BAYESIAN IMBALANCED REGRESSION DEBIASING

Anonymous authors
Paper under double-blind review

ABSTRACT

Imbalanced regression, where the training data has an uneven distribution on its range, is widely encountered in the real world, *e.g.,* age estimation (uni-dimensional regression) and pose estimation (multi-dimensional regression). Compared to imbalanced and long-tailed classification, imbalanced regression has its unique challenges as the regression label space can be continuous, boundless, and high-dimensional. In this work, we present a principled framework, **Bayesian Posterior Debiasing (Bayesian-PD)**, for re-balancing the regression among frequent and rare observations. Our key insight is that a balanced posterior can be obtained by debiasing the conditional probability with a regression label space prior. Importantly, through a normalization reparameterization technique, we derive a general debiasing function between the empirical posterior and the balanced posterior without relying on task-specific assumptions. We show that the Bayesian-PD framework has multiple instantiations in both training and testing time, with either closed-form or numerical implementations. We further uncover that several existing methods in imbalanced classification/regression serve as special cases of our Bayesian-PD framework. Extensive experiments on both uni- and multi-dimensional regression benchmarks demonstrate the effectiveness of the Bayesian-PD framework on various real-world tasks. Notably, Bayesian-PD exhibits strong robustness to different skewness of the training distributions.

1 INTRODUCTION

Imbalanced regression is widely encountered in the real world. For example, in pose estimation, most of the poses in the training set center around a mean pose, while extreme poses like *hand-stand* have few training samples (Rong et al., 2020). Nonetheless, in pursuit of generalizability and fairness, the evaluation of algorithms often bases on a balanced metric or a balanced test set in practice. The train-test label distribution mismatch often leads to inferior performance on less observed labels (Buda et al., 2018; Liu et al., 2019; Gupta et al., 2019). Unlike imbalanced and long-tailed classification that has been widely discussed (Kang et al., 2020; Zhou et al., 2020; Wang et al., 2021b), imbalanced regression has been under-explored. Compared with imbalanced classification, imbalanced regression has several unique challenges: its label space could be continuous, boundless, and high-dimensional. Recent research (Yang et al., 2021; Steininger et al., 2021) makes progress on estimating the underlying training distributions by employing kernel density estimation (KDE). Yet, we lack an effective approach to leverage the estimated label distribution.

In this work, we present a principled framework, **Bayesian Posterior Debiasing (Bayesian-PD)**, to debias the imbalanced regression among frequent and rare observations. Our key insight is that a balanced posterior can be obtained by debiasing the conditional probability with a label space prior, through a normalization reparameterization technique. Essentially, we leverage the following Bayesian relation to translate between the balanced posterior $p_{\text{bal}}(y|x)$ and the training set posterior $p_{\text{train}}(y|x)$ with the regression space label prior $p_{\text{train}}(y)$:

$$p_{\text{bal}}(y|x) = \frac{p_{\text{train}}(y|x) \cdot p_{\text{train}}(y)^{-1}}{\mathbb{E}_{y' \sim p_{\text{train}}(y|x)}[p_{\text{train}}(y')^{-1}]}; \quad p_{\text{train}}(y|x) = \frac{p_{\text{bal}}(y|x) \cdot p_{\text{train}}(y)}{\mathbb{E}_{y' \sim p_{\text{bal}}(y|x)}[p_{\text{train}}(y')]}. \tag{1.1}$$

Eq. 1.1 includes a bi-directional relation: the first part of Eq. 1.1 allows us to first fit a train-set posterior, and then convert to a balanced posterior in a post-processing fashion; the second part of Eq. 1.1 enables us to parameterize the train-set posterior with a balanced posterior, so that we can directly obtain a balanced posterior via empirical risk minimization on the training set.

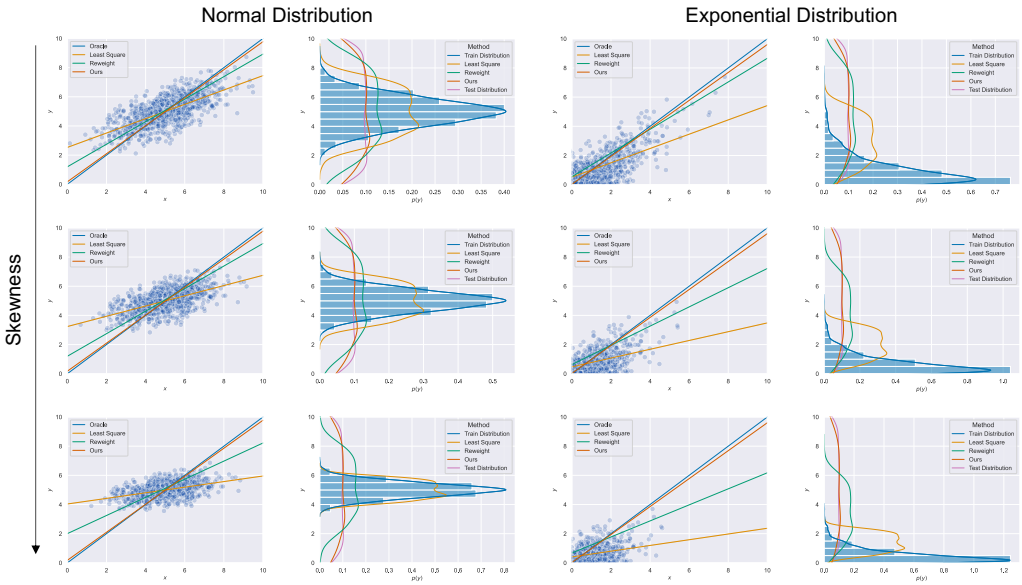

Figure 1: Comparison of Bayesian-PD and existing methods on a 1-D imbalanced linear regression synthetic benchmark. Different label distribution types (Normal & Exponential) and extents of distribution skewness are studied. We show the visualization of the regression results on the left and the marginal label distribution on a balanced test set on the right. Although reweighting (in green) is closer to the oracle (in blue) compared with least square (in yellow), it suffers larger error when the label distribution gets more skewed. In comparison, our method (in red), Bayesian-PD, makes the estimation closest to the oracle and has a uniform marginal label distribution on the test set. Most importantly, Bayesian-PD's performance is invariant to the skewness of the training distribution.

Bayesian-PD shows clear advantages over existing methods both theoretically and practically. As a motivating example, we compare Bayesian-PD with reweighting, a technique employed by state-of-the-art works (Yang et al., 2021; Steininger et al., 2021), in an 1-D linear regression synthetic benchmark shown in Fig. 1. Regressors trained with Bayesian-PD show a consistent performance that is invariant to the skewness of the training label distribution. On the contrary, reweighting shows limited effectiveness in debiasing and results in a significantly larger prediction error when the training label distribution gets more skewed. We show that the advantage of Bayesian-PD extends to nonlinear cases (Fig. 4) and high-dimensional cases (Fig. 3) as well.

We further demonstrate Bayesian-PD's empirical success on existing real-world benchmarks (Yang et al., 2021), including age estimation and depth estimation. Existing imbalanced regression benchmarks only include uni-dimensional label space. In this work, we propose a new multi-dimensional imbalanced regression benchmark on Imbalanced Human Mesh Recovery (IHMR) accompanied by a balanced evaluation metric. We show that Bayesian-PD delivers strong empirical results on both uni- and multi-dimensional benchmarks.

