Title

Label-efficient Bayesian Assessment of Black-box Classifiers

Author

Ji, Disi

Publication Date

2020

UNIVERSITY OF CALIFORNIA,
IRVINE

Label-efficient Bayesian Assessment of Black-box Classifiers

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Computer Science

by

Disi Ji

Dissertation Committee:
Chancellor's Professor Padhraic Smyth, Chair
Assistant Professor Stephan Mandt
Professor Mark Steyvers

2020

DEDICATION

To my parents

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

ACKNOWLEDGMENTS

First and foremost, I would like to thank my advisor Professor Padhraic Smyth. Over the years, Padhraic has been an incredible advisor – wise, patient, and generous with his time and insights. I really appreciate the numerous hours that he spent for me. Padhraic shaped my way of thinking about machine learning problems and helped me learn to identify the important problems to work on. He also taught me how to communicate and collaborate, and how to make progress even when working on challenging problems. His encouragement helped me gradually build confidence in my work. His influence is on every page of this work.

It has been incredibly fortunate for me to work with my collaborators at different stages of my research. This dissertation is greatly benefited from Mark's contribution in the last two years. Discussions with him always provide me new perspectives to think about them. I worked closely with Professor Richard Scheuermann and Professor Yu Qian from J. Craig Venter Institute on cytometry data analysis in the first two years of my graduate program. I have also been lucky to write papers with other datalab members including Eric Nalisnick, Preston Putzel, and Robert Logan.

My gratitude also goes to the other graduate students. Rachel Longjohn, Gavin Kerrigan, Robby Logan, Alex Boyd, Dylan Slack and Zhile Ren have provided valuable suggestions on the contents of this dissertation. I've learned a lot from countless interesting conversations with my labmates: Moshe Lichman, Eric Nalisnick, Dimitris Kotzias, Jihyun Park, Chris Galbraith, Robby Logan, Casey Graff, Preston Putzel, Alex Boyd, Gavin Kerrigan, Rachel Longjohn, Markelle Kelly, Edgar Robles and Sam Showalter. On a personal note, I would like to express my appreciation to Jihyun Unnie, Dimitri, Ren, Dylan and other friends for always being there and for creating so many good memories with me!

Thank the friends from older years who have given me decade-long support and encouragement, especially Xiao, Lingbo, Yue, Maomao, Shuyan, Yinbo and Shiqi. My greatest gratitude is reserved for my parents. They have always believed in me and supported all my decisions. I hope to see them soon when things are back to normal.

This work was supported in part by the National Science Foundation under grants number 1900644 and 1927245 and by a Qualcomm Faculty Award (PS). In addition this work was partially funded by the Center for Statistics and Applications in Forensic Evidence (CSAFE) through Cooperative Agreement 70NANB20H019 between NIST and Iowa State University, which includes activities carried out at the University of California, Irvine. I was also partially supported by NIH/NCATS U01TR001801 (FlowGate), NSF XSEDE allocation MCB170008, and NIH Commons Credits on Cloud Computing CCREQ-2016-03-00006.

VITA

Disi Ji

EDUCATION

Doctor of Philosophy in Computer Science	**2020**
University of California, Irvine	*Irvine, CA*
Master of Science in Computer Science	**2016**
University of California, Irvine	*Irvine, CA*
Bachelor of Science in Mathematics and Applied Mathematics	**2015**
Fudan University	*Shanghai, China*

RESEARCH EXPERIENCE

Graduate Research Assistant	**2015–2020**
University of California, Irvine	*Irvine, California*

TEACHING EXPERIENCE

Teaching Assistant of COMPSCI 260	**2020 Winter**
University of California, Irvine	*Irvine, California*
Teaching Assistant of COMPSCI 273A	**2019 Fall**
University of California, Irvine	*Irvine, California*

PROFESSIONAL EXPERIENCE

Software Engineer Intern	**2019 Summer**
Facebook	*New York City, New York*
Software Engineer Intern	**2018 Summer**
Google	*Cambridge, Massachusetts*

REFEREED JOURNAL PUBLICATIONS

- Ji, Disi, Preston Putzel, Yu Qian, Ivan Chang, Aishwarya Mandava, Richard H. Scheuermann, Jack D. Bui, Huan-You Wang, and Padhraic Smyth. "Machine Learning of Discriminative Gate Locations for Clinical Diagnosis." *Cytometry Part A*, 2020.

REFEREED CONFERENCE PUBLICATIONS

- Ji, Disi, Robert L. Logan IV, Padhraic Smyth, and Mark Steyvers. "Active Bayesian Assessment for Black-Box Classifiers." arXiv preprint arXiv:2002.06532 (2020).
- Ji, Disi, Padhraic Smyth, and Mark Steyvers. "Can I Trust My Fairness Metric? Assessing Fairness with Unlabeled Data and Bayesian Inference." *In Advances in Neural Information Processing Systems*, 2020.
- Ji, Disi, Eric Nalisnick, Yu Qian, Richard H. Scheuermann, and Padhraic Smyth. "Bayesian Trees for Automated Cytometry Data Analysis." *In Machine Learning for Healthcare Conference*, pp. 465-483. 2018.

ABSTRACT OF THE DISSERTATION

Label-efficient Bayesian Assessment of Black-box Classifiers

By

Disi Ji

Doctor of Philosophy in Computer Science

University of California, Irvine, 2020

Chancellor's Professor Padhraic Smyth, Chair

Machine learning classifiers are currently widely used to make decisions about individuals, across a broad variety of societal contexts: education admissions, health insurance, medical diagnosis, court decisions, marketing, face recognition, and more—and this trend is likely to continue to grow. It is now well-recognized that these machine learning models are susceptible to built-in biases that can lead to systematic discrimination against protected groups. The machine learning research community has begun to recognize this important issue and in the past few years had devoted considerable research resources towards developing principles, frameworks, and algorithmic solutions to address these problems.

In this general context, this work addresses the understudied problem of how to assess how accurate, calibrated and fair a model may be, and how much confidence we should have in this assessment given access to a limited amount of labeled data. To be specific, we propose a Bayesian framework for assessing (with uncertainty) performance metrics of black-box classifiers, which is particularly important when only a limited amount of labeled data is available. To improve label-efficiency of the assessment, we develop active Bayesian assessment strategies for an array of fundamental tasks including (1) estimation of model performance; (2) identification of model deficiencies; (3) performance comparison between groups. When unlabeled data is available, we develop a new hierarchical Bayesian methodology that leverages information from both unlabeled and labeled data.

We demonstrate that our proposed approaches need significantly fewer labels than baselines, via a series of experiments assessing the performance of modern neural classifiers (e.g., ResNet and BERT) on several standard image and text classification datasets. One particular example of how the proposed approach can be used is in the increasingly common situation where the user of a blackbox classification model needs to assess its performance from a fairness perspective, in a manner that is separate and independent from the claims made by the entity that trained the model. We demonstrate that the methodology developed in this work is well-suited to such an application.

Chapter 1

Introduction

Complex machine learning models, particularly deep learning models, are now being applied to a variety of practical prediction problems ranging from the diagnosis of medical images [Kermany et al., 2018, Yi et al., 2019] to autonomous driving [Du et al., 2017, Caesar et al., 2020]. As a result, software systems with embedded machine learning components are becoming increasingly common.

It is increasingly important for the user of a model to have accurate and robust assessments of the quality of the model's predictions. However, as an example, "self-confident" estimates provided by machine learning predictors can often be quite unreliable and miscalibrated [Zadrozny and Elkan, 2002, Kull et al., 2017, Ovadia et al., 2019]. In particular, complex models such as deep networks with high-dimensional inputs (e.g., images and text) can be significantly overconfident in practice [Gal and Ghahramani, 2016, Guo et al., 2017, Lakshminarayanan et al., 2017].

Thus, downstream users of black-box predictors will need the capability to carry out assessment separately and independently from the training and evaluation procedures used when fitting the model. This assessment could, for example, be conducted by organizations not involved in training the model, in a manner similar to the assessment of commercial products carried out by regulatory agencies. Additional motivations for independent assessment include legal requirements that may mandate independent model assessment, the need to build human consumers' trust in model predictions, and situations in which the predictor is deployed in an environment with a different distribution over inputs and outputs than the one in which the model was trained. Problems related to detecting and handling unexpected changes in data distributions at deployment time, such as label shift [Lipton et al., 2018], are drawing increasing attention from the machine learning community. Recent work [Recht et al., 2019] has shown that even when closely following the process of creating the original data used to train a model, the performance of classification models on test datasets can differ significantly from the performance on the original dataset. Hendrycks and Dietterich [2019] and Ovadia et al. [2019] found that both accuracy and calibration of classifiers are not robust to common corruptions and perturbations, let alone to worst-case adversarial perturbations.

1.1 Outline & Contributions

In this work, we address the understudied problem of how to assess how accurate, calibrated and fair a model may be, and how much confidence we should have in this assessment given access to a limited amount of labeled data.

In Chapter 2, we develop a Bayesian framework for assessing performance metrics of black-box classifiers. Our contributions are:

- We developed Bayesian techniques for estimating groupwise accuracy, reliability diagram, expected calibration error (ECE), confusion matrix, misclassfication cost, and performance difference;
- We discussed using self-assessment of prediction models as informative priors for

Bayesian assessment;

- We illustrated a number of different ways that the framework can be used to understand performance aspects of widely-used deep learning models and datasets.

Chapter 3 describes a framework for **active Bayesian assessment** of black-box classifiers, using techniques from Bayesian active learning to efficiently select instances to label so that uncertainty of assessment can be reduced for different assessment tasks. Our primary contributions in this chapter are:

- We proposed a general framework for active Bayesian assessment for an array of fundamental tasks including (1) estimation of model performance; (2) identification of model deficiencies; (3) performance comparison between groups;
- We developed a set of Thompson sampling algorithms for label-efficient active assessment;
- We demonstrated that our proposed approaches need significantly fewer labels than baselines, via a series of experiments assessing the performance of modern neural classifiers (e.g., ResNet and BERT) on several standard image and text classification datasets.

In Chapter 4 we study the assessment of black-box classifiers in an algorithmic fairness context.[1] To provide a reliable answer to the question “can I trust my fairness metric", we stress the importance of being aware of the uncertainty in group fairness assessment especially when test size is relatively small. We propose a new framework for combining labeled and unlabeled data to produce lower variance estimates, based on Bayesian calibration of model scores on unlabeled data. The results clearly indicate that the proposed method can

[1] This work has been published as Ji, Smyth, Steyvers (2020) at Conference on Neural Information Processing Systems.

systematically produce significantly more accurate estimates of fairness metrics for different classification models across different datasets and sensitive attributes. In particular, the three primary contributions are

- We proposed a comprehensive Bayesian treatment of fairness assessment that provides uncertainty about estimates of group fairness metrics;
- We developed a new hierarchical Bayesian methodology that leverages information from both unlabeled and labeled examples;
- We demonstrated with systematic large-scale experiments across multiple datasets and models that using unlabeled data can reduce estimation error significantly.

Chapter 2

Bayesian Assessment of Black-Box Classifiers

In this chapter, we develop a general Bayesian framework to assess black-box classifiers with uncertainty estimates. We illustrate the utility of the framework via Bayesian inference with posterior uncertainty for quantities such as groupwise accuracy, reliability diagram, expected calibration error (ECE), confusion matrix, misclassfication cost, and performance difference.

2.1 Preliminaries

2.1.1 Notation and Problem Statement

We consider classification problems with a feature vector $\mathbf{x}$ and a class label $y \in \{1, \ldots, K\}$, e.g., classifying image pixels $\mathbf{x}$ into one of K classes. We assume access to a trained prediction model M that makes predictions of y given a feature vector $\mathbf{x}$. In particular, we assume that the model produces a numerical score for each class, reflecting its confidence, typically in the form of an estimate of the class-conditional probability $p_M(y = k|\mathbf{x})$ for $k = 1, \ldots, K$. Such probability estimates can be obtained from a logistic classifier, from the softmax output layer of a neural network, from averages over leaf nodes in tree-based models, and so on. A notational aside: for probabilities that are being generated by the model we use subscript M, e.g., $p_M(y = k|\mathbf{x})$. When we refer to the true probability with respect to the underlying true distribution $p(\mathbf{x}, y)$ we drop the subscript, e.g., when using terms like $p(y = k|\mathbf{x})$ and $p(\mathbf{x})$ in computing expectations.

2.1.2 Blackbox Classification Models

Many of the classification models are **black boxes** from the perspective of downstream users, such as models developed remotely by commercial entities and hosted as a service in the cloud [Yao et al., 2017, Sanyal et al., 2018]. For a variety of reasons (legal, economic, competitive), users will often have no direct access to the detailed workings of the model, how the model was trained, or the training data.

