Mixed Models with Multiple Instance Learning

Jan P. Engelmann[*]
Helmholtz Munich

Alessandro Palma[*]
Helmholtz Munich

Jakub M. Tomczak
Eindhoven University of Technology

Fabian J. Theis[†]
Helmholtz Munich

Francesco Paolo Casale[†]
Helmholtz Munich

Abstract

Predicting patient features from single-cell data can help identify cellular states implicated in health and disease. Linear models and average cell type expressions are typically favored for this task for their efficiency and robustness, but they overlook the rich cell heterogeneity inherent in single-cell data. To address this gap, we introduce MixMIL, a framework integrating Generalized Linear Mixed Models (GLMM) and Multiple Instance Learning (MIL), upholding the advantages of linear models while modeling cell state heterogeneity. By leveraging predefined cell embeddings, MixMIL enhances computational efficiency and aligns with recent advancements in single-cell representation learning. Our empirical results reveal that MixMIL outperforms existing MIL models in single-cell datasets, uncovering new associations and elucidating biological mechanisms across different domains.

1 INTRODUCTION

Single-cell omics data have been instrumental in unveiling cellular heterogeneity, proving invaluable in studying human health and disease (Perez et al., 2022; Ahern et al., 2022; Vandereyken et al., 2023; Lim et al., 2023). Within these vast datasets, determining which cells are impacted by specific interventions or genetic variations is of utmost importance. However, drawing such associations at the single-cell level is statistically challenging, given the sparse nature of the data and the structured noise introduced by cellular dependencies within a sample (You et al., 2023; Cuomo et al., 2023). Traditional pooling procedures, such as the pseudo-bulk approach in single-cell RNA sequencing, sidestep these challenges but at the expense of overlooking key cell states (Perez et al., 2022; Ahern et al., 2022; Yazar et al., 2022; Janssens et al., 2021; Lafarge et al., 2019; Ljosa et al., 2013).

Multiple instance learning provides a principled framework for modeling cellular heterogeneity through attention and has proven effective in several domains (Ilse et al., 2018; Javed et al., 2022; Li et al., 2021; Cui et al., 2023b; Shao et al., 2021). Yet, its application to single-cell datasets has been limited. This might be attributed to the characteristic low signal-to-noise ratio of these datasets, a setting where simple models based on mean pooling and linear assumptions tend to perform adequately (Crowell et al., 2020).

To bridge this gap, we introduce Mixed Models with Multiple Instance Learning (MixMIL), a new framework integrating the robustness of the Generalized Linear Mixed Model (GLMM) with the MIL ability to model heterogeneity. Designed for robustness and efficiency in single-cell analyses, MixMIL leverages cell embeddings from pre-trained unsupervised models (Theodoris et al., 2023; Cui et al., 2023a; Doron et al., 2023) and integrates a simple attention-based MIL module into GLMMs, synthesizing the strengths of both frameworks. In extensive simulations and evaluations, we benchmarked MixMIL against the GLMM and state-of-the-art MIL architectures. Across a spectrum of applications, spanning single-cell genomics to microscopy and reaching into histopathology, MixMIL consistently outperformed other MIL implementations, underscoring that reduced complexity through principled model design can notably enhance results in these contexts.

Proceedings of the 27[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238.

[*]Equal contribution. [†]Correspondence to: {francescopaolo.casale, fabian.theis}@helmholtz-munich.de

2 RELATED WORK

Multiple Instance Learning MixMIL models cellular heterogeneity within the GLMM framework using MIL. MIL organizes data into collections of instances, called *bags*, and operates on the premise that the label of the collection is known while the labels of individual instances remain unknown. In attention-based MIL (ABMIL), Ilse et al. (2018) trained a bag-level classification model where a neural network learns the influence of single instances on the bag label prediction in the form of attention weights. Later efforts built upon the notion of attention-based deep MIL to improve the interpretability of instance contribution (Javed et al., 2022) and alleviate performance degradation due to class imbalance (Li et al., 2021). More related to our setting, Cui et al. (2023b) used variational inference to estimate the posterior distribution of instance-level weights to enhance model interpretability and uncertainty estimation. In the medical field, MIL has been widely employed to study the heterogeneous effect of disease in large histopathology images in the context of cancer prediction (Ilse et al., 2018; Sudharshan et al., 2019; Zhao et al., 2020; Wagner et al., 2023), segmentation (Lerousseau et al., 2020) and somatic variant detection (Cui et al., 2020). In contrast to earlier approaches, MixMIL uniquely integrates attention-based MIL within GLMMs.

Generalized Linear Mixed Models MixMIL introduces multiple instance learning within the GLMM framework. GLMMs extend linear models with random effects to enable robust regression and hierarchical modeling, and they are equipped to handle a variety of outcome distributions (Breslow and Clayton, 1993; Bates et al., 2014). Specialized GLMMs have become indispensable in genomics, especially in association analysis (Lippert et al., 2011; Bates et al., 2014; Loh et al., 2018) and interaction testing (Casale et al., 2017; Moore et al., 2019; Dahl et al., 2020). In recent advancements, GLMM-based interaction tests have been employed to model cell state heterogeneity in single-cell datasets (Neavin et al., 2021; Cuomo et al., 2022; Gewirtz et al., 2022; Nathan et al., 2022). These tests primarily focus on associating singular patient and genomic features while modeling effect heterogeneity—for instance, exploring how the effect of a genetic variant might regulate the expression of an individual gene based on cell state (Cuomo et al., 2022). In contrast, MixMIL seeks to characterize single patient features using the cell state representations from a group of cells, uniquely incorporating a MIL module in GLMMs for this purpose.

Predefined Embeddings and Single-Cell Atlases MixMIL employs shallow machine learning functions on predefined embeddings for robustness and efficiency. This synergy was first spotlighted in representation learning for computer vision with frameworks such as SimCLR (Chen et al., 2020) and later extended by others (He et al., 2020; Caron et al., 2021). Rapidly, this influence radiated across biological disciplines, advancing representational learning in computational pathology (Ben Taieb and Hamarneh, 2020; Wang et al., 2023), single-cell genomics (Lopez et al., 2018; Theodoris et al., 2023; Cui et al., 2023a), and microscopy (Marin Zapata et al., 2020; Siegismund et al., 2022). While the use of predefined embeddings in MIL has been considered elsewhere (Li et al., 2021; Shao et al., 2021), the synergy with MixMIL is especially timely within the single-cell omics sphere. As the domain leans towards foundational models for comprehensive single-cell atlases (Schiller et al., 2019; Travaglini et al., 2020; Deprez et al., 2020; Wagner et al., 2019; Wilk et al., 2020; Sikkema et al., 2022), MixMIL stands primed, ready to leverage the wealth of emerging high-quality embeddings, equipping researchers for robust, integrated analyses.

3 METHODOLOGY

3.1 Problem Statement

Multiple instance learning is a variation of supervised learning where the training set consists of labeled bags, each containing several instances. Formally, a bag associated with a single label y consists of I unordered and independent instances $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_I\}$, where $\boldsymbol{x}_i \in \mathbb{R}^Q$. We here collectively denote these instances with $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_I]^T \in \mathbb{R}^{I \times Q}$. Notably, the number of instances I can vary across different bags. The primary goal of MIL is to predict the label y from the bag of instances $\boldsymbol{X}$, using a function that is invariant to permutations among instances. This problem could be solved by a model that defines instance embeddings through a function f, $f(\boldsymbol{X}) = \{f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_I)\}$, and then aggregates them into a single bag embedding $\boldsymbol{z}$, which is eventually fed to a predictor. In the following, we focus on two approaches that will serve as an inspiration for our method. First, we present the attention-based deep MIL framework (Ilse et al., 2018) that utilizes deep neural networks and the attention mechanism for aggregation. Second, we outline a GLMM (Breslow and Clayton, 1993), an extension of linear models widely used in genomic analyses.

3.2 Attention-Based MIL

Ilse et al. (2018) introduced an innovative pooling function for MIL, implementing the concept of attention to aggregate instance-level features into bag-level ones. Specifically, they first introduce a neural network func-

tion f to derive low-dimensional instance embeddings from instance features $\boldsymbol{x}_i$ and then use a weighted average pooling function equivalent to the *attention mechanism* to aggregate $\{f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_I)\}$ into bag embeddings $\boldsymbol{z}$. Finally, they consider a classifier to predict the bag label y from $\boldsymbol{z}$. The weighting function was defined as follows:

$$\boldsymbol{z} = f(\boldsymbol{X})^T \boldsymbol{w}, \quad \text{with } w_i > 0 \; \forall i \text{ and } \Sigma_i w_i = 1, \tag{1}$$

where $\boldsymbol{w} \in \mathbb{R}^I$ denotes the vector of importance weights across the I instances. Modeling the importance weights as a two-layer neural network function with softmax activation function on the last layer, $\omega(f(\boldsymbol{X}))$, they can ensure that the constraints on the weights are always satisfied while the entire architecture can be trained end-to-end—i.e., the function f, the importance weight function ω and the bag-level classifier can be jointly optimized.

Following the foundational work by Ilse et al. (2018), more advanced implementations of attention-based MIL have emerged. For example, DSMIL computes attention weights for each observation in a bag based on their similarity with the instance which has the highest classification score for a certain class (Li et al., 2021). Attention weights are then used to aggregate features. In multiclass classification, each class has its own critical instances, which allows DSMIL to use different importance weighting for different classes. Moreover, Cui et al. (2023b) suggested using Bayesian neural networks for attention-based MIL. Such an approach optimizes a posterior on the parameters of the attention function via variational inference and yields calibrated uncertainties for better weight interpretability.

3.3 Generalized Linear Mixed Model for MIL

We can employ a GLMM in the context of MIL to model the relationship between the bag label y and fixed bag embeddings $\boldsymbol{z}(\boldsymbol{X})$ derived from bag $\boldsymbol{X}$ while accounting for bag covariates $\boldsymbol{c}$. Specifically, given a link function g, the expected value of the bag label, $\mu = \mathbb{E}[y|\boldsymbol{X}]$, is linked to a linear predictor of bag embeddings $\boldsymbol{z}(\boldsymbol{X})$ and covariates $\boldsymbol{c}$ through

$$g(\mu) = \boldsymbol{c}^T \boldsymbol{\alpha} + \boldsymbol{z}(\boldsymbol{X})^T \boldsymbol{\beta}, \tag{2}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^K$ and $\boldsymbol{\beta} \in \mathbb{R}^Q$ denote the effects of the K covariates and Q-dimensional bag embeddings, respectively. In the case of high-dimensional bag embeddings, $\boldsymbol{\beta}$ is modeled as a random effect to improve robustness. We note that for average pooled bag embeddings, which are common in single-cell omics analysis (Perez et al., 2022; Ahern et al., 2022; Yazar et al., 2022; Janssens et al., 2021; Lafarge et al., 2019; Ljosa et al., 2013), we have $\boldsymbol{z}(\boldsymbol{X}) = \frac{1}{I} \sum_{i=1}^{I} f(\boldsymbol{x}_i)$.

3.4 Our Approach: The MixMIL Model

We propose a novel integration of the attention-based MIL and GLMM frameworks, as illustrated in Figure 1. Our approach encompasses two main steps: (i) Utilize predefined instance embeddings and model the importance weights as a shallow function of them; (ii) Replace the static pooling function in standard GLMMs (2) with a dynamic, trainable attention-based pooling function that leverages the aforementioned importance weights.