Note that similar Bayesian relations to Bayesian-PD has been discussed under the context of classifier readjustment in earlier literatures (Richard & Lippmann, 1991; Latinne et al., 2001). Recent works (Ren et al., 2020; Tian et al., 2020; Menon et al., 2021) further developed on it and verified its empirical effectiveness on the modern deep learning based classification. However, prior discussions are closely coupled with the pre-assumption on a discrete label space, and hence hard to generalize to the imbalanced regression. In this work, we use a normalization reparameterization trick to present the Bayesian relation in a more general form as Bayesian-PD. Classifier readjustment methods can be viewed as special cases of Bayesian-PD on discrete labels.

In summary, our contributions are three-fold: **1)** We propose a principled framework, Bayesian-PD, for debiasing the imbalanced regression. **2)** We show that Bayesian-PD has multiple instantiations in both training and testing time, with either closed-form or numerical implementations. **3)** Bayesian-PD achieves state-of-the-art results on uni- and multi-dimensional real-world benchmarks. We further demonstrate Bayesian-PD's robustness to different skewness of training distributions.

2 METHODOLOGY

2.1 PRELIMEARIES

We consider an observation space $X = \{x_1, x_2, ..., x_n\}$ and a label space Y, Y is either discrete $\{y_1, y_2, ..., y_k\}$ or contiuous $\mathbb{R}^m$.

Here, we follow the classic probablistic interpretation (McCullagh & Nelder, 1989) for categorical classification and continuous regression. Multi-class classification uses the Softmax function to convert the model output η into probability estimation: $p(y = i|x) = \exp(\eta_i)/\sum_{j=1}^{K}\exp(\eta_j)$, where K is the number of classes. Continuous value regression uses the Identity function to convert the model output y_{pred} to a Gaussian probability estimation:

$$p(y|x) = \mathcal{N}(y; y_{\text{pred}}, \sigma_{\text{pred}}^2\mathrm{I}), \tag{2.1}$$

where σ_{pred} is the scale of an i.i.d. error term $\epsilon \sim \mathcal{N}(0, \sigma_{\text{pred}}^2\mathrm{I})$. Note that the negative log likelihood $-\log p(y|x)$ is the standard Mean Square Error, where σ_{pred} is ignored for being a constant.

Under the imbalanced learning setting, the training set and test set have different label distributions. On the training set, labels are drawn from an imbalanced distribution $y \sim p_{\text{train}}(y)$. On the balanced test set, labels are drawn from a uniform distribution, $y \sim p_{\text{bal}}(y)$. Note that a balanced metric on an imbalanced test set can be effectively equivalent to a balanced test set (Menon et al., 2021). Invariant generative probability $p(x|y)$ is assumed. One may obtain an imbalanced posterior $p_{\text{train}}(y|x)$ by fitting on the training set. We are interested in knowing a balanced posterior $p_{\text{bal}}(y|x)$ that gives an unbiased estimate on the test set.

2.2 BAYESIAN POSTERIOR DEBIASING (BAYESIAN-PD)

By Bayes' Rule, we have

$$p_{\text{train}}(y|x) = \frac{p(x|y)\cdot p_{\text{train}}(y)}{p_{\text{train}}(x)}; \quad p_{\text{bal}}(y|x) = \frac{p(x|y)\cdot p_{\text{bal}}(y)}{p_{\text{bal}}(x)}. \tag{2.2}$$

By change of variables,

$$p_{\text{bal}}(y|x) = p_{\text{train}}(y|x)\cdot\frac{p_{\text{bal}}(y)}{p_{\text{train}}(y)}\cdot\frac{p_{\text{train}}(x)}{p_{\text{bal}}(x)}. \tag{2.3}$$

Eq. 2.3 is a well-known formula. Although the evidence ratio $p_{\text{train}}(x)/p_{\text{bal}}(x)$ is unknown, Eq. 2.3 is sufficient for posterior calibration on the categorical classification. By leveraging the fact that $\exp(\eta_i) \propto p(y|x)$ and $p_{\text{bal}}(y|x) \propto p_{\text{train}}(y|x)\cdot p_{\text{bal}}(y)/p_{\text{train}}(y)$, one may freely re-calibrate the posterior by manipulating with the Softmax logits $\exp(\eta)$, as detaily described in recent works (Ren et al., 2020; Tian et al., 2020; Menon et al., 2021). However, the aforementioned posterior calibration techniques on classification are not transferable to regression for being coupled with Softmax.

To bypass the unknown evidence ratio $p_{\text{train}}(x)/p_{\text{bal}}(x)$ and avoid coupling with at task-specific assumptions, we use a normalization reparameterization technique. Firstly, we normalize the posterior by its expected value of one:

$$p_{\text{bal}}(y|x) = \frac{p_{\text{bal}}(y|x)}{\mathbb{E}_{y'\sim p_{\text{bal}}(y|x)}[1]}; \quad p_{\text{train}}(y|x) = \frac{p_{\text{train}}(y|x)}{\mathbb{E}_{y'\sim p_{\text{train}}(y|x)}[1]} \tag{2.4}$$

Then we re-parameterize the posterior using Eq. 2.3, and have:

$$p_{\text{bal}}(y|x) = \frac{p_{\text{train}}(y|x)\cdot\frac{p_{\text{bal}}(y)}{p_{\text{train}}(y)}}{\mathbb{E}_{y'\sim p_{\text{train}}(y|x)}[\frac{p_{\text{bal}}(y')}{p_{\text{train}}(y')}]}; \quad p_{\text{train}}(y|x) = \frac{p_{\text{bal}}(y|x)\cdot\frac{p_{\text{train}}(y)}{p_{\text{bal}}(y)}}{\mathbb{E}_{y'\sim p_{\text{bal}}(y|x)}[\frac{p_{\text{train}}(y')}{p_{\text{bal}}(y')}]} \tag{2.5}$$

Detailed derivations can be found in Sect. A.1. We name the Bayesian relation described in Eq. 2.5 as Bayesian Posterior Debiasing (Bayesian-PD). Bayesian-PD is a principled framework that translates between an empirical posterior and a balanced posterior without relying on any task-specific assumptions. One may substitute $p_{\text{bal}}(y)$ with arbitrary test label distributions and Eq. 2.5 still holds.

Figure 2: Graphical model illustration for train-time and test-time debiasing, respectively. x, y are from train set and $\tilde{x}, \tilde{y}$ are from test set. ϕ and $\tilde{\phi}$ are parameters of generative distributions that generate x and $\tilde{x}$ respectively. θ is a learnable regressor. In the imbalanced regression setting, the label distribution $p(y|\phi)$ and $p(\tilde{y}|\tilde{\phi})$ are different and known. For train-time debiasing, the generative parameters are taken into account when estimating regressor θ. For test-time debiasing, the generative paramters are considered when predicting $\tilde{y}$.

Here, we adopt the balanced $p_{\text{bal}}(y)$ setting and treat $p_{\text{bal}}(y)$ as a constant. Eq. 2.5 can be thus simplified to Eq. 1.1. We refer Bayesian-PD to Eq. 1.1 in the following sections for brevity. Note that forms similar to Bayesian-PD have been discussed by earlier works (Latinne et al., 2001) under the context of classifier recalibration but not sufficiently investigated. This work exploits Bayesian-PD's generality. We uncover that several existing posterior calibration techniques serve as special cases of it. We further show how Bayesian-PD can shed light on imbalanced regression.