In this chapter, we focus on the problem of assessing the performance of a model on data drawn from some unknown distribution $p(\mathbf{x}, y)$ representing the environment where the model is being used. We are interested in the situation where the model is a black box, i.e. we can observe the inputs $\mathbf{x}$ and the outputs $p_M(y = k|\mathbf{x})$ but don't have any other information about the inner-workings of M. Specifically, rather than learning a model itself we want to learn about the characteristics of a fixed model that is making predictions in a particular environment. To assess the performance of a black-box classifier, we adopt a Bayesian framework and treat the metrics of interest (e.g. classification accuracy and calibration error) as unknown parameters that we estimate from (limited) labeled data drawn from a distribution $p(\mathbf{x}, y)$.

2.1.3 Datasets and Classification Models

Table 2.1: Assessment datasets and models used in our experiments. Size refers to the maximum number of labeled instances available for assessment.

	Mode	Size	Classes	Model
CIFAR-100	Image	10K	100	ResNet-110
ImageNet	Image	50K	1000	ResNet-152
SVHN	Image	26K	10	ResNet-152
20 Newsgroups	Text	7.5K	20	$\text{BERT}_{\text{BASE}}$
DBpedia	Text	70K	14	$\text{BERT}_{\text{BASE}}$

Datasets Throughout this chapter and the next chapter we will use several well-known classification datasets, in combination with large-scale deep network classification models, to illustrate Bayesian assessment.

- *CIFAR-100* [Krizhevsky and Hinton, 2009]: A dataset of 32 × 32 colored images from the web, partitioned into 100 classes. There are 50,000 and 10,000 data points in the train and test sets, respectively. 100 classes are grouped into 20 superclasses, e.g. for superclass *aquatic mammals*, it includes five classes: *beaver*, *dolphin*, *otter*, *seal* and *whale*.

- *Street View House Numbers (SVHN)* [Netzer et al., 2011]: A dataset of 32 × 32 colored images of cropped out house numbers from Google Street View, partitioned into 10 classes. There are 73,257 and 26,032 data points in the train and test sets, respectively.

- *ImageNet* [Russakovsky et al., 2015]: An image dataset of natural objects of variable resolutions from the web, partitioned into 1000 classes. There are 1.2 million and 50,000 data points in the train and test sets, respectively.

- *20 Newsgroups* [Lang, 1995]: A text dataset of news articles, partitioned into 20 categories by content. There are 11,293 and 7,528 data points in the train and test sets, respectively.

- *DBpedia* [Zhang et al., 2015]: A text dataset of structured content from the information created by the Wikipedia project, partitioned into 14 classes. There are 560,000 and 70,000 data points in the train and test sets, respectively.

Prediction Models For image classification we use ResNet [He et al., 2016] architectures with either 110 layers (CIFAR-100) or 152 layers (SVHN and ImageNet). For ImageNet we use the pretrained model provided by PyTorch, and for CIFAR and SVHN we use the pretrained model checkpoints provided at: `https://github.com/bearpaw/pytorch-classification`. For text classification tasks we use fine-tuned $\text{BERT}_{\text{BASE}}$ [Devlin et al., 2019] models.[1]

Prediction models were all trained on standard training sets in the literature, which are independent from the datasets used for assessment. The assessment datasets are based on standard test sets used for each dataset in the literature. Detailed statistics of the test sets are provided in Table 2.1. To facilitate reproducing our results we provide all of the model predictions used in our experiments at: `https://github.com/disiji/bayesian-blackbox`.

2.2 Self-Assessment and Miscalibration

2.2.1 Self-Assessment of Classifiers

Accuracy The *marginal accuracy* of a classification model at $\mathbf{x}$ is defined as $\theta(\mathbf{x}) = p(y = \hat{y}|\mathbf{x})$. We also define *regional accuracy* over local regions of the input space. For any region $\mathcal{R}$ in the input space, regional accuracy is the marginal probability that the predicted label

[1] The text classification models were trained by Robert Logan as part of a collaboration.

matches with the true label, conditioned on $\mathbf{x} \in \mathcal{R}$:

$$\theta_{\mathcal{R}} = \mathbb{E}_{p(\mathbf{x},y|\mathbf{x}\in\mathcal{R})}[\theta(\mathbf{x})] = \int_{\mathcal{R}} p(y = \hat{y}|\mathbf{x})p(\mathbf{x}|\mathbf{x} \in \mathcal{R})d\mathbf{x}. \tag{2.1}$$

To estimate the classwise accuracies $\theta_{\mathcal{R}}$ from data, a standard approach would be to empirically approximate the integral above by sampling $\mathbf{x}, y$ pairs from the conditional distribution $p(\mathbf{x}, y|\mathbf{x} \in \mathcal{R})$. Equivalently, $\theta_{\mathcal{R}}$, conditioned on $\mathbf{x} \in \mathcal{R}$, can be modeled as an unknown Bernoulli parameter with draws $(\mathbf{x}_i, y_i)$ leading to binary outcomes $\mathbb{1}(y_i = \hat{y}_i) \in \{0, 1\}$, where $i = 1, 2, \cdots, S$. In this case, the frequency-based (maximum likelihood) estimate is:

$$\hat{\theta}_{\mathcal{R}} = \frac{1}{S}\sum_{i=1}^{S} \mathbb{1}(y_i = \hat{y}_i). \tag{2.2}$$

Confidence The classifier's label prediction for a particular input $\mathbf{x}$ is $\hat{y} = \arg\max_k p_M(y = k|\mathbf{x})$. We can define $s(\mathbf{x}) = p_M(y = \hat{y}|\mathbf{x})$ as the **score** of a model, which is a function of $\mathbf{x}$, i.e., the class probability that the model produces for its predicted class $\hat{y} \in \{1, \ldots, K\}$ given input $\mathbf{x}$. This is sometimes also referred to as a model's **confidence** in its prediction and can be viewed as a model's **self-assessment** of its accuracy when it predicts $\hat{y}$ given $\mathbf{x}$. We can get a model's self-assessed estimate of its own accuracy from unlabeled data by taking the average of the model's scores on the unlabeled data:

$$s_{\mathcal{R}} = \frac{1}{S}\sum_{i=1}^{S} s(\mathbf{x}). \tag{2.3}$$

Calibration The classification model's output is **calibrated** when it matches with the true probabilities of labels. Meteorologists were among the first to think about calibration from the perspective of forecaster evaluation. Brier [1950] introduced the Brier score to measure the forecasts expressed with probabilities; Murphy and Winkler [1977] proposed reliability diagrams to visually inspect calibration behaviors; DeGroot and Fienberg [1983] discussed the decomposition of classification loss into calibration and refinement losses. In the machine learning literature, previous work mainly studies two types of calibration:

- Most of the previous work studies calibration by comparing model scores with accuracies, e.g. Guo et al. [2017], Kull et al. [2017]. A model is **binary calibrated** when for any input $\mathbf{x}$ its model score $s(\mathbf{x})$ represents the likelihood that its prediction is correct, i.e.

$$\mathbb{1}(y = \hat{y}) \sim \text{Bernoulli}(s(\mathbf{x})), \forall \mathbf{x}. \tag{2.4}$$

- A strictly stronger definition of calibration has recently been discussed in [Vaicenavicius et al., 2019, Kull et al., 2019]. A model is **multi-class calibrated** when the multidimensional model output $p_M(y = k|\mathbf{x})$ matches with the distribution of true label y, i.e.

$$y \sim \text{Cat}(p_M(y = k|\mathbf{x})), \forall \mathbf{x}. \tag{2.5}$$

For a binary calibrated model M, the expectation of its accuracy at $\mathbf{x}$ is the model score $s(\mathbf{x})$. For a multi-class calibrated model M, with $p_M(y = k|\mathbf{x})$ we can compute the the expectation of the confusion probabilities, i.e. the columns of the confusion matrix.

In practice, however, real-world machine learning models, especially deep learning models, are rarely calibrated [Guo et al., 2017], making the self-assessment of classification models unreliable. Thus, it is important to independently assess model performance with labeled data instead of relying on a classifier's self-assessment.

2.2.2 Miscalibration of Classifiers

In this section, we discuss the tools and metrics used to diagnose and measure the binary miscalibration of classifiers.

Reliability Diagram Reliability diagrams are a widely used tool for visually diagnosing model calibration by comparing model scores and empirical accuracies [Murphy and Winkler, 1977, DeGroot and Fienberg, 1983, Niculescu-Mizil and Caruana, 2005]. These diagrams plot the empirical sample accuracy $\theta(\mathbf{x})$ of a model M as a function of the model's confidence scores $s(\mathbf{x})$. For a particular value $s(\mathbf{x}) = s \in [0, 1]$ along the x-axis, the corresponding y value is defined as: $\mathbb{E}_{\mathbf{x}|s(\mathbf{x})=s}[\theta(\mathbf{x})]$. If the model is perfectly calibrated, then $\theta(\mathbf{x}) = s(\mathbf{x})$ and the diagram consists of the identity function on the diagonal. Deviations from the diagonal reflect miscalibration of the model. In particular, if the curve lies below the diagonal with $\theta(\mathbf{x}) < s(\mathbf{x})$ then the model M is overconfident (e.g., see Guo et al. [2017]).

To address data sparsity, scores are often aggregated along the x-axis into bins of equal width or equal frequency, spanning the range $[0, 1]$, e.g., DeGroot and Fienberg [1983], Niculescu-Mizil and Caruana [2005], Guo et al. [2017]. We denote the b-th bin as $\mathcal{R}_b$, and the bins $\{\mathcal{R}_b | b = 1, 2, \cdots, B\}$ form a partition of the input space determined by the model score $s(\mathbf{x})$. For example, with equal-sized binning, $\mathcal{R}_b = \{\mathbf{x}|s(\mathbf{x}) \in [(b-1)/B, b/B)\}$, where $b = 1, \ldots, B$ ($B = 10$ is often used in practice). The accuracy of the model per bin is θ_b, which can be viewed as a marginal accuracy over the region $\mathcal{R}_b$, i.e.,

$$\theta_b = \int_{\mathcal{R}_b} p(y = \hat{y}|\mathbf{x}) p(\mathbf{x}|\mathbf{x} \in \mathcal{R}_b) d\mathbf{x}. \tag{2.6}$$

In practice, θ_b is computed based on a sample of labeled data for examples whose scores fall in the b-th bin.

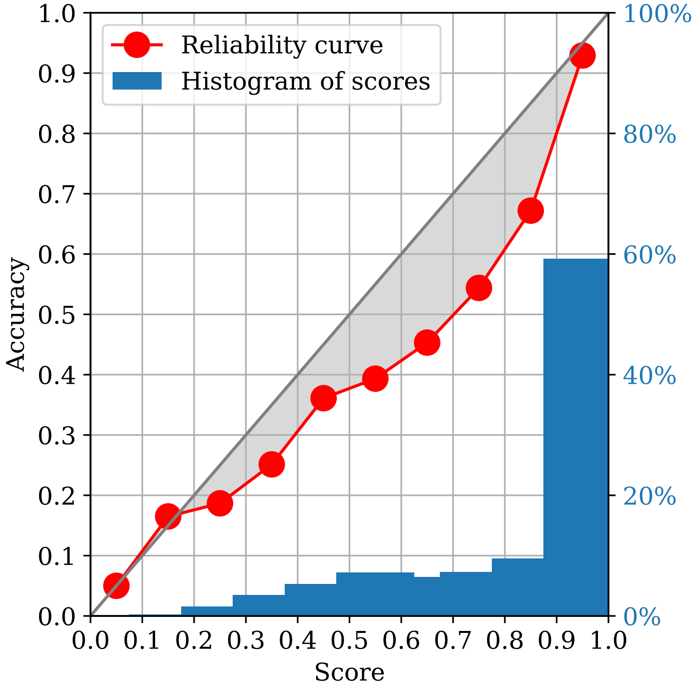


Figure 2.1: Reliability diagram for ResNet-110 on CIFAR-100 and histogram for its model scores. The red circles plot the binwise accuracies of 10 equal-width bins. The gray region shows the deviation of the reliability curve from the diagonal. The blue histogram shows the distribution of the model scores.

Figure 2.1 shows the reliability diagram of ResNet-110 on CIFAR-100. The average score of ResNet-100 is substantially higher than its accuracy for all bins, i.e. ResNet-110 is overconfident in its predictions. Apart from the reliability diagram, we also plot the histogram for model scores (blue) in Figure 2.1. The distribution of model scores is highly skewed, with model scores for about 60% of the data points being greater than 0.9. Guo et al. [2017] shows that this type of miscalibration is common for modern neural networks, and it is influenced by both model architecture and training.