3.4.1 The Model

Predefined Embeddings and Shallow Attention Weight Function Leveraging insights from recent advances in representation learning (see Section 2), we employ predefined instance embeddings from domain-specific unsupervised models as instance features, bypassing the need for end-to-end optimization of a feature extractor f. These embeddings are ubiquitously available across various data modalities (see Section 2). Additionally, we model instance importance weights using a single linear layer with a softmax activation function across instances. With these assumptions, the bag embeddings can be written as follows:

$$\boldsymbol{z}_{\boldsymbol{\gamma}}(\boldsymbol{X}) = \boldsymbol{X}^T \omega_{\boldsymbol{\gamma}}(\boldsymbol{X}) \in \mathbb{R}^Q, \tag{3}$$

$$\text{with } \omega_{\boldsymbol{\gamma}}(\boldsymbol{X}) = \text{softmax}\left(\boldsymbol{X}\boldsymbol{\gamma}\right) = \text{softmax}\left(\begin{bmatrix} \boldsymbol{x}_1^T \boldsymbol{\gamma} \\ \vdots \\ \boldsymbol{x}_I^T \boldsymbol{\gamma} \end{bmatrix}\right) \in \mathbb{R}^I,$$

where $\boldsymbol{X} \in \mathbb{R}^{I \times Q}$ denotes the predefined embeddings across all instances, and we made explicit that both the bag embeddings $\boldsymbol{z}$ and the weight function ω depend on the parameters $\boldsymbol{\gamma}$. This way of aggregating bag embeddings is an *attention mechanism* as defined by Ilse et al. (2018) in Eq. (1).

Modeling Dependencies To model the relationship between the bag label y and bag embeddings $\boldsymbol{z}_{\boldsymbol{\gamma}}(\boldsymbol{X})$ defined in Eq. (3), we consider the GLMM formulation for MIL in Eq. (2):

$$g(\mu) = \boldsymbol{c}^T \boldsymbol{\alpha} + \boldsymbol{z}_{\boldsymbol{\gamma}}(\boldsymbol{X})^T \boldsymbol{\beta}, \tag{4}$$

where now the bag pooling function $\boldsymbol{z}_{\boldsymbol{\gamma}}(\boldsymbol{X})$ (specified in Eq. (3)) is dynamic and end-to-end trainable. To ensure robust regression for small sample sizes or higher-dimensional instance embeddings, we model both $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ as random effects, i.e., we introduce the priors $\boldsymbol{\beta} \sim \mathcal{N}(\boldsymbol{0}, \sigma_\beta^2 \boldsymbol{I}_{Q \times Q})$ and $\boldsymbol{\gamma} \sim \mathcal{N}(\boldsymbol{0}, \sigma_\gamma^2 \boldsymbol{I}_{Q \times Q})$. Here, $\boldsymbol{I}_{Q \times Q}$ denotes the $Q \times Q$ identity matrix, and σ_β^2 and σ_γ^2 are the variances associated with the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$. Given these priors, namely:

$$\text{p}(\boldsymbol{\beta}) = \mathcal{N}\left(\boldsymbol{\beta} \,|\, \boldsymbol{0}, \sigma_\beta^2 \boldsymbol{I}_{Q \times Q}\right) \tag{5}$$

$$\text{p}(\boldsymbol{\gamma}) = \mathcal{N}\left(\boldsymbol{\gamma} \,|\, \boldsymbol{0}, \sigma_\gamma^2 \boldsymbol{I}_{Q \times Q}\right) \;, \tag{6}$$

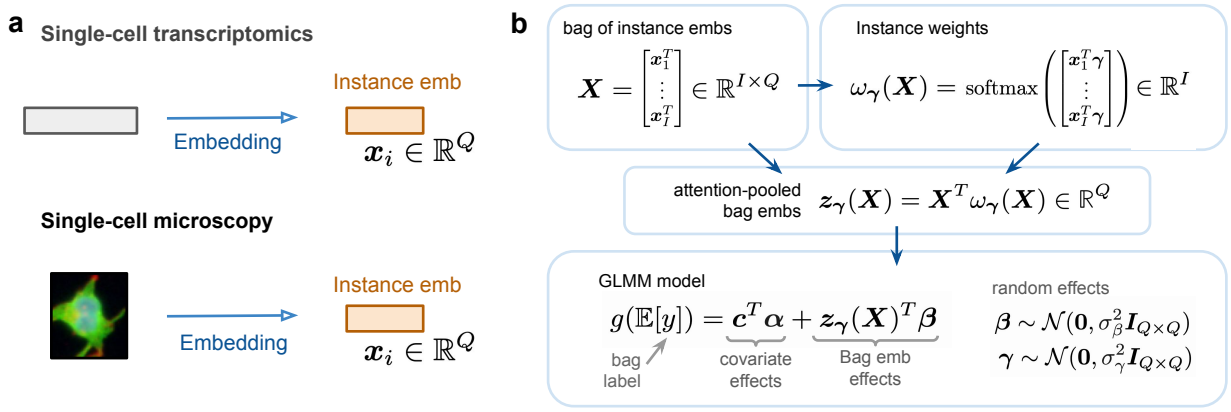


Figure 1: (**a**) MixMIL uses predefined instance embeddings from domain-specific unsupervised models for robustness and efficiency. (**b**) Generalized multi-instance mixed model framework defining MixMIL.

the marginal likelihood of the model is the following:

$$\mathrm{p}(y|\boldsymbol{c}, \boldsymbol{\alpha}, \boldsymbol{X}) = \int \mathrm{p}\left(y \left| \boldsymbol{c}^T\boldsymbol{\alpha} + \boldsymbol{z}_{\boldsymbol{\gamma}}(\boldsymbol{X})^T\boldsymbol{\beta}\right.\right) \mathrm{p}(\boldsymbol{\beta})\mathrm{p}(\boldsymbol{\gamma})\mathrm{d}\boldsymbol{\beta}\mathrm{d}\boldsymbol{\gamma} \tag{7}$$

This integral is generally intractable, but it can be approximated using various techniques such as Monte Carlo methods, Laplace approximation, or variational inference. In this work, we opt to use variational inference.

Instance Importance Heterogeneity The parameter σ_γ^2 primarily controls the heterogeneity of the importance weights across instances. Indeed, when $\sigma_\gamma^2 = 0$, we have $\boldsymbol{\gamma} = \mathbf{0}$, and $\boldsymbol{z_0}(\boldsymbol{X})$ reduces to a simple average across all instances. In this case, MixMIL simplifies to a standard GLMM with average pooled bag features. Conversely, larger values of σ_γ^2 correspond to a more significant disparity in importance weights across instances, with only a few instances contributing the most to bag label predictions.

Model Interpretability Given that both the predictor from bag embeddings and the pooling function are linear in the instance embeddings, the aggregate effect of bag embeddings on bag labels can be expressed as follows:

$$\boldsymbol{z}_{\boldsymbol{\gamma}}(\boldsymbol{X})^T\ \boldsymbol{\beta} = \omega_{\boldsymbol{\gamma}}(\boldsymbol{X})^T\boldsymbol{X}\boldsymbol{\beta} = \omega_{\boldsymbol{\gamma}}(\boldsymbol{X})^T t_{\boldsymbol{\beta}}(\boldsymbol{X}), \tag{8}$$

where $t_{\boldsymbol{\beta}}(\boldsymbol{X}) = \boldsymbol{X}\boldsymbol{\beta} \in \mathbb{R}^I$ can be viewed as the vector of instance-level phenotypic predictions. This interpretable formula provides a way to decompose the overall bag prediction into a weighted sum of instance-level contributions, offering insights into the individual instance influences on the final prediction.

3.4.2 Inference

The aim in inference is to determine the posterior distribution $\mathrm{p}(\boldsymbol{\theta} \,|\, \mathcal{D})$ of the random effect parameters $\boldsymbol{\theta} = \{\boldsymbol{\beta}, \boldsymbol{\gamma}\}$ given the observed data $\mathcal{D}$. As exact inference is intractable for our model, we resort to variational inference, a strategy that approximates the true posterior $\mathrm{p}(\boldsymbol{\theta} \,|\, \mathcal{D})$ by introducing a variational family $\mathrm{q}_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ parameterized by $\boldsymbol{\phi}$, and optimizing $\boldsymbol{\phi}$ to maximize the Evidence Lower Bound (ELBO):

$$\mathrm{ELBO}(\boldsymbol{\phi}, \sigma_\beta^2, \sigma_\gamma^2) = \mathbb{E}_{\mathrm{q}_{\boldsymbol{\phi}}(\boldsymbol{\theta})}\left[\log \mathrm{p}(\mathcal{D}|\boldsymbol{\theta})\right] - D_{\mathrm{KL}}\left(\mathrm{q}_{\boldsymbol{\phi}}||\mathrm{p}\right). \tag{9}$$

Here $D_{\mathrm{KL}}\left(q_{\boldsymbol{\phi}}||\mathrm{p}\right)$ denotes the Kullback-Leibler divergence between the variational approximation $\mathrm{q}_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ and the prior distribution of the parameters $\mathrm{p}(\boldsymbol{\theta})$.

We here consider the variational family of multivariate Gaussian distributions with full rank covariance:

$$\mathrm{q}_{\boldsymbol{\phi}}(\boldsymbol{\theta}) = \mathcal{N}\left(\boldsymbol{\theta} \,|\, \boldsymbol{\mu}_{\boldsymbol{\phi}}, \boldsymbol{\Sigma}_{\boldsymbol{\phi}}\right), \tag{10}$$

parameterized by $2Q$ mean parameters and $Q(2Q+1)$ covariance parameters. In simulations, we also explore a mean field variational family, which assumes a fully factorized posterior across parameter dimensions—leading to $2Q$ mean parameters and $2Q$ variance parameters.

Optimization We jointly optimize the ELBO with respect to fixed effects $\boldsymbol{\alpha}$, variational parameters $\boldsymbol{\phi}$, and prior hyperparameters σ_β^2 and σ_γ^2 using mini-batch gradient descent. This optimization strategy, aligning with the empirical Bayes method (Carlin and Louis, 2000), adjusts the prior distributions in response to observed data. In order to backpropagate through the expectation term in the ELBO, we sample from the variational posterior and utilize the reparameterization trick (Ranganath et al., 2014). In our experiments, we found that using the Adam optimizer with a learning rate of 10^{-3}, a training batch size of 64 bags, and 8 posterior samples to approximate the expectation, produced robust results across all settings.

3.4.3 Predictive Posterior

After optimization, we employ the learned approximate posterior $q(\boldsymbol{\beta}, \boldsymbol{\gamma})$ to predict the label of a new bag from its instance embeddings $\boldsymbol{X}^\star$:

$$\boldsymbol{y}^\star = \mathbb{E}_{q(\boldsymbol{\beta},\boldsymbol{\gamma})}\left[\omega_{\boldsymbol{\gamma}}(\boldsymbol{X}^\star)^T t_{\boldsymbol{\beta}}(\boldsymbol{X}^\star)\right], \tag{11}$$

where we used the formulation in Eq. (8). Moreover, to retrieve important instances, we can leverage the expected value of the importance weights, namely $\mathbb{E}_{q(\boldsymbol{\beta},\boldsymbol{\gamma})}\left[\omega_{\boldsymbol{\gamma}}(\boldsymbol{X}^\star)\right]$.