2.3 Instantiation for Imbalanced Classification

When instantiating for imbalanced classification, the expectation is written into summation, Eq. 1.1 becomes

$$p_{\text{bal}}(y|x) = \frac{p_{\text{train}}(y|x)/p_{\text{train}}(y)}{\sum_{y' \in Y} p_{\text{train}}(y'|x)/p_{\text{train}}(y')}; \quad p_{\text{train}}(y|x) = \frac{p_{\text{bal}}(y|x) \cdot p_{\text{train}}(y)}{\sum_{y' \in Y} p_{\text{bal}}(y'|x) \cdot p_{\text{train}}(y')}. \tag{2.6}$$

This equation can be applied after mapping the model output into a probability distribution. Alternatively, one may incorporate the equation into the mapping functions to simplify the computation. When plugging Softmax into Eq. 2.6, we achieve the same form as posterior calibration techniques described in recent research listed in Tab. 5.

2.4 Instantiation for Imbalanced Regression

When instantiating for imbalanced regression, the expectation is written into integral, Eq. 1.1 becomes

$$p_{\text{bal}}(y|x) = \frac{p_{\text{train}}(y|x) \cdot p_{\text{train}}(y)^{-1}}{\int_Y p_{\text{train}}(y'|x) \cdot p_{\text{train}}(y')^{-1} dy'}; \quad p_{\text{train}}(y|x) = \frac{p_{\text{bal}}(y|x) \cdot p_{\text{train}}(y)}{\int_Y p_{\text{bal}}(y'|x) \cdot p_{\text{train}}(y') dy'}, \tag{2.7}$$

Similar to the classification counterpart, Eq. 2.7 allows to debias the imbalanced regression in both training and testing time. For train-time debiasing, we train the regressor when taking the label distribution shift into account using the second part of Eq. 2.7. For test-time debiasing, we train the regressor normally and explicitly addressing the label distribution shift during inference using the first part of Eq. 2.7. Graphical models illustrating train-time and test-time debiasing are shown in Fig. 2. However, in practice, the integral in Eq. 2.7 introduce difficulty in implementation. We discuss a few feasible implementation variants in the following sections.

2.4.1 Train-time (Closed-form)

In this section, we aim to find a closed-form expression for the integral $\int_Y p_{\text{bal}}(y|x) \cdot p_{\text{train}}(y) dy$. The main challenge is how to express $p_{\text{train}}(y)$ to make the integral tractable. Here, we discuss a viable option, which is to express $p_{\text{train}}(y)$ as a Gaussian Mixture Model (GMM).

GMM-based Analytical Integration (GAI). Recall that the posterior is modeled as Gaussian in regression as aforementioned in Sect. 2.1: $p(y|x) = \mathcal{N}(y; y_{\text{pred}}, \sigma^2_{\text{pred}}\text{I})$. The advantage of employing GMM is the fact that the product of two Gaussians is an unnormalized Gaussian. Concretely, let us have $p_{\text{train}}(y)$ expressed by a Gaussian Mixture:

$$p_{\text{train}}(y) = \sum_{i=1}^{K} \phi_i \mathcal{N}(\mu_i, \Sigma_i), \tag{2.8}$$

where K is the number of Gaussian components, ϕ, μ, Σ are the weights, means and covariances of the GMM. Since the product of two Gaussians is an unnormalized Gaussian, we have:

$$\int_Y \sum_{i=1}^{K} \phi_i \mathcal{N}(\mu_i, \Sigma_i) \cdot \mathcal{N}(y_{\text{pred}}, \sigma_{\text{pred}}^2 \mathbf{I}) dy = \sum_{i=1}^{K} \phi_i S_i \int_Y \mathcal{N}(\tilde{\mu}_i, \tilde{\Sigma}_i) dy. \tag{2.9}$$

where $S_i, \tilde{\mu}_i, \tilde{\Sigma}_i$ are the norm, mean, covariance of the new Gaussian. Now, the integral can be trivially solved. We leave the detailed derivation in Sect. A.2. The final loss form is:

$$L = -\log \mathcal{N}(y_{\text{target}}; y_{\text{pred}}, \sigma_{\text{pred}}^2 \mathbf{I}) + \log \sum_{i=1}^{K} \phi_i \cdot \mathcal{N}(y_{\text{pred}}; \mu_i, \Sigma_i + \sigma_{\text{pred}}^2 \mathbf{I}). \tag{2.10}$$

2.4.2 Train-time (Numerical)

The closed-form solution above imposes a constraint on the modeling of $p_{\text{train}}(y)$. However, in modern deep learning tasks, $p_{\text{train}}(y)$ could be very high-dimensional and has a complex underlying distribution. With the constraint on the distribution modeling, analytically expressing $p_{\text{train}}(y)$ could be challenging. Therefore, we discuss a few numerical approaches, which could be more generally applicable to all types of label data but could bear a larger variance in optimization. In essence, we use Monte Carlo Method (MCM) to approximate $p(y)$:

$$\int_Y p_{\text{bal}}(y|x) \cdot p_{\text{train}}(y) dy = \mathbb{E}_{y' \sim p_{\text{train}}(y)}[p_{\text{bal}}(y'|x)] \approx \frac{1}{N} \sum_{i=1}^{N} p_{\text{bal}}(y = y'_{(i)}|x). \tag{2.11}$$

Batch-based Monte-Carlo (BMC). This variant requires no prior knowledge on $p_{\text{train}}(y)$. It treats all samples in a training batch as random samples from $p_{\text{train}}(y)$. For a training batch $\{y_{(1)}, y_{(2)}, ...y_{(N)}\}$, the debiased loss will be:

$$L = -\log \mathcal{N}(y_{\text{target}}; y_{\text{pred}}, \sigma_{\text{pred}}^2 \mathbf{I}) + \log \sum_{i=1}^{N} \mathcal{N}(y_{(i)}; y_{\text{pred}}, \sigma_{\text{pred}}^2 \mathbf{I}). \tag{2.12}$$

This variant takes minimal implementation efforts. One may also consider increasing the MCM sample number by using a dedicated sampler for MCM. Furthermore, the loss function in Eq. 2.12 can also be rewritten in a temperature-like way:

$$L = -\log \frac{\exp(-\|y_{\text{pred}} - y_{\text{target}}\|_2^2 / \tau)}{\sum_{i=1}^{N} \exp(-\|y_{\text{pred}} - y_{(i)}\|_2^2 / \tau)}, \tag{2.13}$$

where $\tau = 2\sigma_{\text{pred}}^2$ is a temperature coefficient. Eq. 2.13 shows interesting similarity to Softmax with temperature.

Bin-based Numerical Integration (BNI). Although the "bin" based idea mainly applies to univariate label space, it allows us to leverage recent progress on estimating label densities using KDE (Yang et al., 2021; Steininger et al., 2021). These prior works first divide the label space into evenly distributed bins, then use KDE to estimate the $p_{\text{train}}(y)$ at the bin centers. We may directly use their results to make a numerical integration. For B bin centers $\{y_{(1)}, y_{(2)}, ..., y_{(B)}\}$, the loss is:

$$L = -\log \mathcal{N}(y_{\text{target}}; y_{\text{pred}}, \sigma_{\text{pred}}^2 \mathbf{I}) + \log \sum_{i=1}^{B} p_{\text{train}}(y = y_{(i)}) \cdot \mathcal{N}(y_{(i)}; y_{\text{pred}}, \sigma_{\text{pred}}^2 \mathbf{I}). \tag{2.14}$$

2.4.3 Finding optimal noise scale

Unlike the standard MSE loss, the noise scale σ_{pred} makes a difference in our proposed method. Locating an optimal noise scale is thus important. In fact, σ_{pred} can be jointly optimized with y_{pred} during model training when using the aforementioned losses. That is, we can obtain near-optimal σ_{pred} by simply setting σ_{pred} as a learnable parameter and no additional hyper-parameter tuning is required. A comparison between using the ground truth noise scale and using the jointly learned σ_{pred} is shown in Tab. 1. We adopt the joint optimization paradigm in all empirical analyses unless specified. Note that a hyper-parameter search will also be affordable given that σ_{pred} is defined in $\mathbb{R}^+$ and bounded by train-time and test-time MSEs.