Expected Calibration Error (ECE) We can quantify the amount of miscalibration with metrics by measuring the divergence between $s(\cdot)$ and $\theta(\cdot)$. Nixon et al. [2019] provides a comprehensive review of the miscalibration metrics in machine learning literature. In this chapter, we focus on expected calibration error (ECE) given that it is among the widely-used calibration metrics (e.g., Guo et al. [2017], Ovadia et al. [2019]).

ECE is defined as a weighted average of the absolute distance between the true accuracy θ_b and the average score s_b per bin:

$$\text{ECE} = \sum_{b=1}^{B} p_b |\theta_b - s_b| \tag{2.7}$$

where p_b is the probability of a score lying in bin b. The accuracy of the model per bin is θ_b, which can be viewed as a marginal accuracy over the region $\mathcal{R}_b$ in the input space corresponding to $s(\mathbf{x}) \in \text{Bin}_b$, i.e., $\theta_b = \int_{\mathcal{R}_b} p(y = \hat{y}|\mathbf{x}) p(\mathbf{x}|\mathbf{x} \in \mathcal{R}_b) d\mathbf{x}$. When estimating ECE, p_b can be estimated with the model scores of unlabeled data, while estimating θ_b requires true labels of the data points. For example, in Figure 2.1 ECE is 0.098, indicating moderate miscalibration of ResNet-110 on CIFAR-100.

We also note that recent work, e.g. Vaicenavicius et al. [2019], proposed extensions of these to the multi-class definition of miscalibration, but we leave out the discussion in this chapter.

2.3 Bayesian Assessment of Black-box Classifiers

Self-assessment of models is not reliable, where by self-assessment we mean a model's estimate of a performance metric based on its class-probability estimates. For example, in Figure 2.1, the model's self-assessment of its accuracy is 0.84, while the true accuracy of the model is 0.74. In order to obtain an independent and unbiased assessment of a classifier performance metric such as accuracy, this must be done with a labeled test dataset that is independent from that used to train the classifier. In the machine learning research literature it is common to use relatively large test datasets (e.g., see Table 2.1) in order to insure that the empirical estimate of performance (e.g., of accuracy) is reliable. However, in real-world environments when black-box models are deployed, it will frequently be the case that a large labeled test set is not available. For example it may be very expensive to obtain labels for various reasons.

This is the motivation for Bayesian assessment: to make reliable inferences about classifier performance using relatively little data and quantifying uncertainty about these inferences.

We treat classifier performance metrics θ as parameters of interest to estimate with labeled data.[2] It is natural to consider Bayesian inference in this context to represent the uncertainty of the assessment, especially in situations where there is relatively little labeled data available. In the reminder of this chapter, we develop a Bayesian assessment framework to independently assess performance metrics with uncertainty.

We outline below our Bayesian approach to make posterior inferences about a performance metric θ given labeled data $\mathcal{D} = \{(x_i, y_i), i = 1, 2, \cdots, N\}$. (x_i, y_i) are *i.i.d.*. $p(\theta)$ is a prior distribution of θ, i.e. $\theta \sim p(\theta)$. For each data point i, the classification model M generates a predicted label $\hat{y}_i$ for a given x_i. The Oracle is then queried to obtain a label outcome z_i with $z_i = f_M(x_i, y_i)$. For example for accuracy estimation, we have $z_i = \mathbb{1}(y_i = \hat{y}_i)$ where y_i is a stochastic function of x_i and $\hat{y}_i$ is a deterministic function of x_i. We will refer to z_i as the **label outcome** and can define a likelihood (conditional probability of z_i, given the unknown performance metric θ) as $z_i \sim q_\theta(z_i)$, where $q_\theta(z_i) = \text{Bern}(z_i|\theta)$. Later we will extend the definition to z_i to cases other than accuracy.

With Bayes' rule, the posterior distribution of the assessment metric θ after observing labeled data $\mathcal{D} = \{(x_i, y_i), i = 1, 2, \cdots, N\}$ is updated to:

$$p(\theta|\mathcal{D}) = \frac{p(\theta) \cdot \prod_{i=1}^{N} \, q_\theta(z_i)}{\int_\theta p(\theta) \cdot \prod_{i=1}^{N} q_\theta(z_i) \, d\theta}. \tag{2.8}$$

For example for accuracy estimation, θ is the accuracy and $p(\theta) = \text{Beta}(\alpha_0, \beta_0)$ is the prior distribution of θ; $z_i = \mathbb{1}(y_i, \hat{y}_i)$ is the binary outcome of the classifier, i.e. whether the model prediction is correct. We can update the posterior distribution of θ in closed-form to

[2] Here we overload the notation θ. In the previous section, we use θ to represent accuracy. In the remainder of this chapter θ represents the performance metric of interest which includes but is not limited to accuracy.

$\text{Beta}(\alpha_0 + r, \beta_0 + N - r)$ where r is the number of correct label predictions by the model given N trials, i.e. $r = \sum_{i=1}^{N} \mathbb{1}(y_i = \hat{y}_i)$. As the amount of observed labeled data N increases, the *maximum a posteriori estimation* (MPE) $\hat{\theta}$ converges to the true accuracy, and the variance of $\hat{\theta}$ asymptotically decreases. For accuracy estimation, Liu [2008] proves that $\text{Bias}(\hat{\theta})^2$ is of order $\mathcal{O}(\frac{1}{N^2})$; Sawade et al. [2010] proves that $\text{Var}(\hat{\theta})$ is of order $\mathcal{O}(\frac{1}{N})$.

With Bayesian assessment, we can quantify the uncertainty of our assessment with the posterior distribution $p(\theta|\mathcal{D})$, and posterior uncertainty asymptotically decreases as the number of labeled data N increases. Next we demonstrate the importance of uncertainty for model assessment. With an illustrative example, we show that even with real-world large datasets like CIFAR-100, posterior uncertainty of accuracy and ECE are still considerably large.

We then apply our Bayesian assessment framework to assess different performance metrics θ. We do this by defining an appropriate prior $p(\theta)$ and likelihood $q_\theta(z)$ for each metric θ of interest. The metrics we study in this chapter include classwise accuracy, reliability diagram, expected calibration error (ECE), confusion matrix, misclassfication cost and accuracy difference.

2.3.1 Bayesian Groupwise Accuracy

Up to this point we have discussed marginal accuracy of a classifier: it is also natural to consider *groupwise accuracy* such as the accuracy of a model when it predicts a particular class, or when it makes a prediction conditioned on particular attribute values. In this thesis, we assume that group membership of each data point is pre-computed, for example with its predicted label, its model score, or its associated attributes. Given the group membership of each data point, the input space is partitioned into regions $\{R_1, R_2, \cdots, R_g, \cdots, R_G\}$.

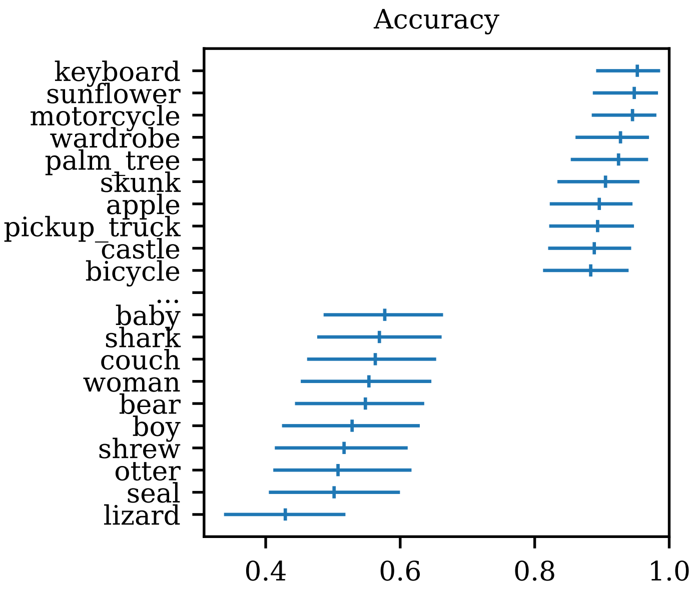


Figure 2.2: Estimated accuracy per class of a ResNet-110 image classifier on the CIFAR-100 test set, using our Bayesian assessment framework, with posterior means and 95% credible intervals per class.

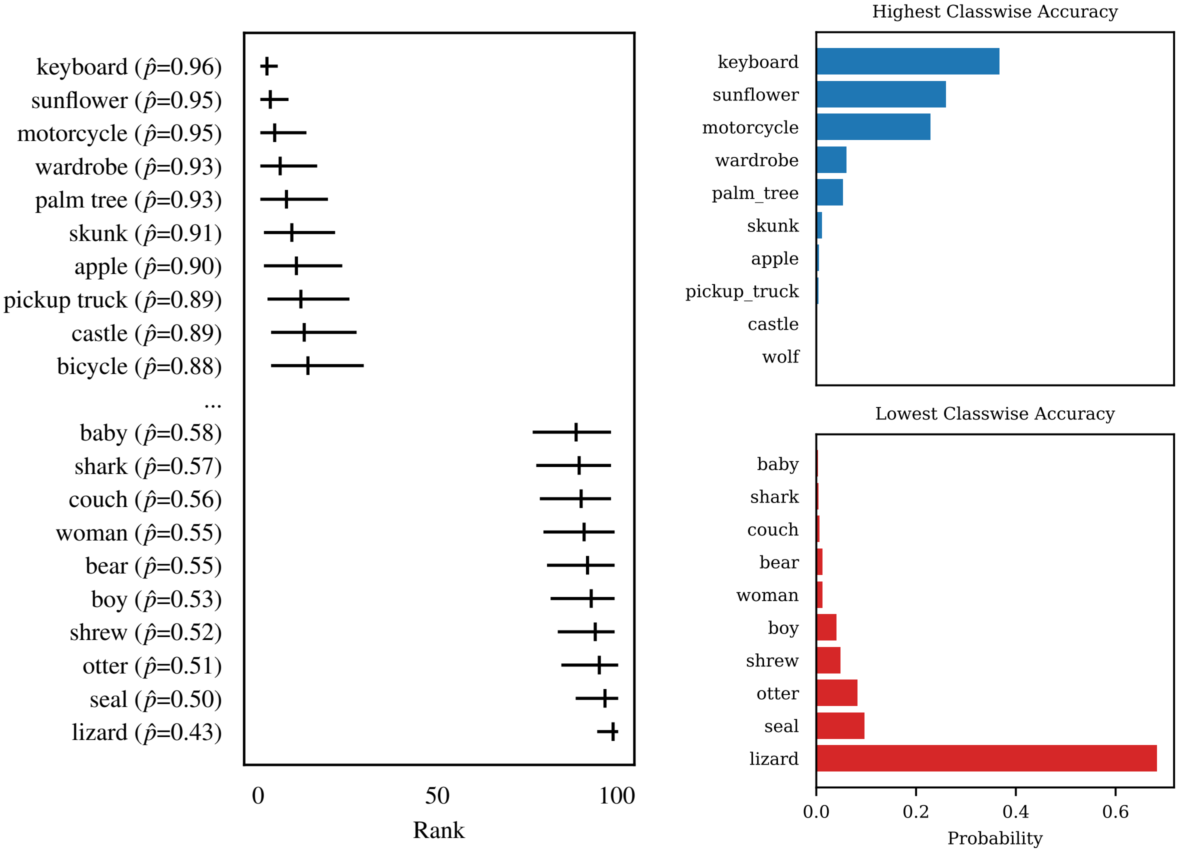
The groupwise accuracies can be treated as G independent unknown Bernoulli parameters $\{\theta_g, g = 1, 2, \cdots, G\}$. Labeled observations $(\mathbf{x}_{gi}, y_{gi}), i = 1, \ldots, N_g$, are sampled randomly per group conditioned on $\mathbf{x}_{gi} \in \mathcal{R}_g$, leading to a binomial likelihood with binary accuracy outcomes $z_{gi} = \mathbb{1}(y_{gi}, \hat{y}_{gi}) \in \{0, 1\}$:

$$\theta_g \sim \text{Beta}(\alpha_g, \beta_g), g = 1, 2, \cdots, G \tag{2.9}$$

$$z_{gi} \sim \text{Bern}(\theta_g), i = 1, 2, \cdots, N_g \tag{2.10}$$

In particular, as illustrated in Figure 2.2 a special case is *classwise accuracy*, the expected accuracy of the model whenever it predicts class k $(\theta_k, k = 1, \ldots, K)$, where the groups are classes, and g and G are replaced notationally by k and K. This corresponds to having the input region be the classifier's decision region $\mathcal{R}_k = \{\mathbf{x} | \hat{y} = k\}$. The results show that there is large amount of uncertainty for the classwise accuracies of CIFAR-100. For example, for the most accurate classes such as *keyboard, sunflower, motorcycle*, etc., with this amount of labeled data we can only state with confidence that the accuracy for these classes lies somewhere between about 0.87 and 0.98.