3.4.4 Likelihood Choices in Experiments

Genotype Labels For genotype labels, where the label represents the minor allele count with possible values in the set {0, 1, 2}, we employ a Binomial likelihood with two trials (Hao et al., 2016). The linear predictor operates in the logit space, and the resulting probability for the binomial distribution is derived from the sigmoid function applied to the logits.

Multiclass Classification For the multiclass classification problem in the microscopy dataset, we employ a categorical likelihood. Specifically, given C classes, we use parameters $\boldsymbol{\alpha} \in \mathbb{R}^{K\times C}$, $\boldsymbol{\beta} \in \mathbb{R}^{Q\times C}$, and $\boldsymbol{\gamma} \in \mathbb{R}^{Q\times C}$ to specify class-specific covariate, feature, and attention effects. The aggregated predictions as per Eq. (11) then produce a C-dimensional logit vector. The class probabilities are then derived from the logit vector using the softmax link function. With this approach, MixMIL can learn different attention mechanisms for different classes. We employ the same prior for feature and attention effects across all classes as outlined in Eq. (5-6).

Binary Classification For the histopathology classification task, we use a Bernoulli likelihood. The linear predictor operates in the logit space, and the probability is determined by the sigmoid function applied to the logits.

3.4.5 Implementation and Complexity

Implementation[1] To facilitate efficient training and inference on both GPU and CPU, we implemented MixMIL using PyTorch. This choice also enabled us to leverage the numerous probability distributions already available in Pytorch for our generalized likelihood framework. For efficient computation of bag-level operations across all bags (e.g., bag-level softmax), we utilized the PyTorch Scatter library. Importantly, our code supports the simultaneous analysis of multiple labels, which we make efficient by tensorizing computations across outcomes.

[1] https://github.com/AIH-SGML/MixMIL

Model Size and Complexity MixMIL employs single linear layers for importance labels and predictions, resulting in $2Q + K$ likelihood parameters, where Q represents the number of instance features and K denotes the number of bag covariates. This is significantly fewer than MIL baselines like ABMIL, DSMIL, and BayesMIL (Table 1). However, we note that MixMIL's variational posterior can notably exceed the likelihood's parameter count, especially with the multivariate Gaussian variational posterior having $2Q + Q(2Q + 1)$ parameters.

4 EXPERIMENTS

We demonstrate the utility of MixMIL by applying it to the task of making predictions for unseen bags of instances. After benchmarking our model in extensive simulations, we selected three applications from diverse domains: (i) predict genetic labels from transcriptional cell embeddings, (ii) predict a compound's Mode of Action (MoA) from morphological cell embeddings, (iii) classify histology slides between cancer vs healthy from histological patch embeddings.

4.1 Methods Considered

MIL Models We considered the established ABMIL (Ilse et al., 2018) and its variation Gated ABMIL. Additionally, we included recently published methods DSMIL (Li et al., 2021) and Bayes-MIL (Cui et al., 2023b). For the MoA prediction tasks, we also considered Additive MIL (Javed et al., 2022), given its reported improved performance in multiclass prediction tasks. For full details on hyperparameter sweeping and selection across the different experiments, see Appendix C.

Traditional ML Models In simulation studies, we also benchmarked MixMIL against a GLMM with a corresponding likelihood and two widely used nonlinear models: Random Forest and XGBoost. All models have predefined bag-level features as inputs. For this, we explored three different strategies: mean pooling, median pooling, and a combination of mean, squared mean, and cubed mean of instance features. Detailed implementation specifics and comparisons with such models on the simulation setting are provided in Appendix C.1 and E.1. We initially also tried to train these models directly on instance features by assigning bag labels to the corresponding instances. However, these models were too slow and underperformed (see Table E.1) and thus were excluded from further comparisons.

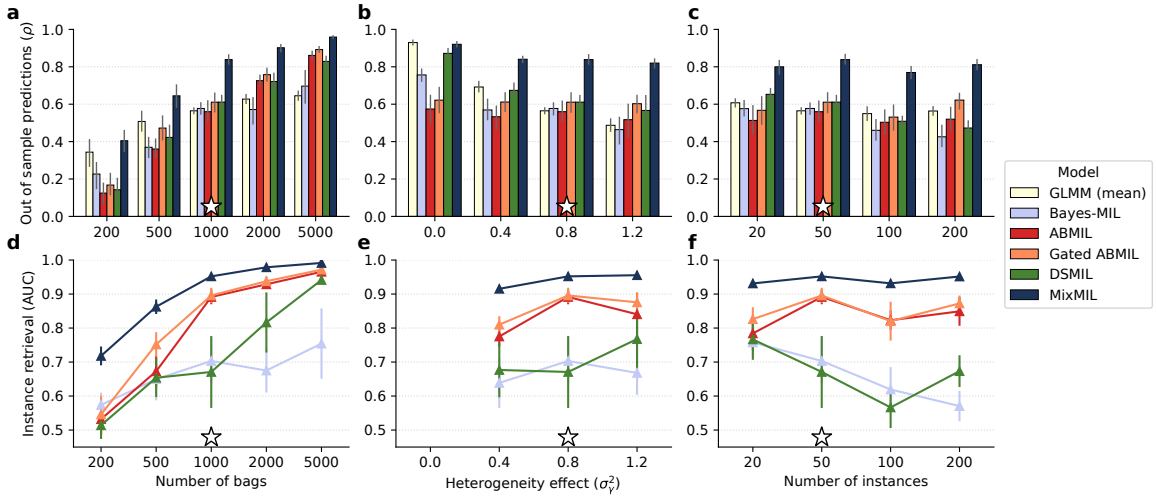


Figure 2: (a-c) Out-of-sample prediction accuracy (Spearman correlation, ρ) for MixMIL, GLMM, and baseline MILs (ABMIL, Gated ABMIL, DSMIL, and Bayes-MIL) varying the sample size (a), the amount of instance importance heterogeneity (b) and the number of instances (c). (d-f) Instance retrieval ROC-AUC of MixMIL and baseline MILs for the top 10% of instances in the same simulated scenarios. GLMM is not shown as it is not designed for instance retrieval. Stars denote default values that were kept constant while varying other parameters. Error bars denote standard errors across 10 repeat experiments. Full results across all methods and scenarios can be found in Section E.1.

4.2 Simulations

Dataset Generation We designed our simulation to emulate the single-cell genomics application. First, we generated instance embeddings $\boldsymbol{X}_{iq} \sim \mathcal{N}(0, 1)$, importance weights ω_γ with $\boldsymbol{\gamma}_q \sim \mathcal{N}(0, \sigma^2_\gamma)$, and bag embedding effects $\boldsymbol{\beta}_q \sim \mathcal{N}(0, \sigma^2_\beta)$. Next, we employed the model in Eq. (4) to generate bag-level logits. Finally, we generated genotypes using a binomial likelihood function with 2 trials taking as input the simulated logits (see also Section 3.4.4). We systematically varied parameters such as the sample size, instance importance heterogeneity (σ^2_γ), the number of instances per bag, the number of instance features, and the variance explained by bag embedding effects. For each parameter configuration, we ran 10 repeat experiments.

Setup Out-of-sample prediction accuracy for all models was measured using Spearman's rank correlation between the actual bag logits and the predicted values in a simulated test set of 200 bags. To evaluate the effectiveness of the MIL models to retrieve the most important instances using the weight posterior, we used ROC-AUC for the top 10% of the simulated instances. Standard errors on all metrics were computed across the 10 repeat experiments.

Results When evaluating models across varying sample sizes, we noticed variable relative performance of the compared models: while the GLMM was superior to baseline MILs at lower sample sizes, baseline MILs gradually improved with more samples, all surpassing the GLMM at around 2,000 bags (Figure 2a). In contrast, MixMIL outperformed all baselines throughout (Figure 2a), emphasizing its reliability in settings where traditional MIL models might be prone to overfitting. As we increased the instance importance heterogeneity, we noted a sharp downturn in the performance of GLMM (Fig 2b), a trend also observed in other conventional ML models (Figure E.1(ii)). In contrast, MIL models maintained their accuracy more effectively. When varying the number of instances per bag, the number of instance features, and the variance attributed to bag embedding effects, MixMIL consistently outperformed baselines (Figure 2c, Figure E.1). We also compared MixMIL with a version utilizing a mean field posterior. As the latter exhibited slightly diminished performance (Figure E.1(i) and E.2), it was not considered in the real data analyses. To conclude, in instance retrieval tasks, MixMIL consistently outperformed baseline MILs, accurately retrieving the top simulated instances (Figure 2d-f).

4.3 Single-Cell Genomics Dataset

Task We here consider the task of predicting genetic variants from transcriptional cell embeddings in the OneK1K dataset (Yazar et al., 2022). This task carries biological significance: Identifying genetic variants associated with cellular transcriptional states can pinpoint cellular processes implicated in health and disease (Consortium, 2017; Westra et al., 2013).

Dataset The OneK1K dataset comprises single-cell RNA sequencing (scRNA-seq) data from approximately 1.3 million peripheral blood mononuclear cells, derived

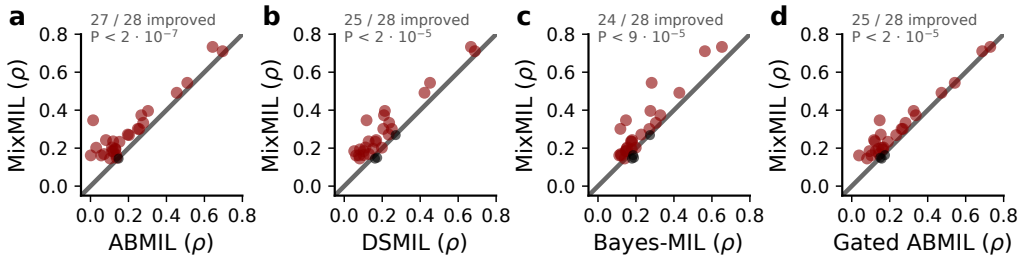


Figure 3: Scatter plots comparing the prediction performance (Spearman correlation, ρ) of MixMIL (y-axis) against baseline MILs (x-axis) for 28 genetic labels: MixMIL vs ABMIL (a), MixMIL vs DSMIL (b), MixMIL vs Bayes-MIL (c), MixMIL vs Gated ABMIL (d). Genetic labels for which MixMIL yielded improved prediction accuracy are highlighted in red. The count of these genes and the P-values from a binomial test (assuming a null of 50/50 performance chance over 28 trials) are reported for each comparison.

from 982 genotyped donors. For our analysis, we considered the sub-lymphoid cells (CD4, CD8, NK cells), spanning cells sharing core lymphoid pathways yet exhibiting distinct functionalities. Regarding genetic labels, we focused on independent variants associated with the average transcriptional state (see next paragraph). The final dataset for our analysis consisted of 981 individuals (bags), 1.1M cells (instances[2]), and 28 genetic labels.

Setup Cell state embeddings were obtained from single-cell expression data using single-cell Variational Inference (scVI) with 30 latent factors (Lopez et al., 2018), a deep generative model for cell state representation learning (see Appendix D.2 for full details). To identify the set of independent genetic labels for our task, we processed 16,718 variants known as cis-eQTLs, using a series of criteria including their association with average transcriptional state and linkage equilibrium considerations. This procedure yielded 28 distinct variants, which we used as labels when comparing alternative MIL implementations. A detailed methodology on this selection process can be found in Appendix D.1. For all models, we controlled for sex and age, and additionally accounted for population structure by using the four leading principal components of genetic data. We used MixMIL's fixed effects for this purpose and regressed these covariates from the data for the baseline MILs. Out-of-sample prediction performance for MIL models was computed utilizing a 5-fold stacked cross-validation procedure. Briefly, we concatenated the out-of-sample predictions on each test fold, forming a single prediction vector for all samples. We then correlated this prediction vector with the observed data using Spearman correlation.

