# Data-Driven Soft Labeling Scales DNA Read Classification to Whole-Body Cell-Type Deconvolution

**Dmytro Rizdvanetskyi** **Nathan Roos** **Pavlo Lutsik**
Department of Oncology
KU Leuven
Leuven, Belgium
{dmytro.rizdvanetskyi, nathanericjeanbaptiste.roos, pavlo.lutsik}@kuleuven.be

## Abstract

Cell-type deconvolution, the task of estimating the proportions of constituent cell types in a heterogeneous biological sample, is a core problem in computational biology. Methods that rely on epigenetic marks such as DNA methylation typically operate on aggregated methylation estimates, discarding the pattern-level information carried by individual DNA reads. Existing read-level approaches that exploit this information are scarce, and all remain restricted to few-class settings; scaling them further is an open problem because, at scale, non-discriminative reads dominate and hard labels conflict with the many-to-many mapping between methylation patterns and cell types, preventing classifier convergence. To overcome this, we propose data-driven soft labels that estimate the conditional cell-type distribution for each read, and integrate this scheme into *Syto*, a new modular framework for read-level classification-based deconvolution. On a whole-body atlas of 39 human cell types, *Syto* reduces MSE by 3.7× over the best examined baseline, with gains transferring to an out-of-distribution dataset spanning 16 tissues. *Syto* lays the foundation for modeling increasingly large cell-type panels, with improved applications in biology and healthcare. The proposed soft-labeling scheme is further translatable to any setting with a many-to-many signal-to-label mapping.

## 1 Introduction

Epigenetic marks, such as histone modifications, chromatin accessibility and DNA methylation, collectively define cell identity and can be used to distinguish between different normal and malignant cells [1, 2]. Among these, DNA methylation stands out as the most stable and cost-effective to profile at scale. Advances in methylome profiling have allowed researchers to exploit this biological signal for crucial clinical tasks: tumor classification, tumor purity estimation, and cell-type deconvolution. Traditionally, these tasks are treated separately. Tumor classification scores a sample against predefined reference tumor cohorts to confirm a diagnosis and guide treatment [3, 4, 5]. In contrast, tumor purity and fraction estimation quantify the proportion of cancer cells in a tissue sample [6, 7] or circulating tumor DNA (ctDNA) in the bloodstream. While these estimates are vital for early diagnostics and monitoring disease progression [8], they often struggle to capture complex cellular heterogeneity, such as the specific makeup of the immune microenvironment [6]. Cell-type deconvolution addresses this gap by determining the precise proportions of all constituent cell types

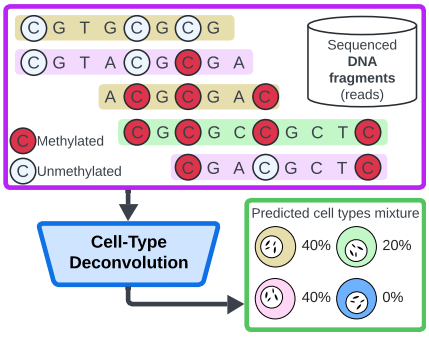


Figure 1: Principal scheme of the methylome-based cell-type deconvolution.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

in an inhomogeneous sample. Because deconvolution fundamentally captures both the target tumor proportion and its specific cellular profile, the narrower problems of tumor classification and purity estimation can be advantageously reduced to it. Thus, accurate deconvolution covers a much broader range of clinical use cases than methods designed solely for classification or purity estimation.

Existing methylome-based deconvolution methods typically operate on aggregated methylation measurements ($\beta$-values [9], $\alpha$-values [10], absolute counts of methylated CpGs [11]) or fragment-level counts [12, 13, 14, 15], omitting the differences in methylation pattern at the level of the individual DNA fragments (or *reads*) that are known to be important for tumorigenesis [16]. Conversely, incorporating local *omics* signatures (potentially containing more than methylation) and the broader nucleotide context may offer a more accurate representation of cell-of-origin and may improve detection capacity [17]. Despite this potential, only three end-to-end classification-based deconvolution methods exist [18, 19, 20] and only two of them preserve the within-read methylation pattern[19, 20]. These methods are either unfit to treat multi-class problems [18, 19], or wrongly apply the Bayesian framework to uncalibrated outputs of neural networks, leading to highly biased deconvolution results [19, 20]. Furthermore, none of these methods has been demonstrated to work across dozens of different cell types [13].

To address those limitations, we introduce *Syto*[1]: a modular framework for read-level pattern-sensitive classification-based cell-type deconvolution that scales to dozens of cell types. Our main contributions are as follows:

- We propose several labeling schemes to enable multi-class classification of omics-enriched DNA reads with a large number of classes. Among these, we introduce a biologically inspired, data-driven soft-labeling scheme – the maximum-likelihood estimator of the true conditional distribution. This approach has potential applications in scenarios where the mapping from signal to class labels is many-to-many rather than many-to-one.
- We propose a linear calibration method for regression models with probability simplex outputs: a simple, hyperparameter-free post hoc correction that outperforms existing calibration baselines in our setting.
- We propose a modular framework with a decoupled classifier, deconvolver, and calibrator, allowing independent improvement of each component. We extend two existing read-level pattern-sensitive classifiers and one based on $\alpha$-values to operate within it and provide five deconvolvers.
- We conduct large-scale experiments on a 39-cell-type reference atlas [13] and on an out-of-distribution dataset of 16 tissues [10]. *Syto* consistently improves deconvolution accuracy over existing baselines.

## 2 Related Work

**Read-level classification-based deconvolution methods.** *CancerDetector* [18] is a tumor detection and fraction estimation method that models the joint methylation states of CpG [2] sites in DNA reads using a Beta-Bernoulli distribution. For every cell-type specific genomic region, the model learns two separate Beta distributions conditioned on $\alpha$-values for tumoral and normal classes; these priors are then used to estimate the tumor-derived cfDNA fraction via maximum likelihood. *Dismir* [19] is another tumor fraction estimation method; it uses a CNN-LSTM architecture with both DNA sequence and methylation state as input to assign each read a *d-score* representing the probability that it originates from a tumor. The sample-level tumor fraction is then obtained by maximizing the posterior probability over all d-scores. *MethylBERT* [20] is an adaptation of the BERT architecture [21] to read-level classification. The model is first pre-trained on 3-mer tokenized reference genome sequences, then fine-tuned by adding methylation embeddings encoding the CpG methylation state at each position and by training a read classifier. Tumor purity is also obtained via maximum likelihood estimation. *MethylBERT* is the only existing read-level pattern-sensitive classification-based model that has demonstrated deconvolution with up to five cell types, using focal loss [22] to mitigate class imbalance.

**Label distribution learning and soft labeling.** Geng [23] has introduced label distribution learning as a computational task of estimating the degree to which each label describes the instance. Xu

[1] *Syto* is the romanization of the Ukrainian word for *sieve*, which fits our framework as it *sieves* the different cell types.

[2] CpG sites are places in the genomic sequence where a Cytosine is followed by Guanine in the 5' to 3' direction. These are the most prevalent methylation targets in mammals [1].

et al. [24] coined the term *label enhancement* (LE) for the task of constructing a label distribution (or *soft labels*) from one-hot encoded labels (or *hard labels*). They proposed Graph Laplacian Label Enhancement (GLLE) to learn the label distribution from hard labels by propagating label information through a graph constructed from the feature space. However, GLLE anchors the label distribution of each instance to its original hard label, which is not suitable for DNA methylation data (cf. Section 3.2). Wang and Geng [25] adapted label distribution learning to classification by, *inter alia*, upweighting the loss of instances whose label distributions have high entropy. Szegedy et al. [26] introduced label smoothing of hard labels to improve the generalization of computer vision models. In their definition, the original hard label is replaced by a soft label constructed as a mixture of the one-hot encoded label and the uniform distribution.

**Calibration.** Müller et al. [27] highlighted that applying label smoothing results in implicit confidence calibration that can be equated to temperature scaling [28] under certain parameters. Kull et al. [29] proposed a class-wise Dirichlet calibration method for classifiers applied on log-probabilities. The calibration transfer function is of the form: $\tilde{\mathbf{p}} \coloneqq \text{softmax}(W \log(\hat{\mathbf{p}}) + \mathbf{b})$ (1), where $\coloneqq$ denotes a definition, $\tilde{\mathbf{p}}$ is the calibrated prediction vector, $\hat{\mathbf{p}}$ is the uncalibrated prediction vector, $\log$ is applied element-wise, and $W$ and $\mathbf{b}$ are the parameters of the calibration model. Dirichlet calibration is a generalization of both temperature scaling ($W = tI$ and $\mathbf{b} = \mathbf{0}$)[28] and vector scaling (diagonal $W$) [28, 29].

# 3 Methods

## 3.1 *Syto*: a modular framework

Existing methods for read-level classification-based cell-type deconvolution follow three steps: selection of informative genomic regions for training and inference, classification of individual reads, and estimation of target cell-type proportions. This common structure offers an opportunity for harmonization. In this work, we propose *Syto*, a modular framework for read-level classification-based deconvolution. *Syto* specifically subsumes the classification and proportions estimation steps; it can be applied out of the box, regardless of the region selection procedure.

**Inference.** *Syto* operates on a sequencing library $\mathcal{D} \coloneqq \{(r_i, g_i, s_i)\}_{1 \le i \le N}$ consisting of $N$ DNA sequences (or *reads*) $r_i$ obtained from a biological bulk sample. Each read $r_i$ belongs to a single group of genomic regions (or *GR group*) $g_i \in \mathcal{G} \coloneqq \{1, \ldots, G\}$. Differentially Methylated Regions (DMRs) are a common choice for GR groups, and we use them in this work. Each read $r_i$ is also assigned an *omics signature* $s_i$. In this work, we focus on methylation, and so the signature is the set of pairs (position, methylation state) for each CpG site in the read. The methylation state is either 1 (methylated) or 0 (unmethylated). The omics signatures can be extended to contain information from gene expression, histone marks, chromatin accessibility, etc. The first step is to classify all reads, producing a vector of probability over the classes $\mathbf{u_i} \coloneqq (u_{i,1}, \ldots, u_{i,C})^T \in \Delta^{C-1}$ for every read $r_i$, where $C \in \mathbb{N}$ is the number of cell types to deconvolve, and $\Delta^{C-1}$ is the $(C-1)$-dimensional probability simplex. The predictions are then aggregated into a *feature matrix* (or *prediction matrix*) $P \in \mathbb{R}^{G \times C}$ (cf. Section D.3 for further details):

$$P_{g,c} \coloneqq \frac{1}{\sum_{i=1}^{N} \delta_{g,g_i} w_i} \sum_{i=1}^{N} \delta_{g,g_i} w_i u_{i,c}, \quad \forall (g,c) \in \mathcal{G} \times \mathcal{C}, \tag{2}$$

where $\mathcal{C} \coloneqq \{1, \ldots, C\}$ is the set of cell type indexes, $w_i$ is a weight equal to the number of CpG sites in $r_i$, and $\delta_{x,y}$ is the Kronecker delta equal to 1 if $x = y$ and 0 otherwise. The prediction matrix is fed into a *deconvolver* model, which outputs mixture proportions $\hat{\mathbf{p}} \in \Delta^{C-1}$. Optionally, a post-hoc calibrator can be applied to obtain calibrated mixture proportions $\tilde{\mathbf{p}} \in \Delta^{C-1}$.

**Training.** *Syto* has three trainable parts: the read-level classifier, the deconvolver, and the calibrator. Each of them can be independently improved. The training dataset additionally contains, for each read $r_i$, the class label $c_i \in \mathcal{C}$, which is the cell type the read originates from. To train the classifier, we assign a label vector to each read. As we explain in Section 3.2, standard one-hot encoded labels (or *hard labels*) are insufficient for model convergence. We propose several labeling methods,

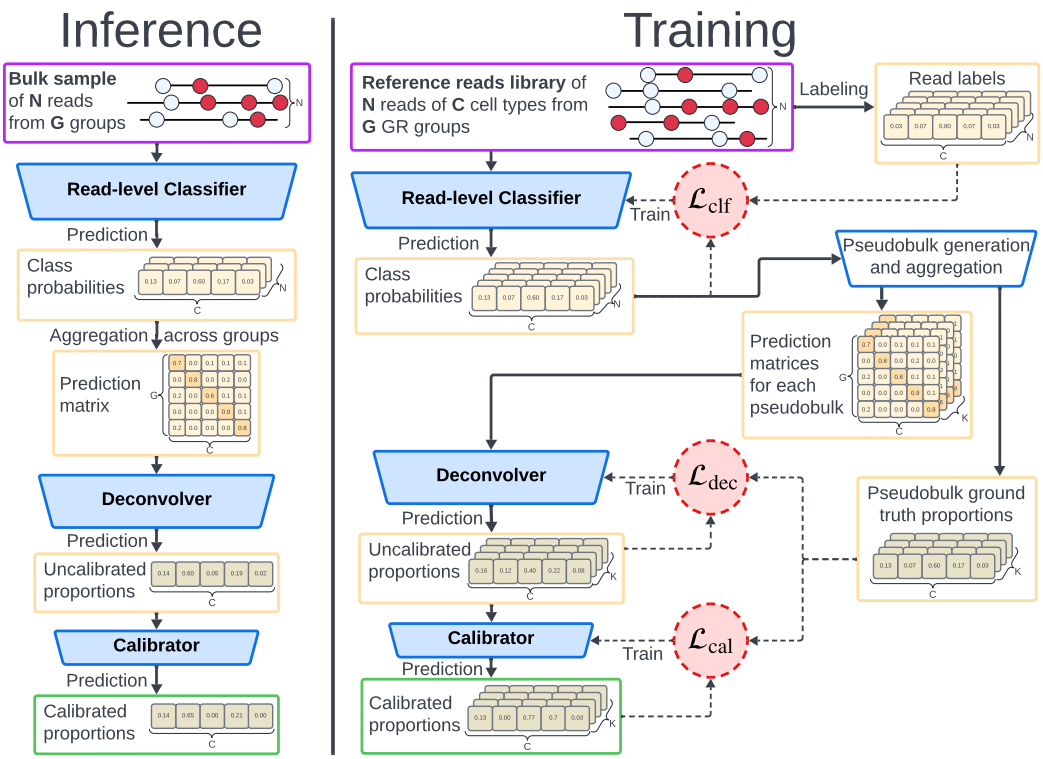


Figure 2: Proposed framework (*Syto*) for read-level classification-based cell-type deconvolution.

namely hard labels with an additional background class, a smoothed version of these hard labels (cf. Section 3.2.1), and a probabilistic labeling called *data-driven soft-labeling* (cf. Section 3.2.2). We extend and modify the three existing read-level classifiers (*CancerDetector*, *Dismir* and *MethylBERT*) and provide custom losses that enable convergence across different label types. Deconvolvers are trained on a dataset of *pseudobulks*, *i.e.* pairs of (ground truth pseudobulk proportions, prediction matrix) obtained by subsampling the original dataset with known proportions. The calibrators are trained on the deconvolvers' outputs, using the pseudobulk ground-truth proportions as targets. The inference and training of *Syto* is summarized in Figure 2.

We deliberately omit the selection of informative regions to emphasize that *Syto* can be applied out of the box with different GR groups, as illustrated in Figure S1. In particular, *Syto* can be applied to long reads by treating the whole genome as a single GR group, as in [17].

### 3.2 Labeling

Using standard one-hot encoded labels (or *hard labels*) as the classification target when the number of cell types is large impedes model convergence. Indeed, when deconvolving $C$ cell-types with an atlas where each GR group $g$ is specific for a single cell type $c^{(g)}$, only about $1/C$ ($\approx 3\%$ when $C = 39$) of reads originate from the cell type for which their GR group is specific (so-called *on-target* reads) and thus carry a distinctive omics signature. The remaining *off-target* reads, however, share similar omics signatures, despite being assigned to different cell-type labels. This yields an impossible learning task and results in degraded models that learn uniform class probabilities regardless of the context. Figure 3 shows clearly that *MethylBERT* does not converge when trained on standard hard labels. In this section, we propose several labeling schemes to overcome this issue.

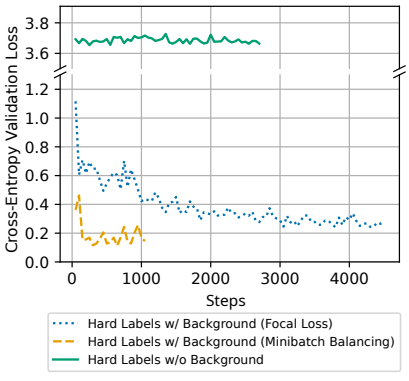


Figure 3: Validation cross-entropy loss for *MethylBERT* using hard labeling with or without background class.

### 3.2.1 Hard labels with background

**Hard labels with background.** To mitigate the detrimental effects of off-target reads, we can assign them to an additional *background* label $\perp \coloneqq C + 1$. Let $r_i$ be a DNA read in GR group $g$, $c_i \in \mathcal{C}$ be its original cell-type label and $c^{(g)}$ be the cell type for which GR group $g$ is specific. We define the hard label with background $y_i$ as $c_i$ if $c_i = c^{(g)}$ (*i.e.* if $r_i$ is on-target) and $\perp$ otherwise. This can be seen as an explicit and more flexible extension of the one-versus-all binary labeling scheme from [20], in which on-target and off-target reads are assigned labels 1 and 0, respectively.

**Canonical label smoothing on hard Labels with background.** We also apply the canonical label smoothing of Szegedy et al. [26] to hard labels with background. Let $\epsilon \in [0, 1]$ be the smoothing parameter. The smoothed hard label of $r_i$ is the vector $\bar{\mathbf{y}}_\mathbf{i} \in \Delta^C$ defined as:

$$\bar{y}_{i,c} \coloneqq (1 - \epsilon)\delta_{c,y_i} + \frac{\epsilon}{C}, \quad \forall c \in \mathcal{C} \cup \{\perp\}. \tag{3}$$

### 3.2.2 Data-driven soft labeling

**Biological rationale.** We argue that data-driven soft labels, constructed as probabilistic representations that capture the conditional distribution over cell types for each omics signature, are more appropriate than hard labels for read-level cell type classification. The main reason is that the mapping between omics signatures and cell types is not many-to-one, as in standard classification, but rather many-to-many, because a single omics signature (and in particular a single methylation pattern) in a given genomic region can be associated with multiple cell types. This is due, among several reasons, to imperfect propagation and maintenance of DNA methylation patterns across cell divisions [30], as well as high methylome variation across the cell cycle, with a sharp drop in average methylation during the S phase [31]. Both issues are further compounded by technical noise: the purification of cell types prior to sequencing is imperfect [13], and the techniques to infer the methylation status of CpG sites are noisy [32]. This ambiguity in the mapping was previously resolved by selecting highly discriminative regions that are hypomethylated in a specific cell type and methylated in all others [33, 34, 13, 35]. However, as we consider an increasing number of cell types, it becomes necessary to resort to less specific regions, where methylation patterns can only discriminate between non-singleton cell-type groups [10]. In this case, assigning a different hard label to reads sharing the same methylation pattern is not biologically meaningful and may impair the models' ability to learn the true conditional distribution of cell types.

**Data-driven soft labels as MLE of the true categorical distribution.** Let $r_i$ be a read with omics signature $s_i$, its *data-driven soft label* is defined as the empirical frequency of each class among the reads sharing that signature. For all $c \in \mathcal{C}$, we denote by $n_{i,c}$ the number of reads of class $c$ with signature $s_i$. The soft label $\mathbf{q_i} \coloneqq (q_{i,1}, \ldots, q_{i,C})^T \in \Delta^{C-1}$ of $r_i$ is the vector defined as:

$$q_{i,c} \coloneqq \frac{n_{i,c}}{\sum_{c' \in \mathcal{C}} n_{i,c'}}, \quad \forall c \in \mathcal{C}. \tag{4}$$

$\mathbf{q_i}$ is the unique maximum likelihood estimator of the true conditional categorical distribution $P(c \mid s_i)$. This results from a direct application of the classical MLE for categorical distributions [36].

**Coverage normalization by pooling.** The construction of soft labels faces two challenges: class imbalance and low per-signature read coverage. We handle imbalance by reweighting each class count by the inverse of its global frequency in the training data. The second issue is addressed by pooling class counts from the nearest signatures[3] within each genomic region, until a minimum read count $\tau$ is reached or a maximum distance threshold $\delta_{\max}$ is exceeded. Importantly, this count transfer between signatures is *not* symmetric. This procedure yields *data-driven soft labels with pooling*. We summarize the soft labeling procedure with class reweighting and pooling in Algorithm 1. Our adaptations differentiate it from the more general *KNN soft labels* by El Gayar et al. [37].

[3] In this work, we use the Jaccard distance over the signatures (cf. Section B.2). The extension of omics signatures to other signals such as chromatin accessibility, histone marks or gene expression will require adapting the distance.

**Algorithm 1** Data-driven soft labeling with pooling

**Input:** A set of signatures $\{s_i\}_{1 \le i \le S}$; for each signature $s_i$, the class counts $\mathbf{n_i} \in \mathbb{N}^C$ of reads having signature $s_i$; minimal count threshold $\tau \in \mathbb{N}$; maximal distance $\delta_{\max} \in \mathbb{R}^+$
**Output:** Soft label $\mathbf{q_i}^{\text{pooled}} \in \Delta^{C-1}$ for each signature $s_i$

**for** each class $c = 1, \ldots, C$ **do**
  $w_c \leftarrow \sum_{i=1}^{S} n_{i,c}$ ▷ compute class weights as global counts
**for** each GR group $g$ **do** ▷ aggregation is performed only within GR groups
  Compute $J[i,j] \leftarrow \text{JACCARDDIST}(s_i, s_j)$ for all pairs of signatures $(s_i, s_j)$ in $g$
  **for** each signature $s_i$ in region $g$ **do**
    $\tilde{\mathbf{n}}_\mathbf{i} \leftarrow (0, \ldots, 0)^T \in \mathbb{N}^C$ ▷ initialize pooled class counts for $s_i$
    **for** $j$ in $\text{argsort}(J[i,:])$ **do** ▷ iterate over signatures ($s_i$ included) from nearest to farthest
      **if** $J[i,j] > \delta_{\max}$ or $\text{sum}(\tilde{\mathbf{n}}_\mathbf{i}) \ge \tau$ **then break**
      $\tilde{\mathbf{n}}_\mathbf{i} \leftarrow \tilde{\mathbf{n}}_\mathbf{i} + \mathbf{n}_j$ ▷ add class counts of $s_j$ to the pooled counts of $s_i$
    $\hat{n}_{i,c} \leftarrow \tilde{n}_{i,c} \cdot w_c$ for each class $c$ ▷ weight the pooled counts
    $q_{i,c}^{\text{pooled}} \leftarrow \hat{n}_{i,c} \,/\, \sum_{c'=1}^{C} \hat{n}_{i,c'}$ for each class $c$ ▷ normalize

**Confidence-weighted Cross Entropy (CE) loss.** When $C$ is large, the majority of reads are off-target and thus have high-entropy soft labels. In this case, the standard CE loss leads the model to predict uniform distributions, which is not desirable. We use a confidence-weighted CE loss. For a read $r_i$ with soft label $\mathbf{t}_i$ of length $C$ and predicted class probabilities $\mathbf{u}_i$, the weight is $\big(\max(\mathbf{t}_i) - C^{-1}\big) / \big(1 - C^{-1}\big)$ and the loss is:

$$\mathcal{L}^{\text{CW}}(\mathbf{u}_i, \mathbf{t}_i) := \left( \frac{\max(\mathbf{t}_i) - C^{-1}}{1 - C^{-1}} \right) \text{CE}(\mathbf{u}_i, \mathbf{t}_i), \tag{5}$$

where $\text{CE}(\mathbf{u_i}, \mathbf{t_i}) := -\sum_c t_{i,c} \log u_{i,c}$ is the standard cross-entropy.

Our data-driven soft labels fit in the general field of Label Enhancement [24]. However, to our knowledge, we are the first to propose such a construction of soft labels. Moreover, in this setting, the most informative reads are those with low entropy, which is the opposite situation of the one in the existing literature [25].

### 3.3 Read-level classifiers

**Extension of existing classifiers.** We adapted *CancerDetector* [18] to the multiclass setting by extending the Beta-Bernoulli read likelihood model to all $C$ cell types, with per-marker, per-class parameter estimation by MLE (cf. Section B.4). We did a major refactoring of *MethylBERT* [20] to harmonize it with the Hugging Face Transformers library [38] and enable faster fine-tuning. We also substituted the original classification head with one with GR group attention (cf. Section B.5). The original *Dismir* [19] implementation does not integrate GR group context, making it untrainable on short reads with many classes, as the context cannot be inferred from the reads themselves. Therefore, we also use the classification head with GR group attention for *Dismir*.

**Lookup classifier as baseline.** To serve as a baseline, we introduced *Lookup Classifier*, a modification of 1-NN. For an input read $r$ with signature $s$, the predicted probability vector under soft labeling corresponds to the soft label computed on the training data. Under hard labeling with background, the prediction is instead the argmax over all reads sharing signature $s$, with ties resolved by averaging the one-hot encoded labels. If $s$ was never seen in the training data, for soft labels, the training signatures at minimal Jaccard distance to $s$ are aggregated, their normalized counts are pooled, and the resulting soft label is returned. For hard labels, we return the average of the one-hot encoded labels of the training signatures at minimal distance to $s$. If $r$ belongs to no GR group in the training data, a uniform distribution is returned for soft labels, and the background class is returned for hard labels.

### 3.4 Deconvolution models

All previous works relied in some way or another on maximum likelihood estimation (MLE) in order to infer the mixture proportions from the classifiers' outputs. However, preliminary experiments showed that MLE methods applied on top of the classifier outputs fail in our large-scale setting (cf. Section B.7). We therefore propose five distinct deconvolution models (or *deconvolvers*) to

learn the transfer function from the classifier output to the mixture proportions, which can all be used in combination with every classifier. Three of those models are supervised machine learning (ML) models, namely *XGBoost* [39] (referred to as *XGB*), and two neural networks: a multi-layer perceptron with three hidden layers (referred to as *MLP*) and a multi-layer perceptron with one wider hidden layer (referred to as shallow wide network or *SWN*), both implemented with PyTorch [40]. *XGB* is trained with RMSE loss, while the neural networks are trained with $loss = MSE + 0.5KL$. The other two methods are the widely used *Non-negative Least Squares* or *NNLS*, and a variant with sum-to-one constraint: *Probability Simplex Least Squares* or *PSLS*; both implemented with cvxpy [41]. *NNLS* and *PSLS* are initialized on the $C$ prediction matrices resulting from mixtures with a single cell type. ML deconvolvers are fitted on the $K$ prediction matrices obtained from the aggregation of the classifier predictions. For the pseudobulk experiment, we have simulated $K = 100{,}000$ pseudobulk mixtures for the training and validation split, and $50{,}000$ for the test split, via the sampling procedure described in Section D.2. We describe the models' architectures and fitting parameters in detail in Section B.8.

### 3.5 Calibration of predicted proportions

The output of deconvolvers is not necessarily well calibrated. Noting that the mapping between predicted and true proportions appears well approximated by a linear function (cf. Figure S2), we propose a simple, hyperparameter-free method we call *linear calibration*. For all cell-type $c \in \mathcal{C}$, we fit $a_c$ and $b_c$ the parameters of the best fitting linear model mapping the validation set predicted proportions of $c$ $(\hat{p}_{k,c})_{k=1}^{K}$ to the true proportions $(p_{k,c})_{k=1}^{K}$: $p_{k,c} = a_c \hat{p}_{k,c} + b_c + \epsilon$, where $\epsilon$ is the error term. The fitted parameters are arranged in two vectors $\mathbf{a} \coloneqq (a_1, \ldots, a_C)^T \in \mathbb{R}^C$ and $\mathbf{b} \coloneqq (b_1, \ldots, b_C) \in \mathbb{R}^C$ and inference is performed as follows:

$$\tilde{\mathbf{p}} \coloneqq f(\mathbf{a} \odot \hat{\mathbf{p}} + \mathbf{b}), \tag{6}$$

where $\odot$ is the Hadamard product, $\hat{\mathbf{p}}$ is the uncalibrated prediction vector, $\tilde{\mathbf{p}}$ is the calibrated prediction vector and $f$ is a function that corrects the small deviations of $\mathbf{a} \odot \hat{\mathbf{p}} + \mathbf{b}$ from the probability simplex. We present two variants of $f$: *clip-normalize*, where $f : \mathbf{x} \in \mathbb{R}^C \mapsto ((x_c)_+ / \sum_{c' \in \mathcal{C}} (x_{c'})_+)_{c \in \mathcal{C}}$ with $(x)_+ \coloneqq \max(0, x)$, and *simplex-projection*, which projects the uncalibrated vector onto the probability simplex using the threshold algorithm by Duchi et al. [42]. We performed preliminary experiments comparing the performance of our method with established calibration methods from the literature, namely Dirichlet calibration, vector scaling and temperature scaling [28, 29]. Results in Section B.9 show that our method and vector scaling yield better MSE. We proceed with these in the main experiments.

## 4 Experiments

### 4.1 Experimental setups

**External baselines.** We compared *Syto* to *UXM* [13], *CelFiE* [11], *Houseman CP* [9] and *EpiDISH* [43]. The motivation and implementation details are described in Section B.10.

**Pseudobulk experiment.** We chose the largest available (by number of cell-types) DNA methylation atlas of healthy adult human cell types for modeling [13, 44]. The dataset comprises 205 WGBS (Whole Genome Bisulfite Sequencing) libraries from 39 major somatic cell types, originating from 137 donors, and is publicly available at GEO under accession number GSE186458. We preprocessed the data as described in Section C.1. We grouped the DMRs of Loyfer et al. [13] into 39 GR groups, each containing up to 25 DMRs specific to a single cell type. To fit the read-level classifiers, we first filtered the dataset to keep only reads that overlapped with at least 4 CpGs in the reference atlas. We split the data into training, validation and test sets following the procedure in Section D.1, ensuring that every GR group has DNA reads in all splits while isolating biological sources whenever possible. Resulting split ratios were 62:19:19. Of the 205 biological samples, 4 were shared across all splits and 2 were shared between the validation and test splits. To mitigate the risk of overestimating the generalization capabilities of the methods, we also evaluated them on an OOD (Out-of-Distribution) RRBS (Reduced Representation Bisulfite Sequencing) dataset (see below). We trained and evaluated 360 possible permutations of methods of *Syto* (cf. Figure S6) and measured the MSE, MAE, KL divergence and Limits of Agreement on the predicted proportions (cf. metrics definitions in Section D.5). The fitting parameters for the classifiers, deconvolvers and calibrators are described in sections Section B.6, Section B.8 and Section B.9, respectively.

**Deconvolution of OOD RRBS dataset.** In the absence of a publicly available alternative dataset, we used the RRBS methylation atlas of 29 human tissues, composed of 521 non-cancer samples, by Li et al. [10] (GEO accession number GSE233417). For each tissue, we constructed the expected cell type proportions using the *Tabula Sapiens* dataset [45, 46] as a joining key (cf. Section C.4) The resulting 16 tissues had a matching rate in the range $88.7\% - 100\%$ and 260 biological samples in total (see tissue composition in Figure S4). *Tabula Sapiens* cell type proportions cannot be treated as ground truth due to technical biases in data collection [46]. Thus, instead of the usual deconvolution metrics, we use a proximal measure based on the insight that cell types not expected in the target tissue should not appear in the deconvolution results, or equivalently, that the predicted proportions of expected cell types should sum to a high value, ideally one. We call "Tissue Concordance Score" (or *TCS*) the sum of predicted proportions of cell types expected in the tissue (cf. Section D.5). To determine the set of expected cell types, we use matched cell types with individual contributions of at least 1% per tissue in *Tabula Sapiens*. For this experiment the train split is the fusion of the training and validation splits of the pseudobulk experiment, the validation split is the test split of the pseudobulk experiment, and the test split is the OOD RRBS dataset. Soft labels are computed using all splits of the pseudobulk experiment. We evaluated the same 360 permutations of *Syto* featured in the pseudobulk experiment. Additionally, we calculated *TCS* for *UXM* operating with a $10\times$ more DMRs, denoted *UXM U250*.

### 4.2 Results

**Pseudobulk experiment.** The main results are summarized in Table 1 and the integrality of results is in Section E.2. *Syto*, equipped with our extended *CancerDetector, MethylBERT* or *Dismir*, consistently outperforms the best baseline, *CelFiE*, achieving a $3.7\times$ reduction of the MSE (from $3.28\text{e}{-4}$ to $0.88\text{e}{-4}$) with the best setting (*MethylBERT* + data-driven soft labels w/ pooling + *NNLS* + Linear calibrator with simplex projection). Our linear calibrator with simplex projection appears as the best calibrator, uniformly improving the MSE, including on baselines. Similarly, our soft labeling with pooling uniformly outperforms alternative labeling schemes in terms of MSE.

**OOD RRBS dataset.** Each labeling scheme's best classifier-deconvolver-calibrator combination yielded significantly higher TCS than the best baseline *CelFiE* with the original reference (Figure 4) with *Syto* variants not using canonical soft labeling surpassing *UXM U250*, which uses $10\times$ more DMRs. Data-driven soft labeling without pooling with *Dismir* delivered a robust improvement: all 20 deconvolver-classifier combinations outperformed *CelFiE*, with 11 at $p \leq 0.05$. *CancerDetector* with training-counts prior was on par, also surpassing *CelFiE* in all 20 combinations and 11 at $p \leq 0.05$. Full counts are in Table S18.

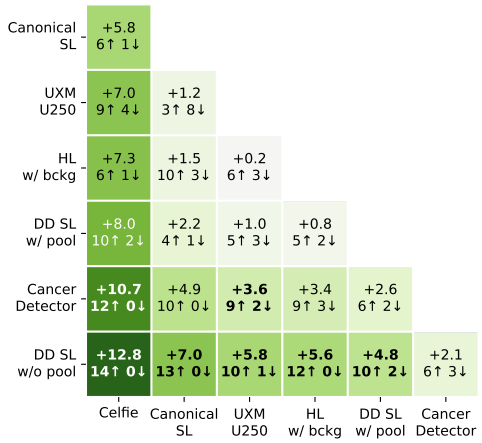


Figure 4: Pairwise mean differences in TCS (row minus column, best configuration per category). The values in parentheses indicate the number of tissues where significantly higher (↑) or lower (↓) TCS is observed for the row method relative to column method (Wilcoxon signed-rank test, $p \leq 0.05$).

## 5 Discussion

A clear pattern supported by the two experiments is that how much work the labeling schemes do depends on the classifier's inductive bias (see Figures S8 and S9). Starting from the pseudobulk experiment, our simple *1-NN Lookup* improves significantly when switched from hard labeling to data-driven soft labels and dramatically when pooling is enabled. *Dismir* improves consistently, but the gain is much smaller. *MethylBERT* is on par with *Dismir* on improvement magnitude, but suffers when pooling is off. In the OOD experiment, *1-NN Lookup* and *MethylBERT* only work with soft labels and *Dismir* is comparatively insensitive to the choice of labeling scheme. *MethylBERT* is consistently better with pooling; however, its results without pooling are confounded by the necessity of using balanced minibatching (see Section B.6). *1-NN Lookup* and *Dismir* reach higher TCS without pooling, yet these gains must be viewed through the lens of TCS limitations (see Section E.7). Importantly, soft labeling can elevate even limited classifiers: uncalibrated *1-NN Lookup* fitted on soft labels with pooling and paired with *SWN* deconvolver beats every examined uncalibrated external baseline in both experiments.

Table 1: Pseudobulk experiment test split deconvolution metrics. Best result in each section is in **bold**, second best is underlined, best across all sections is highlighted. Best result per classifier across labeling schemes is *italicized*. For all metrics, we indicate the 95% confidence interval computed with BCa bootstrap [47] with 10,000 resamples. Results are reported for deconvolvers fitted on the top 156 features unless $^{\dagger}$ is added to the deconvolver name, in which case they are fitted on the combination of diagonal + background column of the feature matrices (cf. Section D.4).

| Classifier | Labeling Scheme (or prior type) | Deconvolver | Calibrator | $R^2$ ($\times 10^{-2}$) | MAE ($\times 10^{-3}$) | MSE ($\times 10^{-4}$) |
|---|---|---|---|---|---|---|
| *(a) Syto variants with ablation of calibration — for each classifier and labeling/prior, the best deconvolver by test MSE is reported* | | | | | | |
| Dismir [19] (CNN+LSTM) | Hard labels w/ bckg. | SWN | None | 97.98 [97.94, 98.01] | 3.44 [3.42, 3.45] | 1.68 [1.65, 1.71] |
| | Canonical Soft Labels | MLP | None | 97.49 [97.45, 97.54] | 3.65 [3.63, 3.67] | 2.08 [2.05, 2.12] |
| | DD Soft Labels w/o pool. | SWN | None | 98.24 [98.20, 98.27] | 3.19 [3.18, 3.20] | 1.46 [1.43, 1.49] |
| | DD Soft Labels w/ pool. | SWN | None | *98.32 [98.29, 98.34]* | *3.15 [3.14, 3.16]* | *1.40 [1.38, 1.42]* |
| MethylBERT [20] | Hard labels w/ bckg. | NNLS$^{\dagger}$ | None | 98.06 [98.03, 98.08] | 3.53 [3.52, 3.55] | 1.61 [1.59, 1.63] |
| | Canonical Soft Labels | NNLS$^{\dagger}$ | None | 98.12 [98.10, 98.14] | 3.55 [3.53, 3.56] | 1.56 [1.54, 1.58] |
| | DD Soft Labels w/o pool. | SWN | None | 97.99 [97.96, 98.01] | 3.49 [3.48, 3.51] | 1.67 [1.65, 1.69] |
| | DD Soft Labels w/ pool. | SWN | None | ***98.49 [98.48, 98.51]*** | *3.08 [3.07, 3.09]* | ***1.25 [1.23, 1.26]*** |
| Lookup Classifier (1-NN) | Hard labels w/ bckg. | SWN$^{\dagger}$ | None | 86.43 [86.31, 86.56] | 10.29 [10.25, 10.32] | 11.26 [11.16, 11.37] |
| | DD Soft Labels w/o pool. | SWN | None | 88.62 [88.49, 88.75] | 9.60 [9.57, 9.64] | 9.45 [9.33, 9.56] |
| | DD Soft Labels w/ pool. | SWN | None | *96.72 [96.67, 96.76]* | *4.42 [4.40, 4.44]* | *2.73 [2.69, 2.77]* |
| CancerDetector [18] | Prior: Training counts | PSLS | None | 98.41 [98.39, 98.44] | 3.05 [3.04, 3.07] | 1.32 [1.30, 1.34] |
| | Prior: Uniform | PSLS | None | 98.44 [98.42, 98.46] | **3.02 [3.01, 3.04]** | 1.30 [1.28, 1.31] |
| *(b) Syto variants with calibration — for each classifier and labeling/prior, the best (deconvolver + calibrator) by test MSE is reported* | | | | | | |
| Dismir [19] (CNN+LSTM) | Hard labels w/ bckg. | NNLS$^{\dagger}$ | Lin. simplex | 98.73 [98.71, 98.74] | 2.92 [2.91, 2.93] | 1.06 [1.05, 1.07] |
| | Canonical Soft Labels | NNLS$^{\dagger}$ | Lin. simplex | 97.98 [97.95, 98.01] | 3.39 [3.38, 3.41] | 1.68 [1.65, 1.70] |
| | DD Soft Labels w/o pool. | NNLS | Lin. simplex | 98.89 [98.88, 98.90] | 2.72 [2.71, 2.73] | 0.92 [0.91, 0.93] |
| | DD Soft Labels w/ pool. | NNLS | Lin. simplex | *98.93 [98.92, 98.94]* | ***2.66 [2.65, 2.67]*** | *0.89 [0.88, 0.90]* |
| MethylBERT [20] | Hard labels w/ bckg. | NNLS$^{\dagger}$ | Lin. simplex | 98.57 [98.56, 98.59] | 3.09 [3.08, 3.10] | 1.18 [1.17, 1.20] |
| | Canonical Soft Labels | PSLS | Lin. simplex | 98.77 [98.76, 98.78] | 3.01 [3.00, 3.02] | 1.02 [1.01, 1.03] |
| | DD Soft Labels w/o pool. | PSLS | Lin. simplex | 98.62 [98.60, 98.63] | 3.01 [2.99, 3.02] | 1.15 [1.14, 1.16] |
| | DD Soft Labels w/ pool. | NNLS | Lin. simplex | ***98.94 [98.93, 98.95]*** | *2.70 [2.69, 2.71]* | ***0.88 [0.87, 0.89]*** |
| Lookup Classifier (1-NN) | Hard labels w/ bckg. | SWN | Lin. simplex | 91.56 [91.46, 91.65] | 7.89 [7.86, 7.92] | 7.01 [6.93, 7.09] |
| | DD Soft Labels w/o pool. | NNLS | Lin. simplex | 96.67 [96.62, 96.72] | 4.56 [4.54, 4.58] | 2.76 [2.72, 2.80] |
| | DD Soft Labels w/ pool. | PSLS | Lin. simplex | *98.27 [98.25, 98.29]* | *3.27 [3.26, 3.28]* | *1.44 [1.42, 1.45]* |
| CancerDetector [18] | Prior: Training counts | PSLS | None | 98.90 [98.89, 98.91] | 2.74 [2.73, 2.75] | 0.91 [0.90, 0.92] |
| | Prior: Uniform | PSLS | None | 98.91 [98.90, 98.92] | 2.72 [2.71, 2.73] | 0.91 [0.90, 0.91] |
| *(c) External baselines with and without our calibration* | | | | | | |
| *CelFiE* [11] | | | None | 96.05 [96.00, 96.10] | 6.42 [6.40, 6.44] | 3.28 [3.24, 3.32] |
| | | | Lin. simplex | **98.03 [98.01, 98.06]** | **3.47 [3.45, 3.48]** | **1.63 [1.61, 1.65]** |
| *EpiDISH RPC* [43] | | | None | 89.41 [89.28, 89.54] | 10.20 [10.16, 10.23] | 8.79 [8.68, 8.90] |
| | | | Vec. scaling | 95.95 [95.92, 95.98] | 6.02 [6.00, 6.04] | 3.36 [3.34, 3.38] |
| *Houseman CP* [43, 9] | | | None | 90.94 [90.83, 91.05] | 9.25 [9.22, 9.28] | 7.52 [7.42, 7.62] |
| | | | Lin. simplex | 96.51 [96.47, 96.55] | 4.91 [4.89, 4.92] | 2.90 [2.86, 2.93] |
| *UXM* [13] | | | None | 94.86 [94.78, 94.93] | 6.49 [6.47, 6.52] | 4.27 [4.20, 4.34] |
| | | | Lin. simplex | 96.74 [96.68, 96.79] | 4.32 [4.30, 4.34] | 2.71 [2.66, 2.76] |

Our labeling-free extension of *CancerDetector*, while operating on $\alpha$-values, yields the second-best uncalibrated combination, challenging the need for pattern-level classifiers. This advantage is likely specific to the present dataset, whose short reads and selected regions make average methylation informative by design. The gap with pattern-sensitive methods is expected to increase for longer reads and more complex methylation patterns [20].