Inferring Statistics of Interest via Monte Carlo Sampling



(a) (b)

Figure 2.3: (a) MCMC-based ranking of accuracy across predicted classes for CIFAR-100 (where 1 corresponds to the class with the highest accuracy. (b) Posterior probabilities of the most and least accurate predictions on CIFAR-100. The class with the highest classwise accuracy is somewhat uncertain, while the class with the lowest classwise accuracy is very likely *lizard*.

An additional benefit of the Bayesian framework is that we can draw samples from the posterior to infer other statistics of interest. Here we illustrate this method with two examples.

Bayesian Ranking via Monte Carlo Sampling We can infer the Bayesian ranking of classes in terms of classwise accuracy or expected calibration error (ECE), by drawing samples from the posterior distributions [Marshall and Spiegelhalter, 1998]. For instance, we can estimate the ranking of classwise accuracy of a model for CIFAR-100, by sampling $\hat{\theta}_k$'s (from their respective posterior beta densities) for each of the classes and then computing the rank of each class with the sampled accuracy. We run this experiment 10,000 times and then for each class we can empirically estimate the distribution of its ranking. The MPE and 95% credible interval of ranking per predicted class for the top 10 and bottom 10 are provided in Figure 2.3a for CIFAR-100.

Posterior probabilities of the most and least accurate predictions We can estimate the probability that a particular class such as *lizard* is the least accurate predicted class of CIFAR-100 by sampling $\hat{\theta}_{k^*}$'s (from their respective posterior beta densities) for each of the classes and then measuring whether $\hat{\theta}_{lizard}$ is the minimum of the sampled values. Running this experiment 10,000 times and then averaging the results, we determine that there is a 68% chance that *lizard* is the least accurate class predicted by ResNet-110 on CIFAR-100. For the most accurate class, there is more uncertainty, with *keyboard, sunflower, and motorcycle* each having probability around 0.3 of being the most accurate predicted class. The posterior probabilities for other classes are provided in Figure 2.3b, along with results for estimating which class has the highest classwise accuracy.

2.3.2 Bayesian Reliability Diagrams

Another application of Bayesian groupwise accuracy estimation is to estimate **reliability diagrams**, in which each group g corresponds to a bin b partitioned by model scores. We can use a beta prior on each θ_b and define a binomial likelihood on the label outcome $z_i = \mathbb{1}(y_i = \hat{y}_i)$ within each bin (i.e., whether a model's predictions are correct or not on each example in a bin), resulting in posterior beta densities for each θ_b.

Figure 2.4 shows Bayesian reliability diagrams for five datasets based on different amounts of labeled data. We used a weak beta prior with pseudocount $\alpha_b + \beta_b = 2$ centered on the diagonal. Column 1 and 2 display reliability diagrams estimated using $N = 100$ and $N = 1000$ randomly selected examples (respectively). Column 3 displays diagrams estimated using the full set of available labeled examples for each dataset (e.g., the *size* column in Table 1).

With the full set of examples (column 3), the posterior means and the posterior 95% credible intervals are generally below the diagonal, i.e., we can infer with high confidence that the models are miscalibrated (and overconfident, to varying degrees) across all five datasets. For some bins where the scores are less than 0.5, the credible intervals are wide due to little data, and there is not enough information to determine with high confidence if the corresponding models are calibrated or not in these regions. With $N = 100$ examples (column 1), the posterior uncertainty captured by the 95% credible intervals indicates that there is not yet enough information to determine whether the models are miscalibrated given only $N = 100$ labeled examples. In addition, the frequency-based and Bayesian estimates often disagree. The frequency-based estimates are noisy and don't provide any notion of uncertainty. The Bayesian MPE estimates are also noisy but are more plausible given the smoothing from the prior. With $N = 1000$ examples (column 2) there is enough information to reliably infer that the CIFAR-100 model is overconfident in all bins for scores above 0.3. For the remaining datasets the credible intervals are generally wide enough to encompass 0.5 for most bins, meaning that we do not have enough data to make reliable inferences about calibration, i.e., the possibility that the models are well-calibrated cannot be ruled out without acquiring more data. When the full test dataset is used (column 3), the frequency based estimates (blue) and Bayesian estimates (red) are in close agreement.

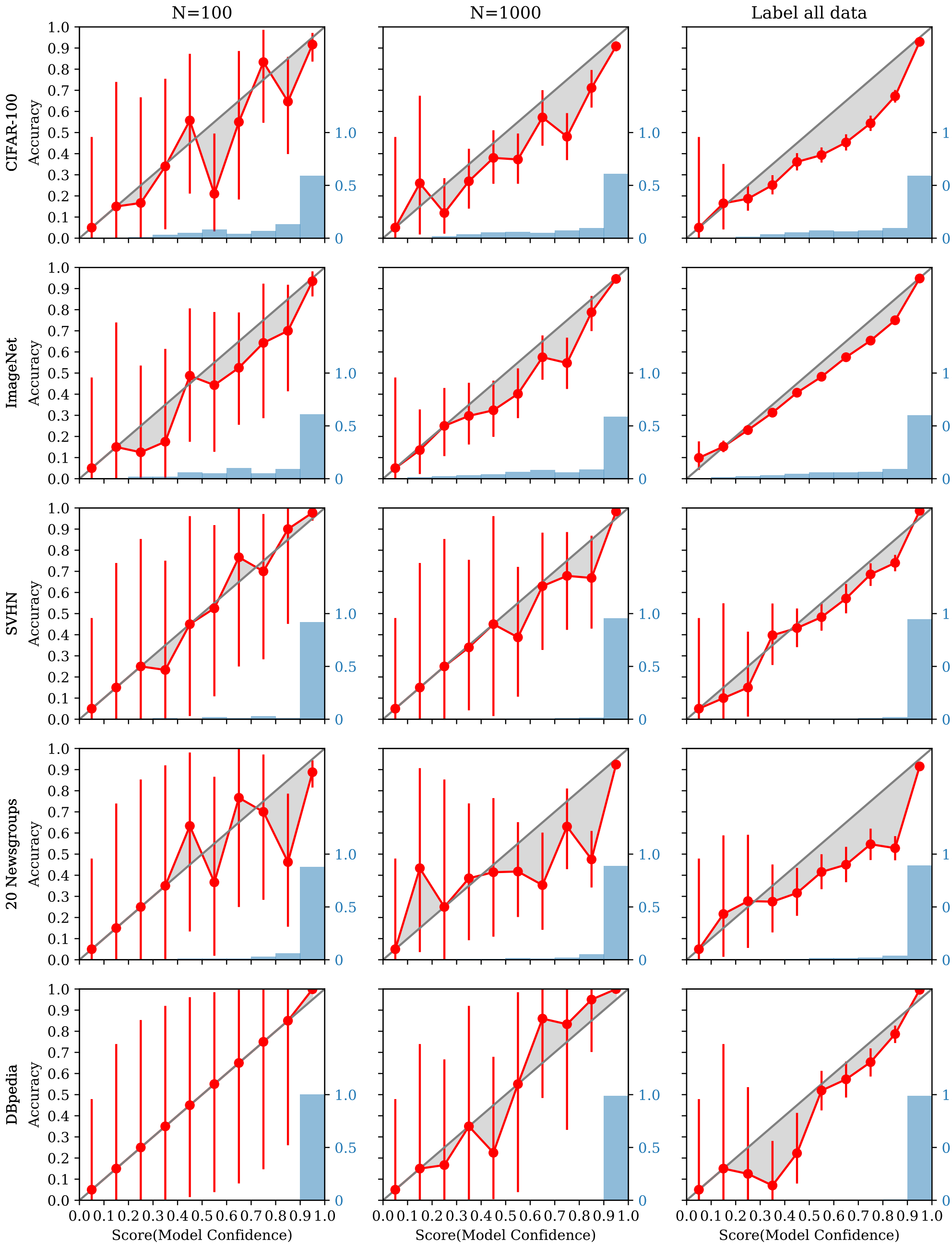


Figure 2.4: Bayesian reliability diagrams for five datasets (rows) estimated using varying amounts of test data (cols). The red circles plot the posterior mean for θ_b under our Bayesian model. Red bars display 95% credible intervals. Shaded gray areas indicate the estimated magnitudes of the calibration errors relative to the Bayesian estimates. The blue histogram shows the distribution of the scores for N randomly drawn samples.

2.3.3 Bayesian Calibration Error

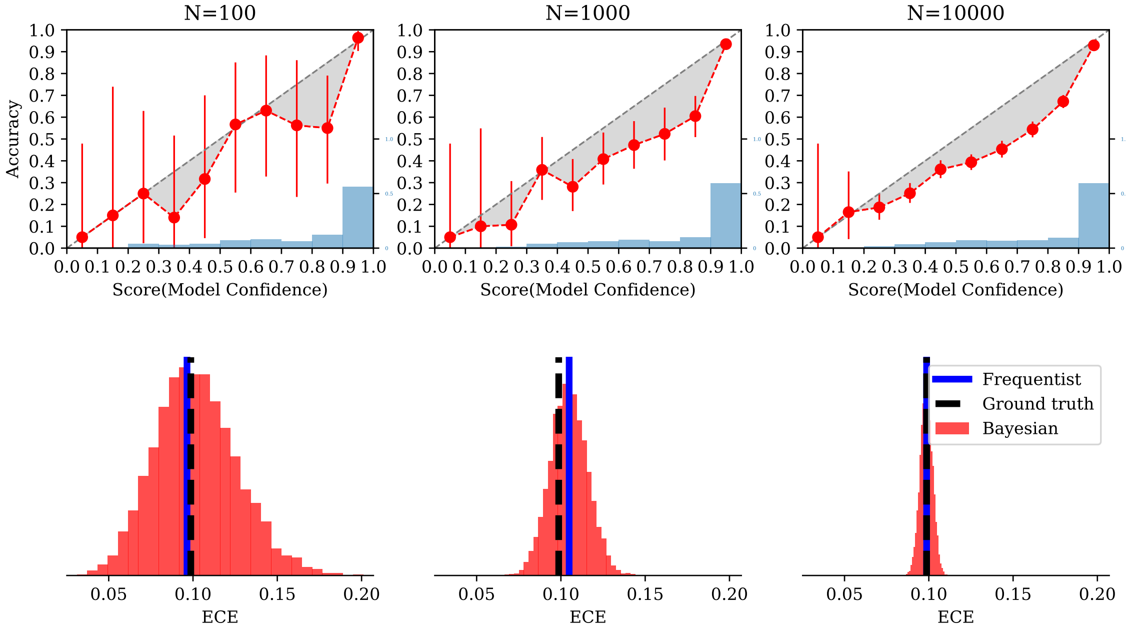


Figure 2.5: Bayesian reliability diagrams (top) and posterior densities for ECE (bottom) for CIFAR-100 as the amount of data used for estimation increases. Vertical lines in the right plots depict the ground truth ECE (black, evaluated with all available assessment data) and frequentist estimates (blue). The red histogram summarizes N=1000 posterior Monte Carlo samples for ECE using the Bayesian procedure.

We can also assess calibration-related metrics for a classifier in a Bayesian fashion using any of the well-known various calibration metrics [Kumar et al., 2019, Nixon et al., 2019]. For example, with a beta prior on θ_b, the posterior distribution over ECE is a weighted average of the absolute value of B shifted beta posterior distributions corresponding to the individual θ_b's:

$$\theta_b \sim \text{Beta}(\alpha_b, \beta_b), b = 1, 2, \cdots, B \tag{2.11}$$

$$\text{ECE} = \sum_{b=1}^{B} p_b |\theta_b - s_b| \tag{2.12}$$

The posterior is not available in closed form but Monte Carlo samples are straightforward to obtain, by drawing samples from the posterior distributions of θ_b and computing ECE with Eqn 2.12.