Results MixMIL demonstrated a consistent enhancement in performance compared to MIL baselines across the majority of the 28 genetic labels (Figure 3). Specifically, MixMIL outperformed ABMIL for 27 out of 28 labels ($P< 2 \cdot 10^{-7}$, from a binomial test; Figure 3), DSMIL for 25 out 28 labels ($P< 2 \cdot 10^{-5}$), Bayes-MIL for 24 out of 28 labels ($P< 9 \cdot 10^{-5}$), and Gated ABMIL for 25 out of 28 labels ($P< 2 \cdot 10^{-5}$). Notably, in addition to improved prediction performance, MixMIL showed a marked reduction in running time, being over 12× faster than ABMIL and over 40× faster than Bayes-MIL (Table 1). Relatedly, MixMIL also has a lower complexity than other MIL models, utilizing less than 7% of the parameters of DSMIL and under 2.4% of the parameters in Bayes-MIL (Table 1). Finally, we leveraged MixMIL's instance retrieval to delve deeper into the transcriptional cell states that are most predictive of specific genetic labels. Notably, some of these states corresponded with known cell types, while others revealed novel biological insights (Figure E.3).

Table 1: Running times and number of parameters for MixMIL and baseline MILs on the genetics dataset. Specifically, we report batch training times (ms) and prediction times (ms) benchmarked on a V100 GPU with 32GB memory, alongside counts of likelihood and variational parameters.

Method	batch time (ms)	predict time (ms)	lik pars (#)	var pars (#)
Bayes-MIL	5.65 ± 0.02	16.55 ± 0.30	2883	1890
ABMIL	1.72 ± 0.02	0.21 ± 0.01	1922	-
Gated ABMIL	2.02 ± 0.02	0.25 ± 0.01	2852	-
DSMIL	1.99 ± 0.06	0.39 ± 0.02	992	-
MixMIL	$\mathbf{0.14 \pm 0.01}$	$\mathbf{0.04 \pm 0.01}$	67	1890

4.4 Microscopy Dataset

Task We considered the task to predict a compound's MoA[3] from morphological cell embeddings using the microscopy-based drug screening dataset BBBC021 (Caie et al., 2010). Accurate MoA classification is critical in drug discovery as it expedites

[2]The number of cells per donor ranged from 139 to 2587

[3]A compound's MoA refers to the specific biological process affected by the compound

the understanding of drug effects. As not all cells in culture respond uniformly to the same perturbation, using MIL to model response heterogeneity can improve MoA prediction accuracy and offer insight into phenotypic responses.

Dataset The BBBC021 dataset contains microscopy images of MCF7 breast cancer cell lines treated with 113 compounds for 24 hours (Caie et al., 2010). Following Ljosa et al. (2013), we focus on 39 compounds with a visible impact on cell morphology, which was associated with 12 distinct MoA labels. The experiments were run on plates with 96 wells, each containing multiple cells. All cells in a well were perturbed by the same compound, and the same compound was used to perturb multiple wells on a single plate and multiple replicate plates. Within this setup, we have 2,526 wells (bags), 133,628 cells (total number of instances), and 12 MoAs (labels).

Setup We extracted morphological cell embeddings from single-cell images using the pre-trained ResNet50 model proposed by Perakis et al. (2021), which was trained on the same dataset using the self-supervised SimCLR framework (Chen et al., 2020). For all models, we used the leading 256 principal components of the SimCLR embeddings as instance representations—increasing the number of principal components did not significantly improve performance. We, furthermore, accounted for plate batch effects using MixMIL's covariate effects. To accommodate the multiclass classification task, we employed a categorical likelihood for MixMIL as described in Section 3.4.4. We then compared its performance with the baseline models in Section 4.1 using the F1 score metric and balanced accuracy. To evaluate model generalization, we held out one plate per compound for testing and optimized the model on the wells of the remaining plates. To compute standard errors, we ran three repeat experiments, each time holding out a different plate.

Table 2: F1 score and balanced accuracy comparison for MixMIL and baseline MILs on the MoA classification task. We report averages and standard errors across three repeat experiments, holding out a different plate per treatment for testing.

Method	Bal. Accuracy	F1 Macro	F1 Micro
Bayes-MIL	0.63 ± 0.02	0.63 ± 0.02	0.70 ± 0.01
ABMIL	0.72 ± 0.02	0.73 ± 0.01	0.76 ± 0.01
Gated ABMIL	0.67 ± 0.03	0.65 ± 0.03	0.70 ± 0.03
Additive ABMIL	0.41 ± 0.00	0.34 ± 0.00	0.47 ± 0.02
DSMIL	0.89 ± 0.02	0.89 ± 0.02	0.90 ± 0.01
MixMIL	$\mathbf{0.94 \pm 0.02}$	$\mathbf{0.94 \pm 0.01}$	$\mathbf{0.95 \pm 0.01}$

Results MixMIL surpassed other MIL models in out-of-sample predictions (Table 2). Furthermore, a visual assessment of instances based on MIL methods' attention weights revealed that MixMIL consistently down-weighted experimental artifacts (Figure 4 and Figure E.4).

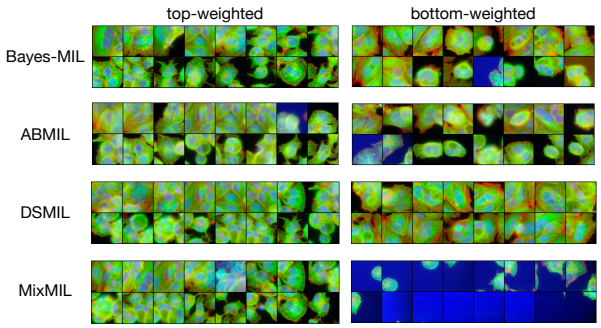


Figure 4: Top and bottom 16 weighted cells for the Latrunculin B drug for different MIL methods.

4.5 Histopathology Dataset

Task We used MixMIL to classify histology slides as cancerous or healthy. Each slide comprises numerous patches, each represented by an embedding. The application of MIL to this problem enables the assessment of individual patch contributions to the overall slide diagnosis (Ilse et al., 2018; Li et al., 2021; Cui et al., 2023b).

Setup and Results Our experiments utilized the widely-referenced Camelyon16 (Ehteshami Bejnordi et al., 2017) histopathology dataset. In particular, we employed the version provided by Li et al. (2021), which offers SimCLR embeddings of 4,886,434 patches at 20x magnification spanning across 399 slides (160 cancerous and 239 healthy), and a predefined train-test split. We earmarked 10% of the training bags for hyperparameter optimization (see Appendix C.3). No covariates were available for this dataset. All MIL models exhibited remarkable accuracy, with MixMIL topping the performance chart (Table 3).

Table 3: AUC classification accuracy results on the test set of the official train-test split of Camelyon16 (Li et al., 2021), evaluated over five different training seeds.

Method	AUC
Bayes-MIL	0.865 ± 0.031
ABMIL	0.958 ± 0.015
Gated ABMIL	0.965 ± 0.006
DSMIL	0.915 ± 0.013
MixMIL	$\mathbf{0.977 \pm 0.001}$

5 DISCUSSION

In this work, we presented Mixed Models with Multiple Instance Learning (MixMIL), a framework merging GLMMs and attention-based MIL. Conceived with genomics analyses in mind, MixMIL achieves robustness and efficiency by utilizing pre-trained embeddings and a shallow function for attention modeling. Our simulations demonstrate the versatility of MixMIL, even in scenarios where simple ML baselines surpassed traditional MIL approaches. This adaptability was further validated in real-world data applications spanning a wide range of tasks—from applications like genomics, characterized by a lower signal-to-noise ratio, to benchmark MIL datasets in histopathology. As a limitation, we note that MixMIL's inherent simplicity could lead to suboptimal performance when working with large datasets on complex tasks. We hope that by sharing the MixMIL framework, the scientific community finds a valuable tool to analyze multi-instance datasets.

Acknowledgments

We thank the reviewers for their constructive comments. We are grateful to Jose Alquicira-Hernandez for providing access to the processed OneK1K dataset. Special thanks go to Fabiola Curion for her valuable insights and helpful discussions.
JPE received support from the European Laboratory for Learning and Intelligent Systems (ELLIS) through their PhD Program. AP was supported by the Helmholtz Association, as part of the joint research school Munich School for Data Science (MUDS).
Co-funded by the European Union (ERC, DeepCell - 101054957). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.
FJT consults for Immunai Inc., Singularity Bio B.V., CytoReason Ltd, Cellarity, and has ownership interest in Dermagnostix GmbH and Cellarity.
FPC was funded by the Free State of Bavaria's Hightech Agenda through the Institute of AI for Health (AIH).

Contributions

JPE and FPC developed the model. JPE performed the simulation, single-cell genomics, and histopathology experiments. AP performed the microscopy experiment. JMT provided valuable technical knowledge and contributed to the interpretation of results. FPC conceived the project with support from FJT. FPC and FJT supervised the study. All authors wrote and contributed to the manuscript. The authors read and approved the final manuscript.

References

David J Ahern, Zhichao Ai, Mark Ainsworth, et al. A blood atlas of covid-19 defines hallmarks of disease severity and specificity. Cell, 185(5), 2022.

Douglas Bates, Martin Mächler, Ben Bolker, et al. Fitting linear mixed-effects models using lme4. arXiv preprint arXiv:1406.5823, 2014.

A Ben Taieb and G Hamarneh. Kimianet: An interpretable deep learning approach for histopathology image classification and retrieval. IEEE Transactions on Medical Imaging, 39(7), 2020.

Norman E Breslow and David G Clayton. Approximate inference in generalized linear mixed models. Journal of the American statistical Association, 88 (421), 1993.

Peter D. Caie, Rebecca E. Walls, Alexandra Ingleston-Orme, et al. High-content phenotypic profiling of drug response signatures across distinct cancer cells. Molecular Cancer Therapeutics, 9(6), June 2010.

Bradley Carlin and Thomas Louis. Empirical bayes: Past, present and future. Journal of The American Statistical Association - J AMER STATIST ASSN, 95, 12 2000.

Mathilde Caron, Hugo Touvron, Ishan Misra, et al. Emerging properties in self-supervised vision transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), October 2021.

Francesco Paolo Casale, Danilo Horta, Barbara Rakitsch, et al. Joint genetic analysis using variant sets reveals polygenic gene-context interactions. PLoS genetics, 13(4), 2017.

Ting Chen, Simon Kornblith, Mohammad Norouzi, et al. A simple framework for contrastive learning of visual representations. In Hal Daumé III and Aarti Singh, editors, Proceedings of the 37th International Conference on Machine Learning, volume 119 of Proceedings of Machine Learning Research. PMLR, 13–18 Jul 2020.

The GTEx Consortium. Genetic effects on gene expression across human tissues. Nature, 550(7675), 2017.

Helena L. Crowell, Charlotte Soneson, Pierre-Luc Germain, et al. muscat detects subpopulation-specific state transitions from multi-sample multi-condition single-cell transcriptomics data. Nature Communications, 11(1), November 2020.

Danni Cui, Yingying Liu, Gang Liu, et al. A multiple-instance learning-based convolutional neural network model to detect the IDH1 mutation in the histopathology images of glioma tissues. Journal of Computational Biology, 27(8), August 2020.