The choice of deconvolver and calibrator is less consequential than that of the labeling scheme and classifier, but is essential to distinguish between good and best combinations. *NNLS* and *PSLS* dominate pseudobulk performance when paired with our linear calibration with simplex projection, but loose most of their advantage in OOD. *SWN* is stronger before calibration and stable across evaluation settings. *XGB* performs worst in pseudobulk but second best in OOD, albeit with the largest variation in TCS. And *MLP* is the only deconvolver absent from the Pareto front of the two metrics (see Figures S7 and S8). The rank correlation between pseudobulk MSE and OOD TCS is only $+0.37$, lower still for *NNLS* and *PSLS*. Therefore, deconvolver selection based on pseudobulk performance alone, as is common in the field, may be suboptimal.

The usefulness of TCS depends on its own limitations. We screened it for the failure modes listed in Section 6 (details in Section E.7) and found two major artifacts: under-counted keratinocytes in skin and potential over-allocation to smooth muscle in the ovary. These penalise all methods equally in the first case and favour *Syto* in the second; yet even when the ovary is excluded, the relative ranking

of the labeling schemes is preserved and *Syto* maintains its advantage over the baselines, albeit with a reduced TCS gap.

Lastly, as discussed at length in Section E.4, the bias correction by calibrating on pseudobulk is useful beyond *Syto*. All four examined baselines improve their scores and benefits translating to OOD. Systematic, linearly correctable bias therefore appears to be a general property of reference-based setups, and the calibrator can be adopted independently of the rest of the framework.

## 6 Limitations

The reported performance for learnable classifiers (*Dismir*, *MethylBERT*) and deconvolvers (*XGB*, *MLP*, *SWN*) should be considered a lower bound, as optimal hyperparameters were not pursued.

We rely on pseudobulk mixtures, a field standard. Evaluation with real samples containing known ground-truth proportions would be preferable, but no suitable read-level dataset was available at the time. We therefore use an additional OOD experiment with TCS as a measure to strengthen our benchmark. TCS has several prominent failure modes: (i) when a cell type with relatively modest contribution is overpredicted (TCS rewards the wrong method), (ii) when a cell type truly present in a tissue is not listed as expected (TCS penalizes the right method), or (iii) when a cell type is listed as expected in a majority of tissues and a method is biased to overly predict it (TCS rewards it). We screened for these issues and found limited impact (see Section E.7).

As noted in [48, 49], the choice of genomic regions affects deconvolution performance. Our main experiments rely on a large set of regions selected with `wgbstools` [50]. We feature a supplementary experiment focusing on a single alternative strategy for the regions selection (see Section E.6), far from an exhaustive sensitivity analysis. Furthermore, our training data comes from a single WGBS study based on FACS-purified samples of 137 donors, many diagnosed with various cancers despite atlas focus on normal human cells (samples use tumour-adjacent normal tissue material). The purity estimates in this data are questionable, confounded by marker selection, and are below 80% for 17 samples out of 205. The OOD evaluation rests on a single RRBS dataset: broader benchmarking data would be desirable.

The 39 modeled cell types come from various levels of the ontological tree, mostly higher in hierarchy and therefore not suitable for clinical analyses where granular assignment is a differentiating factor. As with most reference-based methods, *Syto* in its published version cannot predict cell types absent from the reference. Necessary extensions are foreseeable and are planned for future work. The absence of malignant cell types in the reference limits the application of the *Syto* models trained on it. Such models can in principle flag cancer indirectly [20], but only as a secondary check and not as a reliable estimate. While there is some evidence that *Syto* works well with more heterogeneous and related markers characteristic of malignant cell types (see Section E.6), the reference must be extended with target tumours before models trained on it can be used for tumour purity estimation and cancer classification. We therefore frame these experiments as a methodological proof of concept, leaving translational validation for future work.

## 7 Conclusion

This study presents *Syto*, a modular framework for cell-type classification. It extends and includes existing read-level classifiers to support a broad range of cell types and introduces learnable deconvolvers and post-hoc calibrators. We propose several labeling schemes to enable convergence in large multi-class settings. Our data-driven soft-labeling approach outperforms all other evaluated labeling strategies, including our generalization of hard labels with background, achieving superior results in both pseudobulk and OOD scenarios, and surpassing the best examined baseline *CelFiE* by a substantial margin. Beyond improving deconvolution accuracy, scaling read-level approaches to larger reference panels enables analyses that retain cell-type information at the level of individual reads, providing access to local epigenetic variation that aggregate-based methods cannot capture. The promising results observed across 39 healthy cell types suggest the feasibility of progressively expanding to larger and more complex reference panels. A primary extension involves incorporating malignant cell types, enabling *Syto* to address tumor classification and purity estimation as specific instances of deconvolution. Furthermore, integration of multi-omics signatures and third-generation long-read sequencing data will facilitate the capture of more complex epigenetic patterns and enhance differentiating capabilities, potentially enabling new biological insights and, following appropriate validation, supporting clinical decision-making in diseases associated with epigenetic alterations.

## Acknowledgments and Disclosure of Funding

We acknowledge funding from the KU Leuven BOFZAP starting grant (STG/22/024) and FWO Runner-up project MUTAHIST-PoP (G0ATW25N) to P.L.

## References

[1] Melanie Ehrlich, Miguel A Gama-Sosa, Lan-Hsiang Huang, Rose Marie Midgett, Kenneth C Kuo, Roy A McCune, and Charles Gehrke. Amount and distribution of 5-methylcytosine in human dna from different types of tissues or cells. *Nucleic acids research*, 10(8):2709–2721, 1982.

[2] Miguel A Gama-Sosa, Valerie A Slagel, Ronald W Trewyn, Ronald Oxenhandler, Kenneth C Kuo, Charles W Gehrke, and Melanie Ehrlich. The 5-methylcytosine content of dna from human tumors. *Nucleic acids research*, 11(19):6883–6894, 1983.

[3] David Capper, David TW Jones, Martin Sill, Volker Hovestadt, Daniel Schrimpf, Dominik Sturm, Christian Koelsche, Felix Sahm, Lukas Chavez, David E Reuss, et al. Dna methylation-based classification of central nervous system tumors. *Nature*, 555(7697):469–474, 2018.

[4] C Vermeulen, M Pagès-Gallego, L Kester, MEG Kranendonk, P Wesseling, N Verburg, P de Witt Hamer, EJ Kooi, L Dankmeijer, J Van Der Lugt, et al. Ultra-fast deep-learned cns tumor classification during surgery. *Nature*, 622(7984):842–849, 2023.

[5] Dongsheng Yuan, Robin Jugas, Petra Pokorna, Jaroslav Sterba, Ondrej Slaby, Simone Schmid, Christin Siewert, Brendan Osberg, David Capper, Skarphedinn Halldorsson, et al. crossnn is an explainable framework for cross-platform dna methylation-based classification of tumors. *Nature cancer*, 6(7):1283–1294, 2025.

[6] Syed Haider, Svitlana Tyekucheva, Davide Prandi, Natalie S Fox, Jaeil Ahn, Andrew Wei Xu, Angeliki Pantazi, Peter J Park, Peter W Laird, Chris Sander, et al. Systematic assessment of tumor purity and its clinical implications. *JCO precision oncology*, 4:995–1005, 2020.

[7] Douglas Hanahan. Hallmarks of cancer: new dimensions. *Cancer discovery*, 12(1):31–46, 2022.

[8] Lianjie Wang, Hui Geng, Yujie Liu, Lei Liu, Yanhua Chen, Fanchen Wu, Zhiyi Liu, Shiliang Ling, Yan Wang, and Lihong Zhou. Hot and cold tumors: Immunological features and the therapeutic strategies. *MedComm*, 4(5):e343, 2023.

[9] Eugene Andres Houseman, William P Accomando, Devin C Koestler, Brock C Christensen, Carmen J Marsit, Heather H Nelson, John K Wiencke, and Karl T Kelsey. Dna methylation arrays as surrogate measures of cell mixture distribution. *BMC bioinformatics*, 13:1–16, 2012.

[10] Shuo Li, Weihua Zeng, Xiaohui Ni, Qiao Liu, Wenyuan Li, Mary L Stackpole, Yonggang Zhou, Arjan Gower, Kostyantyn Krysan, Preeti Ahuja, et al. Comprehensive tissue deconvolution of cell-free dna by deep learning for disease diagnosis and monitoring. *Proceedings of the National Academy of Sciences*, 120 (28):e2305236120, 2023.

[11] Christa Caggiano, Barbara Celona, Fleur Garton, Joel Mefford, Brian L Black, Robert Henderson, Catherine Lomen-Hoerth, Andrew Dahl, and Noah Zaitlen. Comprehensive cell type decomposition of circulating cell-free dna with celfie. *Nature communications*, 12(1):2717, 2021.

[12] Pia Keukeleire, Stavros Makrodimitris, and Marcel Reinders. Cell type deconvolution of methylated cell-free dna at the resolution of individual reads. *NAR Genomics and Bioinformatics*, 5(2):lqad048, 2023.

[13] Netanel Loyfer, Judith Magenheim, Ayelet Peretz, Gordon Cann, Joerg Bredno, Agnes Klochendler, Ilana Fox-Fisher, Sapir Shabi-Porat, Merav Hecht, Tsuria Pelet, et al. A dna methylation atlas of normal human cell types. *Nature*, 613(7943):355–364, 2023.

[14] Maísa R Ferro dos Santos, Edoardo Giuili, Andries De Koker, Celine Everaert, and Katleen De Preter. Computational deconvolution of dna methylation data from mixed dna samples. *Briefings in Bioinformatics*, 25(3):bbae234, 2024.

[15] Mary L Stackpole, Weihua Zeng, Shuo Li, Chun-Chi Liu, Yonggang Zhou, Shanshan He, Angela Yeh, Ziye Wang, Fengzhu Sun, Qingjiao Li, et al. Cost-effective methylome sequencing of cell-free dna for accurately detecting and locating cancer. *Nature communications*, 13(1):5566, 2022.

[16] Melanie Ehrlich. Dna hypomethylation in cancer cells. *Epigenomics*, 1(2):239–259, 2009.

[17] Toby M Baker, Nedas Matulionis, Cassidy Andrasz, Dan Gerke, Natalia Garcia-Dutton, David Atkinson, Kami Chiotti, Selina Wu, Suchita Lulla, Jieun Oh, et al. Genome-wide classification of tumor-derived reads from bulk long-read sequencing. *bioRxiv*, pages 2026–03, 2026.

[18] Wenyuan Li, Qingjiao Li, Shuli Kang, Mary Same, Yonggang Zhou, Carol Sun, Chun-Chi Liu, Lea Matsuoka, Linda Sher, Wing Hung Wong, et al. Cancerdetector: ultrasensitive and non-invasive cancer detection at the resolution of individual reads using cell-free dna methylation sequencing data. *Nucleic acids research*, 46(15):e89–e89, 2018.

[19] Jiaqi Li, Lei Wei, Xianglin Zhang, Wei Zhang, Haochen Wang, Bixi Zhong, Zhen Xie, Hairong Lv, and Xiaowo Wang. Dismir: Deep learning-based noninvasive cancer detection by integrating dna sequence and methylation information of individual cell-free dna reads. *Briefings in bioinformatics*, 22(6):bbab250, 2021.

[20] Yunhee Jeong, Clarissa Gerhäuser, Guido Sauter, Thorsten Schlomm, Karl Rohr, and Pavlo Lutsik. Methylbert enables read-level dna methylation pattern identification and tumor deconvolution using a transformer-based model. *Nature Communications*, 16(1):788, 2025.

[21] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[22] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2980–2988, 2017.

[23] Xin Geng. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28(7): 1734–1748, 2016.

[24] Ning Xu, Yun-Peng Liu, and Xin Geng. Label enhancement for label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 33(4):1632–1643, 2021. doi: 10.1109/TKDE.2019.2947040.

[25] Jing Wang and Xin Geng. Classification with label distribution learning. In *IJCAI*, volume 1, page 2, 2019.

[26] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[27] Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? *Advances in neural information processing systems*, 32, 2019.

[28] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

[29] Meelis Kull, Miquel Perello Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. *Advances in neural information processing systems*, 32, 2019.

[30] Toshikazu Ushijima, Naoko Watanabe, Eriko Okochi, Atsushi Kaneda, Takashi Sugimura, and Kazuaki Miyamoto. Fidelity of the methylation pattern and its variation in the genome. *Genome Research*, 13(5): 868–874, 2003. doi: 10.1101/gr.969603.

[31] Christoph Geisenberger, Jeroen van den Berg, Vincent van Batenburg, Buys de Barbanson, Anna Lyubimova, Joe Verity-Legg, Xiufei Chen, Yibin Liu, Chun-Xiao Song, Jeroen de Ridder, and Alexander van Oudenaarden. Single-cell multi-omic detection of dna methylation and histone modifications reconstructs the dynamics of epigenomic maintenance. *Nature Methods*, 22(10):2042–2051, Oct 2025. ISSN 1548-7105. doi: 10.1038/s41592-025-02847-4. URL `https://doi.org/10.1038/s41592-025-02847-4`.

[32] Li Zhou, Hong Kiat Ng, Daniela I. Drautz-Moses, Stephan C. Schuster, Stephan Beck, Changhoon Kim, John Campbell Chambers, and Marie Loh. Systematic evaluation of library preparation methods and sequencing platforms for high-throughput whole genome bisulfite sequencing. *Scientific Reports*, 9(1): 10383, Jul 2019. ISSN 2045-2322. doi: 10.1038/s41598-019-46875-5. URL `https://doi.org/10.1038/s41598-019-46875-5`.

[33] Matthias Zilbauer, Tim F Rayner, Christine Clark, Alison J Coffey, Chris J Joyce, Priit Palta, Aarno Palotie, Paul A Lyons, and Kenneth GC Smith. Genome-wide methylation analyses of primary human leukocyte subsets identifies functionally important cell-type–specific hypomethylated regions. *Blood, The Journal of the American Society of Hematology*, 122(25):e52–e60, 2013.

[34] Michael B Stadler, Rabih Murr, Lukas Burger, Robert Ivanek, Florian Lienert, Anne Schöler, Erik van Nimwegen, Christiane Wirbelauer, Edward J Oakeley, Dimos Gaidatzis, et al. Dna-binding factors shape the mouse methylome at distal regulatory regions. *Nature*, 480(7378):490–495, 2011.

[35] Dirk Schübeler. Function and information content of dna methylation. *Nature*, 517(7534):321–326, 2015.

[36] Kevin P. Murphy. *Machine learning : a probabilistic perspective*. Adaptive computation and machine learning series. MIT, 2012. ISBN 9780262018029 0262018020.

[37] Neamat El Gayar, Friedhelm Schwenker, and Günther Palm. A study of the robustness of knn classifiers trained using soft labels. In *IAPR Workshop on Artificial Neural Networks in Pattern Recognition*, pages 67–80. Springer, 2006.

[38] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45, 2020.

[39] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[40] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[41] Steven Diamond and Stephen Boyd. Cvxpy: A python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

[42] John Duchi, Shai Shalev-Shwartz, Yoram Singer, and Tushar Chandra. Efficient projections onto the l1-ball for learning in high dimensions. *Proceedings of the 25th International Conference on Machine Learning*, pages 272–279, 07 2008. doi: 10.1145/1390156.1390191.

[43] Andrew E. Teschendorff, Charles E. Breeze, Shijie C. Zheng, and Stephan Beck. A comparison of reference-based algorithms for correcting cell-type heterogeneity in epigenome-wide association studies. *BMC Bioinformatics*, 18(1):105, Feb 2017. ISSN 1471-2105. doi: 10.1186/s12859-017-1511-5. URL `https://doi.org/10.1186/s12859-017-1511-5`.

[44] Joshua Moss, Roni Ben-Ami, Ela Shai, Ofer Gal-Rosenberg, Yosef Kalish, Agnes Klochendler, Gordon Cann, Benjamin Glaser, Ariela Arad, Ruth Shemer, et al. Megakaryocyte-and erythroblast-specific cell-free dna patterns in plasma and platelets reflect thrombopoiesis and erythropoiesis levels. *Nature communications*, 14(1):7542, 2023.

[45] Stephen R Quake, Tabula Sapiens Consortium, et al. Tabula sapiens reveals transcription factor expression, senescence effects, and sex-specific features in cell types from 28 human organs and tissues. *bioRxiv*, pages 2024–12, 2025.

[46] The Tabula Sapiens Consortium*, Robert C Jones, Jim Karkanias, Mark A Krasnow, Angela Oliveira Pisco, Stephen R Quake, Julia Salzman, Nir Yosef, Bryan Bulthaup, Phillip Brown, et al. The tabula sapiens: A multiple-organ, single-cell transcriptomic atlas of humans. *Science*, 376(6594):eabl4896, 2022.

[47] Bradley Efron. Better bootstrap confidence intervals. *Journal of the American Statistical Association*, 82 (397):171–185, 1987. ISSN 01621459, 1537274X. URL `http://www.jstor.org/stable/2289144`.

[48] Kobe De Ridder, Huiwen Che, Kaat Leroy, and Bernard Thienpont. Benchmarking of methods for dna methylome deconvolution. *Nature Communications*, 15(1):4134, 2024.

[49] Tongyue Sun, Jinqi Yuan, Yacheng Zhu, Jingqi Li, Shen Yang, Junpeng Zhou, Xinzhou Ge, Susu Qu, Wei Li, Jingyi Jessica Li, et al. Systematic evaluation of methylation-based cell type deconvolution methods for plasma cell-free dna. *Genome biology*, 25(1):318, 2024.

[50] Netanel Loyfer, Jonathan Rosenski, and Tommy Kaplan. wgbstools: a computational suite for dna methylation sequencing data analysis. *Life Science Alliance*, 9(4), 2026.

[51] Edoardo Giuili, Sofie Van de Velde, Sam Kint, Maísa R Ferro dos Santos, Lotte Cornelli, Sofie Roelandt, Kathleen Schoofs, Renske Imschoot, Ruben Van Paemel, Leander Meuris, et al. A benchmark of dna methylation deconvolution methods for tumoral fraction estimation using deconflow. *bioRxiv*, pages 2025–11, 2025.

[52] Yunhee Jeong, Lisa Barros de Andrade e Sousa, Dominik Thalmeier, Reka Toth, Marlene Ganslmeier, Kersten Breuer, Christoph Plass, and Pavlo Lutsik. Systematic evaluation of cell-type deconvolution pipelines for sequencing-based bulk dna methylomes. *Briefings in bioinformatics*, 23(4):bbac248, 2022.

[53] Junyan Song and Pei-Fen Kuan. A systematic assessment of cell type deconvolution algorithms for dna methylation data. *Briefings in bioinformatics*, 23(6):bbac449, 2022.

[54] Xiaolong Guo and Andrew E Teschendorff. Guidelines on optimizing dna methylation reference panels for cell-type deconvolution. *Communications Biology*, 2026.

[55] James McLaughlin, Josh Lagrimas, Haider Iqbal, Helen Parkinson, and Henriette Harmse. Ols4: a new ontology lookup service for a growing interdisciplinary knowledge ecosystem. *Bioinformatics*, 41(5): btaf279, 2025.

# A Source code and experiments data

The production code that can be used to reproduce the paper results is distributed in this frozen release: `https://github.com/CompEpigen/syto/releases/tag/v1.0.0`.

The run configs for every step in each experiment, along with preprocessed data, fitted models, and generated pseudobulks, are deposited in the KU Leuven Research Data Repository (DOI: 10.48804/JM4BOS). The full mirror of the same data can be found at `https://huggingface.co/datasets/CompEpigen/syto.1.0`.

# B Supplementary material on methods

## B.1 General framework

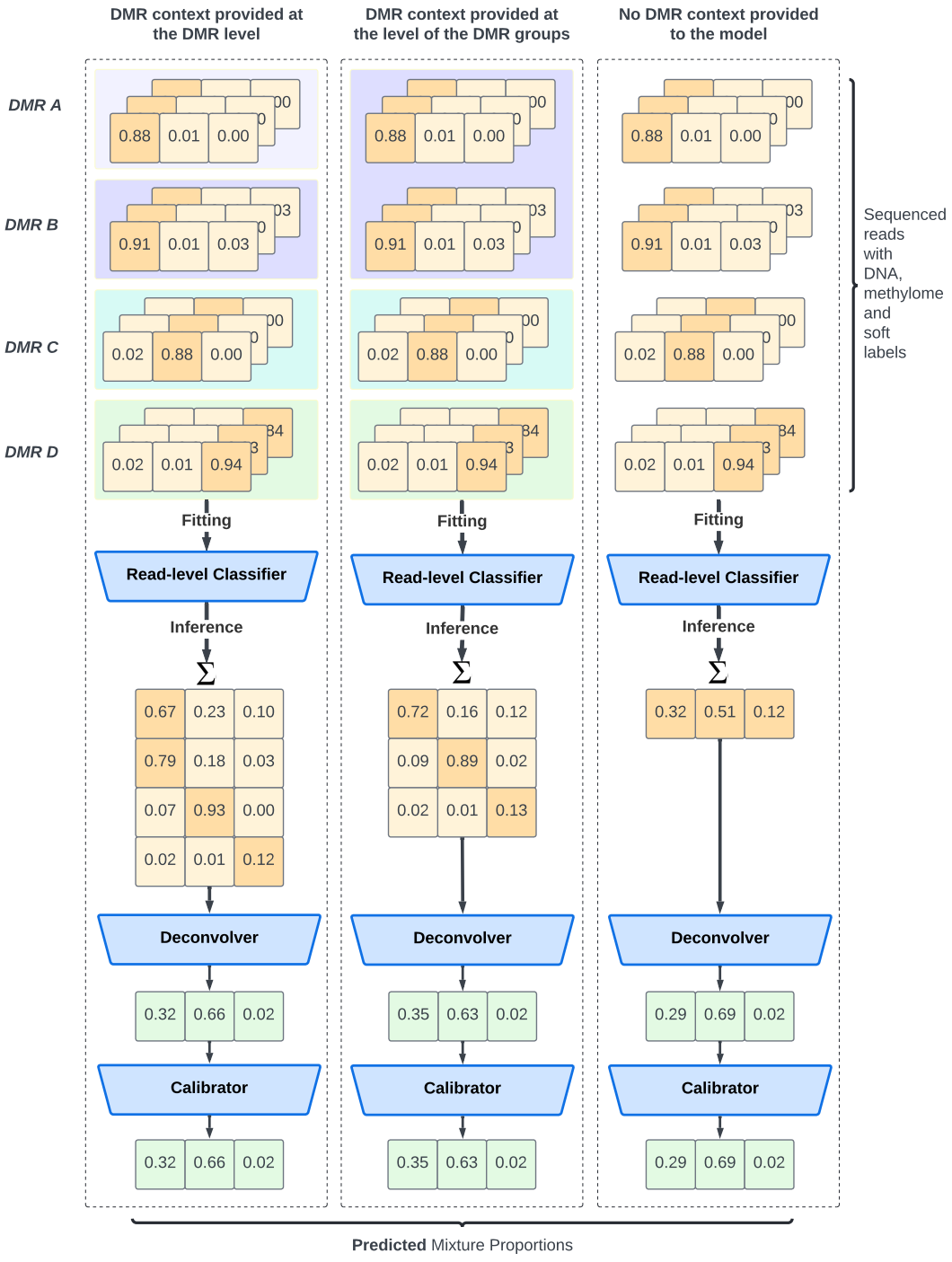


Figure S1: Implementation of proposed framework for different reference aggregation levels. This work utilizes DMR groups (center).

### B.2 Jaccard distance-based smoothing of soft labels

As mentioned in the main paper, for signatures with low read coverage, we pool the counts of similar signatures using the Jaccard distance. In this section, we describe precisely how we apply the Jaccard distance to signatures.

In our setting, we define an omics signature as a set of pairs (position, methylation status) where position is the position of the CpG site in the human genome (in accordance with selected reference) and methylation status is either 0 for unmethylated or 1 for methylated. Since every signature is a set, we can define the Jaccard distance between two signatures $s_1$ and $s_2$ in the classical way as:

$$\textsc{JaccardDist}(s_1, s_2) \coloneqq 1 - \frac{|s_1 \cap s_2|}{|s_1 \cup s_2|} = \frac{|s_1 \Delta s_2|}{|s_1 \cup s_2|}, \tag{7}$$

where $\Delta$ denotes the symmetric difference between sets and $|\cdot|$ denotes the cardinality of a set.

### B.3 Sensitivity of *Syto* to the choice of pooling parameters

In our data-driven soft labeling approach, two parameters control how the pooling is performed: the Jaccard distance threshold $\delta$ and the minimum count threshold $\tau$ to reach per signature.

In this section, we discuss the heuristics that lead to the choice of these parameters and we show the results of a sensitivity analysis on the choice of these parameters.

In all of our experiments, we set $\delta = 0.41$ and $\tau = 30$. Those choices were made based on heuristics derived from the training data:

- Distance threshold of $\delta = 0.41$: More than 80% of reads in our training data contain 4–7 CpGs. A Jaccard distance threshold of 0.41 allows reads in this range to differ by at most one CpG, reflecting the high local correlation of CpG methylation. Differences in two or more CpGs are more likely to represent a biological change. This threshold choice groups 4-CpG reads with methylation levels of 0 and 0.25, as well as 0.75 and 1.0, similarly to how UXM operates.
- Minimum count threshold of $\tau = 30$: We argue that the required read depth should depend on the number of modeled classes while still providing sufficient support. Setting the threshold too high may unnecessarily discard informative variation. We used 30 reads, slightly below the number of cell types.

We additionally perform a sensitivity analysis on the choice of these parameters. We varied each of the two parameters $\delta$ and $\tau$ while keeping the other fixed to the default value. We trained the *Syto* variant Dismir + Soft Labeling with Pooling. To isolate the effect of the pooling parameters as much as possible, we did not apply calibration. All other experimental settings were kept the same as in the main pseudobulk experiment.

Table S1 shows the results of this sensitivity analysis. First of all, we observe that, overall, the variations in performance between different parameter settings are small compared to the difference with external baselines (CelFiE — the previous SoTA — had an MSE of $3.27 \times 10^{-4}$), which hints at the robustness of our method to the pooling parameters.

We also observe that the best pooling parameters are different depending on the deconvolver. However, if we focus on *NNLS* and *PSLS*, which have the desirable property of having no training procedure and thus are not prone to variability due to random initialization and training dynamics (contrarily to *XGB* [4], *SWN* and *MLP*), we see that our heuristic choice landed in a sweet spot for both parameters, confirming our original intuition.

### B.4 Multiclass extension of *CancerDetector*

*CancerDetector* is a method by Li et al. [18] to estimate the proportion of reads originating from tumor cells in a bulk sample of cell-free DNA (cfDNA).

[4] XGB is conditionally determenistic, but it is also consistently worse than *NNLS* and *PSLS*

Table S1: MSE ($\times 10^{-4}$) and 95% confidence intervals for varying distance threshold $\delta$ and minimum count threshold $\tau$ parameters for Dismir + Soft Labels with Pooling for all deconvolvers, without calibration. Best results per deconvolver are in **bold**, second best are underlined.

| Deconvolver | $\delta = 0.41$ $\tau = 15$ | $\delta = 0.2$ $\tau = 30$ | $\delta = 0.41$ $\tau = 30$ | $\delta = 0.6$ $\tau = 30$ | $\delta = 0.41$ $\tau = 45$ |
|---|---|---|---|---|---|
| *NNLS* | 1.53 [1.50, 1.55] | 1.65 [1.62, 1.68] | **1.47 [1.45, 1.50]** | 1.66 [1.64, 1.69] | 1.61 [1.58, 1.64] |
| *PSLS* | 1.60 [1.58, 1.63] | 1.79 [1.75, 1.82] | **1.55 [1.52, 1.57]** | 1.73 [1.70, 1.76] | 1.70 [1.67, 1.74] |
| *MLP* | 2.16 [2.11, 2.20] | **1.60 [1.58, 1.62]** | 2.17 [2.14, 2.19] | 2.22 [2.19, 2.25] | 1.91 [1.89, 1.93] |
| *SWN* | 1.24 [1.23, 1.26] | **1.20 [1.19, 1.22]** | 1.40 [1.38, 1.42] | 1.67 [1.63, 1.70] | 1.49 [1.46, 1.52] |
| *XGB* | 2.10 [2.06, 2.14] | **1.98 [1.95, 2.01]** | 2.12 [2.08, 2.16] | 2.27 [2.23, 2.32] | 2.18 [2.15, 2.22] |

***CancerDetector* for binary deconvolution.** In the original method, the **training phase** proceeds in two steps. First, identifying methylation markers specific to the cancer under scrutiny by identifying eligible markers — *i.e.* markers with sufficient coverage — and then keeping the markers whose methylation levels can differentiate tumor samples from normal tissue samples and normal plasma samples. Second, for each marker, for each class (tumoral or normal), the parameters of the beta distribution of the $\alpha$-values of the reads in that marker from the given class are computed by maximum likelihood estimation (MLE).

In the **deconvolution phase**, for each read in the bulk sample, the likelihood of the read to belong to each class is computed according to the beta distribution of the each class and the marker from which the read originates. Finally, the proportion of each class in the mixture is computed by MLE.

Li et al. [18] also rely on the prior knowledge that they are deconvolving a mixture of normal and tumoral reads where the tumoral reads are expected to be in low proportion. They apply a process called "removal of 'confounding' markers" where they estimate the tumor proportion of each marker and iteratively remove the markers whose tumor proportion is more than one standard deviation above the mean tumor proportion, until convergence.

***CancerDetector* for multiclass classification.** We extend *CancerDetector* to the multiclass classification setting of the Loyfer et al. [13] dataset. The original marker selection is replaced by the selection of DMR already performed for the construction of the dataset [13].

The dataset $\mathcal{D} \coloneqq \{(r_i, g_i, s_i, c_i)\}_{i=1}^{N}$ consists of $N$ independent and identically distributed (i.i.d.) samples, where $r_i$ is the read, $g_i$ is the GR group to each $r_i$ belong, $s_i$ is the omics signature of $r_i$ and $c_i$ is its class label. The independence assumption is common in the literature [18, 20]. Since the particular position is irrelevant in this section, we will model the signatures as the tuples of the methylation states of the reads' CpG sites.

For a given read $r_i$, its signature $s_i \coloneqq (s_{i,1}, \ldots, s_{i,L_i}) \in \{0,1\}^{L_i}$ is modeled by the Beta-Bernouilli distribution, as in [18]. Specifically, the methylation state $s_{i,j}$ is distributed as $s_{i,j} \sim \text{Bernouilli}(p)$ where $p \sim \text{Beta}(\eta_{g_i}^{c_i}, \rho_{g_i}^{c_i})$ is the prior of methylation rates of CpG sites for the reads in GR group $g_i$ and originating from cell type $c_i$.

During the **training phase**, for every GR group $g \in \mathcal{G}$ that contains at least one read in the dataset, and for every classe $c \in \mathcal{C}$, we estimate the parameters $\eta_g^c$ and $\rho_g^c$ of the Beta distribution $\text{Beta}(\eta_g^c, \rho_g^c)$, which models the $\alpha$-values of the reads belonging to GR group $g$ with class label $c$. We estimate the parameters $\eta_g^c$ and $\rho_g^c$ by maximum likelihood estimation (MLE) when there is at least 3 reads for the GR group-class combination, and when all reads do not have the same methylation rate. In case there are at least 3 reads but all of them have the same methylation rate (which happens in our data only with methylation rate 0 and 1), then the MLE cannot converge, and so we fallback to bayesian estimation with uniform prior. In case there is 2 reads or less for the GR group-class combination, we deem the distribution as being uniform.

The **classification phase** consists in:

1. Filtering the reads of the bulk sample to deconvolve by keeping only the reads in GR groups that were seen in the training data, obtaining the following dataset: $\mathcal{D}' \coloneqq \{(r_i, g_i, s_i)\}_{i=1}^{N'}$.

2. Computing, for all $(r_i, g_i, s_i) \in \mathcal{D}'$, the likelihood of the read to belong to each class $c \in \mathcal{C}$ according to the beta distribution parameters of GR group $g_i$ and class $c$:

$$\ell(r_i|c) = \prod_{j=1}^{L_i} P(s_{i,j} \mid \text{Beta}(\eta^c_{g_i}, \rho^c_{g_i})), \tag{8}$$

$$= \prod_{j=1}^{L_i} \int_0^1 p^{s_{i,j}}(1-p)^{1-s_{i,j}} \frac{p^{\eta^c_{g_i}-1}(1-p)^{\rho^c_{g_i}-1}}{B(\eta^c_{g_i}, \rho^c_{g_i})} dp, \tag{9}$$

$$= \prod_{j=1}^{L_i} \frac{B(s_{i,j} + \eta^c_{g_i}, 1 - s_{i,j} + \rho^c_{g_i})}{B(\eta^c_{g_i}, \rho^c_{g_i})}, \tag{10}$$

$$= \frac{B(1 + \eta^c_{g_i}, \rho^c_{g_i})^{\sum_{j=1}^{L_i} s_{i,j}} B(\eta^c_{g_i}, 1 + \rho^c_{g_i})^{\sum_{j=1}^{L_i}(1-s_{i,j})}}{B(\eta^c_{g_i}, \rho^c_{g_i})^{L_i}}, \tag{11}$$

where $B$ is the Beta function.

3. Computing, for all $(r_i, g_i, s_i) \in \mathcal{D}'$, the posterior probability of the read to belong to each class $c \in \mathcal{C}$ according to Bayes' rule:

$$P(c \mid r_i, g_i, s_i) = \frac{\ell(r_i|c)P(c)}{\sum_{c' \in \mathcal{C}} \ell(r_i|c')P(c')}, \tag{12}$$

where $P(c)$ is the prior probability of class $c$.

In our experiments, we use as class prior both the distribution of classes in the training data (which is the mathematically correct choice) and the uniform distribution.

### B.5 GR-group-attention classification head

We replace the vanilla fully connected classification head of *MethylBERT* [20] with a cross-attention mechanism conditioned on the GR group identity. A preliminary comparison of the two classification heads showed slightly better losses for our formulation, which we use in all experiments. The intuition is that the same methylation pattern may be informative for a given cell type in one GR group but uninformative in another; conditioning on the GR group allows the classifier to attend to the relevant sequence positions accordingly.

Let $H = (\mathbf{h}_1, \ldots, \mathbf{h}_L) \in \mathbb{R}^{L \times d}$ be the encoder output for a read of length $L$, and let $g \in \mathcal{G}$ be the GR group label. We embed $g$ into a learnable vector $\mathbf{e}_g = E_{\text{gr}}[g] \in \mathbb{R}^d$, where $E_{\text{gr}} \in \mathbb{R}^{G \times d}$ is the GR embedding table.

**Cross-attention.** We compute a single-head cross-attention in which the GR group embedding serves as the query and the encoder outputs serve as keys and values:

$$\mathbf{q} \coloneqq W_Q\, \mathbf{e}_g \in \mathbb{R}^d, \qquad \mathbf{k}_j \coloneqq W_K\, \mathbf{h}_j \in \mathbb{R}^d, \qquad \mathbf{v}_j \coloneqq W_V\, \mathbf{h}_j \in \mathbb{R}^d, \quad \forall j \in \{1, \ldots, L\}, \tag{13}$$

$$\alpha_j \coloneqq \frac{\exp\big(\mathbf{q}^\top \mathbf{k}_j / \sqrt{d}\,\big)}{\sum_{j'=1}^{L} \exp\big(\mathbf{q}^\top \mathbf{k}_{j'} / \sqrt{d}\,\big)} \in \Delta^{L-1}, \quad \forall j \in \{1, \ldots, L\}, \tag{14}$$

$$\mathbf{z} \coloneqq \sum_{j=1}^{L} \alpha_j \mathbf{v}_j \in \mathbb{R}^d, \tag{15}$$

where $W_Q, W_K, W_V \in \mathbb{R}^{d \times d}$ are learnable projection matrices for queries, keys and values respectively, $\boldsymbol{\alpha}$ is the attention distribution over sequence positions and $\mathbf{z}$ is the attended representation. Positions corresponding to padding tokens are masked by setting their pre-softmax logits to $-10^4$.

**Context fusion.** The attended representation $\mathbf{z}$ is concatenated with the original GR group embedding and passed through a feed-forward fusion block:

$$\mathbf{f} \coloneqq \text{Dropout}\Big(\text{ReLU}\big(\text{LayerNorm}(W_f\,[\mathbf{z};\, \mathbf{e}_g] + \mathbf{b}_f)\big)\Big), \tag{16}$$

where $W_f \in \mathbb{R}^{d \times 2d}$ and $\mathbf{b}_f \in \mathbb{R}^d$ are the parameters of the linear layer, and $[\,\cdot\,;\,\cdot\,]$ denotes concatenation.

**Classifier.** The fused features are projected to class logits via a two-layer *MLP*:

$$\boldsymbol{\ell} \coloneqq W_2 \,\mathrm{Dropout}\big(\mathrm{ReLU}(W_1\,\mathbf{f} + \mathbf{b}_1)\big) + \mathbf{b}_2, \tag{17}$$

where $W_1 \in \mathbb{R}^{(d/2)\times d}$, $W_2 \in \mathbb{R}^{C'\times(d/2)}$, $\mathbf{b}_1 \in \mathbb{R}^{d/2}$ and $\mathbf{b}_2 \in \mathbb{R}^{C'}$ are the parameters of the *MLP*, and $C'$ equals $C$ under soft labeling or $C+1$ under hard labeling with background class.

As indicated in the main paper, we extended use of this mechanism to *Dismir* [19], which lacks any conditioning on GR groups as model input in its original implementation. Depending on the underlying classifier, the attention weights $\boldsymbol{\alpha}$ provide means for interpretability, indicating which encoded positions are most relevant for classification given the GR group context.

### B.6 Read-Level Classifier Training

Both *Dismir* [19] and *MethylBERT* [20] use the GR group-attention-based classification head. This replaces the original vanilla fully connected classifier with a cross-attention mechanism conditioned on GR group label (cf. Section B.5). Input sequences are fixed to a maximum length of 160 nucleotides (*Dismir*) or 160 k-mers (*MethylBERT*).

***Dismir.*** The *Dismir* encoder consists of a 1D convolution (100 filters, kernel size 10) followed by ReLU activation and max-pooling, a bidirectional LSTM with hidden size 75 per direction, a second 1D convolution (100 filters, kernel size 3) with max-pooling and dropout ($p = 0.2$) after each pooling layer. The encoder output is passed to the GR group-attention classifier.

Training used the Adam optimizer with a learning rate of $10^{-3}$, weight decay of $10^{-6}$, batch size of 128, and early stopping with patience of 7 epochs over a maximum of 100 epochs. For the soft-labeling scheme, the model was trained with 39 output classes using confidence-weighted cross-entropy loss. For the hard-labeling scheme, the model was trained with 40 output classes (39 cell types plus a background class) using standard cross-entropy loss.

***MethylBERT.*** We fine-tuned the 12-layer pre-trained *MethylBERT* model (`methylbert_hg19_12l`) with the GR group-attention-based classification head. Fine-tuning used the AdamW optimizer with mixed-precision (FP16) training. The learning rate was $4 \times 10^{-4}$, $(\beta_1, \beta_2) = (0.9, 0.98)$ and $\epsilon = 10^{-6}$. Weight decay was 0.1. A linear warmup of 100 steps was used. Gradient clipping was set at 1.0. The batch size was 512 per device, with evaluation and checkpointing every 200 steps, early stopping patiance of 1000 steps over a maximum of 5 epochs. The best checkpoint was selected based on validation loss.

For the data-driven soft-labeling scheme, the model was trained with 39 output classes using confidence-weighted cross-entropy loss. For the hard-labeling and canonical soft-labeling schemes, the model was trained with 40 output classes (39 cell types plus a background class) using standard cross-entropy loss. We additionally applied minibatch balancing for *MethylBERT* when trained using canonical soft, hard, and data driven soft labels without pooling. Each training batch was composed of 50% on-target reads (reads originating from the cell type corresponding to their GR group) and 50% background reads. We used a custom batch sampler that cycles through the background pool to maintain this ratio throughout training.

### B.7 MLE deconvolvers

In previous works, the mapping from the classification results to the mixture proportions was systematically performed via a form of Maximum Likelihood Estimation (MLE) [18, 19, 20]. In this section, we derive two MLE formulations for multiclass deconvolution and compare their performance with *UXM* and our deconvolvers. The main takeaway is that MLE deconvolvers produce results that are orders of magnitude worse than other deconvolvers, which is why they are not featured in the main paper.