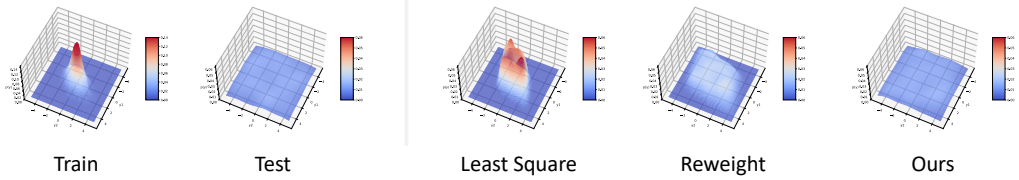

Figure 3: Comparison of marginal label distributions on 2D linear regression. Least square and reweighting show visible bias towards the high-frequency area around the center. In comparison, Bayesian-PD achieves the closest marginal label distribution to the uniform test distribution.

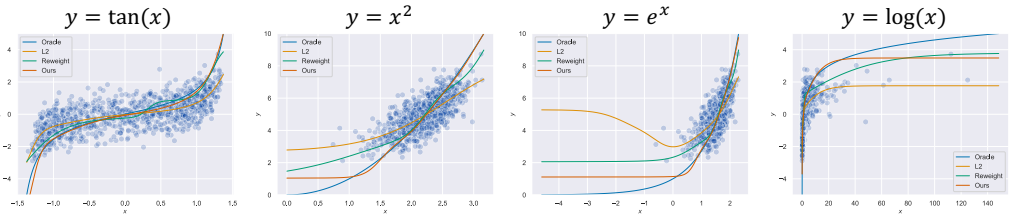

Figure 4: Qualitative comparison for nonlinear regression. Four nonlinear functions are studied. Bayesian-PD (in red) gives the closest estimation to the oracle (in blue).

2.4.4 TEST-TIME

The advantage of test-time debiasing is that it does not require retraining the model. Therefore, the post-hoc classifier re-calibration method has received wide discussion in imbalanced classification research (Richard & Lippmann, 1991; Latinne et al., 2001; Guo et al., 2017; Buda et al., 2018).

Test-Time Adjustment (TTA). Here, we describe a test-time adjustment method for the imbalanced regression. We uniformly select N probes in the label space, $\{y_{(1)}, y_{(2)}, ..., y_{(N)}\}$. Then we select the probe with the highest balanced posterior probability:

$$y_{\text{pred}} = y_{(i^*)}; \quad i^* = \text{argmax}_{1 \le i \le N} p_{\text{train}}(y = y_{(i)}|x)/p_{\text{train}}(y = y_{(i)}). \tag{2.15}$$

where $p_{\text{train}}(y)$ can be approximated by GMM or KDE as mentioned above. Note that, unlike train-time σ_{pred} optimization, the optimal σ_{pred} in test-time debiasing has to be tuned as a hyper-parameter since models have an imperfect estimation of $p_{\text{train}}(y|x)$ in practice. Detailed discussions can be found in classifier re-calibration literatures (Tian et al., 2020; Menon et al., 2021). Moreover, the number of probes grows exponentially with the number of dimensions, which makes the computational cost prohibitive for high-dimensional label spaces.

3 EXPERIMENTS

3.1 SYNTHETIC BENCHMARKS

We construct a simple one-dimensional linear imbalanced regression dataset, with the training label distribution being normal or exponential and skewed to various extents. We train a one-layer linear regressor on the imbalanced training set and test on a uniform test set with no additive noise. We compare three types of regressors: a least-square estimator, a linear regressor inversely reweighted by the true $p(y)$ as described in (Yang et al., 2021), and Bayesian-PD's closed-form variant GAI with true noise scale. We show the visualized results on Fig. 1. We observe that the reweighted regressor shows increasingly larger error when the training distribution becomes more skewed. In comparison, Bayesian-PD gives an unbiased estimation that is robust to different levels of skewness.

We further compare the three methods on a two-dimensional regression. The training label distribution is set as a Multivariate Normal (MVN) distribution. We visualize the marginal label distributions in Fig. 3, where Bayesian-PD achieves a marginal label distribution closest to uniform. For nonlinear regressions, Bayesian-PD achieves a consistent debiasing effectiveness as well, as shown in Fig. 4. We provide another experiment on random seeds in Appendix C to demonstrate Bayesian-PD's robustness to noise.

Despite recent works (Yang et al., 2021; Steininger et al., 2021) focusing on estimating the training label distribution, our synthetic benchmark shows that the bottleneck for existing techniques is with

Table 1: Quantitative results for the case study. †: True noise scale used. For each type of distribution, we evaluate three extents of skewness: Low, Moderate, and High. Best results are bolded.

	Normal			Exponential			MVN		
Method	High	Mod.	Low	High	Mod.	Low	High	Mod.	Low
MSE	5.521	3.275	1.936	18.61	13.14	6.038	5.522	3.809	2.570
Reweight	1.399	0.336	0.092	4.676	1.336	0.128	3.310	1.758	1.001
Ours (GAI)†	**0.031**	**0.001**	0.001	**0.001**	0.002	0.004	**0.122**	0.031	0.011
Ours (BMC)†	0.043	0.004	**0.000**	0.002	**0.000**	**0.000**	0.126	0.033	0.011
Ours (GAI)	0.089	0.008	0.005	0.130	0.082	0.023	0.184	**0.021**	**0.006**
Ours (BMC)	0.141	0.060	0.030	0.122	0.104	0.034	0.142	0.025	0.011

Table 2: Comparison with SOTAs on IMDB-WIKI-DIR. †: MAE metric reported in Yang et al. (2021). Best results are bolded.

	bMAE↓				MAE↓			
Method	All	Many	Med.	Few	All	Many	Med.	Few
Vanilla†	13.92	7.32	15.93	32.78	8.06	7.23	15.12	26.33
RRT†	13.12	**7.27**	14.03	30.48	7.81	**7.07**	14.06	25.13
RRT+LDS†	13.09	7.30	14.05	30.26	**7.79**	7.08	13.76	24.64
Ours (TTA)	13.17	7.63	13.25	30.26	8.13	7.56	12.63	23.79
Ours (BMC)	12.69	7.59	12.90	28.28	8.08	7.52	12.47	23.29
Ours (GAI)	**12.66**	7.65	**12.68**	**28.14**	8.12	7.58	**12.27**	**23.05**

reweighting. Even given the true label distribution, reweighting fails to eliminate the bias in all settings. In comparison, our proposed Bayesian-PD is robust to different skewness of the training distribution and noise, meanwhile applicable to nonlinear and multi-dimensional regressions.

We provide quantitative results for the above described synthetic benchmark in Tab. 1, where we compare different implementation variants and choices of noise scale as well. Tab. 1 shows that the numerical variant achieves comparable results with the closed-form version. Moreover, the jointly-optimized noise scale achieves near-optimal results in most cases.

3.2 Real-World Benchmarks

3.2.1 Datasets & Settings

Age & Depth Estimation. We select two representative tasks from Yang et al. (2021)'s DIR benchmark. We estimate ages from face images on the IMDB-WIKI-DIR dataset and estimate depth maps from images of indoor scenes on the NYUD2-DIR dataset.