An illustrative example for CIFAR-100 is shown in Figure 2.5, the prior distribution of marginal accuracy within each bin is a beta distribution with its mean on the diagonal and pseudocount $\alpha + \beta = 2$. As the amount of data used increases, the credible intervals of the Bayesian reliability diagram (left column) get narrower, the posterior density of ECE (right column) converges to ground truth, and the uncertainty about ECE decreases. When the number of samples is small, with the same set of randomly selected samples 100 samples (row 1), the Bayesian estimation of ECE puts non-negative probability mass on ground truth marginal ECE, where “ground truth" refers to the marginal ECE computed with all labeled assessment data, while the frequentist method significantly overestimates ECE without any notion of uncertainty.

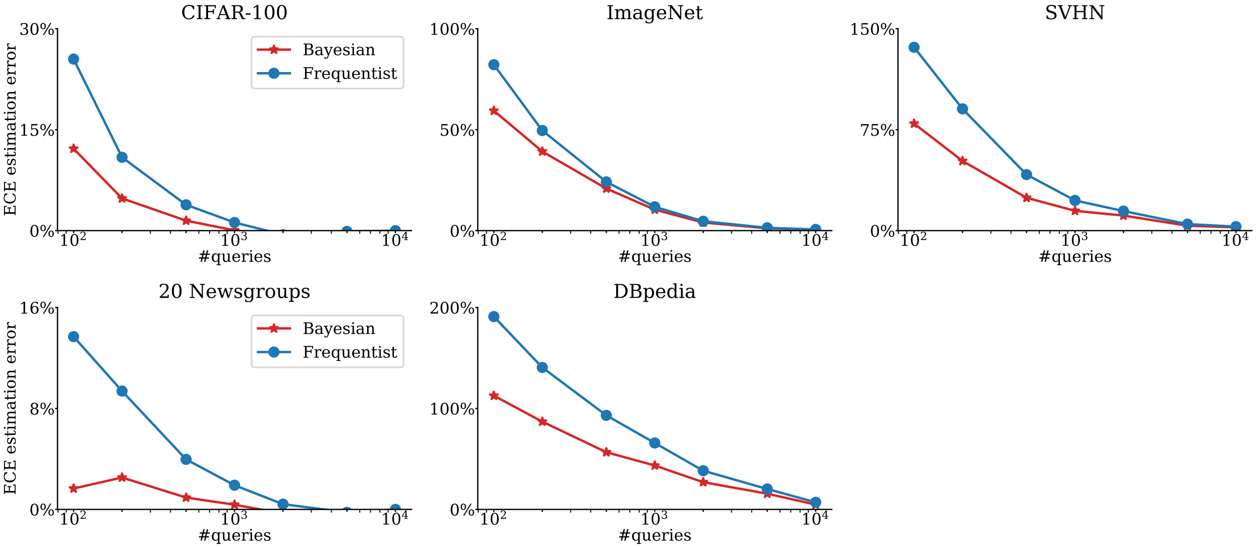


Figure 2.6: Percentage error in estimating expected calibration error (ECE) as a function of dataset size, for Bayesian (red) and frequentist (blue) estimators, across five datasets.

In Figure 2.6 we show the percentage error $(\text{ECE}_N - \text{ECE}^*)/\text{ECE}^*$ obtained for Bayesian mean posterior estimates (MPE) and frequentist estimates of marginal ECE as a function of the number of labeled data points N across five datasets. The percentage is computed relative to the ground truth marginal $\text{ECE} = \text{ECE}^*$, where ECE^* is computed as the ECE on the full test data (Table 2.1) with the number of bins set as 10. The MPE is computed with Monte Carlo samples from the posterior distribution (examples of histograms of such samples are shown in Figure 2.5). At each step, we randomly draw and label N queries from the pool of unlabeled data and compute both a Bayesian and frequentist estimate of marginal calibration error with these labeled data. We run the simulation 1000 times and report the average ECE_N over the N samples. Figure 2.6 shows that the Bayesian MPE estimates are significantly and systematically more accurate in estimating ECE across all five datasets, particularly when there is relatively little labeled data available.

Bayesian Estimation of ECE per Class

Similar to classwise accuracy, we can also model *classwise ECE*,

$$\mathrm{ECE}_k = \sum_{b=1}^{B} p_{kb} |\theta_{kb} - s_{kb}| \tag{2.13}$$

by modifying the model described above to use regions $\mathcal{R}_{kb} = \{\mathbf{x} | \hat{y} = k, s(\mathbf{x}) \in \mathcal{R}_b\}$ that partition the input space by predicted class in addition to partitioning by the model score. p_{kb} is the probability of $\mathcal{R}_{kb}$ conditioned on $\mathcal{R}_k$, i.e. the probability of the b-th bin for predicted class k, and s_{kb} is the expected model scores of $\mathcal{R}_{kb}$. This follows the same procedure as for "marginal ECE" in the previous subsection except that the data is now partitioned by predicted class $k = 1, \ldots, K$, and a posterior density on ECE_k for each class is obtained.

We performed Bayesian inference about both classwise accuracy and ECE performance. We used beta priors with $\alpha_k = \beta_k = 1, k = 1, \ldots, K$ for classwise accuracy, and $\alpha_b = 2s_b, \beta_b = 2(1 - s_b), b = 1, \ldots, K$ for binwise accuracy. Figure 2.7 plots the resulting mean posterior estimates (MPEs) and 95% credible intervals (CIs) for accuracy and ECE values for each of the predicted classes for 5 datasets. The accuracies and ECE values of the model vary substantially across classes, and classes with low accuracy tend to be less calibrated. There is also considerable posterior uncertainty for these metrics even using the whole test set across

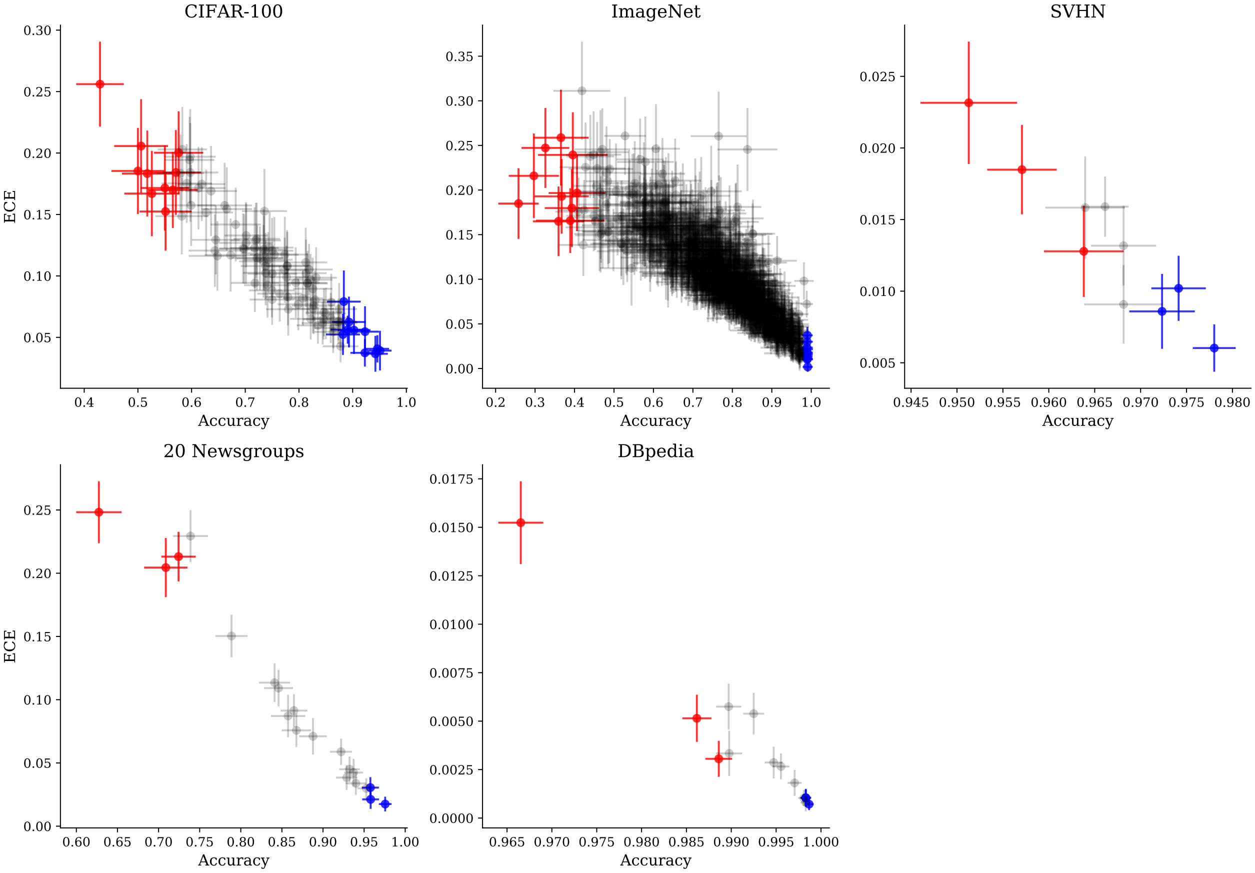


Figure 2.7: Scatter plots of classwise accuracy and ECE for 5 datasets. Each marker represents posterior means and 95% credible intervals of posterior accuracy and ECE for each predicted class. Markers in red and blue represent the top-m least and most accurate predicted classes, markers in gray represent the other classes, with $m = 10$ for CIFAR-100 and ImageNet, and $m = 3$ for the other datasets.

5 datasets. For CIFAR-100, ImageNet and 20 Newsgroups, the variance of classwise accuracy and ECE among all predicted class is considerably greater than the variance of two other datasets. The results also suggest that measuring classwise ECE is important for model performance assessment because of its high variance among predicted classes.

2.3.4 Bayesian Estimation of Confusion Matrices

In a manner similar to using a beta-binomial distribution to model accuracy, we can model confusion probabilities using a Dirichlet-multinomial distribution. Conditioned on a predicted class $\hat{y} = k$, the true class label y has a categorical distribution $\theta_{jk} = p(y = j|\hat{y} = k)$. We will refer to θ_{jk} as confusion probabilities, since they resemble the elements of a confusion matrix. For the i-th data point classified as $\hat{y}_i = k$, we can model the confusion probabilities and the generative process of the label outcome $z_i = y_i$ with a Dirichlet-Multinomial model:

$$\theta_{\cdot k} \sim \text{Dirichlet}(\alpha_{\cdot k}), k = 1, 2, \cdots, K \tag{2.14}$$

$$z_{ki} \sim \text{Cat}(\theta_{\cdot k}), i = 1, 2, \cdots, N_k. \tag{2.15}$$

There are $\mathcal{O}(K^2)$ parameters in total in K Dirichlet distributions, each of which is parameterized with a K dimensional vector $\alpha_{\cdot k}$. We will return to the topic of confusion matrix estimation in Section 2.4 where we discuss the effect of both informative and uninformative priors for this task.

2.3.5 Bayesian Misclassification Costs

Accuracy assessment can be viewed as implicitly assigning a binary cost to model mistakes, i.e. a cost of 1 to incorrect predictions and a cost of 0 to correct predictions. In this sense, the predicted class with the lowest accuracy is equivalent to the class with the greatest expected cost. However, in real-world applications, the costs of different types of mistakes can vary drastically. For example, in autonomous driving applications, misclassifying a pedestrian as a crosswalk can have much more severe consequences than other misclassifications.

To deal with such situations, we extend our approach to incorporate an arbitrary cost matrix $\mathbf{C} = [c_{jk}]$ to assign cost to different misclassifications, where c_{jk} is the cost of predicting class $\hat{y} = k$ for a data point whose true class is $y = j$. The **classwise expected cost** for predicted class k is given by:

$$C^M_{\mathcal{R}_k} = \mathbb{E}_{p(\mathbf{x},y|\mathbf{x}\in\mathcal{R}_k)}[c_{jk}\mathbb{1}(y=j)] = \sum_{j=1}^{K} c_{jk}\theta_{jk}. \tag{2.16}$$

The posterior of $C^M_{\mathcal{R}_k}$ is not available in closed form but Monte Carlo samples are straightforward to obtain, by randomly sampling $\theta_{\cdot k} \sim \text{Dirichlet}(\alpha_{\cdot k})$ and computing $C^M_{\mathcal{R}_k}$ deterministically with the sampled $\theta_{\cdot k}$ and the predefined cost matrix $\mathbf{C}$. We will return to the topic of costs in Chapter 3 where we develop label-efficient algorithms to determine which classes have the largest expected cost for a model M.

2.3.6 Bayesian Estimation of Accuracy Differences

Comparison of performance between two groups is another important assessment task. For example, when assessing a model's group fairness, we are interested in the difference in model accuracy between different ethnic groups and genders [Hardt et al., 2016]; when deciding between 2 clinical trials, the difference of their success rate and the uncertainty of the difference are important to patients. In both cases, obtaining labeled data is expensive, and the uncertainty of the performance metrics is important for interpretability of the assessment.