Haotian Cui, Chloe Wang, Hassaan Maan, et al. scGPT: Towards building a foundation model for single-cell multi-omics using generative AI. bioRxiv, May 2023a.

Yufei Cui, Ziquan Liu, Xiangyu Liu, et al. Bayes-MIL: A new probabilistic perspective on attention-based multiple instance learning for whole slide images. In The Eleventh International Conference on Learning Representations, 2023b.

Anna SE Cuomo, Tobias Heinen, Danai Vagiaki, et al. Cellregmap: a statistical framework for mapping context-specific regulatory variants using scrna-seq. Molecular Systems Biology, 18(8), 2022.

Anna SE Cuomo, Aparna Nathan, Soumya Raychaudhuri, et al. Single-cell genomics meets human genetics. Nature Reviews Genetics, 2023.

Andy Dahl, Khiem Nguyen, Na Cai, et al. A robust method uncovers significant context-specific heritability in diverse complex traits. The American Journal of Human Genetics, 106(1), 2020.

Marie Deprez, Laure-Emmanuelle Zaragosi, Marin Truchi, et al. A single-cell atlas of the human healthy airways. American journal of respiratory and critical care medicine, 202(12), 2020.

Michael Doron, Théo Moutakanni, Zitong S. Chen, et al. Unbiased single-cell morphology with self-supervised vision transformers. bioRxiv, June 2023.

Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes van Diest, et al. Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer. JAMA, 318(22), 12 2017. ISSN 0098-7484.

Ariel DH Gewirtz, F William Townes, and Barbara E Engelhardt. Expression qtls in single-cell sequencing data. bioRxiv, 2022.

Wei Hao, Minsun Song, and John D Storey. Probabilistic models of genetic variation in structured populations applied to global human studies. Bioinformatics, 32(5), 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, et al. Momentum contrast for unsupervised visual representation learning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.

Maximilian Ilse, Jakub Tomczak, and Max Welling. Attention-based deep multiple instance learning. In International conference on machine learning. PMLR, 2018.

Rens Janssens, Xian Zhang, Audrey Kauffmann, et al. Fully unsupervised deep mode of action learning for phenotyping high-content cellular images. Bioinformatics, 37(23), 2021.

Syed Ashar Javed, Dinkar Juyal, Harshith Padigela, et al. Additive mil: Intrinsically interpretable multiple instance learning for pathology. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, Advances in Neural Information Processing Systems, volume 35. Curran Associates, Inc., 2022.

Maxime W Lafarge, Juan C Caicedo, Anne E Carpenter, et al. Capturing single-cell phenotypic variation via unsupervised representation learning. In International Conference on Medical Imaging with Deep Learning. PMLR, 2019.

Marvin Lerousseau, Maria Vakalopoulou, Marion Classe, et al. Weakly supervised multiple instance learning histopathological tumor segmentation. In Medical Image Computing and Computer Assisted Intervention – MICCAI 2020. Springer International Publishing, 2020.

Bin Li, Yin Li, and Kevin W. Eliceiri. Dual-stream multiple instance learning network for whole slide image classification with self-supervised contrastive learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2021.

Jennifer Lim, Venessa Chin, Kirsten Fairfax, et al. Transitioning single-cell genomics into the clinic. Nature Reviews Genetics, 2023.

Christoph Lippert, Jennifer Listgarten, Ying Liu, et al. Fast linear mixed models for genome-wide association studies. Nature methods, 8(10), 2011.

Vebjorn Ljosa, Peter D Caie, Rob Ter Horst, et al. Comparison of methods for image-based profiling of cellular morphological responses to small-molecule treatment. Journal of biomolecular screening, 18 (10), 2013.

Po-Ru Loh, Gleb Kichaev, Steven Gazal, et al. Mixed-model association for biobank-scale datasets. Nature genetics, 50(7), 2018.

Romain Lopez, Jeffrey Regier, Michael B Cole, et al. Deep generative modeling for single-cell transcriptomics. Nature methods, 15(12), 2018.

Paula A Marin Zapata, Sina Roth, Dirk Schmutzler, et al. Self-supervised feature extraction from image time series in plant phenotyping using triplet networks. Bioinformatics, 37(6), 10 2020. ISSN 1367-4803.

Rachel Moore, Francesco Paolo Casale, Marc Jan Bonder, et al. A linear mixed-model approach to study multivariate gene–environment interactions. Nature genetics, 51(1), 2019.

Aparna Nathan, Samira Asgari, Kazuyoshi Ishigaki, et al. Single-cell eqtl models reveal dynamic t cell

state dependence of disease loci. Nature, 606(7912), 2022.

Drew Neavin, Quan Nguyen, Maciej S Daniszewski, et al. Single cell eqtl analysis identifies cell type-specific genetic control of gene expression in fibroblasts and reprogrammed induced pluripotent stem cells. Genome biology, 22(1), 2021.

David Ochoa, Andrew Hercules, Miguel Carmona, et al. The next-generation open targets platform: reimagined, redesigned, rebuilt. Nucleic acids research, 51 (D1):D1353–D1359, 2023.

Alexis Perakis, Ali Gorji, Samriddhi Jain, et al. Contrastive learning of single-cell phenotypic representations for treatment classification. In Machine Learning in Medical Imaging. Springer International Publishing, 2021.

Richard K Perez, M Grace Gordon, Meena Subramaniam, et al. Single-cell rna-seq reveals cell type–specific molecular and genetic associations to lupus. Science, 376(6589), 2022.

Alexander Plotnikov, Eldar Zehorai, Shiri Procaccia, et al. The mapk cascades: Signaling components, nuclear roles and mechanisms of nuclear translocation. Biochimica et Biophysica Acta (BBA) - Molecular Cell Research, 1813(9), 2011. Regulation of Signaling and Cellular Fate through Modulation of Nuclear Protein Import.

Shaun Purcell, Benjamin Neale, Kathe Todd-Brown, et al. Plink: a tool set for whole-genome association and population-based linkage analyses. The American journal of human genetics, 81(3), 2007.

Rajesh Ranganath, Sean Gerrish, and David Blei. Black Box Variational Inference. In Samuel Kaski and Jukka Corander, editors, Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, volume 33 of Proceedings of Machine Learning Research, Reykjavik, Iceland, 22–25 Apr 2014. PMLR, PMLR.

Herbert B Schiller, Daniel T Montoro, Lukas M Simon, et al. The human lung cell atlas: a high-resolution reference map of the human lung in health and disease. American journal of respiratory cell and molecular biology, 61(1), 2019.

Zhuchen Shao, Hao Bian, Yang Chen, et al. Transmil: Transformer based correlated multiple instance learning for whole slide image classification. Advances in neural information processing systems, 34, 2021.

Daniel Siegismund, Mario Wieser, Stephan Heyse, et al. Self-supervised representation learning for high-content screening. In Ender Konukoglu, Bjoern Menze, Archana Venkataraman, Christian Baumgartner, Qi Dou, and Shadi Albarqouni, editors, Proceedings of The 5th International Conference on Medical Imaging with Deep Learning, volume 172 of Proceedings of Machine Learning Research. PMLR, 06–08 Jul 2022.

Lisa Sikkema, Daniel C Strobl, Luke Zappia, et al. An integrated cell atlas of the human lung in health and disease. bioRxiv, 2022.

S. Singh, M.-A. Bray, T.R. Jones, et al. Pipeline for illumination correction of images for high-throughput microscopy. Journal of Microscopy, 256(3), September 2014.

Montgomery Slatkin. Linkage disequilibrium–understanding the evolutionary past and mapping the medical future. Nature Reviews Genetics, 9(6), 2008.

P.J. Sudharshan, Caroline Petitjean, Fabio Spanhol, et al. Multiple instance learning for histopathological breast cancer image classification. Expert Systems with Applications, 117, March 2019.

Poonam Tewary, De Yang, Gonzalo de la Rosa, et al. Granulysin activates antigen-presenting cells through TLR4 and acts as an immune alarmin. Blood, 116 (18), 11 2010.

Christina V. Theodoris, Ling Xiao, Anant Chopra, et al. Transfer learning enables predictions in network biology. Nature, 618(7965), May 2023.

Kyle J Travaglini, Ahmad N Nabhan, Lolita Penland, et al. A molecular cell atlas of the human lung from single-cell rna sequencing. Nature, 587(7835), 2020.

Katy Vandereyken, Alejandro Sifrim, Bernard Thienpont, et al. Methods and applications for single-cell and spatial multi-omics. Nature Reviews Genetics, 2023.

Johanna Wagner, Maria Anna Rapsomaniki, Stéphane Chevrier, et al. A single-cell atlas of the tumor and immune ecosystem of human breast cancer. Cell, 177 (5), 2019.

Sophia J. Wagner, Daniel Reisenbüchler, Nicholas P. West, et al. Transformer-based biomarker prediction from colorectal cancer histology: A large-scale multicentric study. Cancer Cell, 41(9), 2023.

Xiyue Wang, Yuexi Du, Sen Yang, et al. Retccl: Clustering-guided contrastive learning for whole-slide image retrieval. Medical Image Analysis, 83, 2023.

Harm-Jan Westra, Marjolein J Peters, Tonu Esko, et al. Systematic identification of trans eqtls as putative drivers of known disease associations. Nature Genetics, 45(10), 2013.

Aaron J Wilk, Arjun Rustagi, Nancy Q Zhao, et al. A single-cell atlas of the peripheral immune response in patients with severe covid-19. Nature medicine, 26(7), 2020.

Seyhan Yazar, Jose Alquicira-Hernandez, Kristof Wing, et al. Single-cell eqtl mapping identifies cell type-specific genetic control of autoimmune disease. Science, 376(6589), 2022.

Yue You, Xueyi Dong, Yong Kiat Wee, et al. Modeling group heteroscedasticity in single-cell rna-seq pseudo-bulk data. Genome biology, 24(1), 2023.

Yu Zhao, Fan Yang, Yuqi Fang, et al. Predicting lymph node metastasis using histopathological images based on multiple instance learning with deep graph convolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2020.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes/No/Not Applicable]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes/No/Not Applicable]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes/No/Not Applicable]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes/No/Not Applicable]
   (b) Complete proofs of all theoretical results. [Yes/No/Not Applicable]
   (c) Clear explanations of any assumptions. [Yes/No/Not Applicable]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes/No/Not Applicable]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes/No/Not Applicable]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes/No/Not Applicable]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes/No/Not Applicable]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes/No/Not Applicable]
   (b) The license information of the assets, if applicable. [Yes/No/Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes/No/Not Applicable]
   (d) Information about consent from data providers/curators. [Yes/No/Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Yes/No/Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Yes/No/Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Yes/No/Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Yes/No/Not Applicable]

Mixed Models with Multiple Instance Learning: Appendix

A MixMIL TRAINING DETAILS

To enhance numerical stability during training, we reparameterize the embedding-based prediction of our model. Specifically, introducing a bag-level index i to Eq (3-4) in the main text, we denote the embedding-based bag prediction of bag i, composed of I_i instances, as:

$$(\boldsymbol{u})_i = \boldsymbol{z}_{\boldsymbol{\gamma}}(\boldsymbol{X}_i)^T \boldsymbol{\beta} \tag{A.1}$$

where $\boldsymbol{X}_i \in \mathbb{R}^{I_i \times Q}$ collectively denotes the instance embeddings in bag i. During training, we reparameterize the embedding-based prediction $\boldsymbol{u}$ to have sample mean 0 and sample variance $b^2 = \frac{1}{Q}\sum_{j=1}^{Q} \beta_j^2$. Importantly, this reparameterization does not alter the model structure but improves its trainability and interpretability[4]. Specifically, we implement the reparameterization by first introducing

$$(\tilde{\boldsymbol{u}})_i = \boldsymbol{z}_{\boldsymbol{\gamma}}(\boldsymbol{X}_i)^T \underbrace{\boldsymbol{\beta} \,/\, b}_{\boldsymbol{\eta}}, \tag{A.2}$$

where $\boldsymbol{\eta}$ has unit mean square. We then rescale $\tilde{\boldsymbol{u}}$ to:

$$\boldsymbol{u} = b \times \frac{\tilde{\boldsymbol{u}} - \text{mean}(\tilde{\boldsymbol{u}})}{\text{std}(\tilde{\boldsymbol{u}})}. \tag{A.3}$$

This standardization operation can be implemented within the Pytorch framework using a `batchnorm` layer. By using this reparameterization, we introduce stochasticity and stability during training, while keeping track of training set statistics for inference.

