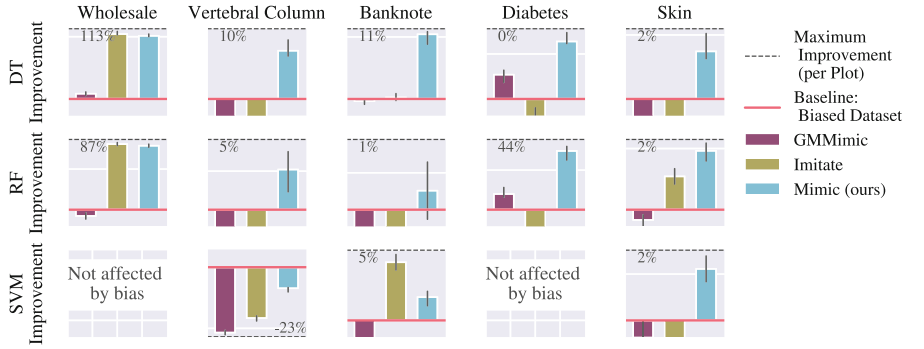

Fig. 4. We compare the degree to which the classifier accuracy can improve when different augmentation techniques are used. The baseline (red line) represents the accuracy when the classifiers are trained on the biased dataset alone. 100% corresponds to training on a ground-truth sample. Note that we omit the y-axes labels and replace them with the dashed line indicating the maximum improvement (maximum y-value) for each plot. The bottom of the plots is cut off unless Mimic's performance is displayed there for an easier comparison. The black lines are 90% confidence intervals and indicate significant differences from the baseline if they do not touch it. (Color figure online)

Mimic show similar performances for a larger number of clusters while Mimic clearly dominates when only a small number of clusters is present, regardless of the dimensionality. Imitate shows strong performance in this case too, but decreases rapidly since it operates with only one cluster.

Cluster Overlap. The center-to-cluster distances directly affect the difficulty of the clustering task as they control the overlap. In experiments (Suppl. C) GMMimic and Mimic both show improvements even for a large number of clusters and high overlaps. Mimic demonstrates its strength particularly for better isolated clusters where it improves the classification accuracy by up to 50%.

Real-Life Datasets. Figure 4 summarizes the results on five real-world datasets. For most datasets, we can see Mimic's potential to improve the classifier accuracy substantially, in most cases more than its competitors. A few observations are noteworthy: On the Wholesale dataset, Imitate performs well since it consists of only one cluster per class. The Vertebral Column dataset seems particularly hard for all methods as the semi-synthetic bias removes 70% of the majority class points (which therefore cannot be reconstructed by any method), leaving an almost balanced classification problem with full overlap and an imbalanced test set. Here, the tree-based methods essentially select the majority class, and Mimic is able to tip the scales favorably, but cannot help the SVM. Overall, although GMMimic demonstrates solid performance on the synthetic dataset, it does not seem to generalize well to the real-world datasets.

Discussion. Overall, the experiments show that application of an augmentation technique can provide a meaningful improvement on a biased dataset. While Imitate is designed for datasets with only one cluster per class, GMMimic and Mimic can improve upon its performance when dealing with multi-cluster datasets. The

experiments on synthetic datasets with artificial biases point towards a similar performance of GMM- and MIMIC-based data augmentation. On the real-world datasets, however, we do not see this confirmed: MIMIC can further improve the classification performance. Further research will investigate where which method tends to be superior and particularly if a symbiosis of both can be beneficial, e.g., with GMM as an initial model and a MIMIC-inspired merging strategy and augmentation.

MIMIC relaxes Imitate's assumption that the ground-truth dataset consists of only one Gaussian per class. Instead, it can model multiple Gaussian clusters or even approximate non-Gaussian clusters with mixture models. This makes MIMIC applicable to a substantially wider range of datasets. However, not all distributions can be approximated well as a mixture of Gaussians. Future extensions should include an automated test of applicability as well as approaches applicable to a wider range of distributions.

6 Conclusion

Machine Learning models inherit selection biases from datasets causing them to predict inaccurately if the biases remain undetected. Existing bias mitigation strategies require certain kinds of knowledge of the bias or the ground-truth. In real-world scenarios, however, this requirement often cannot be met. A first attempt to detect and mitigate selection biases in a "blind" setting has been made with the Imitate algorithm, although it is limited to datasets with only one Gaussian cluster per class.

In this paper, we introduced MIMIC, a technique that uses Imitate as a building block but overcomes these limitations and can model a wider range of datasets exploiting mixtures of Gaussians. As such, multi-cluster modeling of many non-normally distributed datasets is now possible.