Consider two groups g_1 and g_2 with a true accuracy difference $\Delta = \theta_{g_1} - \theta_{g_2}$. Our approach uses the "rope" (region of practical equivalence) method of Bayesian hypothesis testing (e.g., Benavoli et al. [2017]) as follows. The cumulative density in each of three regions $\mu = (P(\Delta < -\epsilon), P(-\epsilon \leq \Delta \leq \epsilon)$,$P(\Delta > \epsilon))$ represents the posterior probability that the accuracy of group g_1 is more than ϵ lower than the accuracy of g_2, that the two accuracies are "practically equivalent," or that g_1's accuracy is more than ϵ higher than that of g_2, where ϵ is user-specified. In our experiments we use $\epsilon = 0.05$ and the cumulative densities μ are estimated with 10,000 Monte Carlo samples. The assessment task is to identify the region

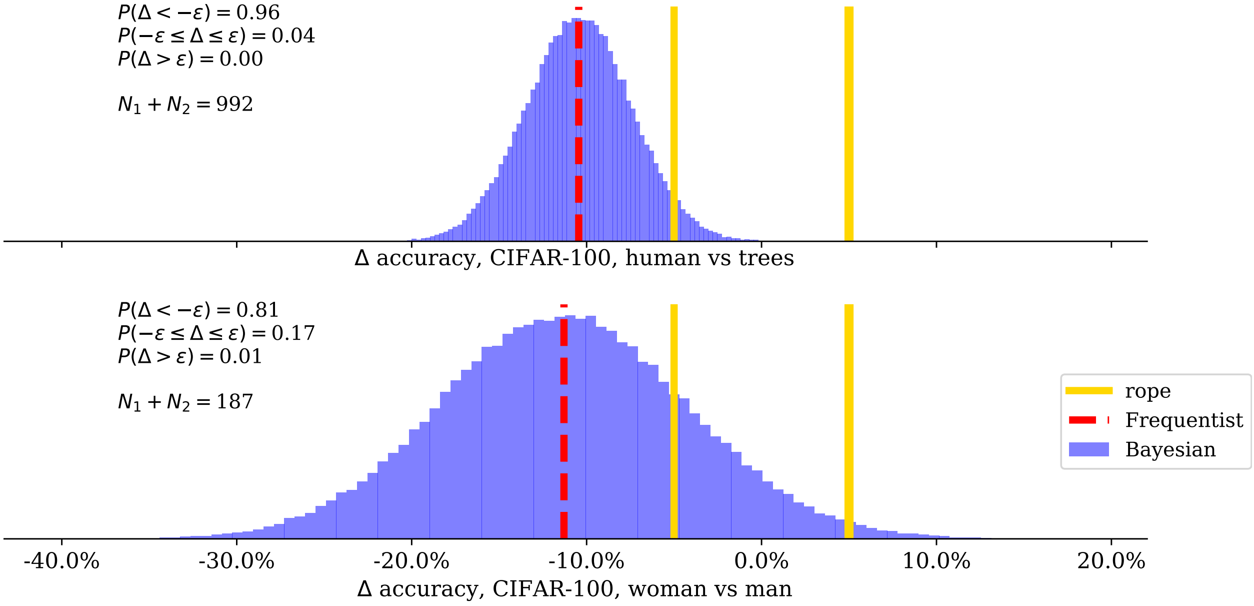


Figure 2.8: Density plot for the differences of accuracy between two superclasses/classes of CIFAR-100. Region of practical equivalence is [-0.05, 0.05]. Vertical solid lines in gold plots the region of practical equivalence, vertical red dashed line plots the frequentist estimate of accuracy difference. Left: two groups are predicted superclass "human" and "trees" in CIFAR100. Right: two groups are predicted classes "woman" and "man" in CIFAR-100.

$\eta = \arg\max(\mu)$ in which Δ has the highest cumulative density, where $\lambda = \max(\mu) \in [0, 1]$ represents the confidence of the assessment.

Bayesian estimation of group differences allows us to compare the performance between two groups with uncertainty. For example, with prior distribution $\text{Beta}(1, 1)$ and the full test set of CIFAR-100, the posterior distribution of groupwise accuracies of ResNet-110 on superclass "human" and "trees" are $\theta_{g_1} \sim \text{Beta}(280, 203)$ and $\theta_{g_2} \sim \text{Beta}(351, 162)$ respectively. The total amount of labeled data for the two superclasses are 481 and 511. With random samples from the posterior distributions of θ_{g_1} and θ_{g_2}, we can simulate the posterior distribution of the difference in accuracy between "human" and "trees" $\Delta = \theta_{g1} - \theta_{g2}$ and compute its cumulative density in each of three regions $\mu = (P(\Delta < -\epsilon), P(-\epsilon \leq \Delta \leq \epsilon), P(\Delta > \epsilon))$. In Figure 2.8, when $\epsilon = 0.05$, ResNet-110 is less accurate on the predicted superclass "human" than on "trees" with 96% probability. Similarly with 82% probability, the accuracy of ResNet-110 on "woman" is lower than "man". Although the point estimates of the two performance differences have values that are both approximately 10%, the assessment of "human" vs. "tree" is more certain because more samples are labeled. We will return to the topic of Bayesian assessment of group differences, in the specific context of algorithmic fairness, in the next chapter.

2.4 Uninformative and Informative Priors

To conduct Bayesian inference for the performance metrics, we need to specify the prior distribution of the metric $p(\theta)$ and the generative model for the label outcome $q_\theta(z)$. While $q_\theta(z)$ is either a Bernoulli or categorical distribution parametrized by θ for the metrics we discussed before, for the priors, which are beta and Dirichlet distributions for all of the metrics we considered, there is freedom in terms of how the parameters of the priors are set. In this section we discuss the use of both informative and uninformative priors in this context.

For accuracy-based metrics, e.g. groupwise accuracy, reliability diagram, ECE, accuracy difference, we use a beta prior for each group g, i.e. $p(\theta_g) = \text{Beta}(\theta_g; \alpha_g, \beta_g)$. For a confusion-based metric, e.g. the confusion matrix and misclassification cost, since we mainly focus on confusion probabilities among different classes, and the data space is partitioned into K groups by predicted labels, we use K instead of G to index the groups. For class k, $\theta_{\cdot k} \sim \text{Dirichlet}(\alpha_{\cdot k})$. $N_0 = \alpha_g + \beta_g$ and $N_0 = \|\alpha_{\cdot k}\|_1$ are the pseudocounts, i.e. *strength* of the beta prior and Dirichlet prior. We assume that the distributions of θ_g are independent across different groups. Modeling the dependency across different groups, which could potentially be useful when the number of groups is large, is an interesting direction for future work.

For setting the location of the mean of the prior, we discuss both uninformative and informative priors in this section. We discuss setting the strength of the prior N_0 in Chapter 3.

2.4.1 Uninformative Priors

An uninformative prior can be created to reflect a balance among outcomes when no information is available. The uninformative prior distribution we use is $\alpha = \beta = N_0/2$. For example, when $N_0 = 2$, the prior distribution $p(\theta)$ is a uniform distribution over $[0, 1]$. When $N_0 = 1$, we have Jeffrey's prior on θ [Jeffreys, 1946]. For the uninformative prior for a confusion matrix, we use $\alpha_{jk} = N_0/K, \forall j, k$.

2.4.2 Use Self-Assessment of Classifiers as Informative Priors

If the classification model M is binary calibrated, model scores $s(\mathbf{x})$ provide the estimated accuracy of the classifier; if the model M is multi-class calibrated, model outputs $p_M(y = k|\mathbf{x})$ can be used to compute the confusion probabilities of the classifier. Even if we believe a model is not calibrated, as is often the case in practice, we can nonetheless use the model scores as a prior and allow the label data to overcome this prior due to any miscalibration. As we will see below, the model scores capture quite a bit of useful information about the structure of the confusion matrix, even for miscalibrated models.

For groupwise accuracy, the informative beta prior for each group is $\text{Beta}(N_0 s_g, N_0(1 - s_g))$ where s_g is the average model confidence for group g. In our experiments, s_g is computed as the empirical average of the model's scores for group g over the unlabeled test data (Table 2.1). As an informative prior for a confusion matrix, we compute $\alpha_{\cdot k}$ as the expectation of model's own outputs over $\mathcal{R}_k$:

$$\alpha_{jk} \propto \Sigma_{\mathbf{x} \in \mathcal{R}_k} p_M(y = j|\mathbf{x}). \tag{2.17}$$

In experiments, $\alpha_{\cdot k}$ is computed with the unlabeled test data (Table 2.1).

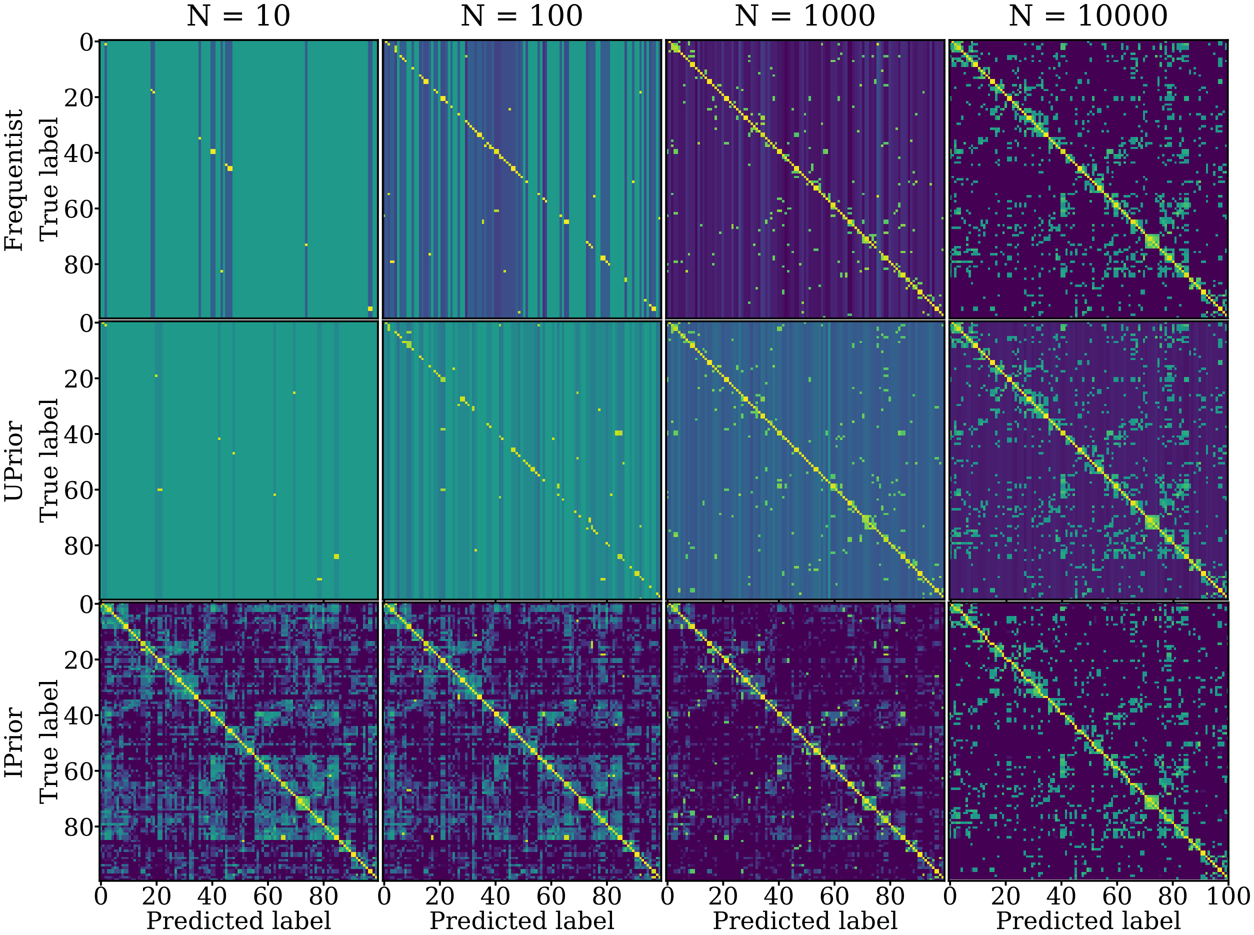


Figure 2.9: Estimates of the confusion probabilities of ResNet-110 on CIFAR-100, comparing the frequentist estimates (Frequentist), and mean posterior estimates of the Bayesian method under uninformative prior(UPrior) and informative priors(IPrior), with $N = 10, 100, 1000, 10000$ randomly selected labeled data point.