Imbalanced Human Mesh Recovery (IHMR). IHMR is a new, multi-dimensional imbalanced regression benchmark. We estimate human meshes from images, where the mesh is represented by a parametric human model known as SMPL (Loper et al., 2015). Typically, SMPL model has two parameters: $\theta \in \mathbb{R}^{24\times 3}$ represents the rotation of 23 body joints and 1 global orientation and $\beta \in \mathbb{R}^{10}$ represents the 10 PCA components for body shape. Therefore, the label space of IHMR is multi-dimensional. Aligned with recent works (Rong et al., 2020), we observe that the distribution of human meshes is long-tailed. We show a visualization of training distribution in Fig. 8. Following Kolotouros et al. (2019), we train on a combination of 3D and 2D human datasets and test on an in-the-wild 3D dataset. Detailed settings can be found in Appendix B.

3.2.2 Evaluation Metrics

Yang et al. (2021) uses primarily overall metric, *e.g.*, Mean Absolute Error (MAE) to report the performance on the benchmark. This is on the assumption that the test dataset is perfectly balanced. However, we observe visible tails in IMDB-WIKI-DIR test sets as shown in Fig. 7. To avoid overlooking the performance on the tail classes, we follow the idea of balanced metrics (Brodersen et al., 2010), and divide the label space into a finite number of even sub-regions, compute the average in-

Table 3: Comparison with SOTAs on NYUD2-DIR. †: reported in Yang et al. (2021).

Method	RMSE↓				δ_1 ↑			
	All	Many	Med.	Few	All	Many	Med.	Few
Vanilla†	1.477	**0.591**	0.952	2.123	0.677	**0.777**	0.693	0.570
Vanilla + LDS†	1.387	0.671	0.913	1.954	0.672	0.701	0.706	0.630
Ours (TTA)	1.267	0.737	1.069	**1.688**	0.698	0.661	0.693	**0.736**
Ours (BNI)	1.283	0.787	**0.870**	1.736	0.694	0.622	**0.806**	0.723
Ours (GAI)	**1.251**	0.692	0.959	1.703	**0.702**	0.676	0.734	0.715

Table 4: Effectiveness on Imbalanced Human Mesh Recovery. †: reported in Rong et al. (2020). SPIN-RT: keep the SPIN's feature extractor fixed and retrain the last linear regression layers.

Method	bMPVPE↓			bMPJPE↓			bPA-MPJPE↓		
	All	10%	5%	All	10%	5%	All	10%	5%
SPIN†	-	130.0	130.6	-	-	-	-	-	-
PM-Net†	-	124.9	**126.4**	-	-	-	-	-	-
SPIN-RT	116.1	127.0	130.5	99.58	113.5	114.5	66.53	77.71	76.66
Ours (BMC)	113.9	128.6	129.6	97.87	113.7	113.0	65.90	77.73	76.35
Ours (GAI)	**112.7**	**122.9**	128.1	**96.70**	**108.8**	**111.9**	**64.69**	**74.04**	**74.35**

side the sub-regions, and take the mean overall sub-regions. We name it "balanced-" ("b-") metric, *e.g.*, bMAE.

Age & Depth Estimation. We primarily report bMAE on IMDB-WIKI-DIR. NYUD2-DIR's test set is balanced, we follow Yang et al. (2021) and report RMSE.

Imbalanced Human Mesh Recovery. We propose to use balanced metrics on HMR. We evenly divide the label space into 100 sub-regions according to their vertex-based distances to the mean parameter, and compute balanced metrics as described above. Following Rong et al. (2020), we primarily report balanced mean per-vertex position error (bMPVPE). We also report balanced mean per-joint position error (bMPJPE) and balanced Procrustes-aligned mean per joint position error (bPA-MPJPE). We include the "tail 5%" metric and the "tail 10%" metric to show performance on extreme poses as well.

3.2.3 Comparison Results

Tab. 2 shows a comparison with state-of-the-art (SOTA) methods on age estimation. Regressor Re-Training (RRT) (Yang et al., 2021) first trains the feature extractor normally and retrain the last linear layer using inverse re-weighting. RRT+LDS is an improved version of RRT, where training label distribution is estimated using Label Distribution Smoothing (Yang et al., 2021). We do not include Feature Distribution Smoothing (FDS) in the comparison since it improves the feature learning and should be complementary to our method. Without FDS, RRT and RRT+LDS are the best performing methods on IMDB-WIKI-DIR in Yang et al. (2021)'s benchmark. Our training-time variants substantially outperform the previous method. Notably, the BMC variant outperforms SOTAs with a large margin without relying on the pre-processed training label distribution. The test-time implementation also achieves comparable results to SOTA with a rough hyper-parameter tuning. We further analyze the bMAE gain in Fig. 5 and observe an effective trade-off between frequent labels and rare labels towards a balanced estimation.

Tab. 3 shows comparison with SOTA on depth estimation. Note that depth map has an inter-pixel dependency, the pixel-wise error σ_{pred} can be under-estimated and the BMC can give an inaccurate estimation to $p_{\text{train}}(y)$. We set a fixed σ_{pred} to 1, and use BNI for numerical variant evaluation. Compared with the SOTA, both train-time and test-time implementation achieve clear improvements.

Tab. 4 shows a comparison between Bayesian-PD and existing HMR methods. Bayesian-PD outperforms the baseline by a large margin on the main metric bMPVPE (-3.4). We show qualitative comparison in Appendix F. PM-Net (Rong et al., 2020) achieves better results on tail-5% bMPVPE, by designing prototypes and adaptively selecting them as the initialization for SMPL regression. PM-Net improves the regression initialization and should be complementary to our method.

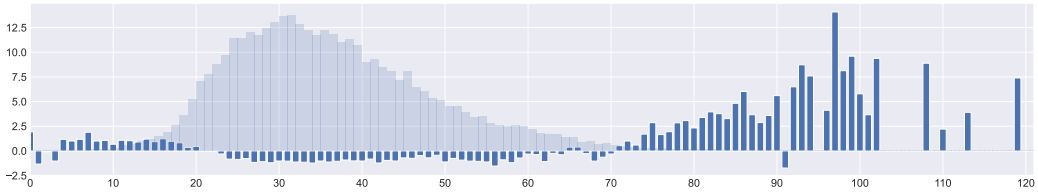

Figure 5: Bayesian-PD's bMAE gain over the baseline. The light blue area in the background shows the training label histogram of IMDB-WIKI-DIR. Bayesian-PD improves the performance on tail labels (age < 20 and > 70) substantially.

Table 5: Summary of variants of posterior calibration in imbalanced classification. τ is a scaling factor to address imperfect model learning (Tian et al., 2020; Menon et al., 2021), which equals one ideally. Listed methods serve as special cases of Bayesian-PD.