B BASELINE MODELS

B.1 MIL Baselines

- **ABMIL.** Ilse et al. (2018) use neural networks and attention to learn instance-specific weights and perform bag-level aggregation to optimize a downstream prediction task. The authors additionally propose a gated version of the model where they combine *tanh* and *sigmoid* activation functions to the attention weights to better learn non-linearities.
- **Additive MIL.** Javed et al. (2022) overcome the lack of weight interpretability by pooling instance-level predictions rather than pooling instance features. In greater detail, attention weights are first estimated similarly to ABMIL and used to weigh single instances in a bag. Successively, a predictor is applied directly to the weighted instances to derive instance-specific logits, which are summed to yield a bag-level prediction.
- **DSMIL.** Li et al. (2021) first model an instance classifier which provides class-specific activation scores for each element in a bag. Max-pooling on the classification scores defines a critical instance per class. To obtain bag-level embeddings, attention weights are learned based on the distance of single elements from the critical instance per class and used to aggregate features.

[4] After this reparameterization, $\sigma^2_{\boldsymbol{\beta}}$ can be directly interpreted as the variance explained by pooled bag embeddings.

- **Bayes-MIL.** Cui et al. (2023b) derive uncertainty over the attention weights of a standard MIL model by learning patch-specific posterior distributions with variational inference. In their application to histopathology screens, the authors introduce a slide regularizer to concentrate attention on either the positive or negative side of patches for precise localization with high confidence. Additionally, the authors encode spatial information between patches in Multiple Instance Learning (MIL) for Whole Slide Image (WSI) recognition and localization, proposing the use of Conditional Random Fields (CRF). Because spatial information is not available for the simulation, genetics and microscopy datasets, we use the Bayes-MIL version without slide regularizer and CRF.

B.2 Traditional ML Models

In simulation studies, we also benchmarked MixMIL against a GLMM with a corresponding likelihood and two widely used nonlinear models: Random Forest and XGBoost. All models use predefined bag-level features as inputs, based on three different strategies: mean pooling, median pooling, and a combination of mean, squared mean, and cubed mean of instance features (Figure E.1 (ii)).

C EXPERIMENTAL SETUP

C.1 Simulations and Genomics Dataset

MIL Model Hyperparameter Search Hyperparameters for all MIL models across the simulations and genomics application were optimized based on the simulation default scenario, which emulates the genomics dataset. Specifically, we used our simulation procedure with default parameters to generate 10 datasets, including training and validation sets. For each simulated dataset, we trained MIL models with different losses, learning rates and regularization parameters (Table C.1) on the training set and evaluated their performance on the validation set. The hyperparameters for each model were then chosen based on the average Spearman correlation between predicted and true values, and are listed in Table C.2. Across the losses that we considered, which included common regression losses (`MSELoss`, `HuberLoss`, `SmoothL1Loss`, `L1Loss`) as well as the negative log-likelihood of the Binomial distribution used in MixMIL, the `HuberLoss` consistently yielded the best results for this use case.

Table C.1: Hyperparameter sweep for Simulations and Genomics dataset

Model	Learning Rate	Weight Decay	Dim. Encoder	Regularization
Bayes-MIL	{5e-3, 5e-4, 1e-4, 1e-5}	{5e-4, 1e-4, 5e-5, 1e-5, 1e-6 }	{30}	`log10space`(1e-6, 1e-10)
DSMIL	{5e-3, 2e-4, 5e-5}	{5e-3, 5e-4, 1e-4}	{30}	-
ABMIL	{5e-3, 5e-4, 5e-5}	{5e-3, 5e-4, 1e-4, 1e-5, 0}	{30}	-
Gated ABMIL	{5e-3, 5e-4, 5e-5}	{5e-3, 5e-4, 1e-4, 1e-5, 0}	{30}	-

Table C.2: Optimized Hyperparameters for Simulations and Genomics dataset

Model	Learning Rate	Weight Decay	Dim. Encoder	Regularization
Bayes-MIL	5e-4	1e-6	30	1e-8
DSMIL	2e-4	5e-3	30	-
ABMIL	5e-4	1e-4	30	-
Gated ABMIL	5e-4	1e-4	30	-

Traditional ML Models Hyperparameter Search To tune the hyperparameters of the baseline models considered in our paper, we performed a randomized search of the hyperparameter space, using a 5-fold cross-validation within the training set and sampling 20 hyperparameter combinations. Specifically, each combination of hyperparameters was evaluated by the average cross-validated Spearman correlation metric, the combination that provided the best performance was chosen, and the final models were retrained on the entire training set before performing out-of-sample predictions on the test set. For full information on hyperparameters and their respective search spaces see Table C.3 and Table C.4.

Table C.3: Random Forest Hyperparameter Search Space

Hyperparameter	Search Space
Number of Estimators (n_estimators)	{50, 100, 150, 200, 500}
Max Features (max_features)	{`sqrt`, `log2`, `None`}
Max Depth (max_depth)	{10, 20, 30, 50}
Min Samples Split (min_samples_split)	{2, 5, 10}
Min Samples Leaf (min_samples_leaf)	{1, 2, 4}

Table C.4: XGBoost Hyperparameter Search Space

Hyperparameter	Search Space
Max Depth (max_depth)	{3, 4, ..., 9}
Learning Rate (learning_rate)	$\{10^{-3}, ..., 10^{-2}, ..., 10^{0}\}$
Number of Estimators (n_estimators)	(100, 1000)
Subsample (subsample)	(0.5, 1)
Col Sample By Tree (colsample_bytree)	(0.5, 1)
Objective (objective)	`binary:logistic`
Evaluation Metric (eval_metric)	`logloss`

C.2 Microscopy Dataset

For the microscopy data, we ran three repeat experiments, each time holding out a different plate as a test set. For each repeat experiment, we trained MIL models with different hyperparameters on 90% of the training set (Table C.5) and evaluated their predictive performance based on the F1 score on the remaining 10%. For each model, we considered hyperparameters that yielded the best performance on this validation set and retrained on the entire training set before computing predictions on the test set. We noticed that the same hyperparameters were selected for each model across the three repeat experiments (Table C.6).

Table C.5: Hyperparameter sweep for Microscopy Dataset

Model	Learning Rate	Dim. Encoder
Bayes-MIL	{1e-3, 5e-4, 1e-4}	{64, 100, 128}
DSMIL	{1e-3, 5e-4, 1e-4}	{64, 100, 128}
ABMIL	{1e-3, 5e-4, 1e-4}	{64, 100, 128}
Additive ABMIL	{1e-3, 5e-4, 1e-4}	{64, 100, 128}
Gated ABMIL	{1e-3, 5e-4, 1e-4}	{64, 100, 128}

Table C.6: Optimized Hyperparameters for Microscopy Dataset

Model	Learning Rate	Weight Decay	Dim. Encoder
Bayes-MIL	1e-4	1e-5	64
DSMIL	1e-4	1e-5	100
ABMIL	5e-4	1e-5	100
Additive ABMIL	5e-4	1e-5	100
Gated ABMIL	5e-4	1e-5	100

C.3 Histopathology Dataset

We employed the version of Camelyon16 provided by (Li et al., 2021), which contained a predefined train-test split. We earmarked 10% of the training bags for hyperparameter optimization and swept MIL model hyperparameters

using a grid-search approach (Table C.7). For each model, we picked hyperparameters based on the validation loss. The final parameters are shown in Table C.8.

Table C.7: Hyperparameter sweep for Camelyon16 Dataset

Model	Learning Rate	Weight Decay	Dim. Encoder	Regularization
Bayes-MIL	{5e-4, 1e-4, 5e-5, 1e-5}	{1e-4, 1e-5, 1e-6, 1e-7}	{30, 60, 120}	`log10space`(1e-6, 1e-10)
DSMIL	{1e-4, 2e-4, 5e-5}	{5e-3}	{30, 60, 120}	-
ABMIL	{1e-4, 5e-4, 5e-5}	{1e-4}	{30, 60, 120}	-
Gated ABMIL	{1e-4, 5e-4, 5e-5}	{1e-4}	{30, 60, 120}	-

Table C.8: Optimized Hyperparameters for Camelyon16 Dataset

Model	Learning Rate	Weight Decay	Dim. Encoder	Regularization
Bayes-MIL	5e-4	1e-5	30	1e-8
DSMIL	1e-4	5e-3	60	-
ABMIL	5e-5	1e-4	60	-
Gated ABMIL	5e-4	1e-4	30	-

D DATA PREPROCESSING

D.1 Variant Filtering Procedure

To identify the 28 variants associated with the average transcriptional state, we adopted the following approach: We started from 16,718 variants associated with proximal gene expression in the primary analysis of this data. These variants are typically known as cis-expression Quantitative Trait Loci (cis-eQTLs). Then, we selected variants that could be predicted from average cell embedding using a GLMM (Spearman $\rho > 0.15$). Next, to mitigate dependencies between genetic labels due to linkage disequilibrium (Slatkin, 2008), we undertook a clumping procedure (Purcell et al., 2007), which yielded the final set of 28 variants.

D.2 Single-Cell Embeddings

Single-cell RNA-seq data consists of a cell-by-gene matrix of RNA counts per cell. For the OneK1K dataset, we have approximately 1.3 million cells and 32,000 genes (Yazar et al., 2022). However, due to technical bias and experimental dropout, many of these genes are uninformative. Therefore, it is common to subset to the most highly variable genes and perform analyses on embedding space. The single-cell Variational Inference (scVI) model (Lopez et al., 2018) is designed with the properties of this data in mind. We used hyperparameters commonly used for this type of dataset size and report them in Table D.1.

Table D.1: Summary of scVI (Lopez et al., 2018) parameters

Parameter	Value
Layers	2
Batch Size	256
Epochs	15
Gene Likelihood	ZINB
Covariate	Sequencing Pool
Highly Variable Genes (HVG)	5000

As described in the main, we performed the scVI integration on the sub-lymphoid cells (CD4, CD8, NK cells).