2.4.3 Illustative Results

To illustrate how the informative prior helps deal with sparsity, we plot the estimates of the confusion matrix of ResNet-110 on CIFAR-110 obtained with the frequentist method and the Bayesian method in Figure 2.9. The number of parameters to estimate $\mathcal{O}(K^2)$ is approximately the same as the number of labeled samples that are available. The strength of both priors are 1. As was shown earlier in Figure 2.1, the informative prior is based on a model that is significantly miscalibrated. For the frequentist and UPrior estimates, when the number of labeled data N is less than 1000, the structure of the confusion matrix is barely captured, due to sparsity of the data relative to the number of parameters to be estimated. On the other hand, the IPrior does a much better job of capturing the structure of the confusion matrix, even with only a small number of labeled examples (e.g., first and second columns).

2.5 Related Work

Assessment of Blackbox Models. The assessment tasks we investigated in this chapter have been studied in different contexts, most of which focus on defining and optimizing the related metrics. For example, many papers, e.g. Guo et al. [2017] show that complex models such as deep networks with high-dimensional inputs (e.g., images and text) can be significantly overconfident in practice. Nixon et al. [2019] provide a review of different ways to measure miscalibration apart from ECE. Different methods have been developed to recalibrate classification models, e.g. Naeini et al. [2015], Guo et al. [2017], Kull et al. [2017]. The work described in this chapter is different to this earlier work in that it focuses on a different question, i.e., assessing a trained blackbox model with Bayesian methods. For example for calibration, our objective is to assess calibration and quantify the uncertainty of our assessment, instead of trying to learn a more calibrated model.

Assess with Uncertainty Prior work on using Bayesian ideas in the context of classifier assessment has tended to focus on very specific types of assessment. Goutte and Gaussier [2005] propose a framework for Bayesian estimation of precision, recall, and F-score in an information retrieval context, and Johnson et al. [2019] use Bayesian mixture models to provide posterior distributions of diagnostic metrics (such as true positive rates) for medical tests. Benavoli et al. [2017] develop a general Bayesian framework for comparing multiple classifiers as an alternative to more traditional null hypothesis significance testing. Our proposed approach shares a similar philosophy with this earlier Bayesian work in terms of treating classifier performance metrics as parameters of interest about which we can perform Bayesian inference. However, our framework is significantly more general than this prior work, encompassing a broader range of metrics such as classwise performance metrics and calibration metrics such as ECE.

Other work has proposed frequentist methods for uncertainty quantification in an assessment context, e.g., resampling approaches such as the bootstrap for generating confidence intervals on calibration performance [Bröcker and Smith, 2007, Vaicenavicius et al., 2019, Kumar et al., 2019]. Our focus in this work is not to supplant these existing techniques, but instead to supplement them by developing a Bayesian approach that includes the ability to incorporate prior knowledge.

2.6 Conclusions

This chapter describes a Bayesian framework for assessing performance metrics of black-box classifiers. Our contributions are:

- We developed Bayesian techniques for estimating groupwise accuracy, reliability diagram, expected calibration error (ECE), confusion matrix, misclassfication cost, and performance difference;
- We discussed using self-assessment of prediction models as informative priors for Bayesian assessment;
- We illustrated a number of different ways that the framework can be used to understand performance aspects of some widely-used deep learning models and datasets.

There are a number of potential extensions of the approach for future work such as Bayesian estimation of continuous functions related to accuracy and calibration. In the next chapter, we show that our Bayesian assessment framework readily lends itself to active assessment of blackbox classifiers.

Chapter 3

Active Bayesian Assessment of Black-Box Classifiers

In the previous chapter, we used Bayesian methods to assess black-box classifiers with uncertainty quantification, and demonstrated that the uncertainty of the estimates is high with a small amount of labeled data. In real-world deployment scenarios, labeled data for assessment is likely to be scarce and costly to collect, e.g., for a model being deployed in a diagnostic imaging context in a particular hospital. However, by allocating labeling budget strategically among different regions of the input space, assessment can be carried out in a more data-efficient manner. With this in mind, in this chapter we develop a framework for **active Bayesian assessment** of black-box classifiers, using techniques from Bayesian active learning to adaptively select data points to label so that uncertainty of assessment can be reduced for different assessment tasks.

This chapter is organized as follows:

- In Section 3.1 we introduce the three types performance assessment tasks that we study in this chapter: estimation, identification and comparison.

- In Section 3.2 we first provide a brief review of multi-armed bandit problems and the Thompson sampling algorithm.

- Then we outline the *active assessment* method by applying Thompson sampling to the Bayesian assessment framework we proposed in Chapter 2, to address the assessment tasks in a label-efficient manner.

- In Section 3.4 we describe the settings for the experiments.

- Then in Section 3.5, 3.6 and 3.7, we discuss how specific reward functions r can be designed for different assessment tasks of interest, and demonstrate with empirical results that our active Bayesian assessment performs better than traditional methods.

- In Section 3.9 we provide additional experimental results, where we compare with other active learning methods, discuss choices for prior distributions, and describe sensitivity analysis results, and so on.

- In Section 3.10 and 3.11 we review the related work, summarize our contributions and discuss future directions.

3.1 Performance Assessment of a Blackbox Classifier

As in Chapter 2, we will use θ to indicate a **performance metric** of interest, such as classification accuracy, true positive rate, expected cost, calibration error, etc. Our approach to assess a metric θ relies on the notion of disjoint partitions of the input space. As discussed in Chapter 2, there are multiple partitions that are of interest in practice. We use a general term **group** to refer to the subsets created by the partition and index them with $g = 1, 2, \cdots, G$.

- When studying classwise performance, one grouping of particular interest is where groups correspond to a model's predicted classes, i.e., $g = k$, and the partition of the input space corresponds to the model's decision regions $\mathbf{x} \in \mathcal{R}_k$ for $k = 1, 2, \cdots, K$. If θ refers to classification accuracy, then θ_k is the accuracy per predicted class k. For prediction problems with costs, θ_k can be defined as the expected cost per predicted class and so on.

- When studying calibration properties of a classification model in Chapter 2, we discussed the groups g that correspond to bins b of a model's score, i.e., $s(\mathbf{x}) \in \text{bin}_b, b = 1 \ldots, B$, or equivalently $\mathbf{x} \in \mathcal{R}_b$ where $\mathcal{R}_b$ is the region of the input space where model scores lie in score-bin b. In this case $g = b$.

- Another example is when studying classwise ECE, we can also model ECE_k of the k-th predicted class by using regions $R_{kb} = \{\mathbf{x} | \hat{y} = k, s(\mathbf{x}) \in \text{Bin}_b\}$ that partition the input space by the predicted class in addition to the model score, i.e. $g = \{kb\}$. In an algorithmic fairness context, for group fairness [Hardt et al., 2016] the groups g can correspond to categorical values of a protected attribute such as gender or race, and θ can be defined (for example) as accuracy or true positive rate per group.

For any instantiation of groups g and metric θ, there are three particular **assessment tasks** we will focus on in this chapter: (1) estimation, (2) identification, and (3) comparison.

3.1.1 Estimation

For "estimation" , we mainly focus on risk estimation of black-box classifiers, which involves measuring the distribution of the disagreement between model predictions and true labels. It is important to assess the risk of a model to make informed decisions about the deployment of a predictive model. For example, two of the tasks we discussed in Chapter 2 fall in this category: (1) *groupwise accuracy estimation*: how often the model makes wrong predictions in different regions of the input space; (2) *confusion matrix estimation*: the classes that a classification model tends to confuse with.

For both of the assessment tasks, the objective is to minimize the estimation error of θ_g across G groups. Let $\theta_1, \ldots, \theta_G$ be the set of true (unknown) values for some metric θ and some grouping g. The goal of estimation is to assess the quality of a set of estimates $\hat{\theta}_1, \ldots, \hat{\theta}_G$ relative to the true values, where the estimates are based on a finite set of labeled data. In this chapter we will focus on the root mean square error (RMSE) loss $\left(\sum_{g=1}^{G} p_g(\theta_g - \hat{\theta}_g)^2\right)^{1/2}$ to measure the estimation quality, where the probability of each group $p_g = p(\mathbf{x} \in \mathcal{R}_g)$ can be estimated from unlabeled data.

3.1.2 Identification

Instead of having good estimates of groupwise metrics for all G groups, sometimes we are more interested in identifying the groups that the model has the best or the worst performance for. This can be motivated for example by task allocation, e.g., finding the m predicted classes that a model is least accurate on, so that whenever the model predicts one of these classes the prediction decision is handed instead to a more accurate predictor (e.g., a human). For example, suppose we have a dataset with $K = 100$ equally-likely classes (e.g., CIFAR-100) and a budget where we can send 10% of our examples to a human to make predictions (and the other 90% are made by our black-box model). One way to address this is to find the set of 10 predicted classes that the model is least accurate for and use the human to make predictions when $\hat{y}$ is in this set. Recent work by Raghu et al. [2019] illustrates this type of task allocation in a medical diagnosis context, showing that improvements in estimating fine-grained model performance can yield significant gains for task allocation automation. In their work the authors used a variety heuristics to identify which examples should be used for prediction by human or machine—in Section 3.6 we illustrates how to address such problems using Bayesian modeling and multi-armed bandits.

The goal of identification tasks is to identify extreme groups, e.g., $g^* = \arg\min_g \theta_g$, such as the predicted class with the lowest accuracy (or the highest cost, swapping max for min). We will also investigate methods for finding the m groups with highest or lowest values of a metric θ.

3.1.3 Comparison

As discussed in Section 2.3.6, the goal of performance comparison is to determine if the difference $\Delta = \theta_{g_1} - \theta_{g_2}$ between two groups g_1 and g_2 is statistically significant, e.g., to assess if accuracy or calibration for one group is significantly better than another group for some black-box classifier.

A measure of the quality of a particular assessment method in this context is to compare how often, across multiple datasets of fixed size, a method correctly identifies if a significant difference exists and, if so, its direction. This is of particular relevance for example in the case of groupwise algorithmic fairness, e.g. whether the model is equally accurate among different demographical groups.

3.1.4 Active Assessment

For each of the three assessment tasks above, we need to make inference about θ_g for all groups g. However, the strategy to allocate labeling budget among the groups is task-specific. For example, for risk estimation, in order to minimize RMSE, the budget allocated to each group should be proportional to the probability of each group p_g; while for extreme class identification, we should allocate more budget to the groups that are close to optimal regardless of the group weights.

In this chapter, we develop a unified framework to strategically allocate labeling budget for three tasks: estimation, identification, and comparison. Rather than relying on a random sample of labeled data points from the input distribution for inference, we propose to improve data efficiency by extending our Bayesian framework to support *active assessment* by actively selecting examples $\mathbf{x}$ for labeling in a data-efficient manner. Efficient active selection of examples for labeling is particularly relevant when we have a potentially large pool of unlabeled examples $\mathbf{x}$ available, and have limited resources for labeling (e.g., a human labeler).

In summary, the question we address in this chapter is: if we can only label N samples from a large pool of unlabeled examples that are partitioned into G groups, how should we select samples from G groups in a manner that is label-efficient for our each task?

3.2 Thompson Sampling for Multi-Armed Bandit Problems

In this section we first introduce the multi-armed bandit problems, and then provide a brief review of the Thompson sampling algorithm, which is the main technique we use for active Bayesian assessment.

3.2.1 Multi-armed Bandit Problems

The multi-armed bandit(MAB) problem was proposed to study the exploration-exploitation trade-off in sequential decision problems, and it has been extensively studied across different disciplines for decades [Lattimore and Szepesvári, 2020]. There is a colorful motivating story for MAB problems: image a gambler enters a casino full of different slot machines, each one with its own distribution of a *reward*. The distribution of the reward for each arm is not told in advance. She needs to choose an arm to pull at each time, i.e. sequentially take *actions*, to maximize her cumulative reward. As the gambler gradually learns about the reward of each arm by observing the outcome of each pull of the arms, she starts to face the *exploration-exploitation trade-off*: by *exploring* the arms that she knows less about their reward, she might find an arm that yields higher reward than the other known arms; or by *exploiting* the arms which she already knows that have high expected reward, she is expected to earn a high reward.