We use the notations introduced in the main paper. We additionaly assume the bulk biological sample from which the set of reads $\{(r_i, g_i)\}_{i\in 1,\dots,N}$ originates has true underlying cell types proportions $\boldsymbol{\theta} \coloneqq (\theta_1, \dots, \theta_C) \in \Delta^{C-1}$, where $\theta_c$ is the proportion of cell type $c$ in the sample, $r_i$ is the read and $g_i$ is the GR group to which $r_i$ belongs. For all GR group $g$, we denote by $c^{(g)}$ the cell type for which it is specific.

**MethylBERT-style MLE.** The way Jeong et al. [20] compute likelihood $L^{MB}$ is by modeling, for each read $r_i$ falling in GR group $g_i$, the probability $P(c^{(g_i)} \mid r_i)$ that the read belong to cell type $c^{(g_i)}$. In our framework, the classifier outputs probabilities for all classes: $\mathbf{u_i} \in \Delta^{C-1}$, and in particular $P(c^{(g_i)} \mid r) = u_{i,c^{(g_i)}}$. We denote by $R_c \coloneqq \{i \mid c^{g_i} = c\}$ the set of indexes of reads belonging to a GR group specific to cell type $c$. By independence of the reads[5] and Bayes' theorem:

$$L^{MB}(\boldsymbol{\theta}) = \prod_{c=1}^{C} \prod_{i \in R_c} \left[\theta_c P(r_i \mid c) + (1-\theta_c) P(r_i \mid \text{not } c)\right], \tag{18}$$

$$= \prod_{c=1}^{C} \prod_{i \in R_c} \left[\theta_c \frac{P(c \mid r_i) P(r_i)}{P(c)} + (1-\theta_c) \frac{P(\text{not } c \mid r_i) P(r_i)}{P(\text{not } c)}\right], \tag{19}$$

$$= \prod_{c=1}^{C} \prod_{i \in R_c} P(r_i) \left[\theta_c \frac{P(c \mid r_i)}{P(c)} + (1-\theta_c) \frac{1 - P(c \mid r_i)}{1 - P(c)}\right]. \tag{20}$$

It should be noted that this formulation of the likelihood is the one used in *MethylBERT implementation*[6], but it differs from what is *written* in Equation (19) of the original paper. In fact, there are two errors related to this Equation (19): the first is that $P(r_i \mid \text{not } c)$ is equated with $1 - P(r_i \mid c)$, which is true iff $P(c) = 0.5$, but which is false in general. The second error is that in the paper, $R_c$ is defined as the set of reads classified as $c$, whereas in the implementation, it is the set of reads that fall into GR groups specific to $c$.

The log-likelihood writes as follows:

$$\log L^{MB}(\boldsymbol{\theta}) = \sum_{c=1}^{C} \sum_{i \in R_c} \log \left(\theta_c \frac{u_{i,c} - P(c)}{P(c)(1-P(c))} + \frac{1 - u_{i,c}}{1 - P(c)}\right) + \sum_{c=1}^{C} \sum_{i \in R_c} \log P(r_i). \tag{21}$$

The $P(r_i)$ terms factor out and do not play any role in the optimization. The prior terms $P(c)$ are computed as the ratios of classes in the data effectively processed by the classifier during training (it is only different from the ratio of class in the training set when we use minibatch balancing). We also experiment with a uniform prior. The maximum likelihood estimates $\boldsymbol{\theta}^*$ are obtained by solving the following convex optimization problem using the Python package SciPy:

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\text{argmax}} \log L^{MB}(\boldsymbol{\theta}), \tag{22}$$

$$\text{s.t.} \quad \sum_{c=1}^{C} \theta_c = 1, \qquad \theta_c \geq 0, \quad \forall c \in \mathcal{C}, \tag{23}$$

We apply the MLE deconvolver to pseudobulks corresponding to pure mixtures (*i.e.*, a single cell type in the mixture), the simplest mixtures to deconvolve. We use 3 different classifiers settings to generate the classifier outputs on which we apply the MLE deconvolver. First is *MethylBERT* trained with focal loss on hard labels with background, second is *MethylBERT* trained with minibatch balancing on hard labels with background and third is *MethylBERT* trained with focal loss on the binary label scheme of Jeong et al. [20] (where the label of a read is 1 if it belongs to the cell type specific to its GR group and 0 otherwise). The last setting is the one used in the *MethylBERT* paper [20]. Results are presented in Table S2.

The performance is worse than *UXM*. At first glance, this may seem surprising, as Jeong et al. [20] report that the MLE deconvolver can deconvolve five cell types with performance similar to *UXM* (panel H, Figure 4 in [20]). However, the experimental setup is not described in sufficient detail for us to reproduce this comparison, and several aspects of the two settings differ. First, our experiments

[5]The independence assumption is common in the literature [18, 20].

[6]https://github.com/CompEpigen/methylbert/blob/main/src/methylbert/deconvolute.py

Table S2: Deconvolution performances of *UXM* and of MLE *MethylBERT*-style deconvolver when applied on 39 pure mixtures composed from the test set predictions of *MethylBERT* for two training regimes (focal loss on hard labels with background class and hard labels with background class with balanced minibatches).

| Deconvolver | Training of MethylBERT | Prior | $\mathbf{R^2}$ ($\times 10^{-2}$) | LoA ($\times 10^{-2}$) | LoA worst-class ($\times 10^{-2}$) | MAE ($\times 10^{-3}$) | MSE ($\times 10^{-4}$) | KL ($\times 10^{-2}$) |
|---|---|---|---|---|---|---|---|---|
| MLE (*MethylBERT*-style) | Hard labels w/ Focal loss | Uniform prior | -10.29 | [-32.55, 32.55] | [-33.95, 28.82] | 49.09 | 275.55 | 1313.22 |
| MLE (*MethylBERT*-style) | Hard labels w/ Focal loss | Train freq prior | -8.02 | [-32.21, 32.21] | [-27.64, 35.76] | 49.08 | 269.87 | 1166.60 |
| MLE (*MethylBERT*-style) | Hard labels w/ Minibatch balancing | Uniform prior | 33.12 | [-25.34, 25.34] | [-33.83, 28.73] | 37.83 | 167.08 | 149.94 |
| MLE (*MethylBERT*-style) | Hard labels w/ Minibatch balancing | Effective train freq prior | 39.56 | [-24.09, 24.09] | [-33.36, 28.75] | 36.82 | 151.00 | 139.39 |
| MLE (*MethylBERT*-style) | Binary labels w/ Focal loss | Train freq prior | 94.76 | [-7.09,7.09] | [-26.80, 23.47] | 6.23 | 13.09 | 16.44 |
| MLE (*MethylBERT*-style) | Binary labels w/ Focal loss | Uniform prior | 92.34 | [-8.58,8.58] | [-31.30, 26.70] | 7.27 | 19.15 | 21.63 |
| *UXM* | N/A | N/A | 98.69 | [-3.54, 3.54] | [-9.85, 8.46] | 3.74 | 3.27 | 7.86 |

consider 39 cell types covering the whole body, whereas the *MethylBERT* study considers only five leukocyte cell types. Second, the reported outputs are compared with *UXM*'s estimates rather than with the ground-truth mixture proportions. Third, the pseudobulk proportions in the *MethylBERT* study span relatively narrow ranges: *Blood-B* and *Blood-NK* are always present at less than 10%, while *Blood-Mono+Macro* is restricted to a narrow range. *Blood-Granul* and *Blood-T* are the only cell types with substantial variation in their proportions, and neither spans the full $[0, 1]$ range. This differs substantially from our setting, in which all 39 cell types can occur at arbitrary proportions.

**General MLE.** The extension of likelihood to multiclass by Jeong et al. [20] is actually based on the binary class setting, and relies on the fact that in the original paper, *MethylBERT* only outputs the probability of a read being of the cell type which is specific to the GR group the read belongs to. We derive the likelihood for multiclass in the context where for a single read $r$, the model outputs a posterior probability distribution $P(c \mid r)$ for all classes $c$, also using the read independence assumption and Bayes' theorem:

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{N} P(r_i \mid \boldsymbol{\theta}) = \prod_{i=1}^{N} \sum_{c=1}^{C} \theta_c P(r_i \mid c), \tag{24}$$

$$= \prod_{i=1}^{N} \sum_{c=1}^{C} \theta_c \frac{P(c \mid r_i) P(r_i)}{P(c)} = \prod_{i=1}^{N} P(r_i) \sum_{c=1}^{C} \theta_c \frac{u_{i,c}}{P(c)}. \tag{25}$$

The $P(r_i)$ terms factorize out in the log-likelihood and thus play no role in the optimization:

$$\log L(\boldsymbol{\theta}) = \sum_{i=1}^{N} \log P(r_i) + \sum_{i=1}^{N} \log \left( \sum_{c=1}^{C} \theta_c \frac{u_{i,c}}{P(c)} \right). \tag{26}$$

We obtain the mixture proportion estimates by minimizing the negative log-likelihood subject to positivity and sum-to-one constraints, using the Python package SciPy. Table S3 shows the performance of the main paper deconvolvers (without calibration) and of this general MLE deconvolver with two different priors: the class frequency in the train set (which is the mathematically correct choice), and the uniform prior. As in the previous experiment, the performance of MLE methods is orders of magnitude worse than that of other methods.

Table S3: Deconvolution performances for UXM and MLE general deconvolver when applied on 39 pure pseudobulks composed from the test set predictions of *MethylBERT* with soft labels with pooling.

| Deconvolver | $\mathbf{R^2}$ ($\times 10^{-2}$) | LoA ($\times 10^{-2}$) | LoA worst-class ($\times 10^{-2}$) | MAE ($\times 10^{-3}$) | MSE ($\times 10^{-4}$) | KL ($\times 10^{-2}$) |
|---|---|---|---|---|---|---|
| MLE (general, train freq prior) | -65.15 | [-39.82,39.84] | [-5.10, 152.04] | 48.20 | 412.61 | 544.13 |
| MLE (general, uniform prior) | -20.20 | [-33.97,33.98] | [38.65, 105.00] | 39.46 | 300.29 | 253.97 |
| *UXM* | 98.69 | [-3.54, 3.54] | [-9.85, 8.46] | 3.74 | 3.27 | 7.86 |

One possible reason why MLE deconvolvers fail is that they are applied to neural network outputs, which are known not to be calibrated and therefore not representative of the true posterior probabilities [28]. Another issue is the high sensitivity of the deconvolvers to the prior over classes. Indeed, we

see in Table S3 that using a uniform prior, even if not mathematically correct, leads to better results. It also appears that each combination of type of output on which the model was trained and prior elicits a specific bias cell type, for which the predicted proportion is always high. For instance, in the case of MLE MethylBERT-style, when applied to the outputs of *MethylBERT* trained on hard labels with focal loss and using the train frequency prior, the average predicted proportion of *Gallbladder* across all pure mixtures is 24.0%. A perfect deconvolver would have given an average prediction across all pure mixtures of $1/39 \approx 2.6\%$ for all cell types. Table S4 summarizes the biases of all MLE settings presented in this section, as well as *UXM* bias.

Table S4: Biases of the different MLE methods.

| MLE flavour | Training of MethylBERT | Prior | Bias cell type | Avg prediction of bias cell type over pure mixtures |
|---|---|---|---|---|
| MethylBERT-style | Hard labels w/ Focal loss | Train frequency | *Gallbladder* | 24.0% |
| MethylBERT-style | Hard labels w/ Focal loss | Uniform | *Blood-T* | 24.0% |
| MethylBERT-style | Hard labels w/ Minibatch balancing | Effective Train frequency | *Neuron* | 15.0% |
| MethylBERT-style | Hard labels w/ Minibatch balancing | Uniform | *Neuron* | 22.7% |
| MethylBERT-style | Binary labels w/ Focal loss | Train frequency | *Dermal-Fibro* | 3.3% |
| MethylBERT-style | Binary labels w/ Focal loss | Uniform | *Endothel* | 7.1% |
| General | Soft Labels w/ Pooling | Train frequency | *Dermal-Fibro* | 76.0% |
| General | Soft Labels w/ Pooling | Uniform | *Bladder-Ep* | 74.3% |
| — | *UXM* | — | *Fallopian-Ep* | 3.0% |

### B.8 Fitting parameters of deconvolvers

All deconvolution models were trained on the same feature-selected input matrices derived from in-silico pseudobulk mixtures (cf. Section D.4). Below we list the hyperparameters used for each deconvolver.

**XGBoost (*XGB*).** A gradient-boosted tree ensemble was trained with 500 estimators, a maximum depth of 15 and a learning rate of 0.05. Row subsampling was set to 0.7, with full-column sampling per tree. Regularization was controlled via a minimum child weight of 5, L1 penalty $\alpha = 0.1$ and L2 penalty $\lambda = 1.0$. Outputs were clipped and normalized to produce valid mixture proportions.

**Single-layer Wide Network (*SWN*).** A single hidden layer of 1024 units with GELU activation and 20% dropout, followed by a softmax output layer. Training ran for up to 5000 epochs with a batch size of 512, a learning rate of $5 \times 10^{-5}$, weight decay of $10^{-3}$ and early stopping with a patience of 1000 epochs based on validation MAE. No learning rate scheduler was used.

**Multi-Layer Perceptron (*MLP*).** Two hidden layers of 512 and 256 units (each with GELU activation and 20% dropout), followed by a projection back to the input dimensionality with 10% dropout and a softmax output layer. Training used the same schedule as *SWN*: up to 5000 epochs, batch size 512, learning rate $5 \times 10^{-5}$, weight decay $10^{-3}$ and early stopping with patience 1000 on validation MAE. No learning rate scheduler was used.

**Non-Negative Least Squares (*NNLS*) and Probability Simplex Least Squares (*PSLS*).** Both solvers used a reference matrix constructed from the training-split pure profiles after applying the feature mask. Both *NNLS* and *PSLS* were solved using cvxpy [41]. Neither method has learnable hyperparameters beyond the reference matrix itself.

### B.9 Calibration methods

**Calibration methods analysis.** In addition to the simple linear calibration scheme that we introduced, we also experimented with the existing calibration schemes for multiclass classification, as described in Section 3.5.

The parameters $W$ and $\mathbf{b}$ from Equation (1) are optimized via Adam with cross-entropy loss between the true and calibrated proportions.

The parameters are regularized with an $\ell_2$ penalty of strength $\lambda$ in the case of temperature scaling and vector scaling. For Dirichlet calibration, when $\mu$ is None, we apply $\ell_2$ regularization on all parameters with strength $\lambda$. When $\mu$ is not None, we apply the ODIR regularization of [29] with strength $\lambda$ on the off-diagonal parameters of $W$ and with strength $\mu$ on $\mathbf{b}$.

Training hyperparameters (learning rate and regularization strengths) are selected by 3-fold cross-validation grid search over the mixture proportions of *UXM* on the validation set, retaining the configuration with the lowest mean validation loss. During training, we apply a reduce-on-plateau learning rate scheduler (factor 0.5, patience 10) and early stopping (patience 15, tolerance $10^{-4}$), both monitored on the validation loss. We show the parameters of the grid search and the selected hyperparameters in Table S5.

The results of the calibrators trained on the whole validation set with the best hyperparameters are shown in Table S6.

Results show that our method and vector scaling achieve the best results in terms of $R^2$, LoA, MAE and MSE. We use them in our main experiments.

Table S5: Hyperparameter selection of the learned calibration functions on the validation set. The selected hyperparameters are underlined.

| **Model** | **Reg.** $\lambda$ | **Reg.** $\mu$ | **LR** |
|---|---|---|---|
| Temperature scaling [28] | [0, 0.0001] | N/A | [0.0001, 0.001, 0.01, 0.1] |
| Vector scaling [28, 29] | [0, 0.001, 0.01, 0.1] | N/A | [0.0001, 0.001, 0.01] |
| Dirichlet calibration [29] | [0, 100, 1000, 2000, 4000, 5000, 10000] | [None, 0, 1, 10] | [1e-4, 1e-5, 1e-6, 1e-7] |

Table S6: Calibrators performance on *UXM* predictions on the test set. Best result is **bold**, second best result is underlined.

| **Calibrator** | $\mathbf{R^2}$ ($\times 10^{-2}$) | **LoA** ($\times 10^{-2}$) | **LoA worst-class** ($\times 10^{-2}$) | **MAE** ($\times 10^{-3}$) | **MSE** ($\times 10^{-4}$) | **KL** ($\times 10^{-2}$) |
|---|---|---|---|---|---|---|
| No calibration | 94.86 | [-4.05, 4.05] | [-10.82, 14.02] | 6.49 | 4.27 | 14.98 |
| Lin. Cal. clip-norm (Ours) | 95.91 | [-3.61, 3.61] | [-10.89, 13.04] | 5.33 | 3.40 | 12.33 |
| Lin. Cal. simplex-projection (Ours) | 96.74 | [-3.23, 3.23] | [-10.78, 12.98] | **4.32** | 2.71 | 10.91 |
| Temperature scaling [28] | 96.21 | [-3.48, 3.48] | [-10.03, 12.83] | 5.33 | 3.14 | 10.24 |
| Vector scaling [28, 29] | **96.90** | **[-3.14, 3.14]** | [-10.44, 11.23] | 4.99 | **2.57** | **9.74** |
| Dirichlet calibration [29] | 95.77 | [-3.67, 3.67] | **[-9.56, 9.61]** | 6.09 | 3.51 | 13.11 |

**Calibrators fitting.** In the main experiments, we fit vector scaling on the validation split and select $\lambda$ and the learning rate for each classifier-labeling-deconvolver combination by 3-fold cross-validation grid search across sets $\{10^{-4}, 10^{-3}, 10^{-2}\}$ for learning rate and $\{0, 10^{-4}, 10^{-3}, 10^{-2}\}$ for $\lambda$. Our method is hyperparameter-free and thus does not require a selection procedure.

### B.10 Baselines

**Motivation for selection.** *UXM* is the method proposed by Loyfer et al. [13] alongside their dataset that we rely on for training. It is also one of the best methods for WGBS data [51]. *CelFiE* [11]. is reported as the best method in the DecoNFlow benchmark [51] and is expected to perform well on the chosen atlas [49]. The *Houseman CP* [9] is one of the earliest methylation-based deconvolution algorithms, the best beta-value-based method in the DecoNFlow benchmark [51], strong in several other benchmarks [52, 53], and is one of the most frequently used for comparison. And *EpiDISH* [43] is reported as the best method in multiple benchmarks [48, 53].

**UXM [13].** *UXM* is an unsupervised reference-based deconvolution method operating on the aggregated counts of labeled DNA fragments. *UXM* requires a reference atlas matrix $A \in \mathbb{R}^{G \times C}$ where $A_{g,c}$ is equal to the ratio of reads in GR group $g$ that are unmethylated (*i.e.* with at most $25\%$ methylated CpGs). At inference, the ratios $b_g$ of unmethylated reads in GR group $g$ are aggregated in a vector $\mathbf{b} \in \mathbb{R}^G$ and the mixture proportions $\mathbf{x} \in \Delta^{C-1}$ are inferred using *NNLS*, minimizing $||A\mathbf{x} - b||_2^2$. $\mathbf{x}$ is normalized to 1 a posteriori. In our work, we use four different atlases, built on the region sets described in Section C.2.

1. In the pseudobulk experiment, we first selected genomic region groups from the *U25* atlas for the *hg38* reference, published by Loyfer et al. [13]. We then computed $A_{g,c}$ using the train split, utilizing a $25\%$ methylation value to label fragments as unmethylated, consistent with the UXM

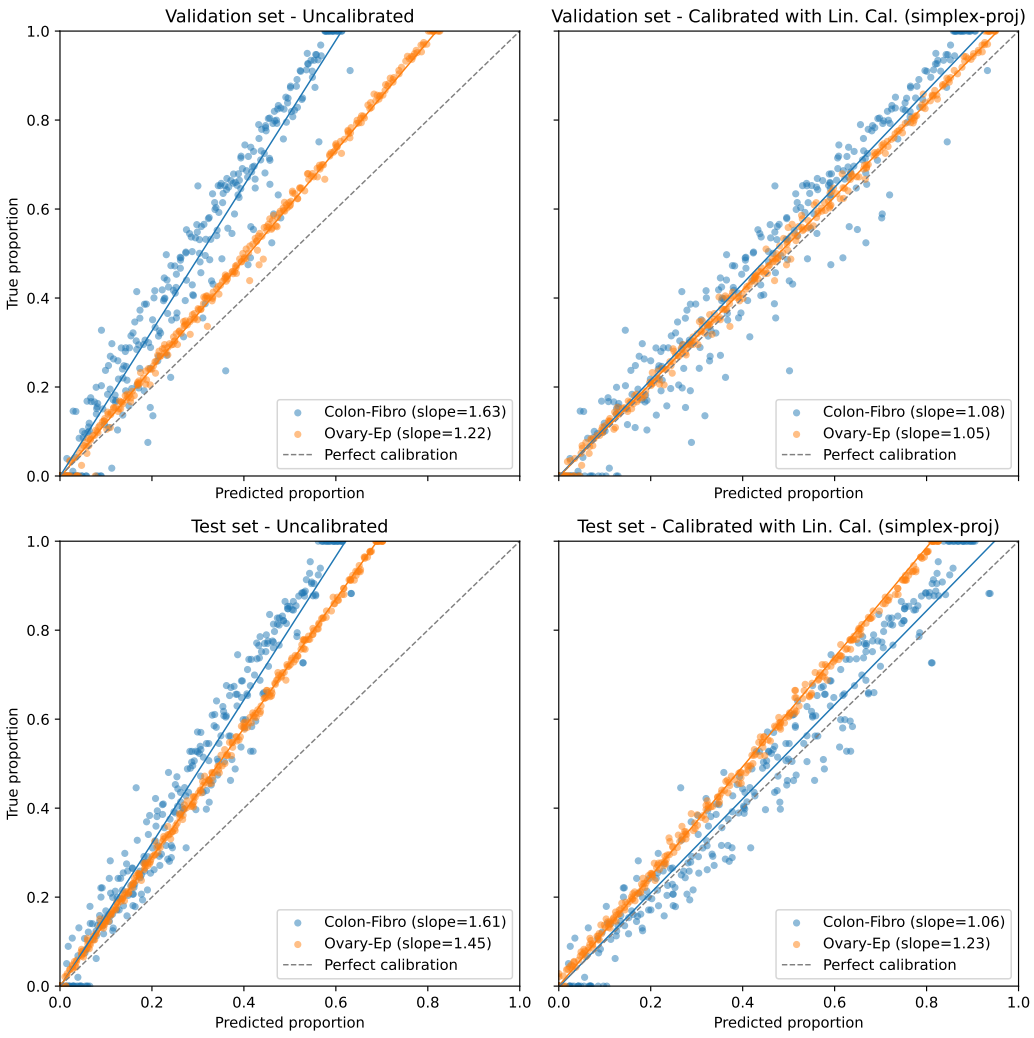


Figure S2: Uncalibrated and calibrated proportions (using linear calibrator with simplex projection) predicted by Dismir + Soft Labels w/ pooling + PSLS on the validation and test set for the two cell types with worst calibration.The regression lines and slopes are computed on the whole sets, but for visualization purposes, a subset of 300 pseudobulks have been selected for each cell type in such a way that the scattered points span uniformly the $[0, 1]$ range.

default threshold. Limiting estimation of the unmethylated reads ratio to the train split is motivated by avoiding information leak that would otherwise be exerted from validation and test sets.

2. In the OOD RRBS experiment, we used *U25* atlas for *hg19* reference, published by Loyfer et al. [13] as our main atlas for inference, with a set of GR groups and corresponding $A_{g,c}$ as in the original work. Additionally, we used *U250* atlas for *hg19* reference, published by Loyfer et al. [13] to compute *UXM U250* TCS. We did it to generate a comparison point where *UXM* has access to $\approx 10\times$ more GR groups and reads at inference and to see if Syto could match this result running on *U25*. Finally, the *U25 GSS sorted* atlas was used to test if better results can be obtained for *UXM* by selecting different GR groups. We reported the score achieved with this atlas under *UXM U25 GSS sorted* (see Figure S13). We also used the *U25 GSS sorted* atlas to run an auxiliary GR group selection sensitivity experiment for *Syto* (see Section E.6).

For running *UXM*, we ported the original Python code (see Section G). Since our pipeline requires converting .pat files into reads, we additionally validated that *UXM* CLI run on .pat and our port run on recovered reads produce the same results by comparing the distribution of TCS. The mean absolute difference across 521 evaluated OOD samples was 0.000369, or $6.6\%$ of the *UXM* uncalibrated pseudobulk MAE which suggests that our port produced results sufficiently close to the original CLI implementation.

**CelFiE [11].** *CelFiE* is an Expectation Maximization (EM) based algorithm, operating on the individual CpGs or GR groups aggregated $\beta$-values. Each row in the reference features the number of methylated CpGs and the total number of CpGs for each cell type $c \in C$. At inference time, *CelFiE* jointly optimizes reference atlas $\beta$-values (parameterized as $\gamma$) and derived cell-type proportions $\alpha$, conditioned on initial atlas and $s \in S$ bulk mixtures to deconvolute. It can also work with an arbitrary number of unknown components $n_{unknown}$, by extending the atlas with $n_{unknown} \times 2$ columns, and the number of methylated CpGs and total number of CpGs for each unknown cell type initialized with zeroes. We used a set of GR groups from the *U25* atlas for the *hg38* reference, published by Loyfer et al. [13], and a training set to compute the atlas for the pseudobulk experiment. For the OOD RRBS experiment, we used GR groups from the *U25* atlas for the *hg19* reference, published by Loyfer et al. [13], and recovered reads overlapping GR groups from all three pseudobulk splits to compute the atlas. For running *CelFiE*, we first ported the original *CelFiE* Python code (see Section G). We kept inference parameters at default values: $random_restarts = 10; convergence_criteria = 0.0001$. We increased the number of EM iterations from the default $1000$ to $10000$ to guarantee convergence. We tried running *CelFiE* using individual CpGs and GR groups aggregated $\beta$-values. Based on a pseudobulk experiment, using aggregated signal yielded significantly stronger results, so we set inference to GR groups, validating a previous finding stated by Caggiano et al. [11]. Despite the aforementioned settings, our initial experiments positioned *CelFiE* inferior to *UXM*, in contrast to [51]. After careful analysis, we have found that optimization of the reference at inference time significantly degrades results. Specifically, a variant of *CelFiE* with enabled optimization of reference atlas yielded MSE of $5.11 \times 10^-4$, while dissabling updates of $\beta$-values reduced MSE to $3.28 \times 10^-4$. Therefore, we freeze $\gamma$ at its initial values across experiments to strengthen the baseline. We argue that re-estimating the atlas is necessary in modeling with unknown components and feasible when several independent samples per deconvolution task are available (the last condition is violated in our pseudobulks). In all our settings, we deconvolute one sample at a time.

**EpiDISH [43].** *EpiDISH* is a singleton method and an R package implementing four different reference-based algorithms, each operating on the individual CpGs $\beta$-values. In this paper, we consider two algorithms: (1) robust partial correlations (RPC) with a posteriori elimination of negative proportions followed by scaling such that the sum of proportions equates to one. (2) linear constraint projection, which enforces proportion positivity and, optionally, normalization to ensure the sum of prportions to one at inference. Algorithm (1) is *EpiDISH* proper, and we refer to it as is throughout our work. Algorithm (2) is an implementation of the Houseman [9] method, and we refer to it as *Houseman CP* or *Houseman Ineq.* interchangeably. Relaxing the inequality constraint produces better results in accordance with [51], so we run it accordingly. To integrate these methods in our experimental pipelines, we translated the original R code (see Section G) into Python. We then generated 5 synthetic samples with 4 cell types each as test fixtures and computed their derived proportions using an R package. We did the same computation. for our Python port and compared the two. Obtained proportions differed at most by 0.000233 and 0.000327 for *EpiDISH* and *Houseman*

*CP*, which we deemed acceptable. For each experiment, we ran *EpiDISH* and *Houseman CP* methods using atlases as derived for *CelFiE* with $\beta$-values computed on the individual CpGs.

# C Data preparation for the experiments

## C.1 Data preprocessing of .pat files

The raw data featured in our experiments is provided in ultra-compact **.pat* format where reads with the same covered CpG pattern are collapsed in single data record without aligned sequences [50]. Since the classification models expect a combination of read-level DNA sequence and methylation pattern, we recovered this information using our modified version of the tool by Jeong et al. [20]. The original version of the tool for recovering mapped reads from raw .pat files kept only the reads lying strictly within the selected genomic regions, discarding reads overlapping region boundaries. In the earlier versions of this paper, we did a whole range of experiments relying on the data filtered this way. Subsequent checks demonstrated that such stringent read inclusion criteria lead to detrimental results across experiments, equally for Syto and baselines. We therefore modified the read recovery tool to keep all reads overlapping target regions, with start and end trimmed to region boundaries. This led to a uniform improvement of MSE in the pseudobulk experiment by 33% for *Syto* and *UXM*, the only baseline we had at a time we introduced this change.

The genomic regions against which the reads were recovered, and the reference assemblies involved, are described in Section C.2.

## C.2 Marker region sets used in the experiments

As regards the cell-type atlas [13], we downloaded all data in the aligned format using both reference assemblies, hg19 and hg38. In the case of the tissue atlas [10], only the hg19 aligned format was available. For each alignment, the reads were recovered from the genomic positions overlapping with the corresponding *"top-250 unmethylated markers atlases"* published by Loyfer et al. [13] (namely *Atlas.U250.l4.hg19.full.tsv* and *Atlas.U250.l4.hg38.full.tsv*[7]). Recovering reads against the *U250* region sets makes a single recovery pass sufficient for all our experiments: every other region set we use is nested within *U250* for the same assembly, so the corresponding reads are obtained by subsetting. Table S7 summarizes the four resulting region sets. Within a given experiment, the same set of GR groups was used for all baseline methods and all Syto permutations. The way each method turns those regions into its own reference is method-specific (cf. Section B.10).

Table S7: Marker region sets used in our experiments. All of them are derived from the unmethylated marker atlases of Loyfer et al. [13].

| Name | Assembly | DMRs per GR group | Used in |
|---|---|---|---|
| *U25* | hg38 | up to 25 | pseudobulk experiment, all methods |
| *U25* | hg19 | up to 25 | OOD RRBS experiment, all methods |
| *U250* | hg19 | up to 250 | OOD RRBS experiment, *UXM U250* only |
| *U25 GSS sorted* | hg19 | up to 25 | OOD RRBS experiment, *UXM U25 GSS sorted* and Syto region selection sensitivity |

We generated *U25 GSS sorted* by subsetting up to 25 top GR groups from the *U250* atlas for *hg19* reference using proxy gap specificity score (see Section C.3).

We initially trained *MethylBERT* on *hg19*-aligned and *hg38*-aligned data and observed no significant differences in the attained losses. Based on this observation and the fact that hg38 had more overlaping reads, we disregarded hg19, and our training effort was done on hg38. For inference in our main experiments, we used *Atlas.U25.l4.hg38.full.tsv* in the pseudobulk experiment and *Atlas.U25.l4.hg19.full.tsv* in the OOD experiment; mixing assemblies was possible for *Dismir* and *MethylBERT*, because we united DMRs into cell-type groups, and for those classifiers training and infering on different alignments was possible. *Lookup Classifier* is not alignment agnostic, because constructed lookup keys are position specific, hence, for this classifier only we used hg38 aligned reads for training in pseudobulk experiment and hg19 aligned reads for training in ODD experiment.

[7] Downloaded from https://github.com/nloyfer/uxm_deconv/tree/main/supplemental

### C.3 Analysis of the DNA methylation atlas of normal human cell types

To better understand the structure of the atlas, we calculated a *proxy gap specificity score* for each cell-type-specific marker region as the difference between the proportion of unmethylated reads for the target cell type (the cell type for which the region is specific) and the maximal fraction of unmethylated reads stemming from other cell-types in the same region (taking inspiration from the work by Guo and Teschendorff [54]). The analysis revealed a previously unreported[8] heterogeneity in the specificity of the reference regions, with a significant tail of lower-quality groups (Figure S3). This proven to be important later when assesed in supplemntary experiment (see Figure S13) and contributed to understanding of the relevance of regions selection to the overall performance.

[8] Sun et al. [49] featured the *overall* specificity of different sets of markers, including those used in this study; but no in-depth analysis *per cell-type group* was done

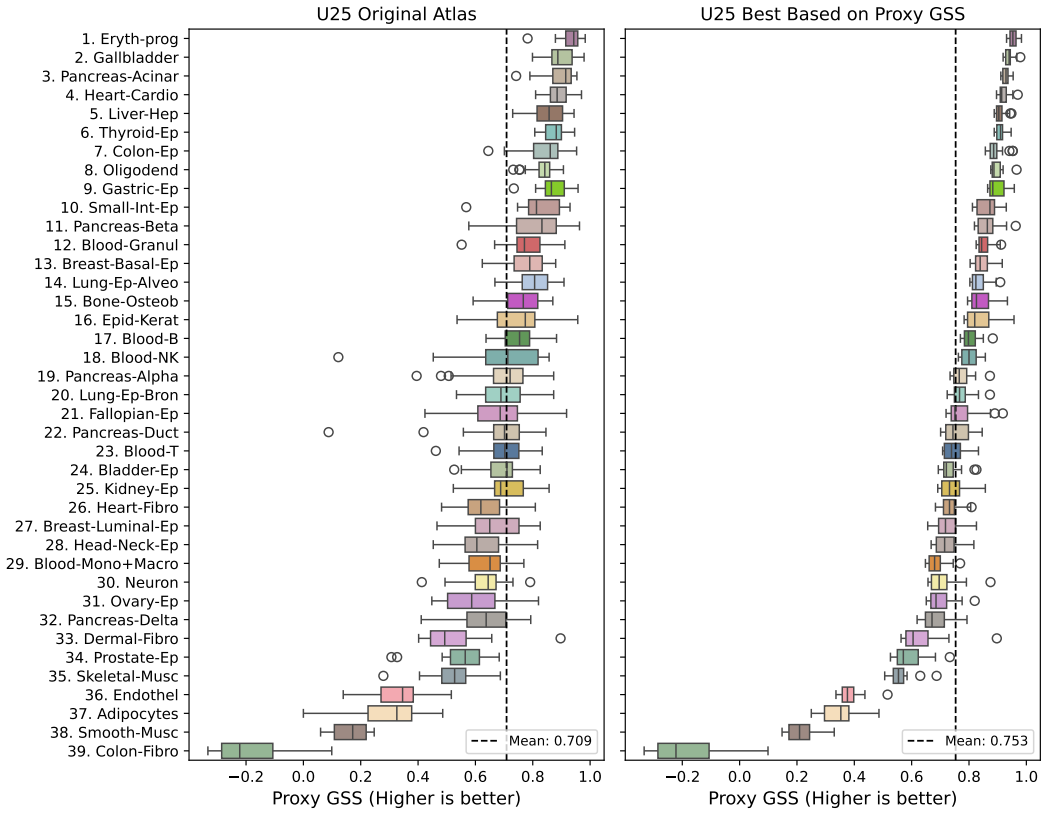


Figure S3: Proxy Gap Specificity Scores or difference between proportion of on-target unmethylated reads and maximal proportion of unmethylated reads from any other cell type in the same region (best appearing on top). We selected U25 Best Based on Proxy GSS (right graph) by subsetting the Top-25 regions per cell type from the U250 atlas, ordered by Proxy GSS.

### C.4 Mapping of cfSort tissues to Tabula Sapiens atlas

To perform mapping, we first matched the tissue types from the OOD dataset to those in the single-cell transcriptomic atlas of human tissues *Tabula Sapiens* [45, 46] using UBERON codes from OLS4 [55]. The intersection yielded 16 exact and 2 putative matches corresponding to 289 samples. We then mapped the cell types from the *Tabula Sapiens* project to the corresponding cell types from our reference atlas [13] using CL codes from OLS4 [55] (cf. Table S8). This additional mapping allowed us to estimate the percentage of cells with direct correspondence between the transcriptomic and methylation atlases for each mapped tissue. Two tissues (muscle and testis) exhibited low mapping rates ($43.6\%$ and $1.5\%$) and were excluded from the analysis, resulting in a final set of 14 exact and 2 putative matches with matching rates in the range $88.7\% - 100\%$, and 260 samples in total. Figure S4 shows the inferred cell type distribution per tissue and Table S8 shows cell types mapping. The mapping was performed only for those *Tabula Sapiens* cell types that contributed to the tissues overlaping with OOD dataset.

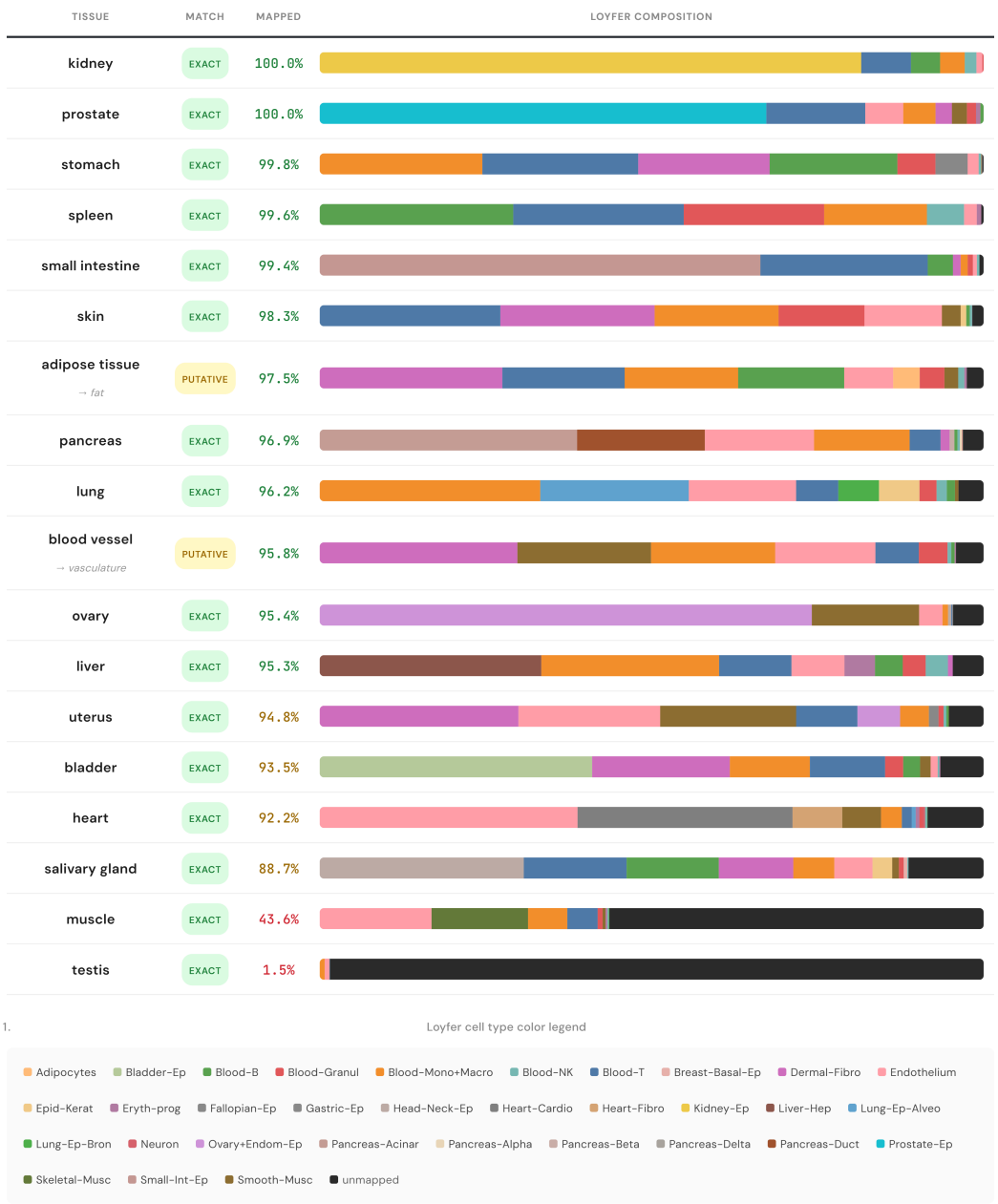


Figure S4: 16+2 mapped cfSort tissues and their derived cell type contributions in terms of Loyfer et al. [13] atlas. 11 tissues had no corresponding mapping and are not displayed here.