Method	Form	Reference
Train-time	$p_{\text{train}}(y\|x) = \frac{\exp(\eta_i) \cdot p_{\text{train}}(y)}{\sum_{y' \in Y} \exp(\eta_j) \cdot p_{\text{train}}(y)}$	Balanced Softmax (Ren et al., 2020) LA Loss (Menon et al., 2021) Seesaw Loss (Wang et al., 2021a)
Test-time	$p_{\text{bal}}(y\|x) = \frac{\exp(\eta_i)/p_{\text{train}}(y)^{\tau}}{\sum_{y' \in Y} \exp(\eta_j)/p_{\text{train}}(y')^{\tau}}$	UNO-IC (Tian et al., 2020) LA Post-hoc (Menon et al., 2021)

4 Related Works

Imbalanced & Long-Tailed Classification. Many techniques have been explored for imbalanced & long-tailed classification, for example, resampling (Chawla et al., 2002; He & Garcia, 2009; Kim et al., 2020; Chu et al., 2020) and reweighting (Huang et al., 2016; Cui et al., 2019; Jamal et al., 2020; Cao et al., 2019). Here, we focus on the posterior calibration techniques, which are the most relevant to this work. Recent works (Ren et al., 2020; Tian et al., 2020; Menon et al., 2021) show that modifying the logits in the mapping function, e.g., Softmax or Sigmoid, by an offset proportional to $\log p_{\text{train}}(y)$ gives the Bayes-optimal estimation of the posterior. The posterior calibration techniques can work as either a train-time loss function or a test-time adjustment. Wang et al. (2021a) further develops an online version that accumulates the statistics of label distribution during training instead of requiring statistics of all training labels ahead of time. We summarize different variants of the label posterior calibration in Tab. 5. In Sect. 2.3, we show that posterior calibration techniques can be viewed as special cases of our proposed Bayesian-PD framework.

Imbalanced Regression. Imbalanced regression is relatively under-explored. Earlier works (Torgo et al., 2013; Branco et al., 2017) focus on resampling and synthesizing new samples for rare labels. Further work (Branco et al., 2018) ensembles regressors trained under different resampling policies. Extending their method towards high-dimensional observations like images is non-trivial. Recent research (Yang et al., 2021; Steininger et al., 2021) proposes to estimate the empirical training distribution with KDE and then apply the standard reweighting technique. Yang et al. (2021) proposes a feature level smoothing as well, which is complementary to this work.

5 Conclusion & Future Works

In conclusion, we propose to debias the imbalanced regression from the Bayesian perspective. We propose a statistically principled framework for imbalanced regression, Bayesian-PD, which does not rely on task-specific assumptions and can be well-connected to existing classification debiasing literature. We further discuss various implementations of Bayesian-PD, including training-time and test-time debiasing, using either closed-form expression or numerical approximation. Bayesian-PD achieves SOTA on various uni- and multi-dimensional imbalanced regression benchmarks.

Future works may use Bayesian-PD as a bridge to introduce more approaches that are developed on the imbalanced classification to the imbalanced regression. For example, Eq. 2.13 can be viewed as Softmax with temperature. Margin-based methods might be introduced to adjust the pair-wise distances as well. One may also leverage deep generative models, *e.g.*, VAE (Kingma & Welling, 2014) and GAN (Goodfellow et al., 2014) , to better model $p_{\text{train}}(y)$.

REFERENCES

Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*, pp. 3686–3693, 2014.

Paula Branco, Luís Torgo, and Rita P Ribeiro. Smogn: a pre-processing approach for imbalanced regression. In *First international workshop on learning with imbalanced domains: Theory and applications*, pp. 36–50. PMLR, 2017.

Paula Branco, Luis Torgo, and Rita P Ribeiro. Rebagg: Resampled bagging for imbalanced regression. In *Second International Workshop on Learning with Imbalanced Domains: Theory and Applications*, pp. 67–81. PMLR, 2018.

Kay Henning Brodersen, Cheng Soon Ong, Klaas Enno Stephan, and Joachim M Buhmann. The balanced accuracy and its posterior distribution. In *2010 20th international conference on pattern recognition*, pp. 3121–3124. IEEE, 2010.

Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018.

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Aréchiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *NeurIPS*, pp. 1565–1576, 2019.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

Peng Chu, Xiao Bian, Shaopeng Liu, and Haibin Ling. Feature space augmentation for long-tailed data. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*, pp. 694–710. Springer, 2020.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9268–9277, 2019.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330. PMLR, 2017.

Agrim Gupta, Piotr Dollár, and Ross B. Girshick. Lvis: A dataset for large vocabulary instance segmentation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5351–5359, 2019.

Haibo He and Edwardo A Garcia. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284, 2009.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Junjie Hu, Mete Ozay, Yan Zhang, and Takayuki Okatani. Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1043–1051. IEEE, 2019.

Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5375–5384, 2016.

Catalin Ionescu, Fuxin Li, and Cristian Sminchisescu. Latent structured models for human pose estimation. In *2011 International Conference on Computer Vision*, pp. 2220–2227. IEEE, 2011.

Muhammad Abdullah Jamal, Matthew Brown, Ming-Hsuan Yang, Liqiang Wang, and Boqing Gong. Rethinking class-balanced methods for long-tailed visual recognition from a domain adaptation perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7610–7619, 2020.

Sam Johnson and Mark Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *BMVC*, pp. 1–11. British Machine Vision Association, 2010.

Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *ICLR*. OpenReview.net, 2020.

Jaehyung Kim, Jongheon Jeong, and Jinwoo Shin. M2m: Imbalanced classification via major-to-minor translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13896–13905, 2020.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2252–2261, 2019.

Patrice Latinne, Marco Saerens, and Christine Decaestecker. Adjusting the outputs of a classifier to new a priori probabilities may significantly improve classification accuracy: evidence from a multi-class problem in remote sensing. In *ICML*, volume 1, pp. 298–305. Citeseer, 2001.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X. Yu. Large-scale long-tailed recognition in an open world. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2532–2541, 2019.

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015.

Peter McCullagh and John A. Nelder. *Generalized Linear Models*. Springer, 1989.

Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *2017 international conference on 3D vision (3DV)*, pp. 506–516. IEEE, 2017.

Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *ICLR*. OpenReview.net, 2021.

Jiawei Ren, Cunjun Yu, Shunan Sheng, Xiao Ma, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Balanced meta-softmax for long-tailed visual recognition. In *NeurIPS*, 2020.

Michael D Richard and Richard P Lippmann. Neural network classifiers estimate bayesian a posteriori probabilities. *Neural computation*, 3(4):461–483, 1991.

Yu Rong, Ziwei Liu, and Chen Change Loy. Chasing the tail in monocular 3d human reconstruction with prototype memory. *arXiv preprint arXiv:2012.14739*, 2020.

Michael Steininger, Konstantin Kobs, Padraig Davidson, Anna Krause, and Andreas Hotho. Density-based weighting for imbalanced regression. *Machine Learning*, pp. 1–25, 2021.

Junjiao Tian, Yen-Cheng Liu, Nathaniel Glaser, Yen-Chang Hsu, and Zsolt Kira. Posterior recalibration for imbalanced datasets. In *NeurIPS*, 2020.

Luís Torgo, Rita P Ribeiro, Bernhard Pfahringer, and Paula Branco. Smote for regression. In *Portuguese conference on artificial intelligence*, pp. 378–389. Springer, 2013.

Timo von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 601–617, 2018.

Jiaqi Wang, Wenwei Zhang, Yuhang Zang, Yuhang Cao, Jiangmiao Pang, Tao Gong, Kai Chen, Ziwei Liu, Chen Change Loy, and Dahua Lin. Seesaw loss for long-tailed instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9695–9704, 2021a.

Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella Yu. Long-tailed recognition by routing diverse distribution-aware experts. In *ICLR*. OpenReview.net, 2021b.

Yuzhe Yang, Kaiwen Zha, Ying-Cong Chen, Hao Wang, and Dina Katabi. Delving into deep imbalanced regression. In *International Conference on Machine Learning (ICML)*, 2021.

Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. BBN: bilateral-branch network with cumulative learning for long-tailed visual recognition. In *CVPR*, pp. 9716–9725. Computer Vision Foundation / IEEE, 2020.