Although limitations still exist as discussed in Sect. 5, we believe that MIMIC is a major step forward towards automated bias identification and mitigation in the case that no knowledge of the bias or the ground-truth exists.

References

1. Abreu, N.: Análise do perfil do cliente Recheio e desenvolvimento de um sistema promocional. Mestrado em marketing, ISCTE-IUL, Lisbon (2011)
2. Bareinboim, E., Tian, J., Pearl, J.: Recovering from selection bias in causal and statistical inference. In: Proceedings of the 28th AAAI Conference on Artificial Intelligence, June 2014 (2014)
3. Bellamy, R.K.E., et al.: AI fairness 360: an extensible toolkit for detecting and mitigating algorithmic bias. IBM J. Res. Develop. **63**(4/5), 4:1–4:15 (2019). https://doi.org/10.1147/JRD.2019.2942287
4. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: LOF: identifying density-based local outliers. In: Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, pp. 93–104 (2000). https://doi.org/10.1145/342009.335388
5. Dost, K., Taskova, K., Riddle, P., Wicker, J.: Your best guess when you know nothing: identification and mitigation of selection bias. In: 2020 IEEE International Conference on Data Mining (ICDM), pp. 996–1001. IEEE (2020). https://doi.org/10.1109/ICDM50108.2020.00115

6. Dua, D., Graff, C.: UCI ML repository (2017). http://archive.ics.uci.edu/ml
7. Goel, N., Yaghini, M., Faltings, B.: Non-discriminatory machine learning through convex fairness criteria. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, April 2018 (2018)
8. Granichin, O., Volkovich, Z.V., Toledano-Kitai, D.: Cluster validation. In: Randomized Algorithms in Automatic Control and Data Mining. ISRL, vol. 67, pp. 163–228. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-642-54786-7_7
9. Hassani, B.K.: Societal bias reinforcement through machine learning: a credit scoring perspective. AI Ethics **1**(3), 239–247 (2020). https://doi.org/10.1007/s43681-020-00026-z
10. Hoeffding, W., Robbins, H.: The central limit theorem for dependent random variables. Duke Math. J. **15**(3), 773–780 (1948). https://doi.org/10.1215/S0012-7094-48-01568-3
11. Hyvärinen, A., Oja, E.: Independent component analysis: algorithms and applications. Neural Netw. **13**(4), 411–430 (2000). https://doi.org/10.1016/S0893-6080(00)00026-5
12. Lavalle, A., Maté, A., Trujillo, J.: An approach to automatically detect and visualize bias in data analytics. In: CEUR Workshop Proceedings of the 22nd International Workshop on Design, Optimization, Languages and Analytical Processing of Big Data, vol. 2572. CEUR (2020)
13. Lyon, A.: Why are normal distributions normal? Br. J. Philos. Sci. **65**(3), 621–649 (2014). https://doi.org/10.1093/bjps/axs046
14. Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., Galstyan, A.: A survey on bias and fairness in machine learning. ACM Comput. Surv. **54**(6), 1–35 (2021). https://doi.org/10.1145/3457607
15. Panch, T., Mattie, H., Atun, R.: Artificial intelligence and algorithmic bias: implications for health systems. J. Glob. Health **9**(2), 010318 (2019). https://doi.org/10.7189/jogh.09.020318
16. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
17. Poulos, J., Valle, R.: Missing data imputation for supervised learning. Appl. Artif. Intell. **32**(2), 186–196 (2018). https://doi.org/10.1080/08839514.2018.1448143
18. Rabanser, S., Günnemann, S., Lipton, Z.: Failing loudly: an empirical study of methods for detecting dataset shift. Adv. Neural Info. Process. Syst. **32**, 1396–1408 (2019)
19. Rezaei, A., Liu, A., Memarrast, O., Ziebart, B.D.: Robust fairness under covariate shift. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 9419–9427 (2021)
20. Smith, A.T., Elkan, C.: Making generative classifiers robust to selection bias. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 657–666 (2007). https://doi.org/10.1145/1281192.1281263
21. Stojanov, P., Gong, M., Carbonell, J., Zhang, K.: Low-dimensional density ratio estimation for covariate shift correction. Proc. Mach. Learn. Res. **89**, 3449–3458 (2019)
22. Strack, B., Deshazo, J., Gennings, C., Olmo Ortiz, J.L., Ventura, S., et al.: Impact of HbA1c measurement on hospital readmission rates: analysis of 70,000 clinical database patient records. BioMed Res. Int. **2014**, 781670 (2014). https://doi.org/10.1155/2014/781670