Table S8: Loyfer cell-type subtypes with their curated Cell Ontology (CL) term and the Tabula Sapiens labels that strictly-match each subtype (**bold** and (e) = exact match with CL term; (e) = exact, (m) = manual match, (em) = both). Exact matches are rare because cell types in Loyfer′s atlas are in fact cell-type groups, while Tabula Sapiens operates at a more granular level. There are exceptions: for example, the Tabula Sapiens *fibroblast* matches the more specific Loyfer Colon-Fibro and Dermal-Fibro subtypes. In such cases, the choice was tissue-specific; for our setup, we used Dermal-Fibro across matched tissues, as it is more generic than Colon-Fibro (which is colon-specific and absent from our OOD set). Several matches were established by considering Tabula Sapiens cell types in the context of their originating tissue: *ciliated epithelial cell* (uterus in Tabula Sapiens) was attributed to Fallopian-Ep, and *epithelial cell* (stomach in Tabula Sapiens) to Gastric-Ep.

| **Loyfer cell type** | **Source tissues** | **CL terms** | **Tabula Sapiens strict matches** |
|---|---|---|---|
| Adipocytes | [Abdominal Subcut.] | [CL:0000136] | [mesenchymal stem cell of adipose tissue (m)] |
| Bladder-Ep | [Bladder] | [CL:1001428] | [**bladder urothelial cell (em)**] |
| Blood-B | [Blood] | [CL:0000236,<br>CL:0000787] | [**B cell (em)**,<br>plasma cell (m)] |
| Blood-Granul | [Blood] | [CL:0000094] | [**granulocyte (em)**,<br>basophil (m),<br>mast cell (m),<br>neutrophil (m)] |
| Blood-Mono+Macro | [Blood,<br>Colon,<br>Liver,<br>Lung alveolar,<br>Lung interstitial] | [CL:0000091,<br>CL:0000576,<br>CL:0000583,<br>CL:0009038,<br>CL:1001603] | [**monocyte (em)**,<br>classical monocyte (m),<br>intermediate monocyte (m),<br>macrophage (m),<br>mononuclear phagocyte (m),<br>non-classical monocyte (m),<br>tissue-resident macrophage (m)] |
| Blood-NK | [Blood] | [CL:0000623] | [**natural killer cell (em)**] |
| Blood-T | [Blood] | [CL:0000084,<br>CL:0000492,<br>CL:0000625,<br>CL:0000895,<br>CL:0000900,<br>CL:0000910,<br>CL:0000913,<br>CL:0001087,<br>CL:0002678] | [CD4-positive alpha-beta T cell (m),<br>**CD8-positive alpha-beta T cell (em)**,<br>**T cell (em)**,<br>**naive thymus-derived CD4-positive alpha-beta T cell (em)**,<br>**naive thymus-derived CD8-positive alpha-beta T cell (em)**,<br>activated CD4-positive alpha-beta T cell (m),<br>activated CD8-positive alpha-beta T cell (m),<br>gamma-delta T cell (m),<br>mature NK T cell (m),<br>regulatory T cell (m)] |
| Bone-Osteob | [Bone] | [CL:0000062] | — |
| Breast-Basal-Ep | [Breast] | [CL:0002324] | [basal cell (m),<br>myoepithelial cell (m)] |
| Breast-Luminal-Ep | [Breast] | [CL:0002326] | — |
| Colon-Ep | [Colon] | [CL:0000164,<br>CL:0011108] | [**enteroendocrine cell (em)**] |
| Colon-Fibro | [Colon] | [CL:4052007] | [fibroblast (m)] |
| Dermal-Fibro | [Ad-dermal] | [CL:0002551] | [fibroblast (m)] |
| Endothelium | [Aorta,<br>Kidney glomerular,<br>Kidney tubular, | [CL:0000115,<br>CL:0002543,<br>CL:1000398, | [**endothelial cell (em)**,<br>**endothelial cell of artery (em)**,<br>**vein endothelial cell (em)**, |

*(continued on next page)*

*(continued)*

| Source Tissue | Cell type | CL term | Tabula Sapiens strict matches |
|---|---|---|---|
| | Liver, Lung alveolar, Pancreas, Vascular saphenous] | CL:1000413, CL:1000892, CL:1001005, CL:4028001] | capillary endothelial cell (m), cardiac endothelial cell (m), endothelial cell of arteriole (m), endothelial cell of lymphatic vessel (m), endothelial cell of vascular tree (m), endothelial cell of venule (m)] |
| Epid-Kerat | [Skin] | [CL:0000312] | [basal cell (m), keratocyte (m), sebocyte (m)] |
| Eryth-prog | [Bone marrow] | [CL:0000038] | [erythrocyte (m), hematopoietic precursor cell (m), hematopoietic stem cell (m)] |
| Fallopian-Ep | [Fallopien tubes] | [CL:4030007] | [ciliated epithelial cell (m)] |
| Gallbladder | [Gallbladder] | [CL:1000415] | — |
| Gastric-Ep | [Gastric, Gastric antrum, Gastric body, Gastric fundus] | [CL:0000164, CL:0002178] | [**enteroendocrine cell (e)**, epithelial cell (m)] |
| Head-Neck-Ep | [Esophagus, Larynx, Pharynx, Tongue, Tonsil palatine, Tonsil pharyngeal] | [CL:0002252, CL:0002633, CL:1001573, CL:1001576, CL:1001577] | [acinar cell of salivary gland (m), glandular secretory epithelial cell (m), salivary gland cell (m)] |
| Heart-Cardio | [Heart] | [CL:0000746] | [regular atrial cardiac myocyte (m), ventricular cardiac muscle cell (m)] |
| Heart-Fibro | [Heart] | [CL:0002548] | [**fibroblast of cardiac tissue (em)**] |
| Kidney-Ep | [Kidney glomerular, Kidney tubular] | [CL:0000653, CL:0002518, CL:1000507] | [**kidney epithelial cell (em)**] |
| Liver-Hep | [Liver] | [CL:0000182] | [**hepatocyte (em)**] |
| Lung-Ep-Alveo | [Lung alveolar, Lung pleural] | [CL:0000077, CL:0002062] | [**mesothelial cell (e)**, **pulmonary alveolar type 1 cell (em)**, pulmonary alveolar type 2 cell (m)] |
| Lung-Ep-Bron | [Lung bronchus] | [CL:0002328] | [club cell (m), lung multiciliated epithelial cell (m), pulmonary ionocyte (m), respiratory tract goblet cell (m), serous cell of epithelium of bronchus (m)] |
| Neuron | [Brain] | [CL:0000540] | [**neuron (em)**] |
| Oligodend | [Brain] | [CL:0000128] | — |
| Ovary+Endom-Ep | [Endometrium, Ovary] | [CL:0002132, CL:4030040] | [**stromal cell of ovary (em)**, epithelial cell of uterus (m), ovarian surface epithelial cell (m), theca cell (m)] |
| Pancreas-Acinar | [Pancreas] | [CL:0002064] | [**pancreatic acinar cell (em)**] |
| Pancreas-Alpha | [Pancreas] | [CL:0000171] | [**pancreatic A cell (em)**] |
| Pancreas-Beta | [Pancreas] | [CL:0000169] | [**type B pancreatic cell (em)**] |
| Pancreas-Delta | [Pancreas] | [CL:0000173] | [**pancreatic D cell (em)**] |
| Pancreas-Duct | [Pancreas] | [CL:0002079] | [**pancreatic ductal cell (em)**] |
| Prostate-Ep | [Prostate] | [CL:0002231] | [basal cell of prostate epithelium (m), luminal cell of prostate epithelium (m)] |

*(continued on next page)*

*(continued)*

| Source Tissue | Cell type | CL term | Tabula Sapiens strict matches |
|---|---|---|---|
| Skeletal-Musc | [Skeletal muscle] | [CL:0000188] | [fast muscle cell (m),<br>skeletal muscle satellite stem cell (m),<br>slow muscle cell (m)] |
| Small-Int-Ep | [Jejunum,<br>Small intestine] | [CL:0000164,<br>CL:1000334] | [BEST4+ enterocyte (m),<br>**enteroendocrine cell (em)**,<br>enterocyte of epithelium proper of duodenum (m),<br>enterocyte of epithelium proper of ileum (m),<br>enterocyte of epithelium proper of jejunum (m),<br>enterocyte of epithelium proper of small intestine (m),<br>enteroendocrine cell of small intestine (m),<br>intestinal crypt stem cell of small intestine (m),<br>intestinal tuft cell (m),<br>paneth cell of epithelium of small intestine (m),<br>small intestine goblet cell (m),<br>transit amplifying cell of small intestine (m)] |
| Smooth-Musc | [Aorta,<br>Bladder,<br>Bronchial,<br>Cornary artery,<br>Prostate] | [CL:0002539,<br>CL:0002592,<br>CL:0002597,<br>CL:0002598,<br>CL:1000487] | [**bronchial smooth muscle cell (em)**,<br>blood vessel smooth muscle cell (m),<br>myometrial cell (m),<br>smooth muscle cell (m),<br>vascular associated smooth muscle cell (m)] |
| Thyroid-Ep | [Thyroid] | [CL:0002257] | — |

# D Details of experimental setups

## D.1 Split construction

To construct the training, validation and test splits, we follow a procedure that ensures every target cell type is represented in all splits while isolating biological sources as much as possible. At the beginning of the procedure, we aim for a 70:15:15 ratio between the splits. Target cell types do not have an equal number of biological samples covering them. Therefore, we establish the following rules, which we follow in the stated order, and in such a way that rule $j$ is only applied in a way that does not contradict all rules $i < j$:

1. If a target cell type is covered by a single sample, then the data from this sample is divided across all three splits.
2. If a target cell type is covered by exactly two samples, one is allocated to train, and the other is split between validation and test sets.
3. If a target cell type is covered by three or more biological samples, those samples must not be individually broken down into separate splits, and the validation and test splits must receive at least one full sample.

Applying the described procedure results in the final train/val/test ratios 62:19:19 with four samples shared across all three splits (GEO *GSM5652204, GSM5652218,GSM5652321, GSM5652358*) and two samples shared across validation and test split (GEO *GSM5652199, GSM5652205*) from a total of 205 samples.

## D.2 Pseudobulk generation procedure

Algorithm 2 describes the sampling procedure used to generate the pseudobulk mixtures employed in this paper. The key parameter is the total size $N$ of the pseudobulk, *i.e.* the number of reads to subsample per split. Across all splits, the number of reads per originating cell type ranged from $4{,}270$ to $464{,}002$, with a mean of $59{,}977$. We used $N = 475{,}000$ across all experiments, resulting in $475{,}000/39 \approx 12{,}179$ reads per GR group, per pseudobulk, per split. This value was set heuristically to balance three competing objectives: sufficient representation of each cell type in the resulting mixtures (for which a higher sample count is beneficial); useful sampling noise to improve generalization of the trained model (for which sample counts that do not match the original cell-type contributions are beneficial); and computational efficiency (for which a lower sample count is beneficial). The generated pseudobulks are aggregated using Algorithm 3.

## D.3 Predictions aggregation

Before deconvolution, we aggregate the model's per-read predictions into a compact matrix, where each row corresponds to a GR group and each column to a candidate cell type of origin. This matrix summarises the cell-of-origin information (or prediction information) that the model extracted from the mixture at the read level.

We aggregate using a weighted average, where each read's weight is proportional to the number of CpGs it covers. The intuition is that reads carrying more CpGs convey more evidence about their cell of origin and are statistically more reliable. For long reads with many CpGs, this weighting is essentially a no-op, but in our setting, most reads cover only a handful of CpGs (between four to seven), so down-weighting the shortest reads has a measurable effect.

A second issue arises at inference time. During training, every GR group is well represented, and aggregation is straightforward. In real samples, however, it is common for no read to map to a particular GR group, leaving an entire row of the matrix undefined. In the ODD dataset analyzed in this work, 235 of 260 samples containe one GR group specific to colon fibroblasts with zero reads, and remaining 25 samples contain no more than five reads from this group. This is because the RRBS protocol produces less output than the WGBS data used for training, and coverage gaps are likely due to our employed atlas having up to 25 DMRs per GR group (with each of our GR groups being specific to a single cell type and only 3 DMRs for colon fibroblasts).

To handle this, we precompute a *uniform prior* matrix that serves as a sensible default for under-observed rows. Concretely, we generate $C$ pure pseudobulk mixtures, one per cell type, each made

**Algorithm 2** Pseudobulk mixture generation

**Input:** Set of splits $\mathcal{S} = \{S_1, \ldots, S_K\}$, each containing read-level predictions; *UXM* atlas $A$ and reference cell names $\mathbf{r}$; Number of GR groups $G$; Total number of reads to sample $N$; Total number of cell types $C \geq 10$

**Output:** The cell types mixed in the pseudobulk, their effective proportions and the sampled reads for each split.

$C_{mix} \sim \text{Uniform}(\{1, \ldots, 10\})$ ▷ Sample number of cell types to mix
$c_1, \ldots, c_{C_{mix}} \leftarrow$ sample $C_{mix}$ elements without replacement from $\{1, \ldots, C\}$
$\boldsymbol{\pi} \leftarrow \textsc{SampleProportions}(C_{mix})$ ▷ $C_{mix}$ uniform samples then normalized s.t. $\sum_j \pi_j = 1$
$n_j \leftarrow \lfloor N \cdot \pi_j \rfloor$ for $j = 1, \ldots, C_{mix}$ ▷ Number of reads to sample for each cell type
$N' \leftarrow \sum_{j=1}^{C_{mix}} n_j$ ▷ Effective total number of reads
$\hat{\boldsymbol{\pi}} \leftarrow \mathbf{0}^{C_{mix}}$ ▷ Effective proportion vector
**for** $j = 1, \ldots, C_{mix}$ **do**
    $\hat{\pi}_{c_j} \leftarrow n_j / N'$
**for all** split $S_k \in \mathcal{S}$ **do**
    $\mathcal{R}^k \leftarrow \emptyset$ ▷ Set of sampled reads for split $S_k$
    **for** $j = 1, \ldots, C_{mix}$ **do** ▷ Iterate over cell types to mix
        $(m_{j,1}, \ldots, m_{j,G}) \sim \text{Multinomial}(n_j;\ \frac{1}{G}, \ldots, \frac{1}{G})$ ▷ Allocate read counts to GR groups
        **for** $g = 1, \ldots, G$ **do** ▷ Iterate over GR groups
            $\mathcal{R}_g^k \leftarrow \{r \in S_k \mid r \text{ originates from } c_j \wedge r \text{ is in GR group } g\}$
            $\mathcal{R}^k \leftarrow \mathcal{R}^k \cup \{\text{sampled } m_{j,g} \text{ reads from } \mathcal{R}_g^k \text{ with replacement}\}$
**return** $(c_1, \ldots, c_{C_{mix}}),\ \hat{\boldsymbol{\pi}},\ \{\mathcal{R}^k\}_{1 \leq k \leq K}$

entirely of reads from a single cell type $c$. Then we aggregate each of them and average the resulting matrices. The result approximates the aggregated matrix one would obtain from a mixture in which all cell types contribute equally, and provides our best uninformed guess for any GR group we fail to observe. Each row of the aggregated matrix obtained from prediction is then shrunk toward the corresponding prior row, with the strength of shrinkage determined by how many reads were actually observed for that GR group. Thus, well-covered rows are barely affected, while empty rows are replaced entirely by the prior. More sophisticated imputation schemes that exploit conditional dependencies between rows are possible, but lie outside the scope of this work.

Algorithm 3 describes the aggregation step in detail, and Algorithm 4 describes how the uniform prior is computed.

### D.4 Deconvolution features selection

As depicted in Figure 2, the prediction aggregation step produces a matrix with dimensions $(G, C)$. In our experiments with soft labels, it yields a $(39, 39)$ matrix and a $(39, 40)$ matrix for the models trained on hard labels with an additional background class. However, not every element in the resulting matrix is equally informative. First, for well-separated cell types where the resulting data-driven soft labels will not be much different from the hard labels for the same instance, the diagonal will be the most informative part of the matrix regardless of the underlying labeling scheme. Second, for the hard labeling scheme, no cross-talk off-diagonal is expected; even for the soft labeling scheme, considering that most of the regions in the atlas are highly specific (cf. Section C.3), only some of the off-diagonal features will be informative. Figure S5 illustrates this considerations. For example, *erythrocyte progenitors*, the most specific entry in the atlas according to the Proxy GSS, also confound GR groups specific to other cell types, as evidenced by higher diagonal scores in the pure mixture across both hard and soft labeling schemes. In contrast, *heart cardiomyocytes*, another well-separated cell type, almost does not "fire" anywhere else except in its target region when the soft labeling scheme is applied. Finally, *dermal fibroblast*, which represents one of the least specific cell types in the reference used, shows some off-diagonal signal captured only by the soft labeling scheme. The hard labeling scheme, however, might provide additional information in the background class.

Considering that most of the data in the resulting matrices is not informative, we decided to apply a filter before deconvolution, specific to each labeling scheme. For soft labeling, we keep only 156 (roughly 10%) of the most relevant features. To score each feature, we computed the feature

**Algorithm 3** Aggregate read-level predictions at the GR group level

**Input:** Dataset $\{(r_i, \mathbf{u_i}, g_i, w_i)\}_{i=1}^{N}$ where $r_i$ is a read, $\mathbf{u_i} \in \Delta^{C-1}$ is the vector of model predictions for that read, $g_i$ is the GR group it belongs to and $w_i$ is the number of CpGs sites covered by the read; Uniform prior matrix $Q \in \mathbb{R}^{G \times C}$ (output of Algorithm 4); Prior weight $w_{\text{prior}} \geq 0$ (default is $w_{\text{prior}} = 1$)
**Output:** Aggregated prediction matrix with weighted-average predictions per GR group, blended with prior

$P \leftarrow$ matrix of dimension $G \times C$ initialized to zero ▷ Aggregated prediction matrix
**for all** GR group g in $\{1, \ldots, G\}$ **do**
  $I_g \leftarrow \{i \mid g_i = g\}$ ▷ Indices of reads in GR group $g$
  $W_g \leftarrow \sum_{i \in I_g} w_i$ ▷ Total weight for GR group $g$
  **if** $W_g > 0$ **then**
    **for all** c in $\{1, \ldots, C\}$ **do**
      $P_{g,c} \leftarrow \frac{1}{W_g} \sum_{i \in I_g} w_i \, u_{i,c}$ ▷ Weighted avg. prediction for GR group $g$ and cell type $c$
      $n_g \leftarrow |I_g|$ ▷ Number of reads in GR group $g$
**// Prior blending: shrink every row towards the uniform prior**
**for all** GR group $g \in \{1, \ldots, G\}$ **do**
  **for all** c in $\{1, \ldots, C\}$ **do**
    $$P_{g,c} \leftarrow \frac{n_g}{n_g + w_{\text{prior}}} P_{g,c} + \frac{w_{\text{prior}}}{n_g + w_{\text{prior}}} Q_{g,c}$$
**return** $P$

**Algorithm 4** Compute uniform prior matrix from pure pseudobulk profiles

**Input:** Number of cell types $C$; Reads per pure profile $N_{\text{pure}}$
**Output:** Uniform prior matrix $Q \in \mathbb{R}^{G \times C}$

**for** $c = 1, \ldots, C$ **do** ▷ Generate one pure prediction matrix per cell type
  $\mathcal{R}_c \leftarrow$ sampled reads from Algorithm 2 where the mixture is made entirely of cell type $c$
  $P^{(c)} \leftarrow$ AGGREGATEPREDICTIONSBYGRGROUP($\mathcal{R}_c$) ▷ Algorithm 3 without prior blending
//Average pure profiles to get uniform prior
$Q \leftarrow \frac{1}{C} \sum_{c=1}^{C} P^{(c)}$
**return** $Q$

matrices for all in-silico mixtures with a single target cell type. We then calculated the maximum and mean values for each matrix element across the resulting 39 profiles. The final score was calculated as the ratio between these two values. We ranked all features by descending score, selected all diagonal features regardless of their scores, and added the remaining top 117 features, resulting in 156 features in total. For hard labeling, in addition to using the *Top 156* features (excluding the background column), we also use a feature selection scheme that selects only the diagonal and the background column. Indeed, as can be seen from Table S9, *Top 156* performs almost as well as the full feature matrix for the soft labeling scheme, while for hard labeling, the performance is no different when either only the diagonal or any number of additional features is used. This corroborates our hypothesis that the hard labeling scheme is steering the classifier to eliminate any cross-talk by concentrating on separating on-target from off-target. Therefore, the *PSLS* deconvolver treats off-diagonal information as noise and ignores it; also, it has to be noted that this is not guaranteed, as one can envision scenarios where spurious correlations may still occur between off-diagonal values and true proportions. Furthermore, by looking at the detailed results (Section E.2), one can observe that another unsupervised deconvolver, *NNLS*, obtains better results under the *Diagonal + Bkg.* feature scheme compared to *Top 156* features ($1.61 \times 10^{-}4$ vs $1.82 \times 10^{-}4$). *PSLS* robustness is likely to stem from the enforced at optimization time sum-to-one constraint, which acts as regularization.

As for the soft labeling scheme, it allows cross-talk information to propagate explicitly, which is why adding more features monotonically improves the deconvolution accuracy. Although using the full matrix would give us slightly better results for the soft labeling scheme, we never use it in our other experiments as the computational cost is prohibitive, especially with the high number of setups to

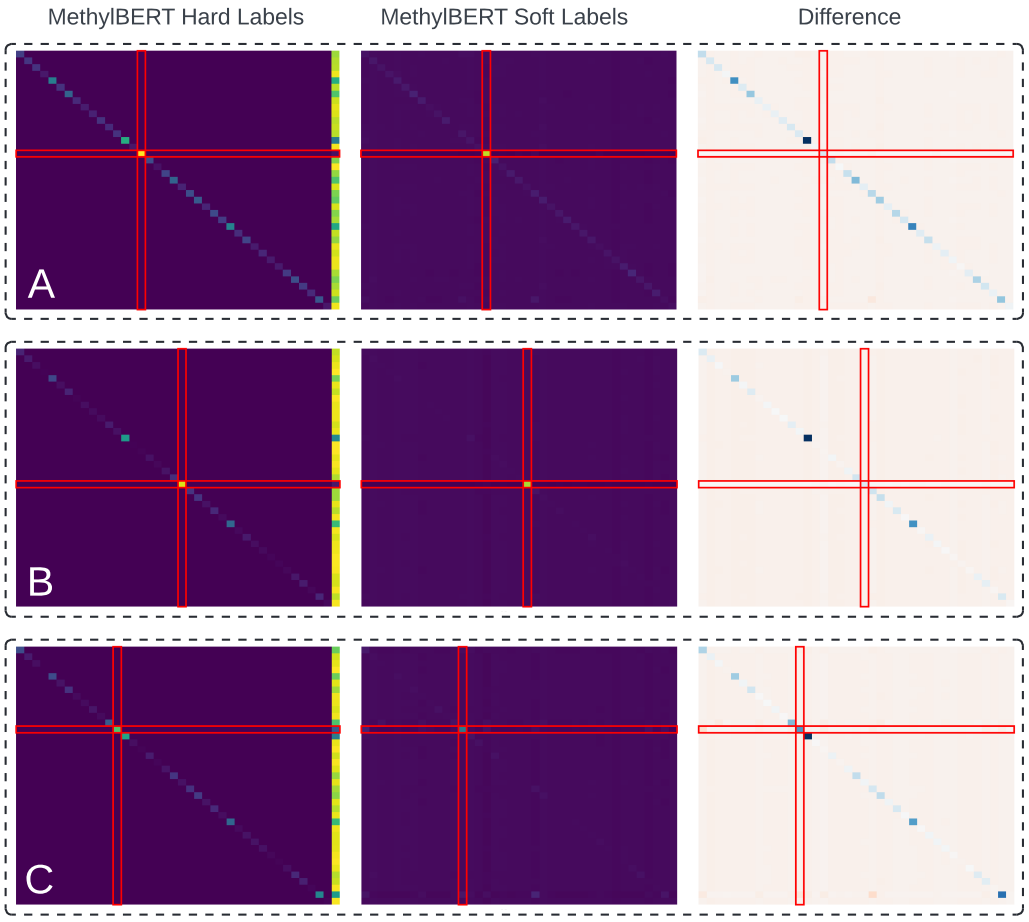


Figure S5: Differences in aggregated prediction matrices for in-silico pure mixtures (the background class column specific to hard labels is omitted). The row and column corresponding to the target cell type are highlighted. **A**: Mixture is composed solely of Erythrocyte Progenitor reads. **B**: Mixture is composed solely of Heart Cardio reads. **C**: Mixture is composed solely of Dermal Fibroblast reads.

test (cf. Figure S6). Instead, selecting the top 156 features is considered asymptotically equivalent to using the entire feature matrix, which is the only setting we utilize across experiments featuring data-driven soft labels.

Table S9: Comparison of feature selection schemes for *MethylBERT* with *PSLS* deconvolver.

| Classifier | Feature Selection Scheme | Deconvolver | Dec. Calib. | $R^2$ ($\times 10^{-2}$) | LoA ($\times 10^{-2}$) | LoA worst-class ($\times 10^{-2}$) | MAE ($\times 10^{-3}$) | MSE ($\times 10^{-4}$) | KL ($\times 10^{-2}$) |
|---|---|---|---|---|---|---|---|---|---|
| *Soft Labels with Pooling* | | | | | | | | | |
| *MethylBERT* | Diagonal | *PSLS* | None | 96.73 | [-3.22, 3.22] | [-12.65, 9.04] | 3.837 | 2.71 | 13.62 |
| | Top 156 | *PSLS* | None | 97.99 | [-2.53, 2.53] | [-9.94, 7.46] | 3.175 | 1.67 | 7.36 |
| | Full Matrix | *PSLS* | None | 98.04 | [-2.50, 2.50] | [-9.62, 7.25] | 3.149 | 1.63 | 7.09 |
| *Hard Labels* | | | | | | | | | |
| *MethylBERT* | Diagonal | *PSLS* | None | 98.03 | [-2.50, 2.50] | [-6.75, 7.36] | 3.584 | 1.63 | 8.73 |
| | Diagonal + Bkg. | *PSLS* | None | 98.03 | [-2.50, 2.50] | [-6.75, 7.36] | 3.585 | 1.63 | 8.71 |
| | Top 156 (w/o background) | *PSLS* | None | 98.03 | [-2.50, 2.50] | [-6.75, 7.37] | 3.584 | 1.63 | 8.71 |
| | Full Matrix | *PSLS* | None | 98.04 | [-2.50, 2.50] | [-6.75, 7.37] | 3.585 | 1.63 | 8.71 |

### D.5 Deconvolution metrics

In this section, we describe the metrics used to evaluate the deconvolution results.

**Deconvolution metrics for the pseudobulk experiment.** We recall the notations used in the main paper. Let $K$ be the number of pseudobulks, $(\mathbf{p}_k)_{1\leq k\leq K}$ be the ground truth mixture proportions and $(\tilde{\mathbf{p}}_k)_{1\leq k\leq K}$ be the final predictions.

$R^2$ is computed as the coefficient of determination across all pseudobulks, variance-weighted per cell-type. It is computed as `sklearn.metrics.r2_score(y_true, y_pred, multioutput="variance_weighted")` in the `scikit-learn` library and the close form writes as:

$$R^2 \coloneqq 1 - \frac{\sum_{k=1}^{K}\sum_{c=1}^{C}(p_{k,c} - \tilde{p}_{k,c})^2}{\sum_{k=1}^{K}\sum_{c=1}^{C}\left(p_{k,c} - \frac{1}{K}\sum_{k'=1}^{K} p_{k',c}\right)^2}. \tag{27}$$

The **MSE** is computed as the average Mean Squared Error over all pseudobulks:

$$\text{MSE} \coloneqq \frac{1}{K}\sum_{k=1}^{K}\frac{1}{C}\sum_{c=1}^{C}(p_{k,c} - \tilde{p}_{k,c})^2. \tag{28}$$

Similarly, the **MAE** is computed as the average Mean Absolute Error over all pseudobulks:

$$\text{MAE} \coloneqq \frac{1}{K}\sum_{k=1}^{K}\frac{1}{C}\sum_{c=1}^{C}|p_{k,c} - \tilde{p}_{k,c}|. \tag{29}$$

The **KL** divergence is also computed as the average Kullback-Leibler divergence over all pseudobulks:

$$\text{KL} \coloneqq \frac{1}{K}\sum_{k=1}^{K}\sum_{c=1}^{C} p_{k,c}\log\left(\frac{p_{k,c}}{\tilde{p}_{k,c}}\right). \tag{30}$$

where both $p_{k,c}$ and $\tilde{p}_{k,c}$ are lower clipped to a small $\epsilon$ for numerical stability in the logarithm (we used $\epsilon = 10^{-8}$).

**LoA** denotes the overall Limits of Agreements defined as $\text{LoA} \coloneqq [\text{LoA}_{\text{lower}}, \text{LoA}_{\text{upper}}]$ where:

$$\text{LoA}_{\text{lower}} \coloneqq \text{bias} - 1.96 \times \text{std}, \tag{31}$$

$$\text{LoA}_{\text{upper}} \coloneqq \text{bias} + 1.96 \times \text{std}, \tag{32}$$

$$\text{bias} \coloneqq \frac{1}{K}\sum_{k=1}^{K}\frac{1}{C}\sum_{c=1}^{C}(p_{k,c} - \tilde{p}_{k,c}), \tag{33}$$

$$\text{std} \coloneqq \sqrt{\frac{1}{KC-1}\sum_{k=1}^{K}\sum_{c=1}^{C}(p_{k,c} - \tilde{p}_{k,c})^2}. \tag{34}$$

**LoA worst-class** denotes the worst Limits of Agreements per class. For all $c \in \mathcal{C}$ we define the class LoA as $\text{LoA}_c \coloneqq [\text{LoA}_{\text{lower},c}, \text{LoA}_{\text{upper},c}]$ where:

$$\text{LoA}_{\text{lower},c} \coloneqq \text{bias}_c - 1.96 \times \text{std}_c, \tag{35}$$

$$\text{LoA}_{\text{upper},c} \coloneqq \text{bias}_c + 1.96 \times \text{std}_c, \tag{36}$$

$$\text{bias}_c \coloneqq \frac{1}{K}\sum_{k=1}^{K}(p_{k,c} - \tilde{p}_{k,c}), \tag{37}$$

$$\text{std}_c \coloneqq \sqrt{\frac{1}{K-1}\sum_{k=1}^{K}(p_{k,c} - \tilde{p}_{k,c})^2}. \tag{38}$$

Then, for all classes $c \in \mathcal{C}$, we then compute the LoA width $\text{LoA}_{\text{width},c} \coloneqq \text{LoA}_{\text{upper},c} - \text{LoA}_{\text{lower},c}$ and we report the class LoA with largest width.

**Deconvolution metrics for the OOD RRBS dataset.** As mentionned in the main paper, the OOD dataset is a dataset of samples originating from known tissues, but with unknown cell type proportions. We inferred the expected cell types in each tissue following the procedure described in Section 4.1 and Section C.4. To evaluate the deconvolution results, we rely on the insight that cell types which are not expected in the sample's tissue should not appear in the deconvolution results. This is equivalent to saying that the predicted proportions of cell types expected in the sample's tissue should sum to a high value, ideally one. We define the "Tissue Concordance Score" (or *TCS*) as the sum of predicted proportions of cell types expected in the sample's tissue. Formally, let $\tilde{\mathbf{p}} \in \Delta^{C-1}$ be the predicted mixture proportions for a sample of tissue $t$ and let $\mathcal{C}_t \subseteq \mathcal{C}$ be the set of cell types expected in tissue $t$ according to the mapping described in Section 4.1 and Section C.4. Then, the Tissue Concordance Score is defined as:

$$\text{TCS} \coloneqq \sum_{c \in \mathcal{C}_t} \tilde{p}_c. \tag{39}$$

### D.6 Possible combinations of elements in *Syto*

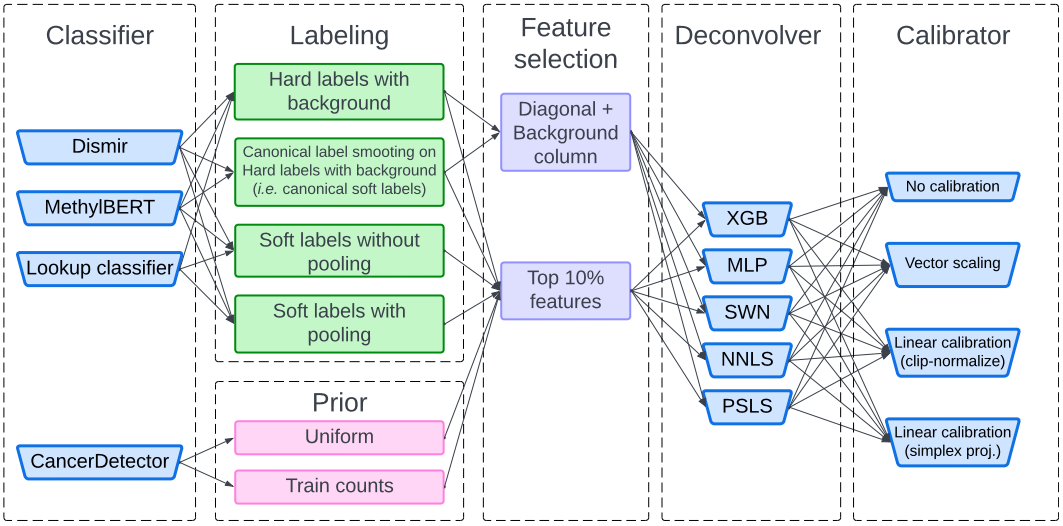


Figure S6: Representation of the design choices when constructing an instance of *Syto*. Each path from classifier to calibrator is a possible choice. There is a total of 360 possible paths, and we explore all of them in our experiments ($360 = (2 \times (2 \times 2 + 2 \times 1) + 1 \times (1 \times 2 + 2 \times 1) + 1 \times 2 \times 1) \times 5 \times 4$).

# E Integral experimental results

## E.1 Joint evaluation of pseudobulk and OOD results

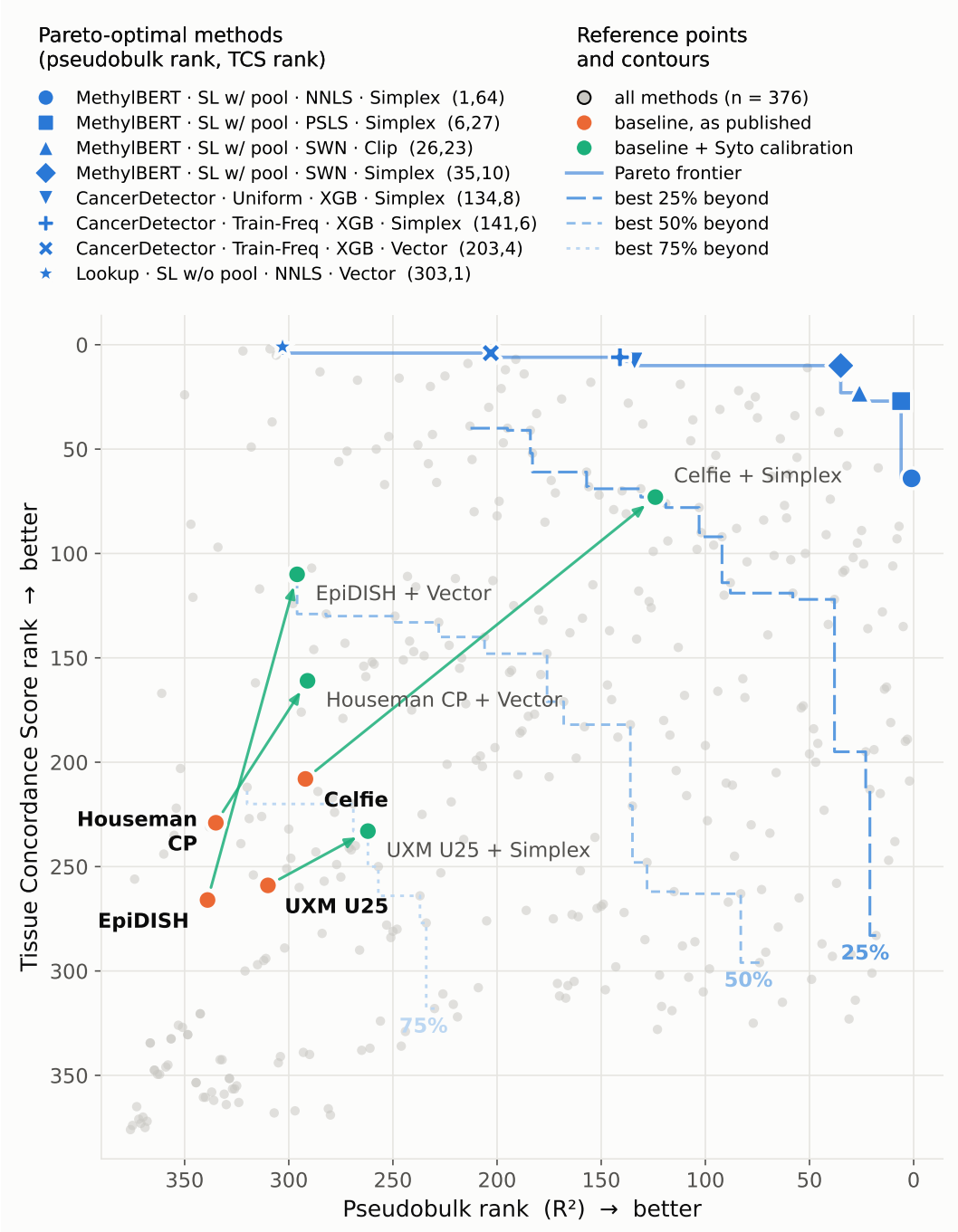


Figure S7: Joint ranking of examined 360 *Syto* variants and *baselines* as measured by pseudobulk's $R^2$ and OOD TCS. The best methods are in the top right corner.

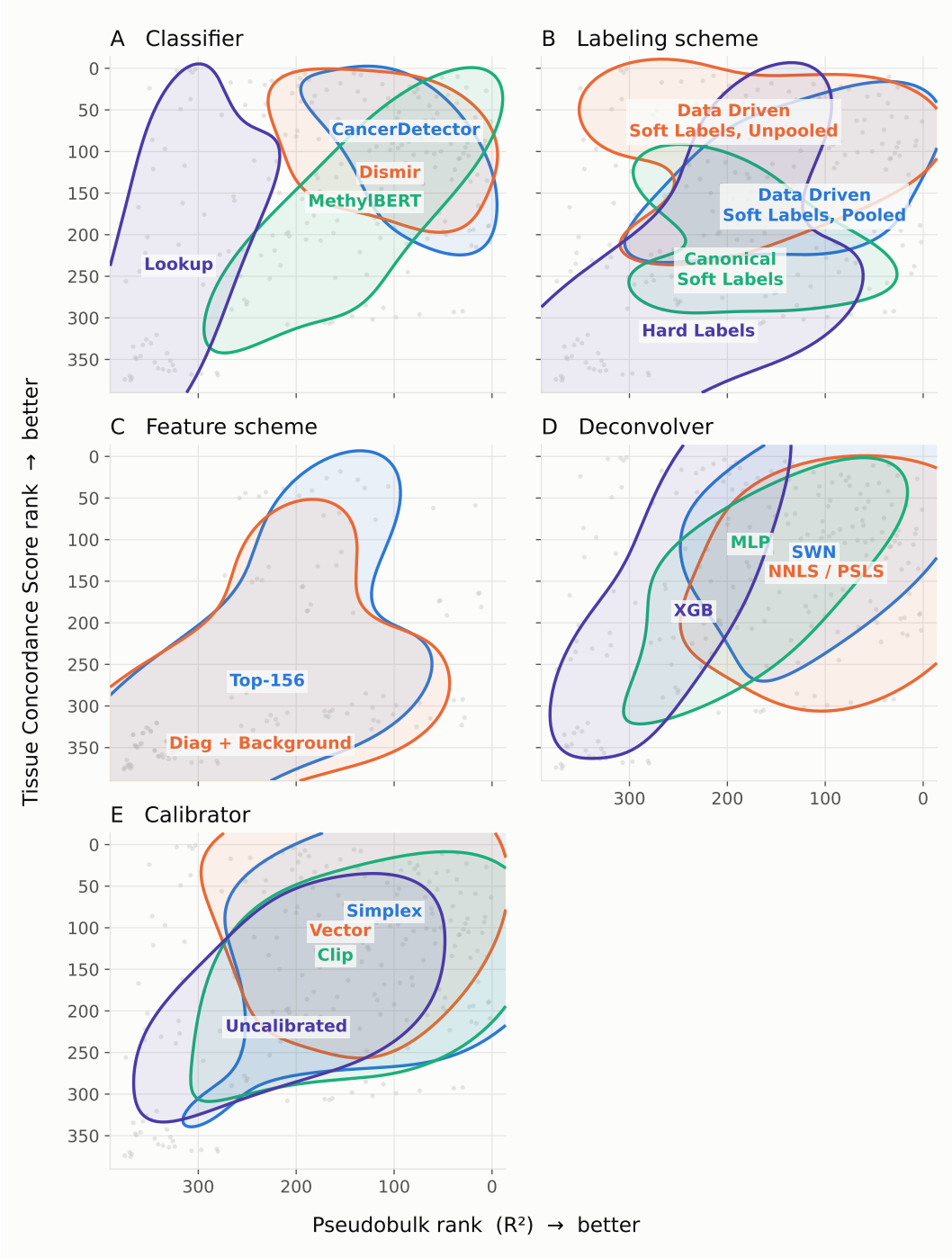


Figure S8: KDE contours of joint metrics plane for the methods' results grouped by different *Syto* configuration choices. Contours outline the highest-density region containing 50% of the fitted KDE probability mass represented on the grid, drawn from a two-dimensional Gaussian kernel density estimate (`scipy.stats.gaussian_kde`, bandwidth factor 0.70, evaluated on a $140 \times 140$ grid). Contour labels are anchored to their corresponding groups' medians. All panels exclude *Diag + Background* feature scheme, except panel **(C)** that is built solely on *Hard Labels* results, where both *Diag + Background* and *Top-156* choices are present. *Canonical Soft Labels* results are only reflected in panel **(B)** and excluded elsewhere. *NNLS* and *PSLS* in panel **(D)** are statistically indistinguishable and therefore represented with the same contour.