APPENDIX

A DERIVATION

A.1 DETAILED DERIVATION FOR BAYESIAN-PD

We show the derivation of Eq. 2.5 in detail. Fisrt, we derive the mapping from $p_{\text{train}}(y|x)$ to $p_{\text{bal}}(y|x)$. We normalize $p_{\text{bal}}(y|x)$ by its expected value of one. Using the simple fact that $\mathbb{E}_{y' \sim p_{\text{bal}}(y|x)}[1] = 1$, we have:

$$p_{\text{bal}}(y|x) = \frac{p_{\text{bal}}(y|x)}{\mathbb{E}_{y' \sim p_{\text{bal}}(y|x)}[1]}. \tag{A.1}$$

Using the Bayesian relation descirbed in Eq. 2.3 to re-paramterize $p_{\text{bal}}(y|x)$, we have:

$$p_{\text{bal}}(y|x) = \frac{p_{\text{train}}(y|x) \cdot \frac{p_{\text{bal}}(y)}{p_{\text{train}}(y)} \cdot \frac{p_{\text{train}}(x)}{p_{\text{bal}}(x)}}{\mathbb{E}_{y' \sim p_{\text{train}}(y|x) \cdot \frac{p_{\text{bal}}(y)}{p_{\text{train}}(y)} \cdot \frac{p_{\text{train}}(x)}{p_{\text{bal}}(x)}}[1]} \tag{A.2}$$

$$= \frac{p_{\text{train}}(y|x) \cdot \frac{p_{\text{bal}}(y)}{p_{\text{train}}(y)} \cdot \frac{p_{\text{train}}(x)}{p_{\text{bal}}(x)}}{\mathbb{E}_{y' \sim p_{\text{train}}(y|x)}[\frac{p_{\text{bal}}(y)}{p_{\text{train}}(y)}] \cdot \frac{p_{\text{train}}(x)}{p_{\text{bal}}(x)}} \tag{A.3}$$

$$= \frac{p_{\text{train}}(y|x) \cdot \frac{p_{\text{bal}}(y)}{p_{\text{train}}(y)}}{\mathbb{E}_{y' \sim p_{\text{train}}(y|x)}[\frac{p_{\text{bal}}(y)}{p_{\text{train}}(y)}]}. \tag{A.4}$$

Symmetrically, we may have the mapping from $p_{\text{bal}}(y|x)$ to $p_{\text{train}}(y|x)$ as well:

$$p_{\text{train}}(y|x) = \frac{p_{\text{bal}}(y|x) \cdot \frac{p_{\text{train}}(y)}{p_{\text{bal}}(y)}}{\mathbb{E}_{y' \sim p_{\text{bal}}(y|x)}[\frac{p_{\text{train}}(y')}{p_{\text{bal}}(y')}]}. \tag{A.5}$$

A.2 LOSS FORM FOR THE GNI VARIANT

We continue our derivation from Eq. 2.9. Note that for a bounded label space, *e.g.*, age, $p_{\text{train}}(y)$ is zero when y is out of the bound. Therefore, we can safely convert $\int_Y p_{\text{bal}}(y|x) \cdot p_{\text{train}}(y)dy$ into $\int_{\mathbb{R}^m} p_{\text{bal}}(y|x) \cdot p_{\text{train}}(y)dy$. We may further simplify Eq. 2.9 into:

$$\sum_{i=1}^{K} \phi_i S_i \int_Y \mathcal{N}(\tilde{\mu}_i, \tilde{\Sigma}_i)dy = \sum_{i=1}^{K} \phi_i S_i \int_{\mathbb{R}^m} \mathcal{N}(\tilde{\mu}_i, \tilde{\Sigma}_i)dy = \sum_{i=1}^{K} \phi_i S_i \tag{A.6}$$

With Eq. A.6, Eq. 2.7 and Eq. 2.8, we may have the debiased negative log-likelihood loss:

$$L = -\log p_{\text{train}}(y|x) = -\log \frac{p_{\text{bal}}(y|x) \cdot p_{\text{train}}(y)}{\int_Y p_{\text{bal}}(y'|x) \cdot p_{\text{train}}(y')dy'} \tag{A.7}$$

$$= -\log \mathcal{N}(y; y_{\text{pred}}, \sigma^2_{\text{pred}}\mathbf{I}) - \log p_{\text{train}}(y) + \log \sum_{i=1}^{K} \phi_i S_i \tag{A.8}$$

Recall that S_i is the norm of the product of two Gaussians. S_i itself is also a Gaussian:

$$S_i = \mathcal{N}(y_{\text{pred}}; \mu_i, \Sigma_i + \sigma^2_{\text{pred}}\mathbf{I}) \tag{A.9}$$

Plug Eq. A.9 back to the negative log likelihood loss, we have:

$$L = -\log \mathcal{N}(y_{\text{target}}; y_{\text{pred}}, \sigma^2_{\text{pred}}\mathbf{I}) + \log \sum_{i=1}^{K} \phi_i \cdot \mathcal{N}(y_{\text{pred}}; \mu_i, \Sigma_i + \sigma^2_{\text{pred}}\mathbf{I}) \tag{A.10}$$

Constants have been ignored.

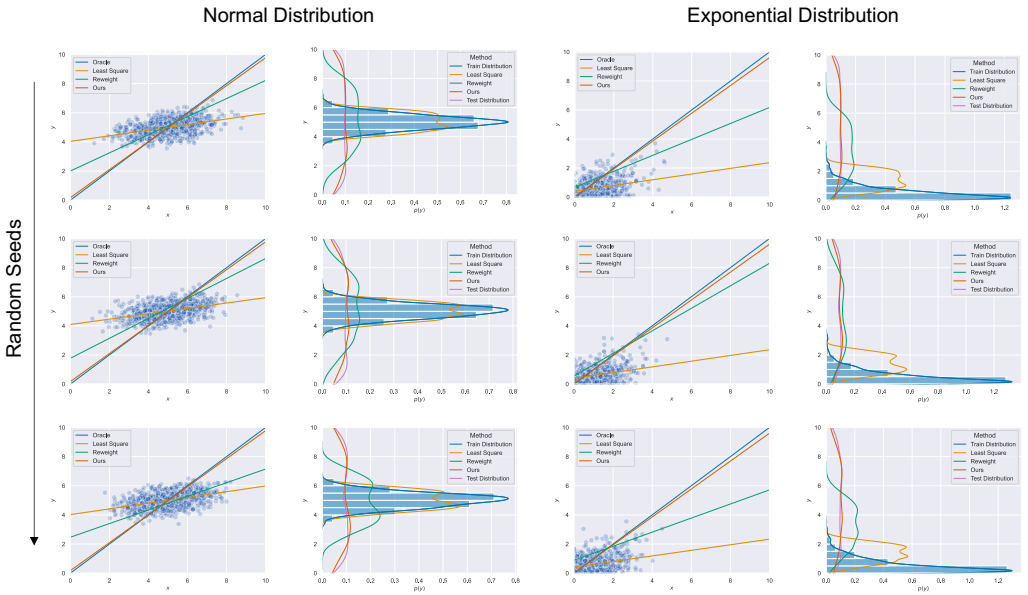

Figure 6: Synthetic benchmark on random seeds. Although the noise scale keeps the same, reweighting's performance varies drastically when different random seeds are used. In comparison, Bayesian-PD is robust to different sampled noises.