D.3 Microscopy Image Feature Extraction

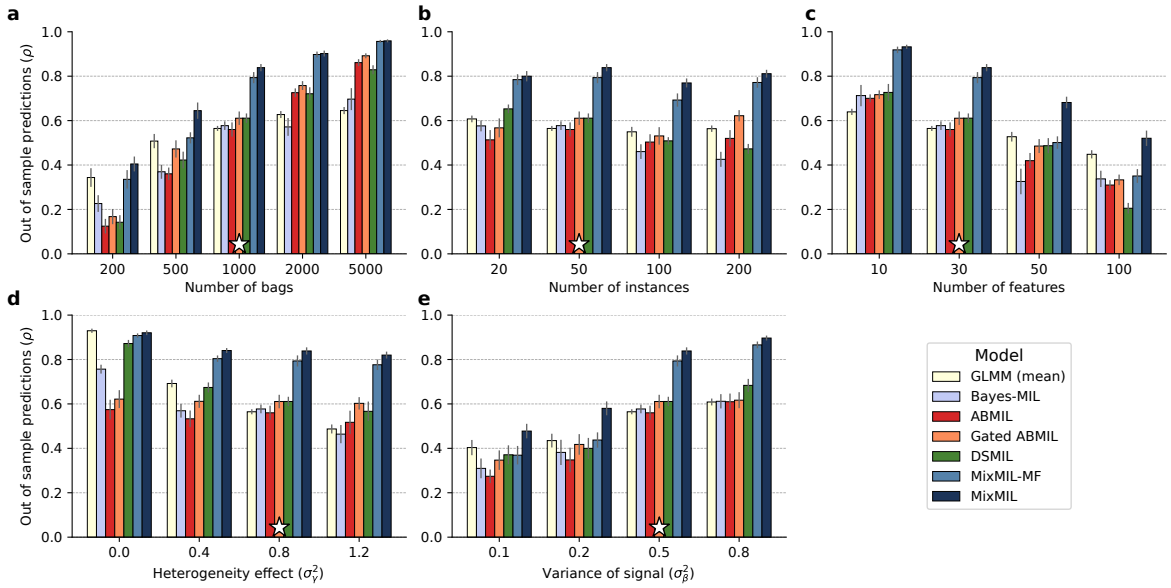
We downloaded cell images from Caie et al. (2010), extracted single cell images using nuclear coordinates made available by Ljosa et al. (2013), and applied plate correction as described in Singh et al. (2014). We then collected the weights of a ResNet50 SimCLR model pre-trained by Perakis et al. (2021) to infer cell embeddings. The extracted embeddings have a dimensionality of 2048, which we reduce using PCA.

D.4 Accounting for Covariates

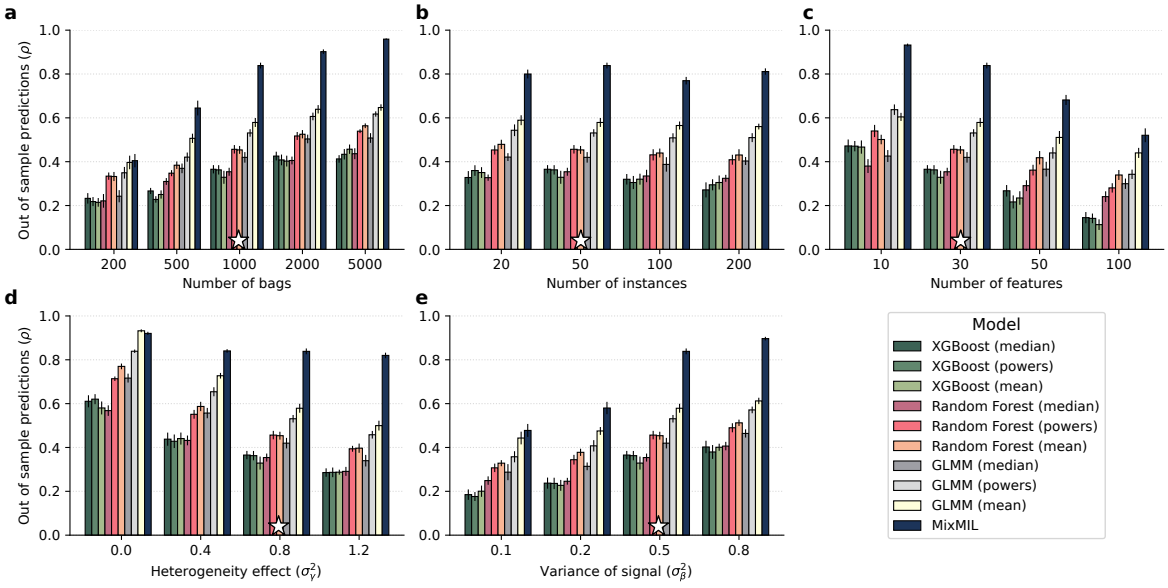
The single-cell genomics and microscopy datasets contain covariates and batches which should not be predictive of the outcome labels but could confound the prediction. We, therefore, accounted for them as fixed effects ($\boldsymbol{c}^T\boldsymbol{\alpha}$), leveraging the GLMM structure of MixMIL and regressed them from the data for the baseline MILs. We used the covariates sex, age, and population structure (4 genetic PCs) for the genetics use case. In the microscopy experiment, we accounted for plate-batch effects. We applied constant intercepts for the histopathology and simulation experiments.

E ADDITIONAL RESULTS

E.1 Simulations



(i) Comparison with MIL models



(ii) Comparison with vs GLMM, Random Forest, and XGBoost

Figure E.1: Out-of-sample prediction performance (Spearman correlation, ρ) in different simulation settings, varying one parameter at a time while keeping the others constant. Specifically, we varied the number of bags (a), the number of instances (b), the number of features (c), the heterogeneity effect (d) and the variance of signal (e).

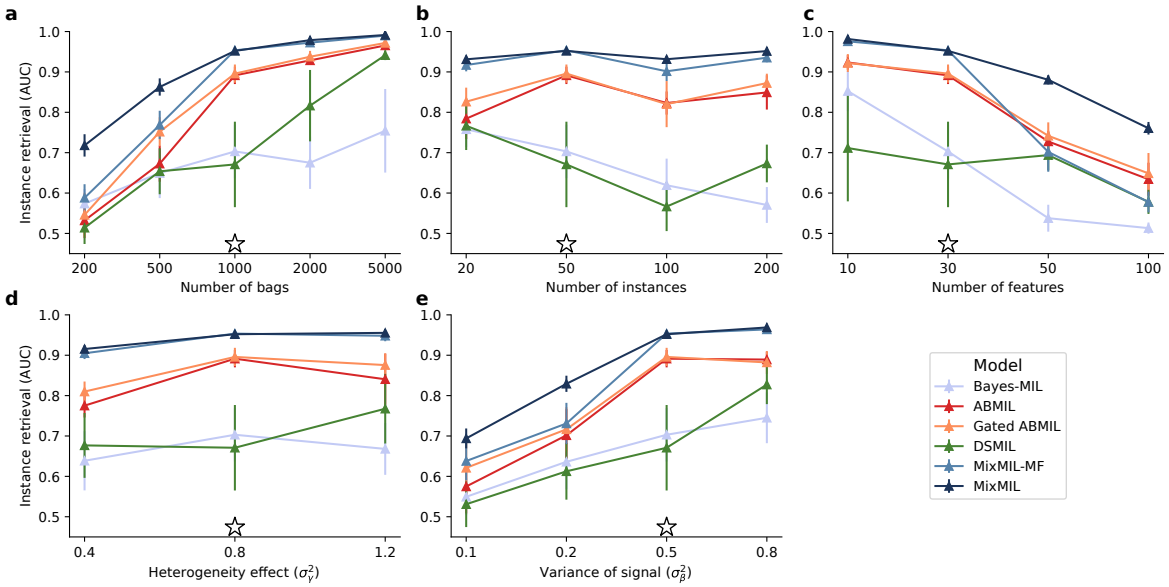


Figure E.2: Instance retrieval ROC-AUC of MixMIL for top 10% of instances in the same simulated scenarios as in Figure E.1. Error bars denote standard errors across 10 repeat experiments.

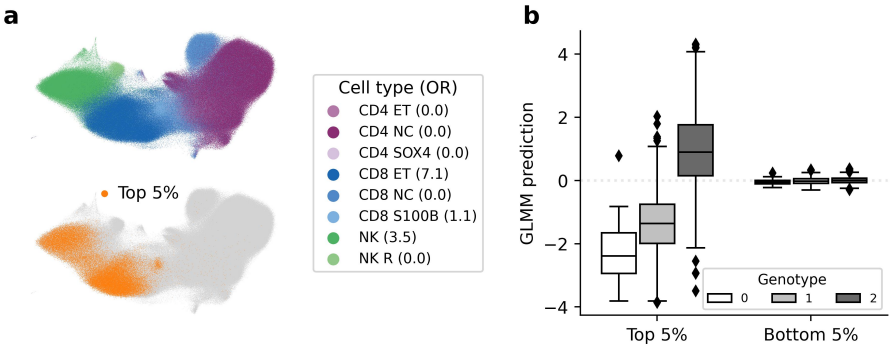
Table E.1: Comparison of instance-level models and their mean-embedding-based counterparts with MixMIL in the default scenario (marked by white stars in Figures 2, E.1, and E.2). The instance-level models, which either matched or underperformed compared to other models and required longer training times, were not included in further analyses.

Model	Type	Out of sample predictions (ρ)
Random Forest	mean	0.45 ± 0.02
Random Forest	instance-level	0.45 ± 0.02
XGBoost	mean	0.36 ± 0.02
XGBoost	instance-level	0.48 ± 0.02
GLMM	mean	0.58 ± 0.02
GLMM	instance-level	0.57 ± 0.02
MixMIL	MIL	$\mathbf{0.84 \pm 0.01}$

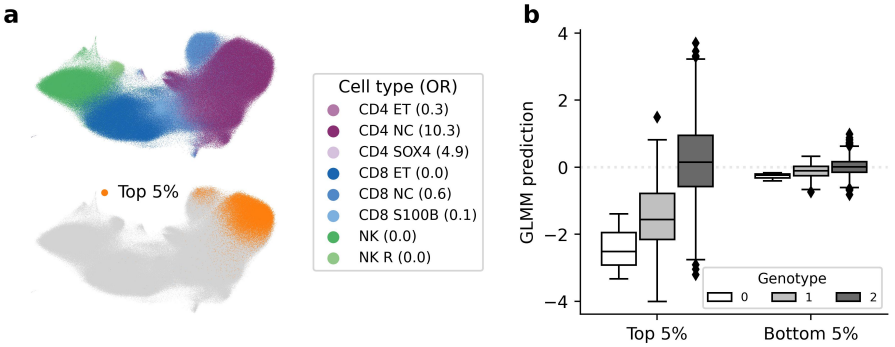
E.2 Single-cell Genomics Dataset

Table E.2: Comparison of prediction performance of MixMIL and baseline MILs on the single-cell genomics data. The table reports the mean and standard deviation of Spearman correlation across 28 genes, and the P-values obtained from a paired t-test, assessing the statistical significance of the differences in performance between MixMIL and each of the baseline MILs.