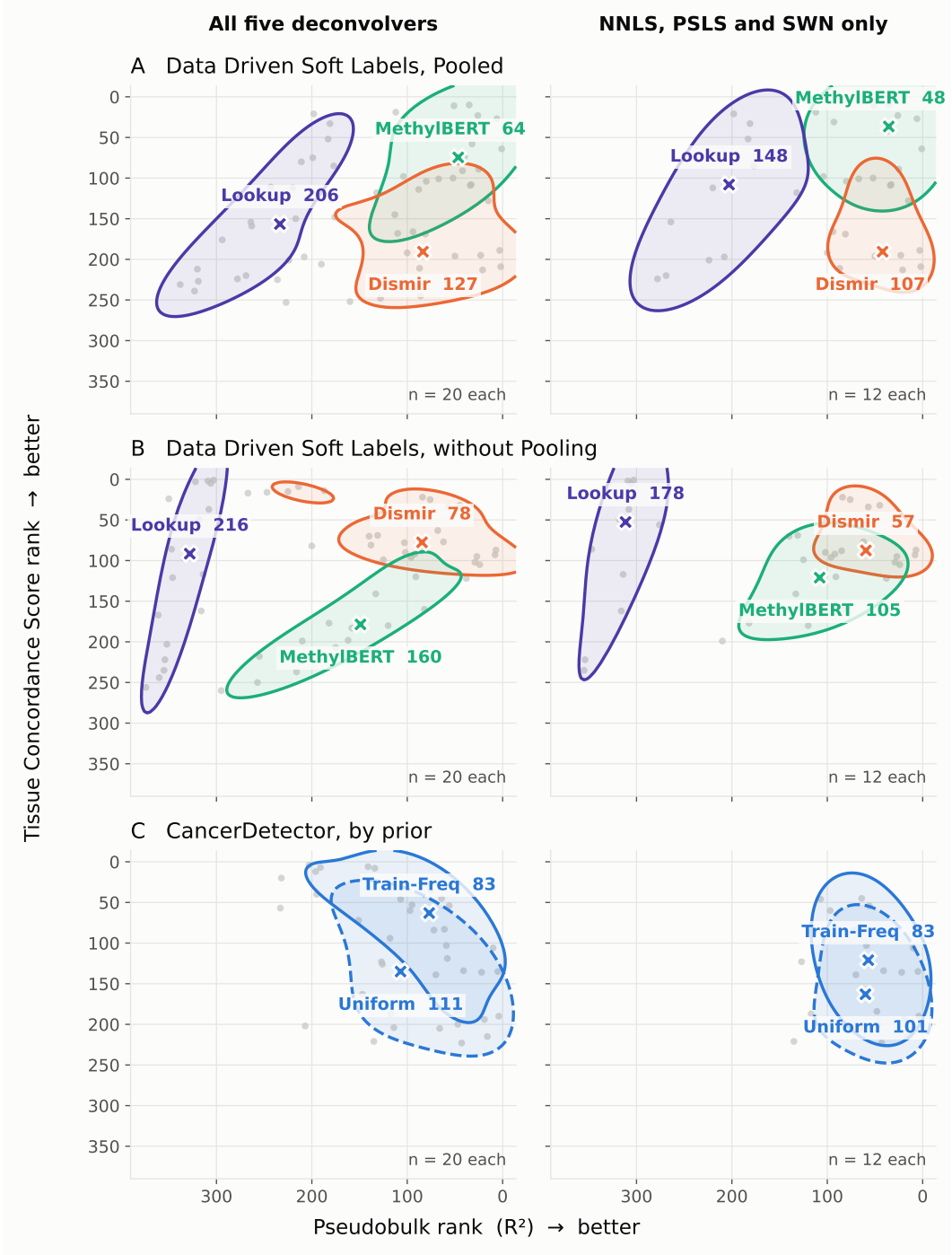


Figure S9: KDE contours of joint metrics plane for a subset of *Syto* configurations. Contours outline the highest-density region containing 50% of the fitted KDE probability mass represented on the grid, drawn from a two-dimensional Gaussian kernel density estimate. (`scipy.stats.gaussian_kde`, bandwidth factor 0.70 for $n = 20$ and 0.95 for $n = 12$, evaluated on a $140 \times 140$ grid). Panels **(A)** and **(B)** represent two best labeling schemes. Panel **(C)** is focused on *CancerDetector* with different priors. The colored $\times$ markers represent each group's median combined rank. *Syto* configurations grounded in *MethylBERT* fitted on *Data Driven Labels with Pooling* outperform options built on any other combination of labeling scheme and classifier.

## E.2 Pseudobulk experiment detailed results

Table S10: Full results of the pseudobulk experiment for all baselines, with and without our calibration methods. For $R^2$, MAE, MSE and KL, we indicated first the value of the metric computed on the whole test set, and then the 95% confidence interval computed with BCa bootstrap [47] with 10,000 resamples.

| Baseline | Dec. Calib. | $\mathbf{R^2}$ ($\times 10^{-2}$) | LoA ($\times 10^{-2}$) | LoA worst-class ($\times 10^{-2}$) | MAE ($\times 10^{-3}$) | MSE ($\times 10^{-4}$) | KL ($\times 10^{-2}$) |
|---|---|---|---|---|---|---|---|
| *CelFiE* [11] | None | 96.05 [96.00, 96.10] | [-3.55, 3.55] | [-9.96, 9.15] | 6.42 [6.40, 6.44] | 3.28 [3.24, 3.32] | 14.29 [14.23, 14.34] |
| | Lin. clip+norm | 97.41 [97.39, 97.44] | [-2.87, 2.87] | [-6.30, 7.98] | 4.33 [4.31, 4.34] | 2.15 [2.12, 2.17] | 10.44 [10.36, 10.53] |
| | Lin. simplex | 98.03 [98.01, 98.06] | [-2.50, 2.50] | [-5.25, 7.11] | 3.47 [3.45, 3.48] | 1.63 [1.61, 1.65] | 11.31 [11.19, 11.44] |
| | Vec. scaling | 96.36 [96.32, 96.39] | [-3.41, 3.41] | [-5.85, 8.52] | 5.52 [5.50, 5.54] | 3.02 [3.01, 3.04] | 8.59 [8.55, 8.63] |
| *EpiDISH* [43] | None | 89.41 [89.28, 89.54] | [-5.81, 5.81] | [-15.79, 14.13] | 10.20 [10.16, 10.23] | 8.79 [8.68, 8.90] | 25.37 [25.25, 25.49] |
| | Lin. clip+norm | 93.53 [93.45, 93.61] | [-4.54, 4.54] | [-11.17, 10.95] | 7.00 [6.97, 7.03] | 5.37 [5.30, 5.45] | 16.57 [16.48, 16.66] |
| | Lin. simplex | 95.90 [95.85, 95.96] | [-3.61, 3.61] | [-9.86, 9.37] | 4.99 [4.97, 5.02] | 3.40 [3.35, 3.45] | 12.75 [12.66, 12.83] |
| | Vec. scaling | 95.95 [95.92, 95.98] | [-3.59, 3.59] | [-7.85, 6.75] | 6.02 [6.00, 6.04] | 3.36 [3.34, 3.38] | 10.54 [10.49, 10.58] |
| *Houseman CP* [43, 9] (no inequality constraint) | None | 90.94 [90.83, 91.05] | [-5.37, 5.37] | [-15.57, 13.22] | 9.25 [9.22, 9.28] | 7.52 [7.42, 7.62] | 22.92 [22.81, 23.02] |
| | Lin. clip+norm | 94.18 [94.11, 94.24] | [-4.31, 4.31] | [-10.71, 10.28] | 6.90 [6.87, 6.92] | 4.83 [4.77, 4.90] | 16.88 [16.79, 16.96] |
| | Lin. simplex | 96.51 [96.47, 96.55] | [-3.34, 3.34] | [-9.16, 8.57] | 4.91 [4.89, 4.92] | 2.90 [2.86, 2.93] | 13.20 [13.11, 13.30] |
| | Vec. scaling | 96.05 [96.02, 96.09] | [-3.55, 3.55] | [-7.71, 6.51] | 5.98 [5.96, 6.00] | 3.28 [3.25, 3.30] | 11.14 [11.08, 11.19] |
| *UXM* [13] | None | 94.86 [94.78, 94.93] | [-4.05, 4.05] | [-10.82, 14.02] | 6.49 [6.47, 6.52] | 4.27 [4.20, 4.34] | 14.98 [14.89, 15.06] |
| | Lin. clip+norm | 95.91 [95.84, 95.97] | [-3.61, 3.61] | [-10.89, 13.04] | 5.33 [5.31, 5.35] | 3.40 [3.34, 3.46] | 12.33 [12.25, 12.42] |
| | Lin. simplex | 96.74 [96.68, 96.79] | [-3.23, 3.23] | [-10.78, 12.98] | 4.32 [4.30, 4.34] | 2.71 [2.66, 2.76] | 10.91 [10.81, 11.00] |
| | Vec. scaling | 96.33 [96.29, 96.38] | [-3.42, 3.42] | [-9.43, 10.33] | 5.20 [5.18, 5.22] | 3.04 [3.01, 3.08] | 9.10 [9.03, 9.16] |

Table S11: Full results for our extension of **CancerDetector** [18]. For $R^2$, MAE, MSE and KL, we indicated first the value of the metric computed on the whole test set, and then the 95% confidence interval computed with BCa bootstrap [47] with 10,000 resamples.

| Prior | Deconvolver | Dec. Calib. | $\mathbf{R^2}$ ($\times 10^{-2}$) | LoA ($\times 10^{-2}$) | LoA worst-class ($\times 10^{-2}$) | MAE ($\times 10^{-3}$) | MSE ($\times 10^{-4}$) | KL ($\times 10^{-2}$) |
|---|---|---|---|---|---|---|---|---|
| Training Counts | XGB | None | 97.32 [97.27, 97.37] | [-2.92, 2.92] | [-8.14, 8.45] | 3.68 [3.66, 3.70] | 2.22 [2.18, 2.26] | 6.47 [6.43, 6.52] |
| | | Lin. clip+norm | 97.66 [97.62, 97.71] | [-2.73, 2.73] | [-8.07, 8.14] | 3.37 [3.35, 3.38] | 1.94 [1.90, 1.98] | 6.47 [6.41, 6.52] |
| | | Lin. simplex | 97.96 [97.92, 98.00] | [-2.55, 2.55] | [-8.00, 7.93] | 3.08 [3.07, 3.10] | 1.69 [1.65, 1.73] | 7.17 [7.10, 7.25] |
| | | Vec. scaling | 97.60 [97.57, 97.64] | [-2.76, 2.76] | [-6.96, 7.14] | 3.89 [3.87, 3.90] | 1.99 [1.96, 2.02] | 5.22 [5.18, 5.25] |
| | MLP | None | 97.91 [97.88, 97.93] | [-2.58, 2.58] | [-5.52, 8.12] | 3.61 [3.59, 3.62] | 1.74 [1.72, 1.76] | 6.23 [6.19, 6.28] |
| | | Lin. clip+norm | 98.17 [98.14, 98.19] | [-2.42, 2.42] | [-5.49, 7.26] | 3.55 [3.54, 3.57] | 1.52 [1.50, 1.54] | 7.48 [7.41, 7.54] |
| | | Lin. simplex | 98.28 [98.26, 98.30] | [-2.34, 2.34] | [-5.44, 7.13] | 3.38 [3.37, 3.39] | 1.43 [1.41, 1.44] | 9.11 [9.02, 9.20] |
| | | Vec. scaling | 98.29 [98.27, 98.31] | [-2.34, 2.34] | [-5.49, 6.03] | 3.46 [3.44, 3.47] | 1.42 [1.41, 1.43] | 5.44 [5.40, 5.48] |
| | SWN | None | 98.02 [98.00, 98.05] | [-2.51, 2.51] | [-6.78, 8.72] | 3.23 [3.21, 3.24] | 1.64 [1.62, 1.66] | 5.43 [5.39, 5.47] |
| | | Lin. clip+norm | 98.29 [98.27, 98.32] | [-2.33, 2.33] | [-7.00, 7.83] | 3.10 [3.09, 3.12] | 1.42 [1.40, 1.44] | 6.34 [6.28, 6.40] |
| | | Lin. simplex | 98.37 [98.35, 98.39] | [-2.28, 2.28] | [-7.20, 7.86] | 3.04 [3.02, 3.05] | 1.35 [1.33, 1.37] | 7.80 [7.72, 7.88] |
| | | Vec. scaling | 98.32 [98.30, 98.34] | [-2.31, 2.31] | [-6.67, 6.86] | 3.32 [3.30, 3.33] | 1.39 [1.38, 1.41] | 4.78 [4.75, 4.82] |
| | NNLS | None | 98.07 [98.04, 98.09] | [-2.48, 2.48] | [-7.25, 7.14] | 3.60 [3.59, 3.61] | 1.61 [1.58, 1.63] | 7.52 [7.46, 7.58] |
| | | Lin. clip+norm | 98.40 [98.38, 98.42] | [-2.26, 2.26] | [-6.27, 6.70] | 3.22 [3.20, 3.23] | 1.32 [1.31, 1.34] | 6.61 [6.55, 6.67] |
| | | Lin. simplex | 98.74 [98.72, 98.75] | [-2.01, 2.01] | [-5.53, 5.99] | 2.82 [2.81, 2.83] | 1.05 [1.04, 1.06] | 6.12 [6.06, 6.19] |
| | | Vec. scaling | 98.10 [98.09, 98.12] | [-2.46, 2.46] | [-4.92, 5.08] | 3.85 [3.83, 3.86] | 1.57 [1.56, 1.58] | 5.62 [5.57, 5.67] |
| | PSLS | None | 98.41 [98.39, 98.44] | [-2.25, 2.25] | [-6.99, 6.10] | 3.05 [3.04, 3.07] | 1.32 [1.30, 1.34] | 6.18 [6.12, 6.23] |
| | | Lin. clip+norm | 98.68 [98.66, 98.70] | [-2.05, 2.05] | [-5.18, 5.31] | 2.96 [2.95, 2.97] | 1.09 [1.08, 1.11] | 6.12 [6.05, 6.18] |
| | | Lin. simplex | 98.90 [98.89, 98.91] | [-1.87, 1.87] | [-4.38, 4.67] | 2.74 [2.73, 2.75] | 0.91 [0.90, 0.92] | 6.17 [6.10, 6.24] |
| | | Vec. scaling | 98.36 [98.35, 98.37] | [-2.29, 2.29] | [-5.34, 5.47] | 3.53 [3.51, 3.54] | 1.36 [1.35, 1.37] | 5.46 [5.41, 5.51] |
| Uniform | XGB | None | 97.31 [97.26, 97.36] | [-2.93, 2.93] | [-7.76, 8.08] | 3.68 [3.66, 3.70] | 2.23 [2.19, 2.27] | 6.51 [6.46, 6.55] |
| | | Lin. clip+norm | 97.67 [97.62, 97.71] | [-2.73, 2.73] | [-7.50, 7.79] | 3.36 [3.34, 3.37] | 1.94 [1.90, 1.98] | 6.42 [6.37, 6.47] |
| | | Lin. simplex | 97.98 [97.94, 98.03] | [-2.54, 2.54] | [-7.33, 7.51] | 3.06 [3.05, 3.07] | 1.67 [1.64, 1.71] | 7.06 [6.99, 7.13] |
| | | Vec. scaling | 97.68 [97.64, 97.72] | [-2.72, 2.72] | [-6.28, 6.81] | 3.84 [3.83, 3.86] | 1.93 [1.90, 1.96] | 5.14 [5.11, 5.17] |
| | MLP | None | 97.57 [97.54, 97.60] | [-2.78, 2.78] | [-6.40, 9.46] | 3.74 [3.73, 3.76] | 2.01 [1.99, 2.04] | 6.64 [6.60, 6.69] |
| | | Lin. clip+norm | 97.93 [97.90, 97.95] | [-2.57, 2.57] | [-6.37, 8.11] | 3.58 [3.56, 3.59] | 1.72 [1.70, 1.74] | 7.84 [7.77, 7.90] |
| | | Lin. simplex | 98.03 [98.00, 98.05] | [-2.51, 2.51] | [-6.52, 8.07] | 3.48 [3.46, 3.49] | 1.64 [1.62, 1.66] | 9.71 [9.62, 9.81] |
| | | Vec. scaling | 98.05 [98.02, 98.07] | [-2.50, 2.50] | [-6.39, 6.79] | 3.57 [3.56, 3.59] | 1.62 [1.60, 1.64] | 5.78 [5.74, 5.82] |
| | SWN | None | 98.05 [98.02, 98.08] | [-2.49, 2.49] | [-7.07, 9.35] | 3.20 [3.19, 3.22] | 1.62 [1.60, 1.64] | 5.52 [5.48, 5.56] |
| | | Lin. clip+norm | 98.31 [98.28, 98.33] | [-2.32, 2.32] | [-7.38, 8.24] | 3.09 [3.08, 3.10] | 1.40 [1.38, 1.43] | 6.50 [6.44, 6.56] |
| | | Lin. simplex | 98.36 [98.33, 98.39] | [-2.29, 2.29] | [-7.65, 8.31] | 3.03 [3.02, 3.04] | 1.36 [1.34, 1.38] | 7.99 [7.91, 8.07] |
| | | Vec. scaling | 98.33 [98.31, 98.35] | [-2.31, 2.31] | [-7.08, 7.18] | 3.28 [3.27, 3.30] | 1.38 [1.37, 1.40] | 4.86 [4.82, 4.89] |
| | NNLS | None | 97.98 [97.96, 98.01] | [-2.54, 2.54] | [-7.49, 7.95] | 3.66 [3.64, 3.67] | 1.67 [1.65, 1.70] | 7.56 [7.50, 7.61] |
| | | Lin. clip+norm | 98.32 [98.29, 98.34] | [-2.32, 2.32] | [-6.97, 7.49] | 3.27 [3.26, 3.29] | 1.40 [1.38, 1.42] | 6.68 [6.63, 6.74] |
| | | Lin. simplex | 98.64 [98.62, 98.65] | [-2.08, 2.08] | [-6.39, 6.82] | 2.87 [2.86, 2.88] | 1.13 [1.12, 1.14] | 6.14 [6.07, 6.20] |
| | | Vec. scaling | 98.14 [98.13, 98.16] | [-2.43, 2.43] | [-5.48, 5.53] | 3.85 [3.84, 3.86] | 1.54 [1.53, 1.55] | 5.58 [5.53, 5.63] |
| | PSLS | None | 98.44 [98.42, 98.46] | [-2.23, 2.23] | [-6.73, 6.46] | 3.02 [3.01, 3.04] | 1.30 [1.28, 1.31] | 6.03 [5.98, 6.08] |
| | | Lin. clip+norm | 98.69 [98.67, 98.70] | [-2.05, 2.05] | [-5.45, 5.91] | 2.95 [2.93, 2.96] | 1.09 [1.08, 1.10] | 5.98 [5.92, 6.04] |
| | | Lin. simplex | 98.91 [98.90, 98.92] | [-1.87, 1.87] | [-4.80, 5.32] | 2.72 [2.71, 2.73] | 0.91 [0.90, 0.91] | 6.00 [5.94, 6.07] |
| | | Vec. scaling | 98.46 [98.45, 98.47] | [-2.22, 2.22] | [-5.39, 5.53] | 3.47 [3.46, 3.48] | 1.28 [1.27, 1.29] | 5.41 [5.36, 5.46] |

Table S12: Full results for **hard labels with background class and Top 156 features for deconvolution**. For $R^2$, MAE, MSE and KL, we indicated first the value of the metric computed on the whole test set, and then the 95% confidence interval computed with BCa bootstrap [47] with 10,000 resamples.

| Classifier | Deconvolver | Dec. Calib. | $\mathbf{R^2}$ ($\times 10^{-2}$) | LoA ($\times 10^{-2}$) | LoA worst-class ($\times 10^{-2}$) | MAE ($\times 10^{-3}$) | MSE ($\times 10^{-4}$) | KL ($\times 10^{-2}$) |
|---|---|---|---|---|---|---|---|---|
| *Dismir* [19] (CNN+LSTM) | XGB | None | 96.99 [96.93, 97.05] | [-3.10, 3.10] | [-8.21, 8.72] | 3.79 [3.77, 3.81] | 2.50 [2.45, 2.55] | 6.69 [6.64, 6.74] |
| | | Lin. clip+norm | 97.36 [97.31, 97.41] | [-2.90, 2.90] | [-7.86, 8.57] | 3.45 [3.43, 3.47] | 2.19 [2.14, 2.24] | 6.47 [6.42, 6.53] |
| | | Lin. simplex | 97.73 [97.68, 97.78] | [-2.69, 2.69] | [-7.67, 8.26] | 3.10 [3.08, 3.11] | 1.88 [1.84, 1.93] | 6.72 [6.65, 6.79] |
| | | Vec. scaling | 97.60 [97.56, 97.64] | [-2.77, 2.77] | [-6.69, 7.80] | 3.87 [3.85, 3.88] | 1.99 [1.96, 2.02] | 5.10 [5.07, 5.13] |
| | MLP | None | 97.39 [97.36, 97.42] | [-2.89, 2.89] | [-7.46, 10.77] | 3.84 [3.82, 3.86] | 2.17 [2.15, 2.19] | 6.91 [6.87, 6.96] |
| | | Lin. clip+norm | 97.74 [97.71, 97.77] | [-2.68, 2.68] | [-7.57, 9.42] | 3.70 [3.69, 3.72] | 1.88 [1.86, 1.90] | 7.98 [7.91, 8.05] |
| | | Lin. simplex | 97.85 [97.83, 97.87] | [-2.62, 2.62] | [-7.89, 9.54] | 3.63 [3.61, 3.64] | 1.78 [1.77, 1.80] | 9.71 [9.62, 9.80] |
| | | Vec. scaling | 97.89 [97.87, 97.91] | [-2.59, 2.59] | [-7.20, 7.92] | 3.75 [3.73, 3.76] | 1.75 [1.73, 1.76] | 5.98 [5.94, 6.03] |
| | SWN | None | 97.98 [97.94, 98.01] | [-2.54, 2.54] | [-7.26, 7.74] | 3.44 [3.42, 3.45] | 1.68 [1.65, 1.71] | 5.33 [5.30, 5.37] |
| | | Lin. clip+norm | 98.12 [98.09, 98.15] | [-2.45, 2.45] | [-7.22, 7.69] | 3.46 [3.44, 3.47] | 1.56 [1.54, 1.58] | 5.92 [5.88, 5.96] |
| | | Lin. simplex | 97.99 [97.96, 98.02] | [-2.53, 2.53] | [-7.48, 7.49] | 3.63 [3.62, 3.65] | 1.67 [1.64, 1.69] | 8.60 [8.52, 8.68] |
| | | Vec. scaling | 97.80 [97.76, 97.83] | [-2.65, 2.65] | [-7.20, 7.86] | 3.71 [3.70, 3.73] | 1.83 [1.80, 1.86] | 5.20 [5.17, 5.23] |
| | NNLS | None | 97.86 [97.83, 97.89] | [-2.61, 2.61] | [-8.69, 7.30] | 3.80 [3.78, 3.81] | 1.78 [1.75, 1.80] | 8.24 [8.16, 8.31] |
| | | Lin. clip+norm | 98.36 [98.34, 98.38] | [-2.29, 2.29] | [-6.01, 6.33] | 3.32 [3.31, 3.34] | 1.36 [1.35, 1.38] | 6.84 [6.76, 6.91] |
| | | Lin. simplex | 98.68 [98.67, 98.70] | [-2.05, 2.05] | [-4.95, 5.60] | 2.94 [2.93, 2.95] | 1.09 [1.08, 1.10] | 6.40 [6.32, 6.48] |
| | | Vec. scaling | 97.89 [97.87, 97.91] | [-2.59, 2.59] | [-5.20, 5.78] | 4.03 [4.01, 4.04] | 1.75 [1.74, 1.76] | 5.99 [5.93, 6.06] |
| | PSLS | None | 97.67 [97.63, 97.72] | [-2.72, 2.72] | [-11.48, 8.75] | 3.44 [3.42, 3.46] | 1.93 [1.90, 1.97] | 9.00 [8.89, 9.12] |
| | | Lin. clip+norm | 98.38 [98.35, 98.40] | [-2.28, 2.28] | [-7.40, 6.80] | 3.18 [3.17, 3.20] | 1.35 [1.33, 1.37] | 6.50 [6.44, 6.56] |
| | | Lin. simplex | 98.70 [98.68, 98.71] | [-2.04, 2.04] | [-6.23, 5.74] | 2.94 [2.93, 2.95] | 1.08 [1.07, 1.09] | 6.60 [6.53, 6.68] |
| | | Vec. scaling | 97.70 [97.67, 97.73] | [-2.71, 2.71] | [-8.64, 8.92] | 3.91 [3.90, 3.93] | 1.91 [1.88, 1.94] | 6.59 [6.52, 6.67] |
| *MethylBERT* [20] | XGB | None | 95.10 [95.03, 95.17] | [-3.95, 3.95] | [-9.21, 9.80] | 4.98 [4.96, 5.01] | 4.07 [4.00, 4.13] | 10.16 [10.09, 10.22] |
| | | Lin. clip+norm | 95.95 [95.88, 96.01] | [-3.60, 3.60] | [-9.22, 9.51] | 4.38 [4.36, 4.40] | 3.37 [3.31, 3.42] | 9.87 [9.80, 9.94] |
| | | Lin. simplex | 96.51 [96.45, 96.56] | [-3.34, 3.34] | [-9.20, 9.22] | 3.96 [3.94, 3.98] | 2.90 [2.85, 2.95] | 11.33 [11.23, 11.43] |
| | | Vec. scaling | 96.51 [96.47, 96.55] | [-3.34, 3.34] | [-7.76, 8.28] | 4.78 [4.76, 4.79] | 2.90 [2.86, 2.93] | 7.83 [7.78, 7.87] |
| | MLP | None | 96.69 [96.64, 96.74] | [-3.25, 3.25] | [-6.49, 9.54] | 4.29 [4.27, 4.31] | 2.75 [2.71, 2.79] | 9.47 [9.39, 9.55] |
| | | Lin. clip+norm | 97.00 [96.94, 97.05] | [-3.10, 3.10] | [-7.36, 7.90] | 4.21 [4.19, 4.22] | 2.49 [2.45, 2.54] | 10.71 [10.61, 10.80] |
| | | Lin. simplex | 97.00 [96.94, 97.05] | [-3.09, 3.09] | [-7.50, 8.18] | 4.08 [4.06, 4.10] | 2.49 [2.45, 2.54] | 13.08 [12.93, 13.24] |
| | | Vec. scaling | 96.99 [96.93, 97.04] | [-3.10, 3.10] | [-7.55, 7.53] | 4.24 [4.22, 4.26] | 2.50 [2.45, 2.55] | 8.39 [8.32, 8.46] |
| | SWN | None | 97.55 [97.52, 97.58] | [-2.79, 2.79] | [-7.11, 9.35] | 3.75 [3.73, 3.76] | 2.03 [2.01, 2.06] | 7.17 [7.12, 7.22] |
| | | Lin. clip+norm | 97.83 [97.80, 97.86] | [-2.63, 2.63] | [-7.27, 7.86] | 3.67 [3.66, 3.69] | 1.80 [1.78, 1.82] | 8.32 [8.25, 8.39] |
| | | Lin. simplex | 97.79 [97.76, 97.82] | [-2.66, 2.66] | [-7.70, 8.01] | 3.63 [3.61, 3.64] | 1.84 [1.81, 1.86] | 10.61 [10.50, 10.71] |
| | | Vec. scaling | 97.95 [97.93, 97.97] | [-2.56, 2.56] | [-6.77, 6.94] | 3.77 [3.76, 3.79] | 1.70 [1.69, 1.72] | 6.34 [6.29, 6.38] |
| | NNLS | None | 97.81 [97.78, 97.84] | [-2.64, 2.64] | [-7.40, 7.88] | 3.77 [3.75, 3.78] | 1.82 [1.80, 1.84] | 9.95 [9.85, 10.05] |
| | | Lin. clip+norm | 98.14 [98.12, 98.17] | [-2.43, 2.43] | [-6.89, 7.46] | 3.55 [3.54, 3.57] | 1.54 [1.52, 1.56] | 8.44 [8.37, 8.51] |
| | | Lin. simplex | 98.47 [98.46, 98.49] | [-2.21, 2.21] | [-6.37, 6.85] | 3.12 [3.11, 3.13] | 1.27 [1.25, 1.28] | 8.36 [8.26, 8.45] |
| | | Vec. scaling | 98.08 [98.06, 98.09] | [-2.48, 2.48] | [-5.63, 5.61] | 3.89 [3.87, 3.90] | 1.60 [1.59, 1.61] | 7.93 [7.84, 8.02] |
| | PSLS | None | 98.04 [98.01, 98.06] | [-2.50, 2.50] | [-6.75, 7.36] | 3.58 [3.57, 3.60] | 1.63 [1.61, 1.65] | 8.73 [8.66, 8.81] |
| | | Lin. clip+norm | 98.29 [98.27, 98.31] | [-2.34, 2.34] | [-6.25, 6.98] | 3.51 [3.50, 3.53] | 1.42 [1.40, 1.44] | 8.19 [8.12, 8.26] |
| | | Lin. simplex | 98.54 [98.53, 98.56] | [-2.16, 2.16] | [-5.80, 6.50] | 3.14 [3.13, 3.15] | 1.21 [1.20, 1.22] | 8.41 [8.32, 8.50] |
| | | Vec. scaling | 98.00 [97.98, 98.02] | [-2.53, 2.53] | [-6.62, 6.77] | 3.90 [3.89, 3.91] | 1.66 [1.65, 1.68] | 7.43 [7.36, 7.50] |
| *Lookup Classifier* (1-NN) | XGB | None | 82.72 [82.60, 82.84] | [-7.42, 7.42] | [-16.41, 13.17] | 12.13 [12.10, 12.17] | 14.34 [14.21, 14.48] | 38.60 [38.37, 38.83] |
| | | Lin. clip+norm | 86.76 [86.66, 86.86] | [-6.50, 6.50] | [-15.20, 12.19] | 10.04 [10.00, 10.07] | 10.99 [10.88, 11.09] | 44.44 [44.07, 44.81] |
| | | Lin. simplex | 88.46 [88.36, 88.56] | [-6.07, 6.07] | [-13.52, 13.91] | 8.86 [8.82, 8.89] | 9.58 [9.49, 9.67] | 54.70 [54.19, 55.21] |
| | | Vec. scaling | 89.25 [89.16, 89.34] | [-5.85, 5.85] | [-12.37, 13.45] | 9.45 [9.42, 9.48] | 8.92 [8.85, 8.99] | 31.14 [30.91, 31.37] |
| | MLP | None | 76.44 [76.15, 76.72] | [-8.67, 8.67] | [-18.02, 26.84] | 12.47 [12.43, 12.52] | 19.55 [19.30, 19.81] | 48.05 [47.68, 48.43] |
| | | Lin. clip+norm | 82.56 [82.33, 82.80] | [-7.46, 7.46] | [-15.75, 16.82] | 10.86 [10.82, 10.90] | 14.47 [14.27, 14.68] | 49.10 [48.69, 49.52] |
| | | Lin. simplex | 82.66 [82.43, 82.89] | [-7.44, 7.44] | [-13.60, 19.09] | 10.44 [10.39, 10.48] | 14.39 [14.19, 14.60] | 57.94 [57.44, 58.45] |
| | | Vec. scaling | 82.27 [82.03, 82.50] | [-7.52, 7.52] | [-15.68, 16.41] | 10.96 [10.92, 11.00] | 14.72 [14.52, 14.93] | 37.43 [37.16, 37.68] |
| | SWN | None | 86.23 [86.11, 86.36] | [-6.63, 6.63] | [-15.91, 24.34] | 10.30 [10.27, 10.33] | 11.43 [11.32, 11.54] | 31.40 [31.22, 31.58] |
| | | Lin. clip+norm | 91.21 [91.12, 91.30] | [-5.29, 5.29] | [-13.65, 14.86] | 8.05 [8.02, 8.08] | 7.29 [7.22, 7.37] | 32.69 [32.42, 32.95] |
| | | Lin. simplex | 91.56 [91.46, 91.65] | [-5.19, 5.19] | [-15.33, 15.56] | 7.89 [7.86, 7.92] | 7.01 [6.93, 7.09] | 41.22 [40.85, 41.60] |
| | | Vec. scaling | 90.91 [90.80, 91.01] | [-5.38, 5.38] | [-14.36, 15.18] | 8.50 [8.47, 8.53] | 7.55 [7.46, 7.63] | 24.32 [24.19, 24.45] |
| | NNLS | None | 83.52 [83.39, 83.65] | [-7.25, 7.25] | [-16.89, 23.27] | 12.95 [12.92, 12.99] | 13.68 [13.56, 13.80] | 51.44 [51.06, 51.83] |
| | | Lin. clip+norm | 89.24 [89.13, 89.34] | [-5.86, 5.86] | [-15.50, 16.15] | 9.98 [9.95, 10.01] | 8.93 [8.84, 9.03] | 41.20 [40.89, 41.53] |
| | | Lin. simplex | 91.38 [91.27, 91.49] | [-5.24, 5.24] | [-16.56, 15.93] | 7.92 [7.89, 7.95] | 7.15 [7.06, 7.24] | 41.74 [41.37, 42.12] |
| | | Vec. scaling | 90.97 [90.89, 91.05] | [-5.37, 5.37] | [-14.21, 15.27] | 8.92 [8.89, 8.94] | 7.50 [7.44, 7.55] | 39.28 [38.91, 39.66] |
| | PSLS | None | 83.55 [83.42, 83.68] | [-7.24, 7.24] | [-16.82, 23.23] | 12.92 [12.89, 12.95] | 13.66 [13.54, 13.78] | 49.96 [49.60, 50.33] |
| | | Lin. clip+norm | 89.30 [89.19, 89.40] | [-5.84, 5.84] | [-15.45, 16.11] | 9.93 [9.90, 9.96] | 8.88 [8.79, 8.98] | 40.97 [40.65, 41.30] |
| | | Lin. simplex | 91.40 [91.30, 91.51] | [-5.24, 5.24] | [-16.51, 15.92] | 7.90 [7.87, 7.93] | 7.14 [7.05, 7.23] | 41.57 [41.20, 41.95] |
| | | Vec. scaling | 87.84 [87.68, 87.99] | [-6.23, 6.23] | [-15.98, 16.61] | 10.06 [10.03, 10.10] | 10.09 [9.97, 10.22] | 36.75 [36.46, 37.06] |

Table S13: Full results for **hard labels with background class and Diag. + Bckg. features for deconvolution**. For $R^2$, MAE, MSE and KL, we indicated first the value of the metric computed on the whole test set, and then the 95% confidence interval computed with BCa bootstrap [47] with 10,000 resamples.

| Classifier | Deconvolver | Dec. Calib. | $\mathbf{R^2}$ ($\times 10^{-2}$) | LoA ($\times 10^{-2}$) | LoA worst-class ($\times 10^{-2}$) | MAE ($\times 10^{-3}$) | MSE ($\times 10^{-4}$) | KL ($\times 10^{-2}$) |
|---|---|---|---|---|---|---|---|---|
| *Dismir* [19] (CNN+LSTM) | XGB | None | 96.95 [96.90, 96.99] | [-3.12, 3.12] | [-7.48, 8.33] | 4.00 [3.98, 4.02] | 2.54 [2.49, 2.58] | 6.90 [6.85, 6.94] |
| | | Lin. clip+norm | 97.37 [97.32, 97.41] | [-2.90, 2.90] | [-7.21, 8.18] | 3.61 [3.59, 3.63] | 2.18 [2.14, 2.22] | 6.50 [6.45, 6.55] |
| | | Lin. simplex | 97.76 [97.71, 97.80] | [-2.67, 2.67] | [-7.04, 7.90] | 3.24 [3.23, 3.26] | 1.86 [1.83, 1.90] | 6.65 [6.59, 6.72] |
| | | Vec. scaling | 97.52 [97.49, 97.55] | [-2.81, 2.81] | [-5.81, 7.22] | 4.07 [4.05, 4.08] | 2.06 [2.04, 2.08] | 5.19 [5.16, 5.23] |
| | MLP | None | 96.22 [96.14, 96.29] | [-3.47, 3.47] | [-13.33, 11.41] | 4.40 [4.38, 4.42] | 3.14 [3.08, 3.20] | 8.54 [8.45, 8.62] |
| | | Lin. clip+norm | 96.64 [96.58, 96.69] | [-3.28, 3.28] | [-11.56, 11.08] | 4.37 [4.34, 4.39] | 2.79 [2.74, 2.84] | 9.55 [9.47, 9.63] |
| | | Lin. simplex | 96.66 [96.61, 96.72] | [-3.26, 3.26] | [-11.23, 10.78] | 4.39 [4.37, 4.41] | 2.77 [2.72, 2.81] | 12.25 [12.13, 12.37] |
| | | Vec. scaling | 96.65 [96.59, 96.70] | [-3.27, 3.27] | [-11.06, 11.17] | 4.45 [4.43, 4.47] | 2.78 [2.74, 2.83] | 7.73 [7.66, 7.80] |
| | SWN | None | 94.81 [94.71, 94.90] | [-4.07, 4.07] | [-11.07, 15.07] | 4.55 [4.52, 4.58] | 4.31 [4.24, 4.39] | 8.31 [8.22, 8.40] |
| | | Lin. clip+norm | 94.58 [94.49, 94.68] | [-4.16, 4.16] | [-12.03, 16.33] | 4.76 [4.73, 4.79] | 4.50 [4.42, 4.57] | 9.22 [9.12, 9.32] |
| | | Lin. simplex | 94.00 [93.90, 94.11] | [-4.37, 4.37] | [-13.55, 18.03] | 4.96 [4.94, 4.99] | 4.98 [4.90, 5.07] | 15.09 [14.81, 15.38] |
| | | Vec. scaling | 93.10 [92.99, 93.22] | [-4.69, 4.69] | [-14.99, 20.06] | 5.19 [5.16, 5.22] | 5.73 [5.63, 5.82] | 9.06 [8.97, 9.16] |
| | NNLS | None | 96.59 [96.51, 96.66] | [-3.30, 3.30] | [-16.60, 12.25] | 3.81 [3.78, 3.83] | 2.83 [2.77, 2.90] | 25.00 [24.57, 25.43] |
| | | Lin. clip+norm | 97.61 [97.57, 97.65] | [-2.76, 2.76] | [-11.89, 10.95] | 3.70 [3.68, 3.72] | 1.98 [1.95, 2.02] | 9.34 [9.25, 9.44] |
| | | Lin. simplex | 97.98 [97.95, 98.01] | [-2.54, 2.54] | [-11.02, 9.78] | 3.39 [3.38, 3.41] | 1.68 [1.65, 1.70] | 8.67 [8.58, 8.76] |
| | | Vec. scaling | 97.19 [97.15, 97.22] | [-2.99, 2.99] | [-11.71, 9.65] | 4.15 [4.13, 4.17] | 2.33 [2.31, 2.36] | 22.38 [21.95, 22.80] |
| | PSLS | None | 96.47 [96.39, 96.55] | [-3.35, 3.35] | [-16.83, 12.49] | 3.87 [3.85, 3.90] | 2.93 [2.86, 3.00] | 23.12 [22.73, 23.54] |
| | | Lin. clip+norm | 97.44 [97.40, 97.49] | [-2.85, 2.85] | [-12.46, 11.46] | 3.78 [3.76, 3.80] | 2.12 [2.08, 2.16] | 9.76 [9.66, 9.86] |
| | | Lin. simplex | 97.81 [97.77, 97.84] | [-2.64, 2.64] | [-11.69, 10.38] | 3.48 [3.46, 3.50] | 1.82 [1.79, 1.85] | 9.10 [9.00, 9.20] |
| | | Vec. scaling | 96.79 [96.75, 96.84] | [-3.20, 3.20] | [-12.64, 12.46] | 4.48 [4.46, 4.49] | 2.66 [2.62, 2.70] | 12.26 [12.04, 12.49] |
| *MethylBERT* [20] | XGB | None | 95.26 [95.19, 95.34] | [-3.89, 3.89] | [-8.46, 9.33] | 4.88 [4.85, 4.90] | 3.93 [3.86, 4.00] | 10.09 [10.01, 10.16] |
| | | Lin. clip+norm | 96.05 [95.98, 96.11] | [-3.55, 3.55] | [-8.45, 8.82] | 4.24 [4.22, 4.27] | 3.28 [3.22, 3.34] | 9.87 [9.79, 9.95] |
| | | Lin. simplex | 96.70 [96.64, 96.76] | [-3.24, 3.24] | [-8.33, 8.54] | 3.78 [3.76, 3.80] | 2.74 [2.68, 2.79] | 11.48 [11.38, 11.58] |
| | | Vec. scaling | 97.13 [97.09, 97.16] | [-3.03, 3.03] | [-7.03, 7.29] | 4.40 [4.38, 4.41] | 2.38 [2.35, 2.42] | 7.15 [7.11, 7.20] |
| | MLP | None | 96.86 [96.82, 96.89] | [-3.17, 3.17] | [-7.65, 11.54] | 4.40 [4.38, 4.42] | 2.61 [2.58, 2.63] | 9.53 [9.45, 9.60] |
| | | Lin. clip+norm | 97.43 [97.41, 97.46] | [-2.86, 2.86] | [-7.45, 8.68] | 4.31 [4.30, 4.33] | 2.13 [2.11, 2.15] | 10.85 [10.75, 10.94] |
| | | Lin. simplex | 97.36 [97.34, 97.39] | [-2.90, 2.90] | [-8.40, 9.23] | 4.29 [4.28, 4.31] | 2.19 [2.17, 2.21] | 13.73 [13.60, 13.85] |
| | | Vec. scaling | 97.39 [97.37, 97.42] | [-2.88, 2.88] | [-7.75, 8.10] | 4.32 [4.30, 4.34] | 2.16 [2.15, 2.18] | 8.46 [8.40, 8.52] |
| | SWN | None | 97.44 [97.41, 97.48] | [-2.86, 2.86] | [-7.75, 10.17] | 3.72 [3.70, 3.73] | 2.12 [2.09, 2.15] | 7.26 [7.21, 7.32] |
| | | Lin. clip+norm | 97.79 [97.76, 97.82] | [-2.65, 2.65] | [-8.01, 8.64] | 3.58 [3.56, 3.59] | 1.83 [1.81, 1.86] | 8.43 [8.36, 8.51] |
| | | Lin. simplex | 97.80 [97.77, 97.83] | [-2.65, 2.65] | [-8.51, 8.84] | 3.49 [3.48, 3.51] | 1.83 [1.80, 1.86] | 10.59 [10.48, 10.69] |
| | | Vec. scaling | 97.92 [97.90, 97.95] | [-2.57, 2.57] | [-7.52, 7.70] | 3.73 [3.72, 3.74] | 1.72 [1.70, 1.74] | 6.32 [6.28, 6.36] |
| | NNLS | None | 98.06 [98.03, 98.08] | [-2.49, 2.49] | [-6.69, 7.33] | 3.53 [3.52, 3.55] | 1.61 [1.59, 1.63] | 9.14 [9.05, 9.22] |
| | | Lin. clip+norm | 98.32 [98.30, 98.34] | [-2.31, 2.31] | [-6.20, 6.95] | 3.45 [3.44, 3.47] | 1.39 [1.37, 1.41] | 8.03 [7.96, 8.10] |
| | | Lin. simplex | 98.57 [98.56, 98.59] | [-2.13, 2.13] | [-5.76, 6.47] | 3.09 [3.08, 3.10] | 1.18 [1.17, 1.20] | 8.24 [8.15, 8.33] |
| | | Vec. scaling | 98.14 [98.12, 98.15] | [-2.44, 2.44] | [-5.33, 5.38] | 3.77 [3.76, 3.79] | 1.55 [1.54, 1.56] | 7.58 [7.50, 7.66] |
| | PSLS | None | 98.04 [98.01, 98.06] | [-2.50, 2.50] | [-6.75, 7.37] | 3.59 [3.57, 3.60] | 1.63 [1.61, 1.65] | 8.71 [8.63, 8.78] |
| | | Lin. clip+norm | 98.29 [98.27, 98.31] | [-2.34, 2.34] | [-6.25, 6.99] | 3.51 [3.50, 3.53] | 1.42 [1.40, 1.44] | 8.19 [8.12, 8.26] |
| | | Lin. simplex | 98.54 [98.52, 98.56] | [-2.16, 2.16] | [-5.81, 6.51] | 3.14 [3.13, 3.15] | 1.21 [1.20, 1.22] | 8.42 [8.32, 8.51] |
| | | Vec. scaling | 98.04 [98.02, 98.06] | [-2.50, 2.50] | [-6.35, 6.49] | 3.88 [3.87, 3.89] | 1.63 [1.61, 1.64] | 7.40 [7.33, 7.47] |
| *Lookup Classifier* (1-NN) | XGB | None | 82.72 [82.60, 82.84] | [-7.42, 7.42] | [-16.41, 13.17] | 12.13 [12.10, 12.17] | 14.34 [14.21, 14.48] | 38.60 [38.37, 38.83] |
| | | Lin. clip+norm | 86.76 [86.66, 86.86] | [-6.50, 6.50] | [-15.20, 12.19] | 10.04 [10.00, 10.07] | 10.99 [10.88, 11.09] | 44.44 [44.07, 44.81] |
| | | Lin. simplex | 88.46 [88.36, 88.56] | [-6.07, 6.07] | [-13.52, 13.91] | 8.86 [8.82, 8.89] | 9.58 [9.49, 9.67] | 54.70 [54.19, 55.21] |
| | | Vec. scaling | 89.25 [89.16, 89.34] | [-5.85, 5.85] | [-12.37, 13.45] | 9.45 [9.42, 9.48] | 8.92 [8.85, 8.99] | 31.14 [30.91, 31.37] |
| | MLP | None | 73.09 [72.80, 73.40] | [-9.26, 9.26] | [-19.10, 29.50] | 13.61 [13.57, 13.66] | 22.33 [22.05, 22.61] | 53.45 [53.04, 53.85] |
| | | Lin. clip+norm | 81.08 [80.80, 81.37] | [-7.77, 7.77] | [-15.22, 22.00] | 11.18 [11.14, 11.23] | 15.70 [15.45, 15.95] | 55.69 [55.22, 56.16] |
| | | Lin. simplex | 81.03 [80.75, 81.32] | [-7.78, 7.78] | [-15.74, 21.97] | 10.85 [10.80, 10.89] | 15.74 [15.49, 15.98] | 69.22 [68.61, 69.86] |
| | | Vec. scaling | 80.65 [80.35, 80.94] | [-7.86, 7.86] | [-15.88, 20.40] | 11.36 [11.31, 11.40] | 16.06 [15.80, 16.32] | 41.45 [41.12, 41.77] |
| | SWN | None | 86.43 [86.31, 86.56] | [-6.58, 6.58] | [-16.04, 24.17] | 10.29 [10.25, 10.32] | 11.26 [11.16, 11.37] | 31.04 [30.86, 31.21] |
| | | Lin. clip+norm | 91.19 [91.10, 91.28] | [-5.30, 5.30] | [-13.66, 15.18] | 8.12 [8.09, 8.14] | 7.31 [7.24, 7.39] | 31.84 [31.59, 32.10] |
| | | Lin. simplex | 91.52 [91.43, 91.62] | [-5.20, 5.20] | [-15.17, 15.76] | 8.02 [7.99, 8.05] | 7.04 [6.96, 7.11] | 39.58 [39.23, 39.94] |
| | | Vec. scaling | 90.86 [90.76, 90.97] | [-5.40, 5.40] | [-14.28, 15.19] | 8.56 [8.53, 8.59] | 7.58 [7.50, 7.67] | 24.28 [24.15, 24.40] |
| | NNLS | None | 83.52 [83.39, 83.65] | [-7.25, 7.25] | [-16.89, 23.27] | 12.95 [12.92, 12.99] | 13.68 [13.56, 13.80] | 51.44 [51.06, 51.83] |
| | | Lin. clip+norm | 89.24 [89.13, 89.34] | [-5.86, 5.86] | [-15.50, 16.15] | 9.98 [9.95, 10.01] | 8.93 [8.84, 9.03] | 41.20 [40.89, 41.53] |
| | | Lin. simplex | 91.38 [91.27, 91.49] | [-5.24, 5.24] | [-16.56, 15.93] | 7.92 [7.89, 7.95] | 7.15 [7.06, 7.24] | 41.74 [41.37, 42.12] |
| | | Vec. scaling | 90.97 [90.89, 91.05] | [-5.37, 5.37] | [-14.21, 15.27] | 8.92 [8.89, 8.94] | 7.50 [7.44, 7.55] | 39.28 [38.91, 39.66] |
| | PSLS | None | 83.55 [83.42, 83.68] | [-7.24, 7.24] | [-16.82, 23.23] | 12.92 [12.89, 12.95] | 13.66 [13.54, 13.78] | 50.49 [50.12, 50.87] |
| | | Lin. clip+norm | 89.30 [89.19, 89.40] | [-5.84, 5.84] | [-15.45, 16.11] | 9.93 [9.90, 9.96] | 8.88 [8.79, 8.98] | 40.97 [40.65, 41.30] |
| | | Lin. simplex | 91.40 [91.30, 91.51] | [-5.24, 5.24] | [-16.51, 15.92] | 7.90 [7.87, 7.93] | 7.14 [7.05, 7.23] | 41.57 [41.20, 41.95] |
| | | Vec. scaling | 88.02 [87.87, 88.17] | [-6.18, 6.18] | [-16.02, 16.56] | 9.99 [9.96, 10.02] | 9.94 [9.82, 10.07] | 36.80 [36.51, 37.11] |