Table 6: Ablation on the choice of noise on IMDB-WIKI-DIR.

	bMAE↓				MAE↓			
Method	All	Many	Med.	Few	All	Many	Med.	Few
Test-time								
Fix. ($\sigma = 4$)	13.24	7.26	14.10	31.08	7.85	7.17	13.41	24.49
Fix. ($\sigma = 5$)	13.17	7.63	13.25	30.26	8.13	7.56	12.63	23.79
Fix. ($\sigma = 6$)	13.42	8.60	12.45	29.07	8.97	8.56	11.89	22.83
Train-time								
Fix. ($\sigma = 6$)	12.85	7.27	13.26	29.79	7.81	7.20	12.78	23.78
Fix. ($\sigma = 7$)	12.67	7.52	12.75	28.67	8.00	7.45	12.32	23.25
Fix. ($\sigma = 8$)	12.68	7.80	12.61	27.83	8.24	7.73	12.21	22.94
Joint.	12.66	7.65	12.68	28.14	8.12	7.58	12.27	23.05

B Implementation Details

B.1 IMDB-WIKI-DIR

We follow the RRT setting in Yang et al. (2021). Concretely, we use ResNet-50 (He et al., 2016) model as the backbone. We train the vanilla model for 90 epochs using Adam optimizer (Kingma & Ba, 2015). We decay the learning rate from 10^{-3} by 0.1 at 60-th epoch and 80-th epoch. We then freeze the backbone, re-initialize and train the last linear layer. For the retraining, we train the last linear layer for 30 epochs with a constant learning rate at 10^{-4}. We use a GMM with 2 components in train time. We use a GMM with 128 components with a standard deviation fixed at 2 in test time. $\sigma_{\text{pred}} = 5$ used for TTA.

B.2 NYUD2-DIR

We follow the settings in Yang et al. (2021). We use a ResNet-50-based encoder-decoder architecture proposed by (Hu et al., 2019). We train the model for 20 epochs using Adam optimizer with an initial learning rate at 10^{-4}. The learning rate decays by 0.1 every 5 epochs. Only direct supervision on depth is used in training. We use a GMM with 16 components in training time. We use a GMM with 128 components with a standard deviation fixed at 2 in test time. $\sigma_{\text{pred}} = 1$ for GAI, BMC, and TTA.

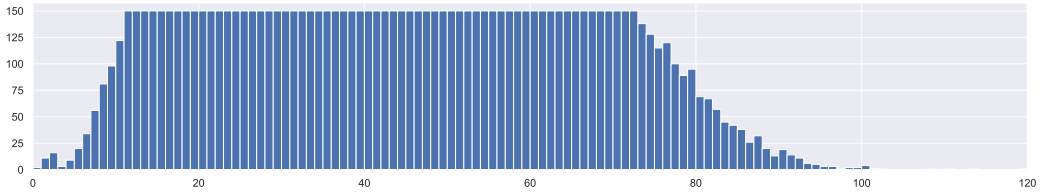

Figure 7: IMDB-WIKI-DIR test set visualization. We observe tail labels on both edges of the test distribution. Overall metrics will not sufficiently assess a model's performance on the senior group (age $<\sim$75) and the youth group (age $>\sim$15).

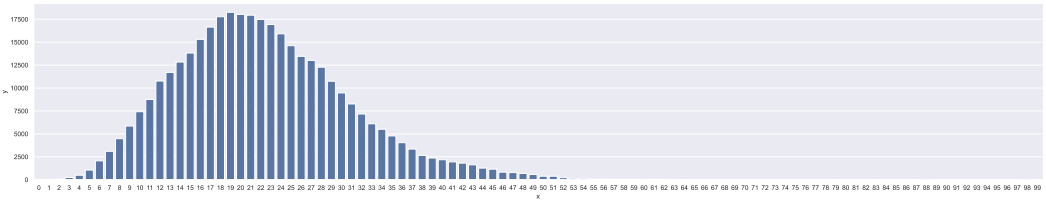

Figure 8: Visualization of the training label distribution of IHMR. The horizontal axis is 100 regions uniformly divided on the pose space according to their geodesic distance to the mean pose.

B.3 IHMR

We use a pretrained SPIN (Kolotouros et al., 2019) model as the feature extractor, and re-train the linear regressor for 20 epochs. We follow SPIN to train on the following 3D datasets: Human3.6M (Ionescu et al., 2011), MPI-INF-3DHP (Mehta et al., 2017); and following 2D datasets: LSP (Johnson & Everingham, 2010); LSP-extended (Kolotouros et al., 2019), MPII (Andriluka et al., 2014), COCO (Lin et al., 2014). We test on 3DPW (von Marcard et al., 2018). Static fits are used to provide supervision on the 2D datasets. We use a constant learning rate at 10^{-4}. We use a GMM with 16 components.

C Synthetic benchmark on noise

We compare least square, reweighting, and Bayesian-PD under different random seeds on the one-dimensional linear regression. A visualization of results is shown in Fig. 6. We observe that reweighting is sensitive to random seeds. Reweighting's performance varies drastically when random seed changes. This may attribute to the fact that reweighting signifies rare labels'noise and the zero mean noise assumption no longer holds. In comparison, Bayesian-PD is robust to different noise sampling results.

D IMDB-WIKI-DIR test set visualization

We visualize the label distribution of IMDB-WIKI-DIR's test set in Fig. 7.

E Ablations

E.1 Effect of the noise scale

We study the effect of σ_{pred} on IMDB-WIKI-DIR, by fixing σ_{pred} at different values. Both train-time and test-time debiasing are studied. We use the GAI variant for train-time debiasing. We also compare fixed σ_{pred} with jointly optimized σ_{pred}. Results are shown in Tab. 6. We observe that larger σ_{pred} trades the performance towards tail labels. We also observe that the jointly optimized σ_{pred} is effective in finding the optimal trade-off point.

E.2 Effect of number of components in GMM

We study the number of components K in GMM on IMDB-WIKI-DIR using the GAI variant. Results are shown in Tab. 7. We notice that the performance reaches optimal when K is larger or equal

Figure 9: Qualitative comparison of Bayesian-PD and the baseline, SPIN-RT. Left: SPIN-RT. Right: Bayesian-PD. We observe that the baseline's predictions are less stretched out. They bias towards the mean pose, particularly for poses like raising arms and bending legs. In comparison, our method effectively eliminates the bias and recovers rare poses.

Table 7: Ablation on the effect of the number of components K in the GMM.

	bMAE↓				MAE↓			
Method	All	Many	Med.	Few	All	Many	Med.	Few
K=1	12.72	7.70	12.94	28.08	8.18	7.63	12.47	23.17
K=2	12.66	7.65	12.68	28.14	8.12	7.58	12.27	23.05
K=4	12.67	7.62	12.68	28.26	8.09	7.55	12.26	23.03
K=128	12.66	7.61	12.87	28.11	8.09	7.53	12.44	23.18

to 2. This may attribute to the fact that the training label distribution of IMDB-WIKE-DIR, as shown in Fig. 5, is relatively simple.

E.3 Effect of the sample size of BMC

Table 8: Ablation on the effect of the sample size of BMC.

Method	B=256	B=16	B=8	B=4	B=2
BMC	0.043	0.001	0.074	0.941	17.18
BMC w/ positive sample	0.043	0.046	0.033	0.037	0.015

We study the effect of sample size in BMC on the one-dimensional linear regression synthetic benchmark. Results are shown in Tab. 8. Smaller sample size leads to a larger variance in the estimation. Therefore, BMC's performance drops when samples are insufficient, particularly when B is less than 16. We observe that the issue can be effectively alleviated by keeping a positive sample, *i.e.*, y_{target}, in the MC batch. The simple tweak achieves substantial improvement when the sample size is small.

F Qualitative comparison on IHMR

In Fig. 9, we show a qualitative comparison of Bayesian-PD and the baseline on the IHMR benchmark.