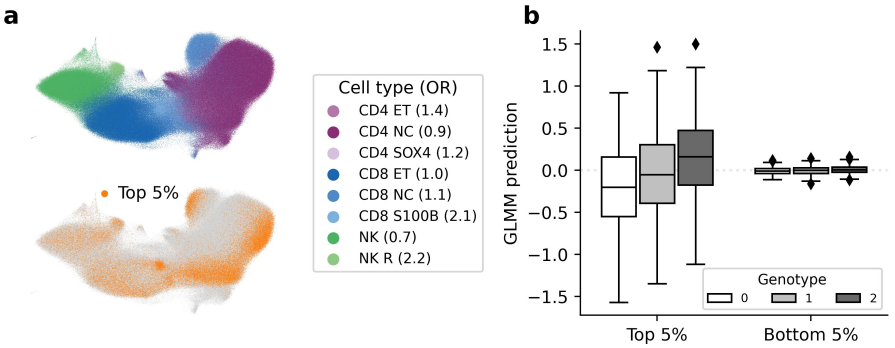
Method	Prediction performance (Spearman)	MixMIL improvement (t-test P-value)
ABMIL	0.21 ± 0.18	$< 4 \cdot 10^{-7}$
DSMIL	0.22 ± 0.17	$< 2 \cdot 10^{-7}$
Bayes-MIL	0.23 ± 0.14	$< 2 \cdot 10^{-4}$
Gated ABMIL	0.24 ± 0.18	$< 5 \cdot 10^{-6}$
MixMIL	$\mathbf{0.29 \pm 0.17}$	-



(i) `rs12151621` (Chromosome 2, near GNLY (Tewary et al., 2010)). Differential gene analysis brought up processes related to defence response (GO:0031349) and regulation of the MAPK cascade (GO:0043408, Plotnikov et al. (2011)).



(ii) `rs9928554` (Chromosome 16, near IL32), associated with Vitamin E measurements (openTargets, Ochoa et al. (2023)). Differential gene analysis brought up processes related to protein transport to the membrane (GO:0072657) and cell regulatory processes (GO:0050728, GO:0043408, GO:0031349).



(iii) `rs7503161` (Chromosome 17, near EIF5A) associated with increased hemoglobin concentration and height (openTargets, Ochoa et al. (2023)). EIF5A is involved in the positive regulation of the apoptosis (programmed cell death) signaling pathway. Differential gene analysis yielded processes directly (GO:0042981, GO:0043069, GO:0043066) and indirectly (GO:0071345, GO:1902531) related to cell death.

Figure E.3: Three examples of genetic variants for which MixMIL improved predictions. (a) UMAP of cell transcriptional embeddings showing cell types (top panel) and top 5% relevant cells according to MixMIL's weights (bottom panel). The odds ratio (OR) quantifies the enrichment of top-weighted cells within each cell type. (b) Box plots showing GLMM genetic predictions vs observed values, where the GLMM was fit either using the top 5% of the instances (left) or the bottom 5% (right) as ranked by MixMIL. Panels (i) and (ii) show variants where MixMIL's importance weights align with known cell types. Conversely, panel (iii) finds a cell state not captured by any individual cell type.

E.3 Microscopy Dataset

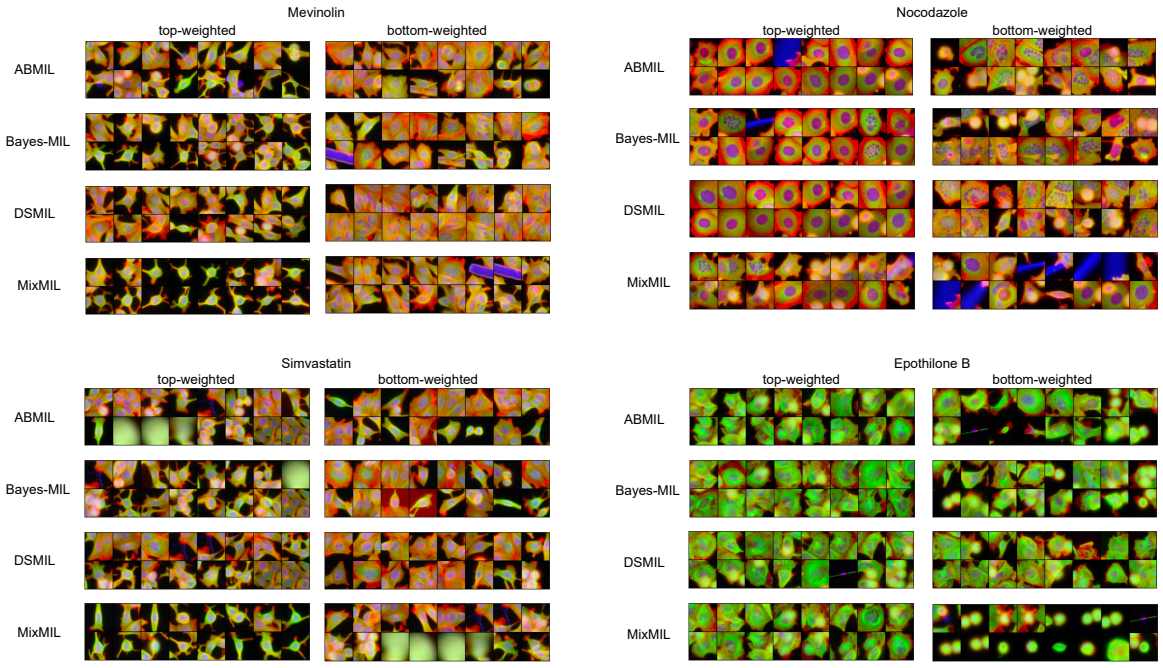


Figure E.4: Top and bottom 16 weighted cells for the Nocodazole, Simvastatin, Epothilone B, and Mevinolin for different MIL methods. While MixMIL down-weights images with technical artifacts and yields consistent phenotypes in the top classes, the competing MIL models tend to miss or up-weight noisy images and produce inconsistent phenotypes for the most important perturbed images.

Table E.3: F1 score comparison between MixMIL and competing MIL models on individual MoA labels. We observe that MixMIL improves on the MIL baselines on most considered classes.

MoA Category	Method	F1
Actin Disruptors	ABMIL	0.841 ± 0.014
	Additive ABMIL	0.235 ± 0.121
	Bayes-MIL	0.733 ± 0.035
	DSMIL	0.920 ± 0.025
	Gated ABMIL	0.730 ± 0.040
	MixMIL	**0.992 ± 0.004**
Aurora Kinase Inhibitors	ABMIL	0.910 ± 0.013
	Additive ABMIL	0.600 ± 0.155
	Bayes-MIL	0.862 ± 0.011
	DSMIL	0.970 ± 0.008
	Gated ABMIL	0.783 ± 0.034
	MixMIL	**1.000 ± 0.000**
Cholesterol-Lowering	ABMIL	0.550 ± 0.026
	Additive ABMIL	0.562 ± 0.023
	Bayes-MIL	0.293 ± 0.059
	DSMIL	0.885 ± 0.008
	Gated ABMIL	0.415 ± 0.011
	MixMIL	**0.914 ± 0.001**
DNA Damage	ABMIL	0.711 ± 0.009
	Additive ABMIL	0.316 ± 0.101
	Bayes-MIL	0.715 ± 0.027
	DSMIL	0.882 ± 0.016
	Gated ABMIL	0.558 ± 0.071
	MixMIL	**0.922 ± 0.019**
DNA Replication	ABMIL	0.677 ± 0.013
	Additive ABMIL	0.595 ± 0.015
	Bayes-MIL	0.712 ± 0.012
	DSMIL	0.838 ± 0.030
	Gated ABMIL	0.542 ± 0.039
	MixMIL	**0.928 ± 0.010**
Eg5 Inhibitors	ABMIL	0.868 ± 0.004
	Additive ABMIL	0.457 ± 0.118
	Bayes-MIL	0.825 ± 0.006
	DSMIL	0.947 ± 0.006
	Gated ABMIL	0.859 ± 0.014
	MixMIL	**0.993 ± 0.004**
Epithelial	ABMIL	0.759 ± 0.023
	Additive ABMIL	0.000 ± 0.000
	Bayes-MIL	0.631 ± 0.026
	DSMIL	0.865 ± 0.012
	Gated ABMIL	0.722 ± 0.026
	MixMIL	**0.929 ± 0.008**
Kinase Inhibitors	ABMIL	0.673 ± 0.040
	Additive ABMIL	0.000 ± 0.000
	Bayes-MIL	0.503 ± 0.048
	DSMIL	**0.956 ± 0.010**
	Gated ABMIL	0.658 ± 0.032
	MixMIL	0.927 ± 0.019
Microtubule Destabilizers	ABMIL	0.769 ± 0.019
	Additive ABMIL	0.621 ± 0.024
	Bayes-MIL	0.701 ± 0.012
	DSMIL	0.932 ± 0.002
	Gated ABMIL	0.816 ± 0.009
	MixMIL	**0.968 ± 0.006**
Microtubule Stabilizers	ABMIL	0.932 ± 0.014
	Additive ABMIL	0.283 ± 0.146
	Bayes-MIL	0.858 ± 0.013
	DSMIL	0.964 ± 0.006
	Gated ABMIL	0.822 ± 0.020
	MixMIL	**1.000 ± 0.000**
Protein Degradation	ABMIL	0.312 ± 0.020
	Additive ABMIL	0.292 ± 0.038
	Bayes-MIL	0.275 ± 0.027
	DSMIL	0.688 ± 0.054
	Gated ABMIL	0.321 ± 0.037
	MixMIL	**0.773 ± 0.026**
Protein Synthesis	ABMIL	0.757 ± 0.023
	Additive ABMIL	0.076 ± 0.039
	Bayes-MIL	0.447 ± 0.041
	DSMIL	0.879 ± 0.008
	Gated ABMIL	0.625 ± 0.016
	MixMIL	**0.960 ± 0.006**

Table E.4: Comparison of MixMIL and MIL baseline models with varying numbers of embedding principal components based on balanced accuracy, F1-macro, and F1-micro, including confidence intervals. Consistent with previous scenarios, MixMIL outperforms other approaches on all evaluated feature dimensions.

Method	**Number of Features**	**Balanced Accuracy**	**F1-macro**	**F1-micro**
ABMIL	64	0.581 ± 0.036	0.577 ± 0.041	0.607 ± 0.047
	256	0.726 ± 0.018	0.730 ± 0.016	0.764 ± 0.015
	512	0.714 ± 0.010	0.709 ± 0.013	0.746 ± 0.009
Additive ABMIL	64	0.298 ± 0.022	0.203 ± 0.020	0.340 ± 0.028
	256	0.410 ± 0.003	0.336 ± 0.002	0.470 ± 0.019
	512	0.475 ± 0.045	0.379 ± 0.051	0.501 ± 0.051
Bayes-MIL	64	0.727 ± 0.042	0.733 ± 0.044	0.758 ± 0.031
	256	0.623 ± 0.018	0.628 ± 0.024	0.699 ± 0.014
	512	0.641 ± 0.005	0.642 ± 0.010	0.693 ± 0.011
DSMIL	64	0.849 ± 0.028	0.850 ± 0.026	0.859 ± 0.022
	256	0.892 ± 0.025	0.894 ± 0.023	0.902 ± 0.019
	512	0.890 ± 0.018	0.895 ± 0.018	0.908 ± 0.014
Gated ABMIL	64	0.528 ± 0.016	0.508 ± 0.016	0.582 ± 0.032
	256	0.669 ± 0.031	0.654 ± 0.032	0.701 ± 0.031
	512	0.674 ± 0.029	0.676 ± 0.027	0.707 ± 0.027
MixMIL	**64**	**0.912 ± 0.019**	**0.915 ± 0.018**	**0.924 ± 0.017**
	256	**0.939 ± 0.017**	**0.942 ± 0.015**	**0.950 ± 0.013**
	512	**0.939 ± 0.019**	**0.944 ± 0.016**	**0.951 ± 0.014**