Table S14: Full results for **Canonical Soft Labels** ($\epsilon = 0.1$) **and Top 156 features for deconvolution**. For $R^2$, MAE, MSE and KL, we indicated first the value of the metric computed on the whole test set, and then the 95% confidence interval computed with BCa bootstrap [47] with 10,000 resamples.

| Classifier | Deconvolver | Dec. Calib. | $\mathbf{R^2}$ ($\times 10^{-2}$) | LoA ($\times 10^{-2}$) | LoA worst-class ($\times 10^{-2}$) | MAE ($\times 10^{-3}$) | MSE ($\times 10^{-4}$) | KL ($\times 10^{-2}$) |
|---|---|---|---|---|---|---|---|---|
| *Dismir* [19] (CNN+LSTM) | XGB | None | 97.15 [97.10, 97.19] | [-3.02, 3.02] | [-7.43, 8.27] | 3.89 [3.87, 3.91] | 2.37 [2.33, 2.41] | 6.68 [6.64, 6.73] |
| | | Lin. clip+norm | 97.57 [97.53, 97.61] | [-2.78, 2.78] | [-7.10, 8.07] | 3.48 [3.47, 3.50] | 2.01 [1.98, 2.05] | 6.25 [6.20, 6.30] |
| | | Lin. simplex | 97.93 [97.90, 97.97] | [-2.57, 2.57] | [-6.89, 7.79] | 3.13 [3.12, 3.15] | 1.71 [1.68, 1.75] | 6.41 [6.35, 6.47] |
| | | Vec. scaling | 97.68 [97.65, 97.70] | [-2.72, 2.72] | [-5.79, 7.09] | 3.94 [3.92, 3.95] | 1.93 [1.91, 1.95] | 5.01 [4.98, 5.04] |
| | MLP | None | 97.49 [97.45, 97.54] | [-2.83, 2.83] | [-9.23, 8.10] | 3.65 [3.63, 3.67] | 2.08 [2.05, 2.12] | 6.18 [6.12, 6.24] |
| | | Lin. clip+norm | 97.80 [97.76, 97.83] | [-2.65, 2.65] | [-7.37, 7.55] | 3.57 [3.56, 3.59] | 1.83 [1.80, 1.86] | 7.27 [7.20, 7.35] |
| | | Lin. simplex | 97.95 [97.92, 97.98] | [-2.56, 2.56] | [-6.68, 6.97] | 3.49 [3.47, 3.50] | 1.70 [1.68, 1.73] | 8.58 [8.49, 8.67] |
| | | Vec. scaling | 97.74 [97.70, 97.77] | [-2.69, 2.69] | [-7.12, 7.63] | 3.69 [3.68, 3.71] | 1.88 [1.85, 1.90] | 5.68 [5.63, 5.73] |
| | SWN | None | 96.52 [96.47, 96.58] | [-3.33, 3.33] | [-10.28, 9.34] | 4.23 [4.21, 4.25] | 2.88 [2.84, 2.93] | 6.98 [6.92, 7.03] |
| | | Lin. clip+norm | 96.58 [96.53, 96.63] | [-3.30, 3.30] | [-8.31, 11.03] | 4.36 [4.34, 4.38] | 2.84 [2.80, 2.88] | 7.80 [7.73, 7.87] |
| | | Lin. simplex | 96.40 [96.35, 96.46] | [-3.39, 3.39] | [-8.83, 11.46] | 4.46 [4.44, 4.48] | 2.99 [2.94, 3.03] | 10.01 [9.92, 10.11] |
| | | Vec. scaling | 95.79 [95.73, 95.85] | [-3.66, 3.66] | [-10.07, 13.15] | 4.63 [4.61, 4.65] | 3.50 [3.45, 3.54] | 7.13 [7.08, 7.19] |
| | NNLS | None | 96.11 [96.01, 96.22] | [-3.52, 3.52] | [-18.72, 13.85] | 4.08 [4.05, 4.11] | 3.22 [3.14, 3.31] | 33.10 [32.55, 33.67] |
| | | Lin. clip+norm | 97.23 [97.17, 97.28] | [-2.97, 2.97] | [-13.73, 12.91] | 3.98 [3.96, 4.00] | 2.30 [2.26, 2.35] | 9.57 [9.48, 9.67] |
| | | Lin. simplex | 97.61 [97.57, 97.65] | [-2.76, 2.76] | [-12.96, 11.35] | 3.55 [3.53, 3.57] | 1.98 [1.95, 2.02] | 8.83 [8.74, 8.93] |
| | | Vec. scaling | 96.61 [96.56, 96.66] | [-3.29, 3.29] | [-13.29, 9.96] | 4.48 [4.46, 4.50] | 2.81 [2.78, 2.85] | 29.75 [29.22, 30.31] |
| | PSLS | None | 95.92 [95.81, 96.03] | [-3.61, 3.61] | [-18.83, 14.12] | 4.00 [3.97, 4.02] | 3.38 [3.29, 3.48] | 30.42 [29.89, 30.99] |
| | | Lin. clip+norm | 96.75 [96.68, 96.83] | [-3.22, 3.22] | [-15.42, 14.09] | 3.97 [3.95, 4.00] | 2.69 [2.63, 2.76] | 11.14 [11.02, 11.27] |
| | | Lin. simplex | 97.02 [96.94, 97.09] | [-3.08, 3.08] | [-15.22, 13.39] | 3.70 [3.68, 3.72] | 2.48 [2.42, 2.54] | 10.51 [10.39, 10.64] |
| | | Vec. scaling | 96.16 [96.09, 96.23] | [-3.50, 3.50] | [-15.25, 14.41] | 4.69 [4.67, 4.72] | 3.19 [3.13, 3.24] | 16.40 [16.08, 16.73] |
| *MethylBERT* [20] | XGB | None | 95.71 [95.64, 95.78] | [-3.70, 3.70] | [-8.10, 8.71] | 4.68 [4.66, 4.70] | 3.56 [3.50, 3.63] | 9.35 [9.28, 9.41] |
| | | Lin. clip+norm | 96.37 [96.31, 96.44] | [-3.40, 3.40] | [-7.62, 8.32] | 4.11 [4.09, 4.13] | 3.01 [2.95, 3.07] | 9.04 [8.97, 9.11] |
| | | Lin. simplex | 97.04 [96.98, 97.10] | [-3.07, 3.07] | [-7.31, 7.81] | 3.62 [3.60, 3.64] | 2.46 [2.41, 2.51] | 10.08 [9.99, 10.17] |
| | | Vec. scaling | 97.26 [97.21, 97.30] | [-2.96, 2.96] | [-6.24, 7.06] | 4.24 [4.22, 4.25] | 2.27 [2.24, 2.31] | 6.60 [6.56, 6.65] |
| | MLP | None | 97.59 [97.56, 97.62] | [-2.77, 2.77] | [-5.35, 7.50] | 3.91 [3.90, 3.93] | 2.00 [1.98, 2.02] | 7.65 [7.60, 7.70] |
| | | Lin. clip+norm | 97.85 [97.82, 97.87] | [-2.62, 2.62] | [-5.55, 6.30] | 3.84 [3.83, 3.85] | 1.79 [1.77, 1.81] | 9.06 [8.98, 9.13] |
| | | Lin. simplex | 97.92 [97.89, 97.94] | [-2.58, 2.58] | [-6.00, 6.54] | 3.70 [3.68, 3.71] | 1.73 [1.71, 1.75] | 11.06 [10.96, 11.16] |
| | | Vec. scaling | 97.91 [97.88, 97.93] | [-2.58, 2.58] | [-5.99, 5.78] | 3.77 [3.76, 3.78] | 1.74 [1.72, 1.76] | 6.81 [6.76, 6.85] |
| | SWN | None | 97.90 [97.88, 97.93] | [-2.59, 2.59] | [-6.06, 7.56] | 3.69 [3.68, 3.71] | 1.74 [1.72, 1.76] | 6.59 [6.55, 6.63] |
| | | Lin. clip+norm | 98.14 [98.12, 98.16] | [-2.44, 2.44] | [-6.41, 6.57] | 3.61 [3.60, 3.62] | 1.55 [1.53, 1.56] | 7.54 [7.48, 7.59] |
| | | Lin. simplex | 98.10 [98.08, 98.12] | [-2.46, 2.46] | [-6.95, 6.86] | 3.59 [3.58, 3.60] | 1.58 [1.56, 1.60] | 9.49 [9.41, 9.58] |
| | | Vec. scaling | 98.22 [98.20, 98.23] | [-2.39, 2.39] | [-5.99, 5.88] | 3.67 [3.66, 3.69] | 1.48 [1.47, 1.49] | 5.82 [5.78, 5.85] |
| | NNLS | None | 97.91 [97.89, 97.94] | [-2.58, 2.58] | [-7.71, 6.56] | 3.71 [3.69, 3.73] | 1.73 [1.71, 1.75] | 9.12 [9.05, 9.20] |
| | | Lin. clip+norm | 98.37 [98.35, 98.39] | [-2.28, 2.28] | [-5.84, 5.72] | 3.39 [3.38, 3.40] | 1.35 [1.33, 1.37] | 7.84 [7.77, 7.91] |
| | | Lin. simplex | 98.71 [98.70, 98.72] | [-2.03, 2.03] | [-4.91, 5.00] | 2.99 [2.98, 3.00] | 1.07 [1.06, 1.08] | 7.72 [7.63, 7.80] |
| | | Vec. scaling | 98.31 [98.29, 98.32] | [-2.32, 2.32] | [-4.78, 4.67] | 3.71 [3.70, 3.72] | 1.40 [1.39, 1.41] | 6.99 [6.92, 7.07] |
| | PSLS | None | 98.06 [98.04, 98.09] | [-2.48, 2.48] | [-7.40, 6.08] | 3.57 [3.55, 3.58] | 1.61 [1.59, 1.63] | 8.24 [8.17, 8.30] |
| | | Lin. clip+norm | 98.49 [98.47, 98.51] | [-2.19, 2.19] | [-5.17, 4.96] | 3.35 [3.34, 3.36] | 1.25 [1.24, 1.27] | 7.72 [7.65, 7.79] |
| | | Lin. simplex | 98.77 [98.76, 98.78] | [-1.98, 1.98] | [-4.26, 4.30] | 3.01 [3.00, 3.02] | 1.02 [1.01, 1.03] | 7.75 [7.66, 7.83] |
| | | Vec. scaling | 98.28 [98.27, 98.30] | [-2.34, 2.34] | [-5.36, 5.15] | 3.68 [3.67, 3.69] | 1.42 [1.41, 1.44] | 6.65 [6.59, 6.71] |

Table S15: Full results for **Canonical Soft Labels** ($\epsilon = 0.1$) **and Diag. + Bckg. features for deconvolution**. For $R^2$, MAE, MSE and KL, we indicated first the value of the metric computed on the whole test set, and then the 95% confidence interval computed with BCa bootstrap [47] with 10,000 resamples.

| Classifier | Deconvolver | Dec. Calib. | $\mathbf{R^2}$ ($\times 10^{-2}$) | LoA ($\times 10^{-2}$) | LoA worst-class ($\times 10^{-2}$) | MAE ($\times 10^{-3}$) | MSE ($\times 10^{-4}$) | KL ($\times 10^{-2}$) |
|---|---|---|---|---|---|---|---|---|
| *Dismir* [19] (CNN+LSTM) | *XGB* | None | 95.45 [95.38, 95.52] | [-3.81, 3.81] | [-10.15, 10.75] | 4.84 [4.82, 4.86] | 3.78 [3.72, 3.85] | 9.45 [9.39, 9.51] |
| | | Lin. clip+norm | 96.32 [96.26, 96.38] | [-3.43, 3.43] | [-9.96, 10.10] | 4.07 [4.05, 4.10] | 3.06 [3.01, 3.11] | 8.82 [8.76, 8.89] |
| | | Lin. simplex | 96.99 [96.94, 97.05] | [-3.10, 3.10] | [-10.04, 10.07] | 3.54 [3.52, 3.56] | 2.50 [2.45, 2.55] | 9.30 [9.19, 9.41] |
| | | Vec. scaling | 97.41 [97.37, 97.44] | [-2.88, 2.88] | [-8.63, 8.63] | 3.87 [3.85, 3.88] | 2.15 [2.12, 2.18] | 6.23 [6.18, 6.27] |
| | *MLP* | None | 95.53 [95.45, 95.61] | [-3.78, 3.78] | [-13.98, 12.74] | 4.99 [4.97, 5.02] | 3.71 [3.64, 3.78] | 10.30 [10.21, 10.39] |
| | | Lin. clip+norm | 96.03 [95.96, 96.10] | [-3.56, 3.56] | [-12.98, 12.73] | 4.78 [4.76, 4.80] | 3.30 [3.24, 3.36] | 11.71 [11.62, 11.81] |
| | | Lin. simplex | 96.33 [96.27, 96.40] | [-3.42, 3.42] | [-12.90, 12.39] | 4.62 [4.60, 4.64] | 3.05 [2.99, 3.10] | 14.18 [14.04, 14.31] |
| | | Vec. scaling | 96.39 [96.33, 96.45] | [-3.39, 3.39] | [-12.35, 12.41] | 4.60 [4.57, 4.62] | 3.00 [2.95, 3.05] | 9.32 [9.24, 9.39] |
| | *SWN* | None | 94.56 [94.38, 94.73] | [-4.17, 4.17] | [-15.27, 14.86] | 4.65 [4.62, 4.67] | 4.52 [4.38, 4.67] | 8.90 [8.81, 9.00] |
| | | Lin. clip+norm | 95.86 [95.72, 95.99] | [-3.64, 3.64] | [-14.43, 14.53] | 4.17 [4.15, 4.19] | 3.44 [3.33, 3.55] | 9.13 [9.05, 9.21] |
| | | Lin. simplex | 96.01 [95.89, 96.14] | [-3.57, 3.57] | [-14.57, 14.06] | 4.15 [4.13, 4.18] | 3.31 [3.21, 3.42] | 11.20 [11.09, 11.31] |
| | | Vec. scaling | 96.13 [95.98, 96.26] | [-3.52, 3.52] | [-13.50, 13.51] | 3.86 [3.83, 3.88] | 3.22 [3.10, 3.34] | 7.04 [6.97, 7.11] |
| | *NNLS* | None | 95.59 [95.49, 95.68] | [-3.75, 3.75] | [-18.15, 13.50] | 4.58 [4.56, 4.61] | 3.67 [3.59, 3.75] | 34.62 [34.06, 35.18] |
| | | Lin. clip+norm | 97.08 [97.02, 97.13] | [-3.06, 3.06] | [-13.59, 13.00] | 3.85 [3.83, 3.87] | 2.43 [2.39, 2.47] | 11.03 [10.93, 11.13] |
| | | Lin. simplex | 97.80 [97.76, 97.85] | [-2.65, 2.65] | [-13.01, 12.03] | 3.18 [3.16, 3.20] | 1.83 [1.79, 1.86] | 10.40 [10.29, 10.51] |
| | | Vec. scaling | 97.09 [97.04, 97.14] | [-3.05, 3.05] | [-14.34, 11.25] | 3.94 [3.92, 3.96] | 2.42 [2.38, 2.46] | 30.57 [30.02, 31.11] |
| | *PSLS* | None | 95.64 [95.54, 95.73] | [-3.73, 3.73] | [-17.80, 13.28] | 4.58 [4.55, 4.61] | 3.63 [3.55, 3.70] | 29.03 [28.57, 29.50] |
| | | Lin. clip+norm | 97.07 [97.02, 97.12] | [-3.06, 3.06] | [-13.44, 12.85] | 3.84 [3.82, 3.86] | 2.44 [2.39, 2.48] | 11.73 [11.61, 11.84] |
| | | Lin. simplex | 97.79 [97.75, 97.83] | [-2.66, 2.66] | [-12.89, 11.91] | 3.19 [3.18, 3.21] | 1.84 [1.80, 1.87] | 10.59 [10.47, 10.70] |
| | | Vec. scaling | 96.80 [96.74, 96.86] | [-3.20, 3.20] | [-13.74, 13.82] | 4.29 [4.27, 4.31] | 2.66 [2.61, 2.70] | 12.33 [12.13, 12.53] |
| *MethylBERT* [20] | XGB | None | 95.37 [95.29, 95.44] | [-3.84, 3.84] | [-8.81, 9.27] | 4.83 [4.80, 4.85] | 3.84 [3.78, 3.91] | 9.83 [9.76, 9.90] |
| | | Lin. clip+norm | 96.07 [96.00, 96.13] | [-3.54, 3.54] | [-8.38, 8.95] | 4.22 [4.20, 4.25] | 3.26 [3.20, 3.33] | 9.55 [9.47, 9.62] |
| | | Lin. simplex | 96.75 [96.69, 96.81] | [-3.22, 3.22] | [-8.07, 8.52] | 3.72 [3.70, 3.74] | 2.69 [2.64, 2.75] | 10.76 [10.66, 10.86] |
| | | Vec. scaling | 97.09 [97.04, 97.13] | [-3.05, 3.05] | [-7.01, 7.65] | 4.35 [4.33, 4.37] | 2.42 [2.38, 2.45] | 6.97 [6.93, 7.02] |
| | MLP | None | 97.30 [97.27, 97.33] | [-2.93, 2.93] | [-6.80, 9.84] | 4.11 [4.10, 4.13] | 2.24 [2.21, 2.27] | 8.48 [8.41, 8.54] |
| | | Lin. clip+norm | 97.78 [97.76, 97.80] | [-2.66, 2.66] | [-6.56, 7.32] | 4.05 [4.03, 4.06] | 1.84 [1.82, 1.86] | 9.50 [9.42, 9.58] |
| | | Lin. simplex | 97.72 [97.70, 97.75] | [-2.69, 2.69] | [-7.59, 7.92] | 4.07 [4.06, 4.09] | 1.89 [1.87, 1.91] | 12.16 [12.04, 12.27] |
| | | Vec. scaling | 97.81 [97.80, 97.83] | [-2.64, 2.64] | [-6.68, 6.74] | 4.04 [4.02, 4.05] | 1.81 [1.80, 1.83] | 7.35 [7.29, 7.40] |
| | SWN | None | 97.97 [97.94, 97.99] | [-2.54, 2.54] | [-6.52, 7.92] | 3.53 [3.51, 3.54] | 1.69 [1.66, 1.71] | 6.39 [6.35, 6.43] |
| | | Lin. clip+norm | 98.22 [98.19, 98.24] | [-2.38, 2.38] | [-6.83, 7.05] | 3.35 [3.34, 3.36] | 1.48 [1.46, 1.50] | 7.25 [7.20, 7.31] |
| | | Lin. simplex | 98.25 [98.23, 98.28] | [-2.36, 2.36] | [-7.20, 7.20] | 3.29 [3.28, 3.30] | 1.45 [1.43, 1.47] | 9.14 [9.05, 9.23] |
| | | Vec. scaling | 98.29 [98.27, 98.31] | [-2.33, 2.33] | [-6.43, 6.33] | 3.49 [3.48, 3.50] | 1.42 [1.40, 1.43] | 5.55 [5.51, 5.58] |
| | NNLS | None | 98.12 [98.10, 98.14] | [-2.45, 2.45] | [-6.37, 6.38] | 3.55 [3.53, 3.56] | 1.56 [1.54, 1.58] | 8.63 [8.56, 8.70] |
| | | Lin. clip+norm | 98.42 [98.40, 98.44] | [-2.24, 2.24] | [-5.57, 5.91] | 3.41 [3.39, 3.42] | 1.31 [1.29, 1.32] | 7.64 [7.57, 7.70] |
| | | Lin. simplex | 98.68 [98.67, 98.69] | [-2.05, 2.05] | [-5.02, 5.42] | 3.05 [3.04, 3.06] | 1.09 [1.08, 1.11] | 7.76 [7.68, 7.84] |
| | | Vec. scaling | 98.32 [98.30, 98.33] | [-2.32, 2.32] | [-4.70, 4.61] | 3.66 [3.65, 3.67] | 1.40 [1.39, 1.41] | 6.93 [6.86, 7.00] |
| | PSLS | None | 98.10 [98.08, 98.13] | [-2.46, 2.46] | [-6.40, 6.47] | 3.59 [3.58, 3.61] | 1.57 [1.56, 1.59] | 8.27 [8.21, 8.34] |
| | | Lin. clip+norm | 98.40 [98.38, 98.41] | [-2.26, 2.26] | [-5.65, 6.02] | 3.45 [3.44, 3.47] | 1.33 [1.32, 1.35] | 7.78 [7.72, 7.84] |
| | | Lin. simplex | 98.66 [98.65, 98.67] | [-2.07, 2.07] | [-5.11, 5.54] | 3.09 [3.08, 3.10] | 1.11 [1.10, 1.12] | 7.90 [7.82, 7.98] |
| | | Vec. scaling | 98.30 [98.28, 98.31] | [-2.33, 2.33] | [-5.18, 5.09] | 3.71 [3.70, 3.72] | 1.41 [1.40, 1.42] | 6.80 [6.74, 6.86] |

Table S16: Full results for **data-driven soft-labels (with pooling)**. For $R^2$, MAE, MSE and KL, we indicated first the value of the metric computed on the whole test set, and then the 95% confidence interval computed with BCa bootstrap [47] with 10,000 resamples.

| Classifier | Deconvolver | Dec. Calib. | $R^2$ ($\times 10^{-2}$) | LoA ($\times 10^{-2}$) | LoA worst-class ($\times 10^{-2}$) | MAE ($\times 10^{-3}$) | MSE ($\times 10^{-4}$) | KL ($\times 10^{-2}$) |
|---|---|---|---|---|---|---|---|---|
| *Dismir* [19] (CNN+LSTM) | XGB | None | 97.45 [97.40, 97.49] | [-2.85, 2.85] | [-7.32, 8.31] | 3.67 [3.65, 3.69] | 2.12 [2.08, 2.16] | 6.35 [6.31, 6.39] |
| | | Lin. clip+norm | 97.74 [97.70, 97.78] | [-2.68, 2.68] | [-7.16, 7.97] | 3.37 [3.36, 3.39] | 1.88 [1.84, 1.91] | 6.16 [6.11, 6.21] |
| | | Lin. simplex | 98.07 [98.03, 98.10] | [-2.48, 2.48] | [-6.96, 7.69] | 3.07 [3.05, 3.08] | 1.60 [1.57, 1.64] | 6.43 [6.37, 6.49] |
| | | Vec. scaling | 97.74 [97.70, 97.77] | [-2.69, 2.69] | [-6.09, 6.64] | 3.81 [3.80, 3.83] | 1.88 [1.85, 1.91] | 4.96 [4.93, 4.99] |
| | MLP | None | 97.39 [97.36, 97.42] | [-2.89, 2.89] | [-7.60, 11.85] | 3.78 [3.76, 3.79] | 2.17 [2.14, 2.19] | 6.93 [6.88, 6.97] |
| | | Lin. clip+norm | 97.85 [97.82, 97.87] | [-2.62, 2.62] | [-7.47, 10.23] | 3.55 [3.54, 3.57] | 1.79 [1.77, 1.81] | 7.70 [7.63, 7.76] |
| | | Lin. simplex | 98.00 [97.98, 98.02] | [-2.52, 2.52] | [-7.55, 10.14] | 3.43 [3.42, 3.45] | 1.66 [1.64, 1.68] | 8.84 [8.76, 8.92] |
| | | Vec. scaling | 98.23 [98.21, 98.24] | [-2.38, 2.38] | [-6.59, 7.77] | 3.45 [3.44, 3.47] | 1.47 [1.46, 1.48] | 5.66 [5.62, 5.70] |
| | SWN | None | 98.32 [98.29, 98.34] | [-2.32, 2.32] | [-6.05, 7.76] | 3.15 [3.14, 3.16] | 1.40 [1.38, 1.42] | 4.84 [4.81, 4.87] |
| | | Lin. clip+norm | 98.52 [98.50, 98.54] | [-2.17, 2.17] | [-5.97, 6.91] | 3.07 [3.06, 3.08] | 1.23 [1.21, 1.24] | 5.41 [5.37, 5.45] |
| | | Lin. simplex | 98.50 [98.48, 98.52] | [-2.19, 2.19] | [-6.06, 6.73] | 3.16 [3.14, 3.17] | 1.24 [1.23, 1.26] | 7.41 [7.34, 7.48] |
| | | Vec. scaling | 98.38 [98.36, 98.39] | [-2.27, 2.27] | [-5.82, 6.66] | 3.32 [3.30, 3.33] | 1.35 [1.33, 1.36] | 4.54 [4.51, 4.57] |
| | NNLS | None | 98.23 [98.20, 98.26] | [-2.38, 2.38] | [-8.37, 7.05] | 3.21 [3.20, 3.23] | 1.47 [1.45, 1.50] | 6.78 [6.72, 6.85] |
| | | Lin. clip+norm | 98.64 [98.62, 98.66] | [-2.08, 2.08] | [-5.72, 5.86] | 2.95 [2.94, 2.96] | 1.13 [1.11, 1.15] | 5.99 [5.93, 6.06] |
| | | Lin. simplex | 98.93 [98.92, 98.94] | [-1.85, 1.85] | [-4.57, 4.94] | 2.66 [2.65, 2.67] | 0.89 [0.88, 0.90] | 5.59 [5.53, 5.66] |
| | | Vec. scaling | 98.39 [98.38, 98.40] | [-2.27, 2.27] | [-4.71, 4.93] | 3.56 [3.55, 3.58] | 1.34 [1.33, 1.34] | 5.37 [5.32, 5.43] |
| | PSLS | None | 98.14 [98.11, 98.17] | [-2.44, 2.44] | [-8.85, 7.15] | 3.17 [3.15, 3.18] | 1.55 [1.52, 1.57] | 6.66 [6.59, 6.73] |
| | | Lin. clip+norm | 98.63 [98.61, 98.65] | [-2.09, 2.09] | [-5.79, 5.63] | 2.94 [2.92, 2.95] | 1.14 [1.12, 1.16] | 6.32 [6.25, 6.40] |
| | | Lin. simplex | 98.92 [98.90, 98.93] | [-1.86, 1.86] | [-4.62, 4.72] | 2.72 [2.71, 2.73] | 0.90 [0.89, 0.91] | 5.73 [5.67, 5.79] |
| | | Vec. scaling | 98.26 [98.24, 98.28] | [-2.36, 2.36] | [-6.29, 6.85] | 3.58 [3.57, 3.60] | 1.45 [1.43, 1.46] | 5.67 [5.62, 5.72] |
| *MethylBERT* [20] | XGB | None | 97.27 [97.23, 97.31] | [-2.95, 2.95] | [-7.33, 7.51] | 3.82 [3.80, 3.84] | 2.27 [2.23, 2.30] | 6.51 [6.47, 6.55] |
| | | Lin. clip+norm | 97.70 [97.66, 97.73] | [-2.71, 2.71] | [-6.78, 7.27] | 3.43 [3.42, 3.45] | 1.91 [1.88, 1.94] | 6.08 [6.03, 6.12] |
| | | Lin. simplex | 98.09 [98.06, 98.13] | [-2.46, 2.46] | [-6.46, 6.88] | 3.07 [3.06, 3.09] | 1.58 [1.55, 1.61] | 6.22 [6.16, 6.27] |
| | | Vec. scaling | 97.65 [97.62, 97.67] | [-2.74, 2.74] | [-5.74, 8.53] | 3.89 [3.88, 3.90] | 1.95 [1.93, 1.98] | 4.93 [4.90, 4.95] |
| | MLP | None | 98.22 [98.20, 98.24] | [-2.38, 2.38] | [-5.11, 6.48] | 3.28 [3.27, 3.29] | 1.48 [1.46, 1.49] | 5.25 [5.21, 5.28] |
| | | Lin. clip+norm | 98.44 [98.42, 98.46] | [-2.23, 2.23] | [-5.28, 5.76] | 3.18 [3.16, 3.19] | 1.30 [1.28, 1.31] | 6.69 [6.63, 6.75] |
| | | Lin. simplex | 98.52 [98.51, 98.54] | [-2.17, 2.17] | [-5.55, 5.85] | 3.14 [3.13, 3.15] | 1.23 [1.21, 1.24] | 8.71 [8.63, 8.80] |
| | | Vec. scaling | 98.46 [98.45, 98.48] | [-2.21, 2.21] | [-5.24, 5.05] | 3.27 [3.26, 3.28] | 1.28 [1.27, 1.29] | 4.84 [4.81, 4.87] |
| | SWN | None | 98.49 [98.48, 98.51] | [-2.19, 2.19] | [-6.36, 4.69] | 3.08 [3.07, 3.09] | 1.25 [1.23, 1.26] | 4.43 [4.41, 4.46] |
| | | Lin. clip+norm | 98.59 [98.58, 98.61] | [-2.12, 2.12] | [-5.35, 4.98] | 3.12 [3.11, 3.13] | 1.17 [1.15, 1.18] | 5.10 [5.06, 5.13] |
| | | Lin. simplex | 98.50 [98.48, 98.51] | [-2.19, 2.19] | [-5.88, 5.02] | 3.28 [3.26, 3.29] | 1.25 [1.23, 1.26] | 7.62 [7.55, 7.69] |
| | | Vec. scaling | 98.38 [98.36, 98.40] | [-2.27, 2.27] | [-5.05, 4.70] | 3.35 [3.34, 3.36] | 1.34 [1.33, 1.36] | 4.34 [4.32, 4.37] |
| | NNLS | None | 98.17 [98.14, 98.20] | [-2.42, 2.42] | [-9.04, 6.81] | 3.14 [3.13, 3.16] | 1.52 [1.50, 1.55] | 7.04 [6.96, 7.13] |
| | | Lin. clip+norm | 98.70 [98.68, 98.72] | [-2.04, 2.04] | [-5.33, 4.58] | 2.88 [2.87, 2.89] | 1.08 [1.07, 1.10] | 6.05 [5.97, 6.13] |
| | | Lin. simplex | 98.94 [98.93, 98.95] | [-1.84, 1.84] | [-4.07, 3.69] | 2.70 [2.69, 2.71] | 0.88 [0.87, 0.89] | 5.55 [5.49, 5.62] |
| | | Vec. scaling | 98.18 [98.17, 98.20] | [-2.41, 2.41] | [-4.36, 6.08] | 3.64 [3.63, 3.65] | 1.51 [1.50, 1.52] | 5.74 [5.66, 5.81] |
| | PSLS | None | 97.99 [97.95, 98.02] | [-2.53, 2.53] | [-9.94, 7.46] | 3.18 [3.16, 3.19] | 1.67 [1.64, 1.70] | 7.36 [7.28, 7.45] |
| | | Lin. clip+norm | 98.61 [98.59, 98.63] | [-2.10, 2.10] | [-5.99, 5.24] | 2.95 [2.94, 2.96] | 1.15 [1.14, 1.17] | 5.69 [5.64, 5.74] |
| | | Lin. simplex | 98.89 [98.88, 98.91] | [-1.88, 1.88] | [-4.77, 4.25] | 2.79 [2.78, 2.80] | 0.92 [0.91, 0.93] | 5.91 [5.84, 5.98] |
| | | Vec. scaling | 98.07 [98.05, 98.10] | [-2.48, 2.48] | [-7.11, 6.93] | 3.65 [3.64, 3.67] | 1.60 [1.58, 1.62] | 5.78 [5.73, 5.84] |
| *Lookup Classifier* (1-NN) | XGB | None | 90.65 [90.54, 90.77] | [-5.46, 5.46] | [-17.10, 15.16] | 7.26 [7.23, 7.30] | 7.76 [7.66, 7.86] | 19.58 [19.42, 19.74] |
| | | Lin. clip+norm | 92.47 [92.37, 92.56] | [-4.90, 4.90] | [-16.58, 15.16] | 6.27 [6.24, 6.30] | 6.25 [6.17, 6.34] | 18.08 [17.91, 18.25] |
| | | Lin. simplex | 93.31 [93.22, 93.41] | [-4.62, 4.62] | [-17.09, 14.29] | 5.61 [5.58, 5.64] | 5.55 [5.47, 5.63] | 22.98 [22.70, 23.28] |
| | | Vec. scaling | 93.22 [93.13, 93.31] | [-4.65, 4.65] | [-16.05, 15.03] | 6.45 [6.42, 6.47] | 5.63 [5.56, 5.70] | 14.16 [14.05, 14.28] |
| | MLP | None | 96.04 [95.98, 96.10] | [-3.55, 3.55] | [-10.36, 12.62] | 4.89 [4.87, 4.92] | 3.29 [3.24, 3.34] | 8.61 [8.55, 8.66] |
| | | Lin. clip+norm | 96.73 [96.68, 96.77] | [-3.23, 3.23] | [-10.28, 11.13] | 4.28 [4.26, 4.30] | 2.72 [2.68, 2.76] | 9.15 [9.08, 9.23] |
| | | Lin. simplex | 97.14 [97.09, 97.18] | [-3.02, 3.02] | [-10.28, 10.91] | 3.99 [3.98, 4.01] | 2.38 [2.34, 2.41] | 10.79 [10.69, 10.88] |
| | | Vec. scaling | 97.55 [97.52, 97.58] | [-2.80, 2.80] | [-9.50, 9.26] | 3.96 [3.94, 3.97] | 2.04 [2.01, 2.06] | 6.62 [6.58, 6.66] |
| | SWN | None | 96.72 [96.67, 96.76] | [-3.24, 3.24] | [-10.69, 11.18] | 4.42 [4.40, 4.44] | 2.73 [2.69, 2.77] | 7.36 [7.31, 7.40] |
| | | Lin. clip+norm | 97.44 [97.40, 97.48] | [-2.86, 2.86] | [-10.52, 10.28] | 3.63 [3.61, 3.65] | 2.13 [2.09, 2.16] | 7.13 [7.08, 7.18] |
| | | Lin. simplex | 97.74 [97.70, 97.78] | [-2.68, 2.68] | [-10.42, 9.88] | 3.51 [3.49, 3.52] | 1.88 [1.84, 1.91] | 9.05 [8.97, 9.12] |
| | | Vec. scaling | 97.73 [97.71, 97.76] | [-2.69, 2.69] | [-9.58, 9.33] | 3.84 [3.83, 3.86] | 1.88 [1.86, 1.90] | 5.92 [5.89, 5.96] |
| | NNLS | None | 96.51 [96.47, 96.56] | [-3.33, 3.33] | [-10.84, 10.88] | 4.97 [4.95, 4.99] | 2.89 [2.85, 2.94] | 10.56 [10.50, 10.62] |
| | | Lin. clip+norm | 97.40 [97.37, 97.44] | [-2.88, 2.88] | [-9.36, 10.01] | 4.02 [4.00, 4.04] | 2.16 [2.12, 2.19] | 8.49 [8.43, 8.55] |
| | | Lin. simplex | 98.12 [98.10, 98.15] | [-2.45, 2.45] | [-8.46, 9.00] | 3.31 [3.30, 3.33] | 1.56 [1.54, 1.58] | 7.16 [7.09, 7.22] |
| | | Vec. scaling | 97.66 [97.64, 97.68] | [-2.73, 2.73] | [-7.06, 7.07] | 4.41 [4.40, 4.42] | 1.94 [1.93, 1.96] | 6.49 [6.44, 6.53] |
| | PSLS | None | 96.65 [96.61, 96.70] | [-3.27, 3.27] | [-10.65, 10.36] | 4.85 [4.83, 4.87] | 2.78 [2.74, 2.82] | 10.24 [10.18, 10.29] |
| | | Lin. clip+norm | 97.56 [97.52, 97.59] | [-2.79, 2.79] | [-8.72, 9.34] | 3.97 [3.96, 3.99] | 2.03 [2.00, 2.06] | 8.41 [8.35, 8.46] |
| | | Lin. simplex | 98.27 [98.25, 98.29] | [-2.35, 2.35] | [-7.66, 8.23] | 3.27 [3.26, 3.28] | 1.44 [1.42, 1.45] | 7.01 [6.95, 7.07] |
| | | Vec. scaling | 97.73 [97.71, 97.75] | [-2.69, 2.69] | [-7.16, 7.23] | 4.36 [4.35, 4.38] | 1.88 [1.87, 1.89] | 6.60 [6.55, 6.64] |

Table S17: Full results for **data-driven soft-labels (without pooling)**. For $R^2$, MAE, MSE and KL, we indicated first the value of the metric computed on the whole test set, and then the 95% confidence interval computed with BCa bootstrap [47] with 10,000 resamples.

| Classifier | Deconvolver | Dec. Calib. | $\mathbf{R^2}$ ($\times 10^{-2}$) | LoA ($\times 10^{-2}$) | LoA worst-class ($\times 10^{-2}$) | MAE ($\times 10^{-3}$) | MSE ($\times 10^{-4}$) | KL ($\times 10^{-2}$) |
|---|---|---|---|---|---|---|---|---|
| *Dismir* [19] (CNN+LSTM) | XGB | None | 97.08 [97.03, 97.13] | [-3.05, 3.05] | [-10.44, 10.21] | 3.77 [3.75, 3.79] | 2.42 [2.38, 2.46] | 6.72 [6.67, 6.77] |
| | | Lin. clip+norm | 97.40 [97.35, 97.44] | [-2.88, 2.88] | [-9.94, 9.94] | 3.45 [3.43, 3.47] | 2.16 [2.12, 2.20] | 6.41 [6.35, 6.46] |
| | | Lin. simplex | 97.71 [97.67, 97.75] | [-2.70, 2.70] | [-9.77, 9.59] | 3.15 [3.14, 3.17] | 1.90 [1.87, 1.94] | 6.68 [6.62, 6.75] |
| | | Vec. scaling | 97.51 [97.47, 97.54] | [-2.82, 2.82] | [-8.89, 9.17] | 3.89 [3.87, 3.90] | 2.07 [2.04, 2.10] | 5.24 [5.20, 5.27] |
| | MLP | None | 97.64 [97.61, 97.67] | [-2.74, 2.74] | [-6.81, 9.71] | 3.70 [3.68, 3.71] | 1.96 [1.94, 1.98] | 6.42 [6.37, 6.46] |
| | | Lin. clip+norm | 97.97 [97.95, 97.99] | [-2.54, 2.54] | [-6.85, 8.95] | 3.48 [3.47, 3.49] | 1.68 [1.67, 1.70] | 7.22 [7.17, 7.29] |
| | | Lin. simplex | 98.12 [98.10, 98.15] | [-2.45, 2.45] | [-6.82, 8.82] | 3.35 [3.34, 3.36] | 1.56 [1.54, 1.58] | 8.56 [8.48, 8.63] |
| | | Vec. scaling | 98.31 [98.30, 98.33] | [-2.32, 2.32] | [-6.28, 7.01] | 3.34 [3.33, 3.35] | 1.40 [1.39, 1.41] | 5.28 [5.24, 5.32] |
| | SWN | None | 98.24 [98.20, 98.27] | [-2.37, 2.37] | [-6.74, 7.44] | 3.19 [3.18, 3.20] | 1.46 [1.43, 1.49] | 4.89 [4.86, 4.92] |
| | | Lin. clip+norm | 98.42 [98.39, 98.44] | [-2.25, 2.25] | [-6.64, 6.85] | 3.18 [3.17, 3.19] | 1.31 [1.29, 1.34] | 5.35 [5.31, 5.38] |
| | | Lin. simplex | 98.37 [98.34, 98.39] | [-2.28, 2.28] | [-7.04, 6.86] | 3.29 [3.28, 3.30] | 1.36 [1.34, 1.38] | 7.47 [7.40, 7.54] |
| | | Vec. scaling | 98.29 [98.27, 98.31] | [-2.34, 2.34] | [-6.11, 6.35] | 3.41 [3.40, 3.43] | 1.42 [1.41, 1.44] | 4.70 [4.67, 4.73] |
| | NNLS | None | 98.13 [98.10, 98.16] | [-2.44, 2.44] | [-8.72, 7.33] | 3.33 [3.32, 3.35] | 1.55 [1.53, 1.58] | 6.96 [6.89, 7.02] |
| | | Lin. clip+norm | 98.58 [98.56, 98.60] | [-2.13, 2.13] | [-5.86, 6.08] | 3.02 [3.00, 3.03] | 1.18 [1.16, 1.19] | 6.03 [5.97, 6.09] |
| | | Lin. simplex | 98.89 [98.88, 98.90] | [-1.88, 1.88] | [-4.65, 5.11] | 2.72 [2.71, 2.73] | 0.92 [0.91, 0.93] | 5.60 [5.53, 5.66] |
| | | Vec. scaling | 98.33 [98.31, 98.34] | [-2.31, 2.31] | [-4.80, 5.12] | 3.63 [3.62, 3.64] | 1.39 [1.38, 1.40] | 5.32 [5.26, 5.37] |
| | PSLS | None | 98.03 [98.00, 98.07] | [-2.50, 2.50] | [-9.57, 7.71] | 3.26 [3.24, 3.28] | 1.63 [1.60, 1.66] | 6.88 [6.81, 6.95] |
| | | Lin. clip+norm | 98.57 [98.55, 98.59] | [-2.14, 2.14] | [-6.14, 6.01] | 3.00 [2.99, 3.01] | 1.19 [1.17, 1.21] | 6.35 [6.27, 6.42] |
| | | Lin. simplex | 98.88 [98.87, 98.89] | [-1.89, 1.89] | [-4.85, 5.01] | 2.77 [2.76, 2.78] | 0.93 [0.92, 0.94] | 5.76 [5.70, 5.82] |
| | | Vec. scaling | 98.23 [98.21, 98.25] | [-2.38, 2.38] | [-6.27, 6.89] | 3.64 [3.62, 3.65] | 1.47 [1.46, 1.48] | 5.62 [5.57, 5.67] |
| *MethylBERT* [20] | XGB | None | 95.98 [95.92, 96.03] | [-3.58, 3.58] | [-10.20, 8.45] | 4.68 [4.66, 4.70] | 3.34 [3.29, 3.39] | 9.20 [9.14, 9.26] |
| | | Lin. clip+norm | 96.85 [96.81, 96.90] | [-3.17, 3.17] | [-6.57, 7.05] | 4.10 [4.08, 4.12] | 2.61 [2.57, 2.65] | 8.30 [8.24, 8.36] |
| | | Lin. simplex | 97.45 [97.41, 97.49] | [-2.85, 2.85] | [-5.96, 6.60] | 3.64 [3.62, 3.65] | 2.12 [2.08, 2.15] | 9.56 [9.46, 9.66] |
| | | Vec. scaling | 96.90 [96.86, 96.93] | [-3.15, 3.15] | [-6.03, 7.93] | 4.60 [4.58, 4.62] | 2.58 [2.55, 2.61] | 6.72 [6.68, 6.76] |
| | MLP | None | 97.43 [97.40, 97.46] | [-2.86, 2.86] | [-6.93, 9.22] | 3.94 [3.92, 3.95] | 2.13 [2.11, 2.16] | 7.29 [7.24, 7.33] |
| | | Lin. clip+norm | 97.75 [97.73, 97.78] | [-2.68, 2.68] | [-7.30, 7.83] | 3.80 [3.78, 3.81] | 1.86 [1.84, 1.89] | 8.64 [8.57, 8.72] |
| | | Lin. simplex | 97.85 [97.83, 97.88] | [-2.62, 2.62] | [-7.82, 8.12] | 3.67 [3.65, 3.68] | 1.78 [1.76, 1.80] | 11.25 [11.15, 11.36] |
| | | Vec. scaling | 97.83 [97.81, 97.86] | [-2.63, 2.63] | [-7.59, 7.01] | 3.84 [3.83, 3.86] | 1.80 [1.78, 1.82] | 6.64 [6.59, 6.68] |
| | SWN | None | 97.99 [97.96, 98.01] | [-2.53, 2.53] | [-6.70, 7.70] | 3.49 [3.48, 3.51] | 1.67 [1.65, 1.69] | 5.94 [5.90, 5.98] |
| | | Lin. clip+norm | 98.21 [98.19, 98.23] | [-2.39, 2.39] | [-7.18, 6.75] | 3.35 [3.34, 3.37] | 1.49 [1.47, 1.50] | 6.77 [6.72, 6.83] |
| | | Lin. simplex | 98.20 [98.18, 98.23] | [-2.39, 2.39] | [-7.86, 7.10] | 3.38 [3.36, 3.39] | 1.49 [1.47, 1.51] | 9.33 [9.24, 9.42] |
| | | Vec. scaling | 98.17 [98.15, 98.18] | [-2.42, 2.42] | [-6.94, 6.31] | 3.60 [3.59, 3.61] | 1.52 [1.51, 1.54] | 5.54 [5.50, 5.57] |
| | NNLS | None | 97.55 [97.51, 97.59] | [-2.79, 2.79] | [-9.46, 8.02] | 3.99 [3.97, 4.01] | 2.03 [2.00, 2.06] | 9.40 [9.33, 9.48] |
| | | Lin. clip+norm | 98.04 [98.02, 98.07] | [-2.50, 2.50] | [-7.70, 7.06] | 3.57 [3.56, 3.59] | 1.62 [1.60, 1.65] | 8.39 [8.31, 8.47] |
| | | Lin. simplex | 98.48 [98.46, 98.49] | [-2.20, 2.20] | [-6.78, 6.25] | 3.07 [3.06, 3.08] | 1.27 [1.25, 1.28] | 7.57 [7.49, 7.66] |
| | | Vec. scaling | 97.97 [97.95, 97.98] | [-2.55, 2.55] | [-6.25, 5.40] | 4.07 [4.06, 4.09] | 1.69 [1.68, 1.70] | 7.03 [6.97, 7.10] |
| | PSLS | None | 97.73 [97.70, 97.77] | [-2.69, 2.69] | [-9.71, 7.75] | 3.67 [3.65, 3.69] | 1.88 [1.85, 1.91] | 8.51 [8.44, 8.59] |
| | | Lin. clip+norm | 98.26 [98.23, 98.28] | [-2.36, 2.36] | [-7.15, 6.24] | 3.38 [3.37, 3.40] | 1.45 [1.43, 1.47] | 8.04 [7.97, 8.13] |
| | | Lin. simplex | 98.62 [98.60, 98.63] | [-2.10, 2.10] | [-6.06, 5.32] | 3.01 [2.99, 3.02] | 1.15 [1.14, 1.16] | 7.53 [7.45, 7.61] |
| | | Vec. scaling | 97.99 [97.97, 98.02] | [-2.53, 2.53] | [-7.29, 6.61] | 3.93 [3.92, 3.94] | 1.67 [1.65, 1.68] | 6.88 [6.82, 6.94] |
| *Lookup Classifier* (1-NN) | XGB | None | 76.78 [76.60, 76.96] | [-8.60, 8.60] | [-19.02, 16.31] | 14.89 [14.85, 14.93] | 19.27 [19.09, 19.46] | 45.76 [45.55, 45.98] |
| | | Lin. clip+norm | 85.35 [85.17, 85.52] | [-6.84, 6.84] | [-18.20, 17.52] | 10.25 [10.22, 10.29] | 12.16 [12.00, 12.33] | 31.67 [31.45, 31.90] |
| | | Lin. simplex | 87.97 [87.79, 88.15] | [-6.19, 6.19] | [-19.03, 16.54] | 8.31 [8.27, 8.35] | 9.99 [9.83, 10.15] | 33.51 [33.17, 33.85] |
| | | Vec. scaling | 88.15 [87.97, 88.32] | [-6.15, 6.15] | [-17.67, 16.93] | 9.42 [9.39, 9.45] | 9.84 [9.70, 9.98] | 23.90 [23.74, 24.05] |
| | MLP | None | 84.10 [83.93, 84.28] | [-7.12, 7.12] | [-13.78, 14.22] | 11.79 [11.74, 11.83] | 13.20 [13.04, 13.36] | 28.86 [28.71, 29.01] |
| | | Lin. clip+norm | 89.01 [88.88, 89.14] | [-5.92, 5.92] | [-13.16, 13.26] | 8.87 [8.84, 8.91] | 9.12 [9.00, 9.24] | 22.09 [21.96, 22.22] |
| | | Lin. simplex | 90.95 [90.84, 91.06] | [-5.37, 5.37] | [-13.38, 12.88] | 7.76 [7.73, 7.79] | 7.51 [7.42, 7.61] | 21.89 [21.74, 22.03] |
| | | Vec. scaling | 92.72 [92.64, 92.79] | [-4.82, 4.82] | [-10.65, 11.35] | 7.40 [7.37, 7.42] | 6.04 [5.98, 6.11] | 15.45 [15.38, 15.53] |
| | SWN | None | 88.62 [88.49, 88.75] | [-6.02, 6.02] | [-14.10, 12.73] | 9.60 [9.57, 9.64] | 9.45 [9.33, 9.56] | 21.71 [21.60, 21.82] |
| | | Lin. clip+norm | 93.34 [93.26, 93.43] | [-4.61, 4.61] | [-11.53, 11.72] | 6.60 [6.57, 6.63] | 5.52 [5.45, 5.60] | 14.92 [14.83, 15.01] |
| | | Lin. simplex | 95.08 [95.02, 95.14] | [-3.96, 3.96] | [-11.57, 10.92] | 5.91 [5.89, 5.93] | 4.08 [4.03, 4.14] | 17.93 [17.80, 18.06] |
| | | Vec. scaling | 95.13 [95.08, 95.17] | [-3.94, 3.94] | [-8.23, 9.44] | 6.32 [6.30, 6.34] | 4.04 [4.00, 4.08] | 12.11 [12.05, 12.16] |
| | NNLS | None | 87.51 [87.39, 87.63] | [-6.31, 6.31] | [-14.58, 13.60] | 11.90 [11.87, 11.93] | 10.37 [10.26, 10.48] | 30.05 [29.93, 30.16] |
| | | Lin. clip+norm | 94.15 [94.07, 94.22] | [-4.32, 4.32] | [-11.29, 12.17] | 6.63 [6.60, 6.66] | 4.86 [4.79, 4.93] | 15.23 [15.14, 15.31] |
| | | Lin. simplex | 96.67 [96.62, 96.72] | [-3.26, 3.26] | [-10.60, 10.72] | 4.56 [4.54, 4.58] | 2.76 [2.72, 2.80] | 11.07 [10.99, 11.15] |
| | | Vec. scaling | 95.46 [95.42, 95.49] | [-3.81, 3.81] | [-6.90, 8.13] | 6.48 [6.46, 6.50] | 3.77 [3.75, 3.79] | 10.82 [10.77, 10.87] |
| | PSLS | None | 86.86 [86.74, 86.99] | [-6.47, 6.47] | [-14.60, 13.86] | 12.02 [11.98, 12.06] | 10.90 [10.79, 11.02] | 30.86 [30.74, 30.98] |
| | | Lin. clip+norm | 93.81 [93.74, 93.89] | [-4.44, 4.44] | [-11.76, 12.69] | 6.81 [6.78, 6.84] | 5.14 [5.06, 5.21] | 15.74 [15.65, 15.83] |
| | | Lin. simplex | 96.56 [96.51, 96.62] | [-3.31, 3.31] | [-11.20, 11.43] | 4.65 [4.63, 4.67] | 2.85 [2.81, 2.90] | 11.20 [11.12, 11.28] |
| | | Vec. scaling | 94.91 [94.86, 94.96] | [-4.03, 4.03] | [-10.39, 11.96] | 6.76 [6.74, 6.78] | 4.22 [4.18, 4.26] | 11.87 [11.81, 11.92] |

### E.3 OOD experiment detailed results

Table S18: Number of (deconvolver, calibrator) configurations achieving higher TCS than *Celfie U25* per classifier, labeling scheme and feature selection strategy. Top section shows classifiers trained on labeled data; bottom section shows labeling-independent classifiers. Each cell of the table reports the count as *total* *(strict)*, where *total* is the number of configurations with higher average TCS (maximum is 20) and *strict* is the number of configurations with mean significantly different based on Wilcoxon signed-rank test ($pvalue \leq 0.05$). The TCS is averaged per tissue and averaged again to ensure equal weight of each tissue in the final score. Feature selection strategies: **DiagBckg** uses the combination of diagonal + background column of the feature matrices; **Top156** uses the top-156 features (cf. Section D.4).

| **Classifier** | **Hard Labels w/ bckg.** | | **Canonical Soft Labels** | | **DD Soft Labels w/o pool.** | **DD Soft Labels w/ pool.** |
|---|---|---|---|---|---|---|
| | DiagBckg | Top156 | DiagBckg | Top156 | Top156 | Top156 |
| *(a) Classifiers trained on labeled data* | | | | | | |
| *Dismir*[19] (CNN+LSTM) | 20 (0) | 20 (1) | 12 (0) | 16 (0) | 20 (11) | 13 (4) |
| *MethylBERT*[20] | 0 (0) | 0 (0) | 0 (0) | 0 (0) | 15 (0) | 19 (10) |
| *Lookup Classifier* (1-NN) | 0 (0) | 0 (0) | N/A | N/A | 16 (10) | 14 (0) |
| *(b) Labeling-independent classifiers* | | | | | | |
| | Uniform prior | Training counts prior | | | | |
| *CancerDetector*[18] | 17 (4) | 20 (11) | | | | |

Table S19: Comparison of TCS (mean and standard deviation in parentheses) against *Celfie U25* for Hard Labeling and Canonical Soft Labeling schemes using the **top 156 features**. Each row corresponds to a deconvolver-calibrator combination, evaluated with three downstream classifiers under hard labels (*Lookup Classifier*, *MethylBERT*, *Dismir*) and two under canonical soft labels (*MethylBERT*, *Dismir*). Cells shaded in magenta indicate combinations whose mean TCS is significantly *lower* than that of *Celfie U25*, while cells shaded in teal indicate combinations whose mean TCS is significantly *higher* (Wilcoxon signed-rank test, $p \leq 0.05$). Color intensity scales with the magnitude of the difference.

| | | Hard Labels | | | Canonical Soft Labels | |
|---|---|---|---|---|---|---|
| Deconvolver | Calibrator | *Lookup Classifier* | *MethylBERT* | *Dismir* | *MethylBERT* | *Dismir* |
| *XGB* | — | 40.16 | 25.65 | 70.65 (17.07) | 53.20 (12.61) | 66.95 (16.16) |
| *XGB* | Lin. clip+norm | 40.25 | 25.95 | 70.68 (17.08) | 55.16 (13.61) | 66.99 (16.16) |
| *XGB* | Lin. simplex | 40.56 | 25.98 | 70.73 (17.09) | 56.43 (14.08) | 67.02 (16.18) |
| *XGB* | Vec. scaling | 44.70 | 25.13 | 71.20 (17.43) | 59.46 (16.01) | 67.49 (16.52) |
| *MLP* | — | 14.12 | 53.09 | 67.09 (16.11) | 57.06 (16.27) | 66.28 (19.42) |
| *MLP* | Lin. clip+norm | 13.89 | 54.61 | 67.13 (16.14) | 57.86 (17.12) | 66.29 (19.45) |
| *MLP* | Lin. simplex | 15.11 | 55.44 | 67.14 (16.14) | 58.76 (17.82) | 66.35 (19.46) |
| *MLP* | Vec. scaling | 20.07 | 56.52 | 67.68 (16.22) | 58.31 (18.33) | 66.73 (19.58) |
| *SWN* | — | 34.26 | 48.15 | 71.23 (19.01) | 62.31 (17.7) | 60.81 (25.59) |
| *SWN* | Lin. clip+norm | 30.91 | 50.18 | 70.97 (18.87) | 63.14 (18.19) | 60.67 (25.35) |
| *SWN* | Lin. simplex | 30.96 | 50.12 | 70.96 (18.88) | 64.04 (18.85) | 60.54 (25.29) |
| *SWN* | Vec. scaling | 32.11 | 52.03 | 71.26 (19.15) | 64.41 (19.58) | 60.74 (25.86) |
| *NNLS* | — | 34.25 | 49.98 | 69.76 (17.02) | 58.75 (16.69) | 67.95 (17.96) |
| *NNLS* | Lin. clip+norm | 33.02 | 51.86 | 69.68 (17.07) | 59.35 (17.28) | 67.60 (17.96) |
| *NNLS* | Lin. simplex | 33.94 | 52.34 | 69.87 (17.1) | 61.11 (18.19) | 67.75 (17.99) |
| *NNLS* | Vec. scaling | 37.02 | 53.96 | 71.73 (18.42) | 62.64 (20.65) | 70.26 (19.6) |
| *PSLS* | — | 33.95 | 50.53 | 66.74 (18.69) | 60.01 (17.82) | 67.51 (17.45) |
| *PSLS* | Lin. clip+norm | 32.45 | 52.17 | 66.21 (18.76) | 60.49 (18.41) | 66.85 (17.51) |
| *PSLS* | Lin. simplex | 32.97 | 52.58 | 66.50 (18.67) | 61.76 (19.14) | 67.14 (17.34) |
| *PSLS* | Vec. scaling | 42.12 | 54.42 | 65.69 (19.88) | 62.93 (20.9) | 66.90 (19.09) |

Table S20: Comparison of TCS (mean and standard deviation in parentheses) against *Celfie U25* for Hard Labeling and Canonical Soft Labeling schemes using the **Diag. + Bckg. features**. Each row corresponds to a deconvolver-calibrator combination, evaluated with two downstream classifiers under hard labels (*MethylBERT*, *Dismir*) and two under canonical soft labels (*MethylBERT*, *Dismir*). Cells shaded in magenta indicate combinations whose mean TCS is significantly *lower* than that of *Celfie U25*, while cells shaded in teal indicate combinations whose mean TCS is significantly *higher* (Wilcoxon signed-rank test, $p \leq 0.05$). Color intensity scales with the magnitude of the difference.

| | | Hard Labels | | | Canonical Soft Labels | |
|---|---|---|---|---|---|---|
| Deconvolver | Calibrator | *Lookup Classifier* | *MethylBERT* | *Dismir* | *MethylBERT* | *Dismir* |
| *XGB* | — | 40.16 (16.75) | 36.94 (8.69) | 70.30 (16.84) | 37.20 (9.18) | 69.52 (17.73) |
| *XGB* | Lin. clip+norm | 40.25 (17.93) | 38.77 (9.66) | 70.33 (16.85) | 38.65 (9.97) | 69.55 (17.73) |
| *XGB* | Lin. simplex | 40.56 (18.41) | 38.91 (9.75) | 70.38 (16.87) | 39.10 (10.22) | 69.59 (17.75) |
| *XGB* | Vec. scaling | 44.70 (20.77) | 41.02 (12.07) | 70.85 (17.18) | 39.99 (12.1) | 70.05 (18.09) |
| *MLP* | — | 13.44 (7.98) | 42.78 (13.8) | 66.94 (16.37) | 57.00 (15.9) | 61.40 (21.62) |
| *MLP* | Lin. clip+norm | 15.09 (8.66) | 45.04 (14.88) | 66.89 (16.37) | 57.65 (16.33) | 61.46 (21.6) |
| *MLP* | Lin. simplex | 16.48 (7.8) | 44.89 (15.03) | 66.86 (16.35) | 58.21 (16.85) | 61.52 (21.62) |
| *MLP* | Vec. scaling | 27.49 (13.76) | 46.52 (15.64) | 66.87 (16.28) | 57.96 (16.86) | 61.45 (21.43) |
| *SWN* | — | 36.87 (9.62) | 44.50 (12.74) | 69.51 (18.72) | 58.11 (15.71) | 52.29 (28.02) |
| *SWN* | Lin. clip+norm | 32.72 (9.97) | 46.21 (13.48) | 69.28 (18.61) | 58.92 (16.36) | 52.17 (27.81) |
| *SWN* | Lin. simplex | 32.98 (10.53) | 46.14 (13.53) | 69.21 (18.58) | 59.90 (17.11) | 52.12 (27.75) |
| *SWN* | Vec. scaling | 32.92 (11.77) | 47.87 (14.91) | 69.43 (18.82) | 60.13 (17.82) | 51.81 (28.17) |
| *NNLS* | — | 34.25 (14.54) | 44.74 (14.7) | 66.07 (19.42) | 54.35 (16.26) | 66.03 (18.99) |
| *NNLS* | Lin. clip+norm | 33.02 (15.38) | 45.76 (15.65) | 65.61 (19.46) | 54.34 (17.03) | 65.57 (19.0) |
| *NNLS* | Lin. simplex | 33.94 (16.54) | 45.92 (15.97) | 65.84 (19.4) | 55.06 (17.74) | 65.76 (18.89) |
| *NNLS* | Vec. scaling | 37.02 (20.33) | 47.43 (18.4) | 65.65 (20.53) | 56.51 (19.81) | 66.21 (20.29) |
| *PSLS* | — | 33.95 (14.01) | 41.64 (14.44) | 66.75 (18.68) | 50.37 (16.08) | 67.28 (17.77) |
| *PSLS* | Lin. clip+norm | 32.45 (15.11) | 42.52 (15.3) | 66.21 (18.75) | 50.24 (16.78) | 66.77 (17.81) |
| *PSLS* | Lin. simplex | 32.97 (16.26) | 42.79 (15.71) | 66.50 (18.67) | 51.11 (17.67) | 67.02 (17.67) |
| *PSLS* | Vec. scaling | 42.09 (20.08) | 44.97 (17.88) | 65.71 (19.88) | 52.87 (19.46) | 66.80 (19.18) |

Table S21: Comparison of TCS (mean and standard deviation in parentheses) against *Celfie U25* for Data-Driven Soft-Labels with pooling and Data-Driven Soft-Labels without pooling using the **top 156 features**. Each row corresponds to a deconvolver-calibrator combination, evaluated with three classifiers under Data-Driven Soft-Label (*Lookup Classifier*,*MethylBERT*, *Dismir*), with and without pooling. Cells shaded in magenta indicate combinations whose mean TCS is significantly *lower* than that of *Celfie U25*, while cells shaded in teal indicate combinations whose mean TCS is significantly *higher* (Wilcoxon signed-rank test, $p \leq 0.05$). Color intensity scales with the magnitude of the difference.

| | | Data-Driven Soft-Labels (with pooling) | | | Data-Driven Soft-Labels (without pooling) | | |
|---|---|---|---|---|---|---|---|
| Deconvolver | Calibrator | *Lookup Classifier* | *MethylBERT* | *Dismir* | *Lookup Classifier* | *MethylBERT* | *Dismir* |
| *XGB* | — | 62.71 (16.86) | 63.26 (12.3) | 66.87 (19.83) | 60.58 (13.41) | 60.18 (15.53) | 72.10 (17.02) |
| *XGB* | Lin. clip+norm | 61.58 (16.1) | 64.65 (13.01) | 66.89 (19.84) | 61.32 (14.83) | 60.77 (16.38) | 72.13 (17.02) |
| *XGB* | Lin. simplex | 63.14 (17.14) | 66.40 (13.73) | 66.90 (19.84) | 64.85 (15.84) | 62.25 (17.15) | 72.15 (17.03) |
| *XGB* | Vec. scaling | 64.33 (18.08) | 69.34 (14.88) | 67.36 (20.17) | 71.29 (18.91) | 63.95 (19.25) | 72.54 (17.21) |
| *MLP* | — | 66.20 (16.67) | 67.70 (16.16) | 60.71 (18.14) | 66.42 (16.89) | 63.94 (16.32) | 69.01 (16.1) |
| *MLP* | Lin. clip+norm | 66.72 (16.72) | 68.63 (16.34) | 60.73 (18.12) | 67.55 (16.7) | 64.63 (16.75) | 69.09 (16.07) |
| *MLP* | Lin. simplex | 67.86 (17.25) | 69.90 (16.72) | 60.83 (18.18) | 68.43 (15.43) | 65.73 (17.48) | 69.25 (16.11) |
| *MLP* | Vec. scaling | 69.16 (18.07) | 69.36 (16.68) | 61.25 (18.22) | 75.44 (17.14) | 65.35 (17.52) | 69.61 (16.04) |
| *SWN* | — | 66.78 (16.07) | 70.71 (15.51) | 68.18 (20.44) | 68.80 (16.82) | 66.96 (15.7) | 71.33 (17.66) |
| *SWN* | Lin. clip+norm | 67.76 (15.89) | 71.30 (15.64) | 67.94 (20.32) | 70.31 (16.38) | 67.56 (16.06) | 71.07 (17.57) |
| *SWN* | Lin. simplex | 68.82 (16.34) | 72.50 (16.11) | 67.90 (20.34) | 70.97 (14.62) | 68.57 (16.76) | 71.04 (17.58) |
| *SWN* | Vec. scaling | 70.22 (17.05) | 72.47 (16.15) | 68.26 (20.56) | 74.96 (17.16) | 68.44 (16.8) | 71.27 (17.78) |
| *NNLS* | — | 63.54 (15.63) | 66.49 (17.21) | 64.38 (19.03) | 63.72 (14.81) | 65.33 (16.82) | 68.65 (17.26) |
| *NNLS* | Lin. clip+norm | 65.23 (15.88) | 67.32 (17.09) | 64.23 (18.98) | 67.60 (14.8) | 65.87 (17.04) | 68.44 (17.24) |
| *NNLS* | Lin. simplex | 68.43 (17.29) | 69.60 (17.49) | 64.42 (18.97) | 71.83 (14.85) | 67.54 (17.68) | 68.78 (17.23) |
| *NNLS* | Vec. scaling | 71.34 (18.99) | 71.19 (18.71) | 65.43 (19.79) | 77.28 (18.97) | 69.48 (19.03) | 69.26 (17.85) |
| *PSLS* | — | 63.81 (15.87) | 67.58 (16.34) | 65.57 (18.2) | 62.50 (15.49) | 66.17 (17.45) | 68.34 (17.04) |
| *PSLS* | Lin. clip+norm | 65.39 (15.9) | 68.71 (16.0) | 65.44 (18.1) | 66.58 (15.07) | 66.61 (17.49) | 68.15 (16.99) |
| *PSLS* | Lin. simplex | 68.11 (16.88) | 71.24 (16.19) | 65.64 (18.08) | 69.91 (14.65) | 68.09 (18.05) | 68.49 (16.97) |
| *PSLS* | Vec. scaling | 71.04 (18.35) | 71.63 (17.28) | 66.37 (18.62) | 76.39 (18.63) | 69.51 (19.05) | 68.71 (17.43) |

Table S22: Comparison of TCS (mean and standard deviation in parentheses) against *Celfie U25* for *CancerDetector* with different priors. Cells shaded in magenta indicate combinations whose mean TCS is significantly *lower* than that of *Celfie U25*, while cells shaded in teal indicate combinations whose mean TCS is significantly *higher* (Wilcoxon signed-rank test, $p \leq 0.05$). Color intensity scales with the magnitude of the difference.

| Deconvolver | Calibrator | Training Counts | Uniform |
|---|---|---|---|
| *XGB* | — | 71.44 (17.56) | 69.90 (18.55) |
| *XGB* | Lin. clip+norm | 72.31 (17.92) | 70.79 (18.99) |
| *XGB* | Lin. simplex | 74.34 (18.77) | 72.66 (19.94) |
| *XGB* | Vec. scaling | 75.13 (19.68) | 73.43 (21.25) |
| *MLP* | — | 69.36 (15.79) | 65.16 (15.3) |
| *MLP* | Lin. clip+norm | 70.18 (15.96) | 66.52 (15.94) |
| *MLP* | Lin. simplex | 71.22 (16.19) | 67.43 (16.56) |
| *MLP* | Vec. scaling | 71.03 (16.68) | 68.47 (17.02) |
| *SWN* | — | 67.53 (16.02) | 65.67 (15.61) |
| *SWN* | Lin. clip+norm | 68.92 (16.28) | 67.01 (15.98) |
| *SWN* | Lin. simplex | 70.11 (16.87) | 68.12 (16.56) |
| *SWN* | Vec. scaling | 70.56 (16.85) | 68.94 (16.87) |
| *NNLS* | — | 64.83 (16.98) | 63.73 (16.89) |
| *NNLS* | Lin. clip+norm | 65.72 (17.24) | 64.76 (17.13) |
| *NNLS* | Lin. simplex | 68.04 (18.1) | 67.04 (18.0) |
| *NNLS* | Vec. scaling | 70.46 (18.98) | 69.77 (19.35) |
| *PSLS* | — | 65.31 (18.63) | 63.63 (17.94) |
| *PSLS* | Lin. clip+norm | 65.51 (18.66) | 64.13 (17.9) |
| *PSLS* | Lin. simplex | 67.04 (19.21) | 65.64 (18.53) |
| *PSLS* | Vec. scaling | 67.56 (19.8) | 67.07 (19.13) |

### E.4 Calibration as a standalone add-on to the baseline methods

We propose that calibration can be applied on the existing deconvolution algorithm as a standalone method to influence derived cell-type fractions. To asses effects of different calibration schemes, we have computed TCS for each option evaluated in this paper. Additionaly, we looked at distribution of allocated mass outside of the set of expected cell-types in accordance with reference and inclusion treshold of 1%. Finally, we meassured sparsity of each solution by counting fraction of cell-types recieveing zero proportion as well as tendency of each calibration method to avard a component with highest mass in the baseline results with even higher mass. The corresponding pseudobulk results for all four baselines, with and without our calibrators, are reported in Table S10.

Based on conducted analysis, we make four observations:

1. Applying calibration consistently improves TCS with progressively bigger gains in the next order: *Uncalibrated -> Clip -> Simplex -> Vector*.
2. All calibration methods are tend to allocate even higher mass to the most dominant component of uncalibrated input, with *Vector* method resulting in the most dramatic effect (see Figure S11). Furthermore, based on the gentler slope, *Vector* calibration tend to benefit less for each point increase in the most dominant component than other methods, suggesting that more mass is redistributed within expected set of cell types for this method comparing to others.
3. Applying *Simplex* method consistently increases sparsity of the resulting deconvolved proportions, while *Vector* method has almost no effect (see Figure S10).
4. Calibration tend to increase *leaked* proprtions for some of the cell types and decrease for other across all baselines (see Figure S12). Consistenly growing fraction of *adipocytes*may indicate reinforcing a bias present in the reference atlas; it may also suggest that *adipocytes* are under-detected in the Tabula Sapines reference, hence are not making into expected cell types for the relevant tissues despite being truly present as all baseline methods detect.

Overall, calibration seems to be usefull with a choice of calibration methods depending on the application context. We see *Simplex* method as a suitable choice for scenarious where one is willing to sacrifice detection limit for reducing number of falsely predicted cell types. *Vector* method may be used where expected cell type profile does not feature a single dominant component which this method is likely to exploit and overestimate.

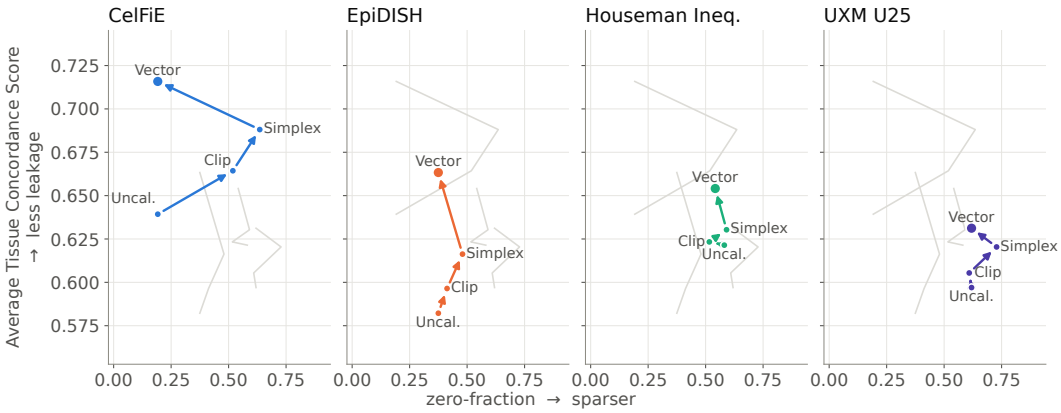


Figure S10: The sample-averaged TCS vs Sparsity trajectories for baseline deconvolvers and calibration methods. *Leakage* used as a term to describe allocation of deconvoluted proportion to cell types that are not present in the constructed reference set of expected cell types per tissue. *Sparsity* is computed as fraction of cell types with zero proportions.

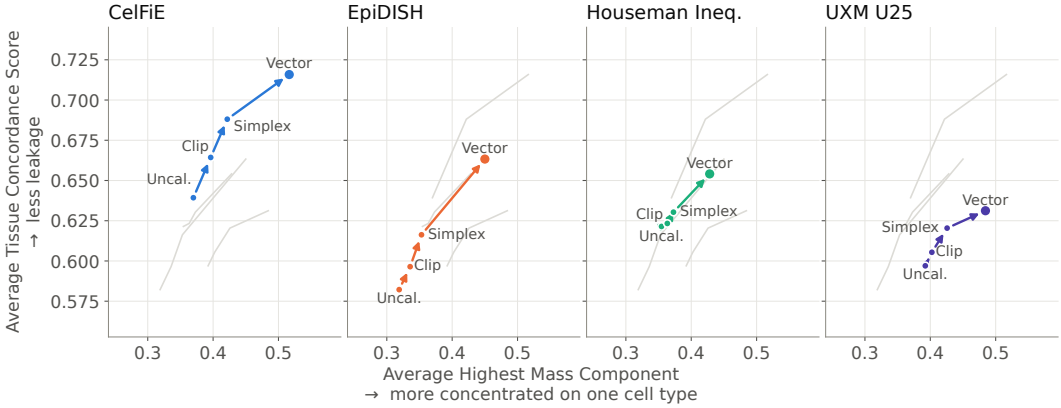


Figure S11: The sample-averaged TCS vs Proportion allocated to the cell type with highest mass trajectories for baseline deconvolvers and calibration methods. *Leakage* used as a term to describe allocation of deconvoluted proportion to cell types that are not present in the constructed reference set of expected cell types per tissue. *Average Highest Mass Component* is computed as average of maximal proportions across all deconvoluted samples.

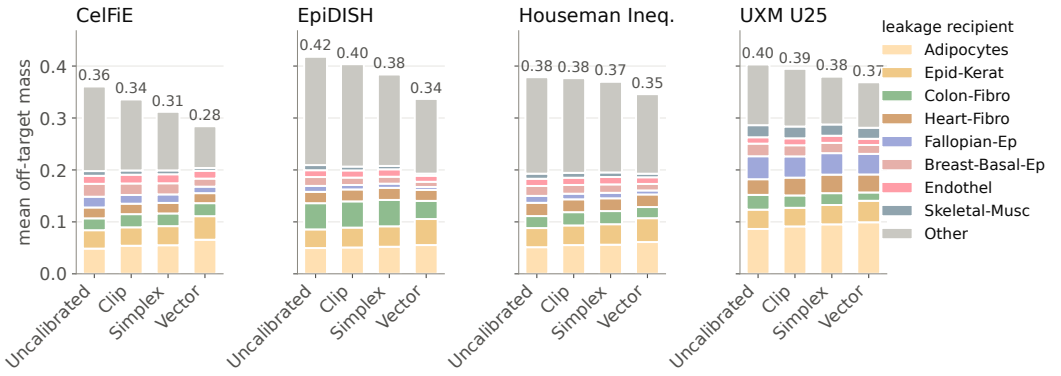


Figure S12: Distributions of the predicted cell types proportions that are not in the expected set of cell types (also refered here as *leaked* proportions). The values in stacked bar plots are averaged per 16 tissues. The value on top of each bar is a total *leakage*, also equivalent to $1 - TCS$.

### E.5 Effect of the reference atlas choice on the *UXM* baseline

We analyzed how the *UXM* baseline method fares against three variants of itself: calibrated versions, a version using a same-sized reference but with markers selected for higher specificity and a version using 250 marker regions per cell type instead of the original 25. As Figure S13 reveals, linear calibration with clip-normalize, linear calibration with simplex projection and vector scaling slightly improve upon the original baseline scores (+0.84, $p \leq 0.001$; +2.34, $p \leq 0.001$ and +3.42, $p \leq 0.001$ respectively). The best results are achieved with the $10x$ larger atlas (+10.69, $p \leq 0.001$); comparable gains can be obtained with a same-sized subset of the U250 atlas optimized for marker specificity (+8.36, $p \leq 0.001$).

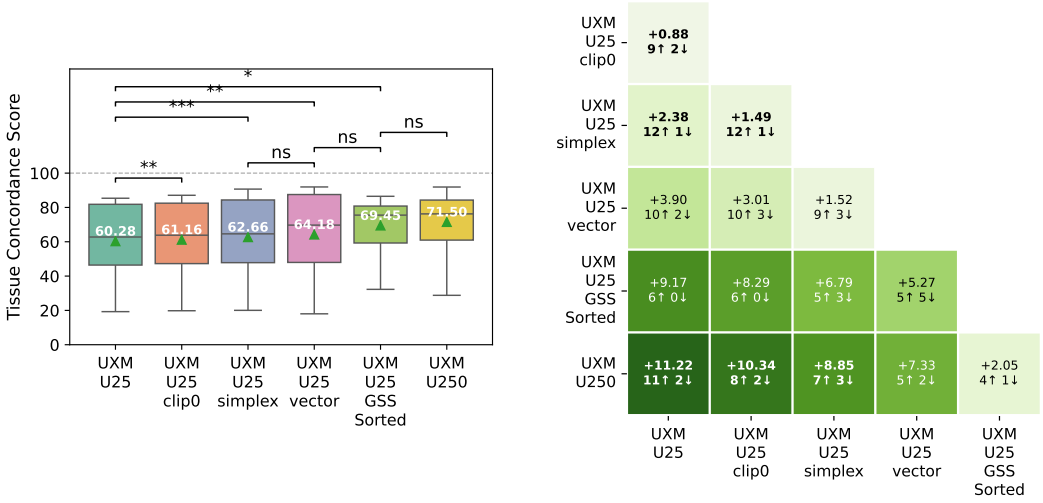


Figure S13: Tissue-averaged TCS for the baseline *UXM* setting (*UXM U25*), various calibration schemes, modified atals optimized for proxy GSS (*UXM U25 GSS sorted*, cf. Section C.3) and extended $10\times$ atlas (*UXM* U250). Left: Box-plot of sample-level TCS. The mean is reported in white text and the significance is derived from the pvalue of the Wilcoxon signed-rank test (*scipy.stats.wilcoxon*). "*", "**" and "***" mean p-value $\leq 0.05$, $\leq 0.01$ and $\leq 0.001$, respectively. Right: Heatmap of differences of tissue-averaged TCS. The values in parentheses indicate the number of tissues where significantly higher (↑) or lower (↓) TCS is observed for the row method relative to column method, using wilcoxon two sided test and significance threshold of 0.05.

### E.6 Analysis of *Syto* sensitivity to the choice of reference atlas

The major part of the results in this paper is built based on two versions of the same reference atlas of cell-type-specific GR groups, one for each of hg19 and hg38 genomic assemblies (Section C.2). By design, we refer to GR groups in the broader sense of informative genomic regions where underlying signal of interest has discriminative properties. When it comes to methylome-based deconvolution, more specific terms like Differentially Methylated Region (DMR) or Differentially Methylated Loci (DML) and their synonyms are frequently used. Another frequent choice is to refer to *markers* as elements of the reference atlases, where each *marker* represents a characteristic localized methylation signature that is expressed in the form of a metric, like, for example, a fraction of unmethylated reads in a certain genomic region that is expected in different cell types, as in the atlas by Loyfer et al. [13].

As it has been highlighted in the literature, the *marker* selection method and the number of *markers* employed in the reference affect deconvolution performance, with the magnitude and direction of effect depending on the particular combination of *markers* and deconvolution method [49, 48, 51]. In *Syto*, such *markers* are not explicitly given at any of the three stages (classification, deconvolution, calibration); however, we use the genomic regions where the *markers* are found as a criterion to filter data at training and inference and as an additional context for classification. Therefore, *Syto* is expected to have its own sensitivity profile with respect to reference atlas composition.

In Section C.3 we demonstrated heterogeneity in the specificity of the GR groups in *U25.l4.hg38* [13] and proposed an alternative atlas of the same size, composed from regions maximizing per-region specificity. We then showed in Section E.5 that using the proposed alternative atlas (here and after *U25.l4.hg38.GSS*) elevates *UXM* by +9.17 on the TCS scale (look for *UXM U25 GSS sorted* in Figure S13). We hypothesised that *Syto* will also benefit from using a more specific atlas, as the pseudobulk experiment offered some evidence that it will.

First, we have observed that per-individual cell-type deconvolution performance is associated with a measure of the *markers* specificity. Indeed, Spearman correlation between the proxy GSS and per-cell-type deconvolution $R^2$ was 0.61 for *Syto* with hard labels, 0.62 for *Syto* with data-driven soft labels [9], and 0.58 for *UXM*. Moreover, on the three cell types with the least specific regions, per-cell-type $R^2$ was 0.49 / 0.76 / 0.84 for *UXM* versus 0.83 / 0.88 / 0.95 for Syto w/ hard labels and 0.84 / 0.92 / 0.95 for Syto w/ DD soft labels, showing that *Syto* performance degrades much less on genomic regions with low specificity, and that soft labeling improves performance on low-specificity regions.

To put our assumptions to the test, we have conducted a focused experiment with an alternative atlas, limiting our choice of classifier to *Dismir* and evaluation to *OOD RRBS* context. We opted out of using a *pseudobulk* experiment as the attained performance was already saturated. We gave priority to *Dismir* owing to its robustness to the labeling scheme choice (see Table S18).

Our main findings are:

- Switching to *U25.l4.hg38.GSS* atlas improves results across different labeling schemes. The improvement magnitude differs, but direction is concordant with *UXM* (Table S23).
- The TCS gains are reinforced by calibration (Table S24).
- The *UXM* improvement partially stems from eliminating Adipocytes and Fallopian Epithelial bias. As can be seen from Table S23, the change in TCS score is not as dramatic for *Syto* as for *UXM*. Table S25 reveals that this is not only due to the higher baseline for *Syto*, but also because *Syto* doesn't have the same bias to begin with, which takes us back to the argument that *Syto* is more robust to the choice of less specific regions.

Additionally, we observed higher Pearson correlation between *UXM* and *Syto* predicted proportions under the *U25.l4.hg38.GSS* atlas ($0.894$–$0.907$ vs $0.775$–$0.809$ under *U25.l4.hg38* for the matched SWN deconvolver; $0.837 - 0.910$ vs $0.778 - 0.829$ across tested 15 Labeling-Deconvolver combinations). We see two possible explanations: either both solutions are closer to the ground truth, or GSS-atlas increased specificity prompts both methods towards a common solution more aggressively.

Our interpretation is subject to the limitations of TCS, discussed in Section E.7. One notable example arises under the *U25.l4.hg38.GSS* atlas: *UXM* dramatically increases the fraction of Smooth Muscle

[9] The reported numbers for *Syto* are for the *Dismir* classifier with the *SWN* deconvolver. Similar numbers were observed across other *Syto* combinations.

Table S23: Comparison of TCS gains for *UXM* and **uncalibrated** result for each labeling scheme using *Dismir* classifier and *SWN* deconvolver. TCS values are averaged within each of the 16 mapped tissues first, then over the 16 tissues. Confidence intervals are computed by drawing 16 tissues with replacement and averaging $\Delta$ 4000 times. *UXM* improves in 11 of the 16 tissues; *Syto* trades bigger gains in roughly half of the tissues over smaller losses in the rest. *U25.l4.hg38* atlas is denoted as U25. *U25.l4.hg38.GSS* atlas is denoted as GSS.

| Method | TCS U25 | TCS GSS | $\Delta$ | 95% CI | Tissues improved |
|---|---|---|---|---|---|
| *UXM* | 60.28 | 69.45 | **+9.17** | [+2.98, +16.01] | 11/16 |
| *Syto* — Hard | 71.23 | 74.48 | +3.25 | [−2.63, +10.76] | 8/16 |
| *Syto* — Soft | 68.18 | 71.83 | +3.66 | [−1.62, +9.96] | 9/16 |
| *Syto* — Soft-NoPool | 71.33 | 72.80 | +1.47 | [−3.80, +7.25] | 7/16 |

Table S24: Relative TCS gains per calibration scheme for *Dismir* classifier + *SWN* deconvolver. TCS values are averaged within each of the 16 mapped tissues first, then over the 16 tissues. The reported number is computed as $\overline{TCS}_{U25.l4.hg38.GSS} - \overline{TCS}_{U25.l4.hg38}$.

| Labeling Scheme | Uncalibrated | Clip | Simplex | Vector |
|---|---|---|---|---|
| Hard Labels | +3.25 | +4.00 | +2.33 | **+4.52** |
| Soft Labels w/ pooling | +3.66 | +4.83 | +4.94 | **+5.96** |
| Soft Labels w/o pooling | +1.47 | +2.50 | +2.15 | **+4.00** |

allocated to uterus and ovary tissues, and *Syto* does the same for uterus and shows a lesser increase for ovary, albeit from an already elevated base ( see Table S26). Since Smooth Muscle is moderately present in both tissues, overall allocation unfairly boosts TCS. Among *Syto* varinats, *Syto* under hard labels "gains" the most from switching to *U25.l4.hg38.GSS* atlas, while *Syto* under soft labels with pooling seems to be more robust than *UXM*. If one were to remove uterus and ovary tissues from analysis, the TCS gain from switching to *U25.l4.hg38.GSS* calculated on the remaining 14 tissues will be considerably lower for *UXM* (+5.2 vs +9.2 on the full tissue set). As for *Syto*, recomputing TCS gain on 14 remaining tissues effectively nullifies gains for uncalibrated variants, reducing maximal gain for calibrated ones (e.g +3.3 vs +6.0 for vector calibrated *SWN* under soft labeling scheme).

To conclude, switching to *U25.l4.hg38.GSS* atlas seems to help *UXM* disproportionately more than *Syto*. While it helps *UXM* to reduce the performance gap, *Syto* maintains an advantage. The results reinforce the idea that *Syto* is more forgiving of the selection of less specific *markers* than *UXM*.

Table S25: Per cell type $\Delta$ of the off-target mass fraction, ranked by pooled $|\Delta|$. Negative sign means that the method using the *U25.l4.hg38.GSS* atlas leaks less mass to that cell type than the method operating on *U25.l4.hg38*.

| Recipient | UXM | Hard | Soft | Soft-NoPool | Direction |
|---|---|---|---|---|---|
| Adipocytes | **−0.049** | −0.014 | −0.026 | +0.012 | mixed |
| Colon-Fibro | **+0.030** | −0.014 | −0.009 | −0.025 | mixed |
| Breast-Basal-Ep | −0.021 | −0.013 | −0.013 | −0.013 | **all −** |
| Fallopian-Ep | **−0.050** | +0.002 | 0.000 | +0.003 | mixed |
| Pancreas-Delta | +0.015 | +0.011 | +0.012 | +0.014 | **all +** |
| Smooth-Musc | −0.002 | +0.011 | +0.008 | +0.015 | mixed |
| Pancreas-Duct | −0.016 | −0.004 | −0.004 | −0.003 | **all −** |
| Lung-Ep-Bron | −0.001 | −0.006 | −0.013 | −0.005 | **all −** |
| Liver-Hep | +0.003 | +0.004 | +0.007 | +0.008 | **all +** |
| Kidney-Ep | +0.004 | +0.004 | +0.005 | +0.006 | **all +** |
| Oligodend | −0.004 | −0.004 | −0.005 | −0.003 | **all −** |
| Thyroid-Ep | −0.006 | −0.001 | +0.007 | −0.002 | mixed |
| Breast-Luminal-Ep | +0.005 | −0.002 | −0.003 | −0.003 | mixed |
| Endothel | +0.001 | +0.003 | +0.001 | −0.006 | mixed |

Table S26: Predicted Smooth Muscle mass fraction in the two tissues where switching the GSS atlas produces the biggest gain. *Syto* results are for *Dismir* classifier + *SWN* deconvolver. Values are averaged over the samples of each tissue; the reference fraction in the header is the Tabula Sapiens composition mapped onto the Loyfer vocabulary. Smooth Muscle is on the expected list of both tissues. *U25.l4.hg38* atlas is denoted as U25. *U25.l4.hg38.GSS* atlas is denoted as GSS.

| | Ovary (ref. $0.162$, $n$=16) | | | Uterus (ref. $0.205$, $n$=12) | | |
|---|---|---|---|---|---|---|
| Method | U25 | GSS | $\Delta$ | U25 | GSS | $\Delta$ |
| UXM | 0.103 | 0.565 | $+0.462$ | 0.266 | 0.683 | $+0.417$ |
| Syto — Hard | 0.658 | 0.785 | $+0.127$ | 0.212 | 0.800 | $+0.587$ |
| Syto — Soft | 0.508 | 0.587 | $+0.078$ | 0.129 | 0.555 | $+0.426$ |
| Syto — Soft-NoPool | 0.398 | 0.627 | $+0.230$ | 0.157 | 0.562 | $+0.404$ |

### E.7 Detailed analysis of TCS failure modes and recovered proportions in the OOD RRBS data

TCS is a conceptually simple metric with several limitations. As mentioned in the main section of this paper, TCS cannot distinguish within expected cell-type errors and thus represents an imperfect upper bound on deconvolution accuracy. It cannot be used to quantify the method's performance absolutely. Nonetheless, we claim that, combined with specific evidence from additional analysis, it can serve as a measure of relative performance. In this supplementary section, we list TCS failure modes and highlight them in experimental data. We then show that, despite several prominent examples of failure modes being observed in our data, their effects are mostly uniform across all compared methods, including *Syto* and baseline variants. For a few cases where failures specific to concrete variants are suspected but cannot be confidently called, we lay out our arguments on why their relative effects are still marginal.

**TCS failure modes.** TCS can both overestimate and underestimate algorithms' performance. Among reasons imaginable, three failure modes stand out as likely to account for most of the occurring errors:

1. **Spurious expected cell type / sensible overpredicted cell type** — a cell type with negligible proportion in the tissue is listed as expected, rewarding a method that overpredicts it. A variation of the same failure mode is when the initial proportion is moderate, so the cell is expected but its proportion is significantly overestimated.
2. **Missing expected cell type** — an abundant cell type may be under-detected by the scRNA-seq behind Tabula Sapiens, so a method that correctly recovers it is penalised.
3. **Ubiquitous cell type** — a cell type is truly present in a majority of tissues, rewarding any method biased toward overpredicting it.

One can control for (1) by setting a threshold for a minimum fraction that cell type must contribute to the tissue composition in order to be added to the set of expected tissue cell types. A higher threshold reduces the risk of (1) but simultaneously increases the risk of penalty for (2). Notably, **Missing expected cell type** can occur even when the threshold for a minimum fraction is set to zero, purely because the reference method completely fails to detect truly present cell types. One cannot control for (3) by adjusting the metric; if the underlying method is biased towards ubiquitous cell types, it will gain more points on TCS scale. Endothelial cells are a good example; being present in most of the tissues (14 out of 16 in our OOD experiment), any method that allocates 100% to these cell types will have a very high average TCS (exactly 87.5 for our data). Luckily, one can easily check if that is the case by examining actual proportions.

In our work, we set a threshold for a minimum fraction that a cell type must contribute to the tissue composition at $1\%$. We also checked how setting different thresholds influences ranking across best methods and found that varying the threshold between $0.5\%$ and $5\%$ does not affect the main claims of the paper (see Figure S14).

Next, we examined how much mass methods allocated to ubiquitous cell types, in absolute terms and in relation to the Tabula Sapiens reference (see Figure S16). Additionally, we modeled the ranking of the methods based on the full set of expected cell types for a given threshold of $1\%$ and two reduced sets: one including only cell types that present no more than in 13 out of 16 tissues and a more conservative set of cell types that are present in at most 7 tissues (see Figure S15).

Baselines and *Syto* methods consistently allocate less mass to the three most ubiquitous cell types than specified in the reference. Three baselines, *UXM U25*, *EpiDISH* and *Houseman CP* as well as *UXM U250* and *UXM U250 GSS sorted* and best *Syto* variant all allocate roughly between $-10$ to $-12$ points to ubiquitous cell types comparing to the reference. The best baseline method, *CelFiE*, and the majority of remaining best *Syto* variants across labeling schemes allocate roughly between $-7$ and $-6$ points to ubiquitous cell types compared to the reference. The closest to the reference method, *Cancer Detector* with train frequencies prior, allocates $\approx -3$ pp. relative to Tabula Sapiens.

Examining TCS of reduced cell type sets allows us to see how much of the TCS gain is spread across the most unique, moderately and ubiquitously present cell types. Excluding the three most ubiquitous cell types from TCS computation repositions *UXM U250* and *UXM U250 GSS sorted* as second and third best solutions after *DD SL w/o pool·Lookup·NNLS·vector*. It also makes *Uniform* prior a better

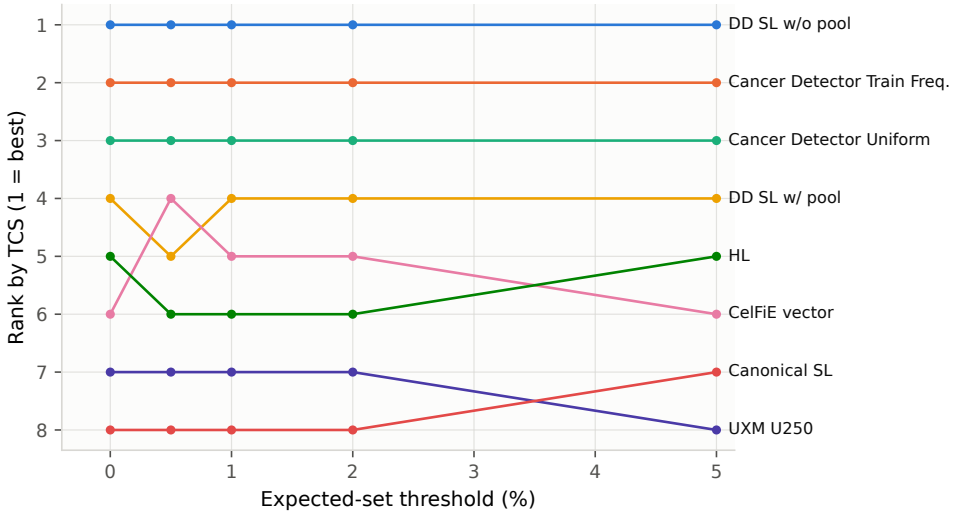


Figure S14: Dependency between TCS rankings of eight top-performing methods and cell type proportion inclusion threshold. Tested values of threshold are: 0%, 0.5%, 1%, 2%, and 5%

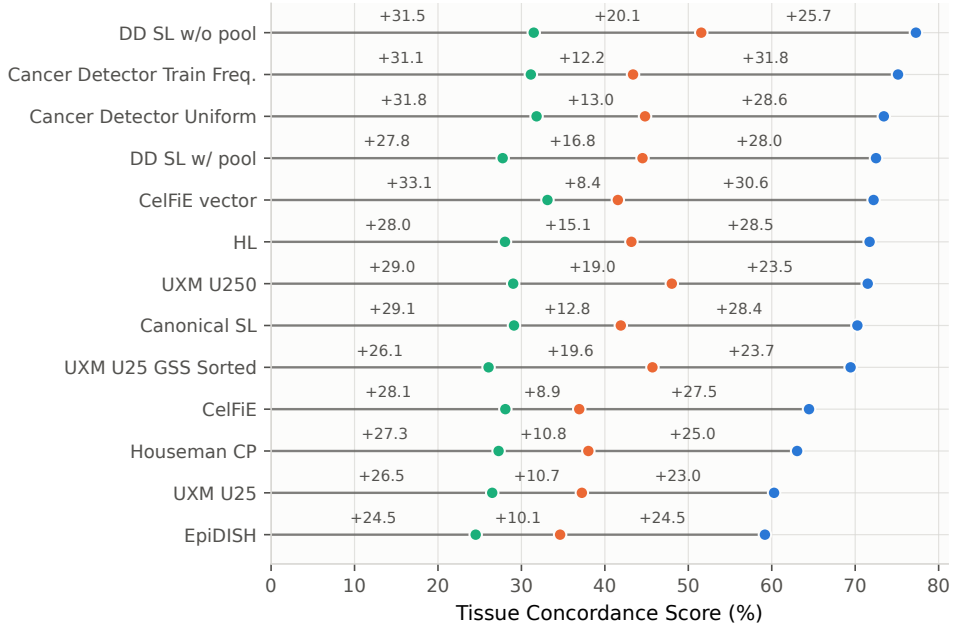


Figure S15: TCS for all baselines, best *Syto* variants, *UXM* using alternative atlases, and best calibrated variant of the baseline as a function of cell types included in the set of expected cell types for computing TCS. Threshold for a minimum fraction is set to 1%. blue = all expected cell types; orange = excluding the 3 types expected in $\geq 14/16$ tissues (Blood-Mono+Macro, Blood-T, Endothel); green = excluding the 7 expected in $\geq 8/16$ (those three plus Blood-B, Blood-Granul, Dermal-Fibro, Smooth-Musc)

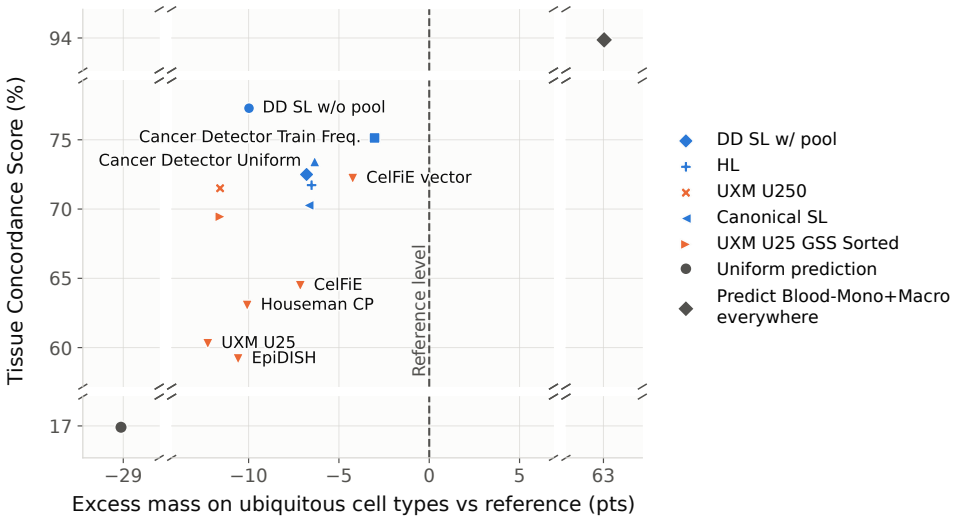


Figure S16: Coordinate plot projecting selected attained TCS vs mass allocated to the cell types present in at least 14/16 tissues. "uniform over all cell types" - TCS for a method that predicts cell type proportions uniformly. Reference levele line corresponds to the average total fraction of Top-3 ubiquitous cell types in Tabula Sapiens reference. Baselines and their variants are colored in orange. *Syto* methods are colored in blue. Uniform prediction results and results steming from predicting Blood-Mono+Macro everywhere are in grey and added for perspective.

Table S27: Dominant cell type called per tissue by each labelling scheme, against the Tabula Sapiens reference. Each entry is the cell type that most configurations in the group call dominant, with the number that agree. Shading: matches Tabula Sapiens; ∗ differs, but the called type is in the expected set, so TCS is unaffected; † differs and lies outside the expected set.

| Tissue | Tabula Sapiens | Baseline | Cancer Detector | HL | Canonical SL | DD SL w/o pool | DD SL w/ pool |
|---|---|---|---|---|---|---|---|
| Adipose Tissue | Dermal-Fibro | ∗ Adipocytes (4/4) | ∗ Endothel (37/40) | ∗ Adipocytes (24/40) | ∗ Adipocytes (28/40) | ∗ Endothel (47/60) | ∗ Adipocytes (40/60) |
| Bladder | Bladder-Ep | Bladder-Ep (3/4) | Bladder-Ep (40/40) | Bladder-Ep (24/40) | Bladder-Ep (20/40) | Bladder-Ep (51/60) | Bladder-Ep (46/60) |
| Blood Vessel | Dermal-Fibro | † Heart-Fibro (2/4) | ∗ Endothel (28/40) | ∗ Blood-Mono+Macro (19/40) | † Heart-Fibro (20/40) | ∗ Endothel (30/60) | ∗ Blood-Mono+Macro (24/60) |
| Heart | Endothel | Endothel (4/4) | Endothel (40/40) | Endothel (21/40) | Endothel (34/40) | Endothel (60/60) | Endothel (59/60) |
| Kidney | Kidney-Ep | Kidney-Ep (4/4) | Kidney-Ep (40/40) | Kidney-Ep (36/40) | Kidney-Ep (40/40) | Kidney-Ep (60/60) | Kidney-Ep (60/60) |
| Liver | Liver-Hep | Liver-Hep (4/4) | Liver-Hep (39/40) | Liver-Hep (40/40) | Liver-Hep (40/40) | Liver-Hep (60/60) | Liver-Hep (57/60) |
| Lung | Blood-Mono+Macro | ∗ Endothel (4/4) | ∗ Endothel (39/40) | † Fallopian-Ep (13/40) | ∗ Endothel (24/40) | ∗ Endothel (56/60) | ∗ Endothel (42/60) |
| Ovary | Ovary-Ep | ∗ Smooth-Musc (2/4) | ∗ Smooth-Musc (33/40) | ∗ Smooth-Musc (39/40) | ∗ Smooth-Musc (40/40) | ∗ Smooth-Musc (49/60) | ∗ Smooth-Musc (60/60) |
| Pancreas | Pancreas-Acinar | Pancreas-Acinar (4/4) | Pancreas-Acinar (40/40) | Pancreas-Acinar (38/40) | Pancreas-Acinar (40/40) | Pancreas-Acinar (60/60) | Pancreas-Acinar (60/60) |
| Prostate | Prostate-Ep | Prostate-Ep (4/4) | Prostate-Ep (40/40) | Prostate-Ep (40/40) | Prostate-Ep (40/40) | Prostate-Ep (60/60) | Prostate-Ep (60/60) |
| Salivary Gland | Head-Neck-Ep | Head-Neck-Ep (4/4) | ∗ Endothel (16/40) | † Fallopian-Ep (13/40) | † Adipocytes (16/40) | ∗ Endothel (39/60) | Head-Neck-Ep (35/60) |
| Skin | Blood-T | † Epid-Kerat (4/4) | † Epid-Kerat (40/40) | † Epid-Kerat (39/40) | † Epid-Kerat (40/40) | † Epid-Kerat (60/60) | † Epid-Kerat (60/60) |
| Small Intestine | Small-Int-Ep | ∗ Blood-B (4/4) | ∗ Blood-B (40/40) | ∗ Blood-B (33/40) | ∗ Blood-B (36/40) | ∗ Blood-B (60/60) | ∗ Blood-B (60/60) |
| Spleen | Blood-B | ∗ Endothel (3/4) | Blood-B (25/40) | ∗ Endothel (16/40) | Blood-B (24/40) | Blood-B (31/60) | Blood-B (41/60) |
| Stomach | Blood-Mono+Macro | ∗ Gastric-Ep (4/4) | ∗ Gastric-Ep (40/40) | ∗ Gastric-Ep (36/40) | ∗ Gastric-Ep (36/40) | ∗ Gastric-Ep (60/60) | ∗ Gastric-Ep (60/60) |
| Uterus | Dermal-Fibro | ∗ Endothel (2/4) | † Adipocytes (31/40) | † Adipocytes (39/40) | † Adipocytes (32/40) | † Adipocytes (47/60) | † Adipocytes (46/60) |

choice for *CancerDetector* and *EpiDISH* and *Houseman CP* better baselines than *CelFiE*. Focusing on cell types present in no more than seven tissues perturbs ranking, positioning *vector*-calibrated *CelFiE* as first and ranking uncalibrated *CelFiE* above all other baselines again and slightly above *DD SL w/ pool·MethylBERT·SWN·simplex* and *HL·Dismir·NNLS·vector*, still below other *Syto* variants.

Without access to the ground truth, the failure mode (3) cannot be fully excluded. Nonetheless, based on collected evidence, it is possibly influencing the ranking within *Syto* methods and is much less likely to affect conclusions for *Syto* advantage over examined baselines.

To screen for failure modes (1) and (2), for each method we selected the cell type with the highest predicted proportion per tissue and compared it with the expected dominant cell type in the reference. We then computed within-group agreement for each group of methods by counting how many methods allocated the highest proportion to the same cell type. The results are summarized in Table S27. Finally, we looked at each individual predicted cell type proportion in comparison to the Tabula Sapiens reference and computed method group-level means and standard deviations for each tissue cell-type combination. We then projected them onto 2D scatter to see what biases are shared and differ between different method groups (see Figure S17).

First and foremost, a clear case of failure mode (2) emerged in *skin* tissue recovered proportions. *Epidermis-Keratinocytes* constitute less than 1% of skin, slightly above 0.5% according to the Tabula

Table S28: Best *Syto* configuration per labelling scheme and the uncalibrated baselines, with and without ovary. Tissue Concordance Score (%), averaged within each tissue and then across tissues. Subscript 16 is the full set; subscript 15 excludes ovary. Configurations are those selected as best per labelling scheme on all 16 tissues, so the rows match the figures. Ranks run over all 378 (360 variants of *Syto* + 4 uncalibrated baselines, 12 calibrated baselines variants, *UXM U25 GSS Sorted* and *UXM U250*) methods; $\Delta\text{Rank} > 0$ means the method ranks higher once ovary is removed.

| Method | $\text{TCS}_{16}$ | $\text{TCS}_{15}$ | $\text{Rank}_{16}$ | $\text{Rank}_{15}$ | $\Delta\text{Rank}$ |
|---|---|---|---|---|---|
| *Best configuration per labelling scheme* | | | | | |
| DD SL w/o pool·Lookup·NNLS·vector | 77.28 | 76.59 | 1 | 1 | 0 |
| Cancer Detector Train Freq.·XGB·vector | 75.13 | 75.43 | 4 | 4 | 0 |
| Cancer Detector Uniform·XGB·vector | 73.43 | 73.70 | 7 | 8 | $-1$ |
| DD SL w/ pool·MethylBERT·SWN·simplex | 72.50 | 72.42 | 10 | 16 | $-6$ |
| HL·Dismir·NNLS·vector | 71.73 | 71.21 | 18 | 35 | $-17$ |
| Canonical SL·Dismir·NNLS·vector | 70.26 | 70.12 | 52 | 58 | $-6$ |
| *Uncalibrated baselines, CelFiE vector, and UXM variants using alternative atlases* | | | | | |
| CelFiE vector | 72.20 | 74.51 | 13 | 7 | $+6$ |
| UXM U250 | 71.50 | 72.83 | 20 | 11 | $+9$ |
| UXM U25 GSS Sorted | 69.45 | 70.19 | 72 | 54 | $+18$ |
| CelFiE | 64.47 | 67.02 | 210 | 150 | $+60$ |
| Houseman CP | 63.04 | 65.33 | 231 | 210 | $+21$ |
| UXM U25 | 60.28 | 63.01 | 261 | 237 | $+24$ |
| EpiDISH | 59.18 | 61.86 | 268 | 247 | $+21$ |

Sapiens reference, hence never making it into the set of expected cell types for this tissue. All examined methods, baselines and *Syto* variants alike, predict an average fraction of *Epidermis-Keratinocytes* in skin tissue samples $\approx 40\%$ with variance depending on the method group. A double-digit fraction of *Epidermis-Keratinocytes* in skin is expected. Therefore, across our benchmark, all methods have approximately the same penalty for this tissue, lowering the global TCS score but not affecting final ranking. Another interesting case is a fraction of *Adipocytes* in the uterus. The spread between method groups is higher than in the previous example; however, *Adipocytes* receives roughly 20% to 40% on average in each method group, highlighting another possible case of failure mode (2).

Moving on to the examples where TCS correctly penalized all methods in the benchmark, we can highlight failure to detect *Ovary Epithelial* in ovarian tissue and *Small Intestinal Epithelial* in small intestine samples. All methods recover near-zero fractions for these cases, well below the reference, with minimal variance within method groups. However, *Syto* allocated to *Smooth Muscle* in ovarian tissue roughly from 18% to 30% on average per method group, with proportions as high as 75.9% for *DD SL w/ pool·Lookup·NNLS·vector* and reference value of 16.9%. In contrast, baseline methods put slightly less than the reference, between 8.6% and 14.4% depending on the method. Therefore, *Syto* gains a higher TCS score for overpredicting moderately present cell types, which can be considered as evidence for the failure mode (1). Notably, *UXM U250* and *UXM U25 GSS Sorted* allocate 46.5% and 56.5% to *Smooth Muscle* in this example, partially explaining their TCS superiority over *U25*-based baselines and also suggesting that the true fraction of *Smooth Muscle* might be higher than in the reference. Another potential example of overpredicting moderately present cell type is in recovered proportions of *Gastric Epithelial* in stomach samples, which is roughly 30% higher on average across all baselines and *Syto* variants. The exerted influence of this potential overprediction is, however, uniform.

Given that ovarian tissue favours *Syto* highly, we also recomputed TCS ranks using a set of 15 tissues, ovary excluded (Table S28). While it tightens the gap between the baselines and *Syto*, it does not change the relative positioning of *Syto* labeling schemes, nor does it change the fact that *Syto* achieves higher *TCS*. Overall, Data Driven Soft Labels with pooling agree with the reference most of the time with respect to the most dominant cell type (for 8 tissues out of 16). The Hard Labeling scheme produces the most discordant results (agrees on 6 tissues out of 16), and the rest of the methods, including baselines, are in between (agree on 7 tissues out of 16).

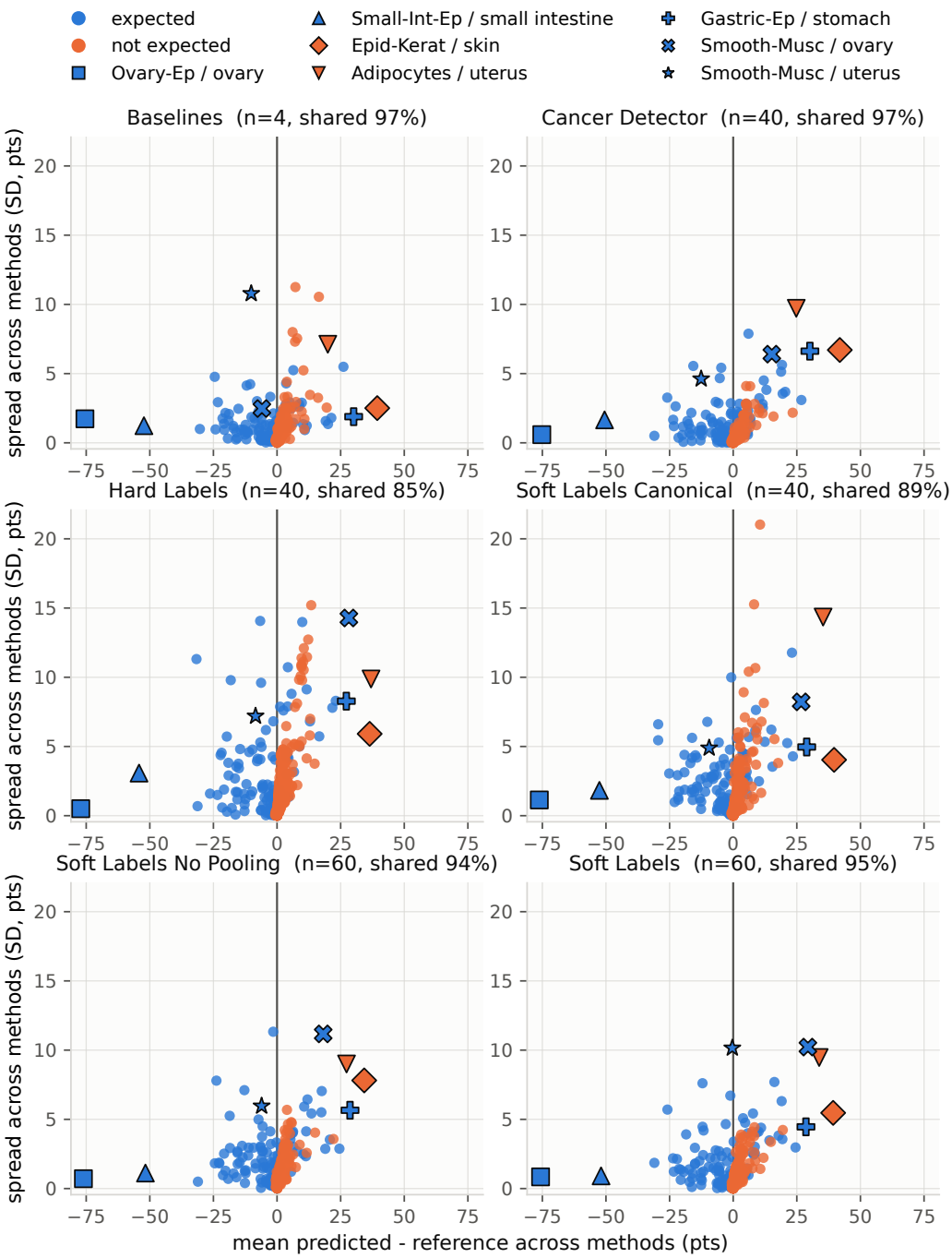


Figure S17: Re-distribution of mass compared to Tabula Sapiens reference per method group. Each dot represented the aggregated prediction per each combination of cell type and tissue. The values in parentheses indicate: n - number of methods in the group; shared - percentage of total variation relative to the reference that is shared across methods within one group. For Hard Labels Lookup classifier-based results were excluded. The feature selection scheme is *Top 156* across *Syto* methods.

## F Experiments compute resources

The majority of experiments were conducted on a single workstation and are fully reproducible on equivalent hardware. To accelerate batch jobs, pseudobulk generation and deconvolver fitting were additionally offloaded to a high-performance computing cluster; however, the primary workstation is sufficient to run the entire pipeline end-to-end.

**Primary workstation.** An HP Z2 Tower G9 equipped with an Intel Core i9-12900K (16 cores, 24 threads, up to 5.2 GHz), 64 GB DDR5 RAM, an NVIDIA RTX A4500 GPU (20 GB VRAM) and a 2 TB Samsung NVMe SSD, running Ubuntu 22.04 LTS. This machine was used for all stages of the pipeline: data preparation (restoring reads from `.pat` files and constructing train/validation/test splits), training and inference of read-level classifiers (*Dismir*, *MethylBERT*, *Lookup Classifier*), pseudobulk mixture generation, deconvolver fitting and inference, and linear calibrator fitting and inference.

**Secondary workstation.** A Dell Precision 7540 laptop equipped with an Intel Core i7-9850H (6 cores, 12 threads, up to 4.6 GHz), 32 GB DDR4 RAM, an NVIDIA Quadro T1000 Mobile GPU (4 GB VRAM) and a 512 GB Western Digital NVMe SSD, running Linux Mint 22.3 (Ubuntu 24.04 base). This machine was used in parallel to fit and evaluate the *CancerDetector* and *Lookup Classifier* baselines, along with their corresponding calibrators.

**HPC cluster.** Batch jobs for pseudobulk generation and deconvolver fitting were offloaded to an HPC cluster.

All details about the compute resources and wall-clock time for each stage of the pipeline are provided in the supplementary spreadsheet named ‘Compute Resources.ods‘.

## G Licensing

**Data.** This work does not produce new data assets. As stated in the main text, all the data used is publicly available, and is downloadable from Gene Expression Omnibus (GEO), a public functional genomics data repository, using accession numbers: GSE186458 for WGBS data used for training and pseudobulk experiment; GSE233417 RRBS data for OOD experiment. There is no explicit Licensing Agreement under which this data is distributed, however *"Unless otherwise stated, documents and files on NCBI Web servers may be freely downloaded and reproduced."* (see the official disclaimer https://www.ncbi.nlm.nih.gov/geo/info/disclaimer.html). The reference cell type counts from Tabula Sapiens consortium were accessed via its public API and the access is compliant with data release policy (https://tabula-sapiens.sf.czbiohub.org/whereisthedata).

**Code.** In this work we have reused or repurposed the next software:

- *MethylBERT*: Distibuted under MIT License (https://github.com/CompEpigen/methylbert/blob/main/LICENSE).
- *UXM*: Distributed under its own License (https://github.com/nloyfer/*UXM*_deconv/blob/main/LICENSE.m), allows non-commercial, research use.
- wgbs_tools: Distributed under its own License (https://github.com/nloyfer/wgbs_tools/blob/master/LICENSE.md).
- *CelFiE*: Distibuted under GNU Affero General Public License v3.0 (https://github.com/christacaggiano/celfie/blob/master/LICENSE.txt).
- *EpiDISH*: Distibuted under GNU General Public License v2.0 (GPL-2) (https://bioconductor.posit.co/packages/3.20/bioc/html/EpiDISH.html).
- HuggingFace Transformers: Distributed under Apache License 2.0 (https://github.com/huggingface/transformers/blob/main/LICENSE).
- PyTorch: Distributed under its own License (https://github.com/pytorch/pytorch/blob/main/LICENSE).

We implemented *Dismir* and *CancerDetector* from scratch and did not reuse any existing code for these implementations.

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The paper's main contributions are enumerated at the end of the introduction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See the limitation paragraph in the main paper (Section 6).

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: The only theoretical result of the paper is the fact that normalized-count soft labels are the unique maximum likelihood estimates of the true conditional categorical distribution (cf. Section 3.2), as a direct application of a textbook result. The reference is provided alongside the result [36].

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We list all relevant details of the experimental setups, models architecture and models' hyperparameters in Section D, Section C and Section B.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code, models and processed data are released open-source as stated in Section A. As mentionned in Section 4.1, the source data used in both experiments is publicly available at GEO under accession numbers GSE186458 and GSE233417 respectively.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The construction of the data splits is described in Section D.1, the pseudobulk generation is described in Section D.2, the predictions aggregation is described in Section D.3, the deconvolution feature selection is described in Section D.4, the deconvolution metrics are described in Section D.5 and the combination of possible elements of *Syto* is described in Section D.6. The model hyperparameters are listed in Section B.8 and Section B.6.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: For the pseudobulk experiments, we report the 95% confidence interval over the followed metrics. The interval is computed using 10,000 bootstrap samples and the Bias Corrected accelerated bootstrap method [47]. For the OOD experiment, we report significance using Wilcoxon signed-rank test using p-value threshold of 0.05.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide the necessary compute resources and wall-clock times for all stages of the experiments in the supplementary spreadsheet named ‘Compute Resources.ods‘, as well as give a brief summary in Section F.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn’t make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The authors have reviewed the NeurIPS Code of Ethics and believe that the conducted research in the paper conforms, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We state potential positive socieatal impact in conclusions. We do not foresee any negative societal impact.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The licences of the codes and datasets used in the paper are listed in Section G. The original papers the produced the code packages and datasets are always cited.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer:[Yes]

    Justification: We released trained models, processed data, and accompanying documentation in the KU Leuven Research Data Repository, as stated in Section A. To facilitate access, we also released the same package as a mirror on the HuggingFace platform under a CC BY 4.0 license.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.