Apart from action and reward, *budget* is the other important concept for multi-armed bandit problems. If the gambler has unlimited budget, i.e. she can stay in the casino forever and pull the multi-armed bandit machine for an infinite amount of times, one of the trivial yet optimal solution is to keep exploring until she finds one arm with a positive expected reward. Then by continuing to pull this arm, her expected cumulative reward is guaranteed to be positive infinite. In this story, the budget is the number of times that the gambler can pull the arms. However, when the budget is finite, as is always in the case in the real world, it is no longer a trivial problem to decide when she should switch from exploration to exploitation, such that she can walk out the casino with the maximum amount of cumulative reward. Mathematically, the multi-armed bandit problem can be summarized as follows:

Definition 3.2.1. *Suppose there are G actions, for $g = 1, 2, \cdots, G$, the reward of each action g is a random variable $\mathbf{r_g}$ whose distribution is unknown at the beginning. At the i-th step, if the selected action $a_i = \hat{g}$, by applying the action, a reward r_i, which is a realization of $\mathbf{r_{\hat{g}}}$, is received. The objective of the multi-armed bandit problem is to maximize the cumulative reward $\Sigma_{i=1}^{N} r_i$ over N actions.*

In general, instead of modeling $\mathbf{r_{\hat{g}}}$ directly, we assume there is a stochastic *outcome* $\mathbf{z_i}$ for the action $a_i = \hat{g}$. Given the outcome z_i, the corresponding reward is determined by the reward function $r_i = r(z_i|\hat{g})$. In the casino example, z_i would be the binary outcome of whether the gambler wins when she pulls the $\hat{g}$-th arm at the i-th step, and r_i is the corresponding amount of reward.

The multi-armed bandit problem has been studied since World War II and finding optimal algorithms has proven to be challenging [Russo et al., 2018]. The first steps towards this problem were taken shortly after the war. Robbins [1952] developed the "win stay, lose shift" strategy, which is simple, non-optimal, but with provable guarantee that it is better than random. Gittins and Jones [1979] proposed an algorithm named "the Gittins index", which gives an optimal solution under Bayesian setting when the prior is known and the reward geometrically discounts as time passes. However, this method is mathematically intractable.

Since then an enormous body of work has been accumulated. Lattimore and Szepesvári [2020] provides a throughout review. Among all the methods, Thompson sampling [Thompson, 1933, 1935] is a natural randomized Bayesian algorithm to heuristically balance between exploration and exploitation, which has recently gained its popularity because of its strong empirical performance on a spectrum of applications [Scott, 2010, Chapelle and Li, 2011, Agrawal and Goyal, 2012].

3.2.2 Thompson Sampling

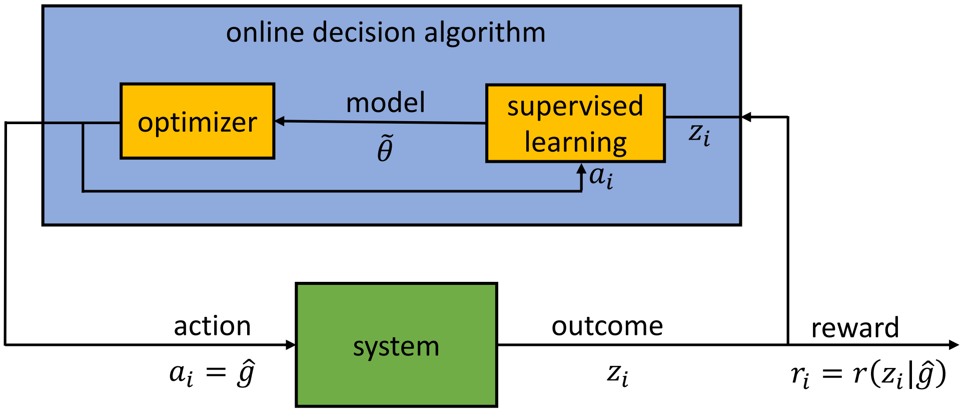


Figure 3.1: Online decision algorithm.

Online Decision Algorithms For multi-armed bandit problems, since exploration is gathering information and exploitation is using the information to get an expected good reward, the decisions made early on influences how informed the later decisions would be. *Online decision algorithms* have been studied to serve as mechanisms to make this type of sequential decisions.

Figure 3.1[1] illustrates how an online decision algorithm works. At the i-th step, the previous $i-1$ actions together with their outcomes $H_{i-1} = \{(a_j, z_j) | j = 1, 2, \cdots, i-1\}$ form the training data for the *supervised learning* model, where a_j and z_j are the action and the outcome at the j-th step.

The supervised learning model parametrized by θ predicts the outcome z of each action g, $g = 1, 2, \cdots, G$. For example, in the casino example θ is a the set of G unknown probabilities, where θ_g is the chance of success for the g-th arm of the slot machine, and the predictive distribution of the outcome is $z \sim \text{Bern}(z|\theta_g)$ according to the supervised model. For each possible outcome z of action g, the corresponding reward is $r(z|g)$, where $r(\cdot|g)$ is the deterministic reward function for the g-th action.

By fitting to H_{i-1}, the supervised learning model generates a point estimation $\widetilde{\theta}$ of the model parameter. For a frequentist supervised learning model, θ is already a point estimation of the model parameter, i.e. $\widetilde{\theta} = \theta$. For a Bayesian supervised learning model, $\widetilde{\theta}$ can be the mean posterior expectation or a posterior sample of the posterior distribution of θ. In this chapter, the Thompson sampling method we use sets $\widetilde{\theta}$ as a posterior sample of θ.

Given the estimated rewards computed with $\widetilde{\theta}$, the *optimizer* selects the action a_i that maximizes the expected reward computed calculated by taking the integral over all possible outcomes z. By applying the action $a_i = \hat{g}$, the *system* generates an outcome z_i, and the agent receives the corresponding reward $r_i = r(z_i|\hat{g})$.

[1] Adapted from Figure 2.1 of Russo et al. [2018].

Thompson Sampling: a Bayesian Online Decision Algorithm Thompson sampling is a Bayesian online decision algorithm. Since it was first proposed by Thompson [1933], it has been reinvented under different contexts multiple times [Wyatt, 1998, Strens, 2000]. The basic idea of Thompson sampling is to assume a simple prior distribution on the parameters of the reward distribution of every arm, and at each step, play an arm according to its posterior probability of being the best arm.

At each step i, the prior distribution of the model parameter θ is its posterior at the previous step, i.e. $\theta \sim p_{i-1}(\theta)$. The outcome z of an action g is modeled by a conditional distribution parametrized by θ, i.e. $z \sim q_\theta(z|g)$ for $\forall g = 1, 2, \cdots, G$.

Given the model parameter θ, the expected reward of the action can be calculated by taking the integral over all possible outcome z as $\mathbb{E}_{q_\theta}[r(z|g)]$. Thompson sampling randomly selects an action a_i according to its posterior probability of being the arm with the highest expected reward. The posterior probability can be expensive to compute yet easy to sample to from:

1. At each step i, we sample an estimate of the model parameter θ from its prior distribution for trial i (which is the posterior having seen rewards up to trial $i-1$), i.e. $\widetilde{\theta} \sim p_{i-1}(\theta)$.
2. Conditioned on the sampled model parameter $\widetilde{\theta}$, the optimizer then selects the action $\hat{g} = \arg\max_g \mathbb{E}_{q_{\widetilde{\theta}}}[r(z|g)]$ that maximizes the expected reward, where $r(z|g)$ is task-specific.

The first step is the key difference between Thompson sampling and alternative such as greedy methods which in most cases use a point estimate to represent the current belief. The randomness in the first steps allows the online decision algorithm to explore actions that do not have the highest intermediate expected reward according to the current belief.

After applying the action $a_i = \hat{g}$, the outcome z_i is used in return to update the posterior distribution of the model parameters with Bayes' theorem: $p_i(\theta_{\hat{g}}) \propto p_{i-1}(\theta_{\hat{g}}) q_\theta(z_i|\hat{g})$ And the

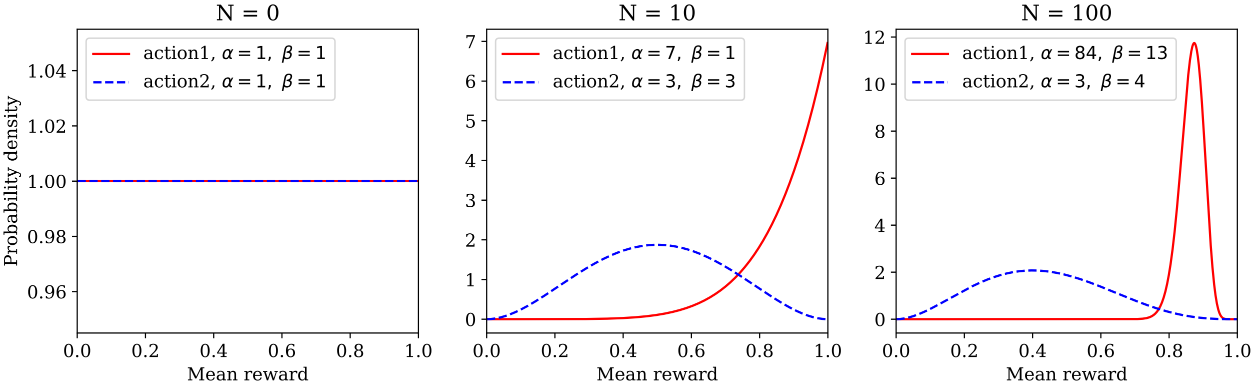


Figure 3.2: Probability density functions over mean rewards for action1 (red solid curve) and action2 (blue dashed curve) with budget $N = 0, 10, 100$, with the prior distribution of expected reward set as Beta(1, 1) for both action1 and action2.

agent gets the corresponding reward $r_i = r(z_i|\hat{g})$.

Balance between Exploration and Exploitation By estimating θ with a Bayesian model, Thompson sampling naturally balances between exploration and exploitation via quantifying and updating the uncertainty of its belief about the expected reward of each action. We demonstrate this property by simulating a simple online decision system with Thompson sampling. This online decision system needs to sequentially decide between action1 and action2 with budget N. We assume the reward of each action is binary, with the true mean reward set as 0.8 for action1 and 0.2 for action2.

In Figure 3.2, we plot the posterior distribution of the expected reward of two actions as the budget N increases from 0 to 10 and 100. At the beginning of the sampling process($N = 0$), with a weak prior $Beta(1, 1)$ for both θ_0 and θ_1, the probabilities of two actions being selected by the Thompson sampling algorithm are equal because their posterior distributions of the expected reward are identical. After applying $N = 10$ actions, action1 and action2 have been selected for 6 and 4 times respectively, among which a positive reward has been received for 6 out of 6 times for action1, and 2 out of 4 times for action2. Comparing their posterior distributions, we show that with the information gathered from 10 actions, the posterior

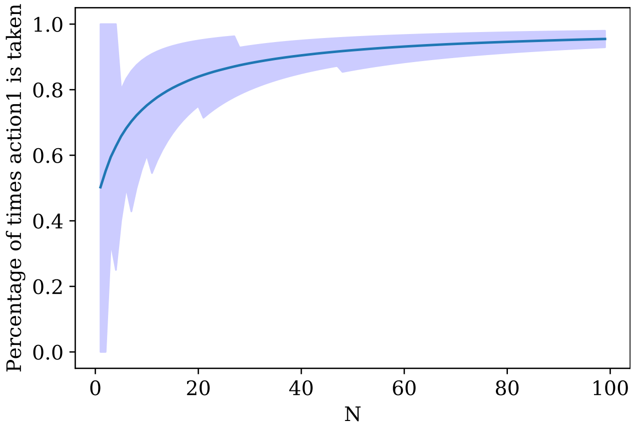


Figure 3.3: Percentage of the times that action1 is chosen over action2 by Thompson sampling averaged over 10000 independent runs, as a function of budget N. For each run, the setup is the same as Figure 3.2. Shading indicates 95% error bar across 10000 runs.

distribution of the mean reward of action1 has greater maximum posterior estimation (MPE) and lower variance. Thus in the next step, the Thompson sampling algorithm is more likely to select action1 over action2. As N increases, the MPE of the posterior distribution gradually converges to 0.8 for action1 and 0.2 for action2, and the uncertainty of the posterior distribution drops. The online decision system is increasingly more certain that action1 provides greater expected reward than action2. To optimize the cumulative reward, the probability of action1 being selected increases. The simulation shows that when $N = 100$, the budget spent on action1 and action2 are 95 and 5.

In Figure 3.3 we plot the percentage of the times that action1 is selected by Thompson sampling, averaged over 10000 independent simulations. When $N = 0$, the mechanism selects two actions with equal probability for exploration; as N increase, the mechanism selects action1 increasingly more frequently for exploitation.

3.3 Active Assessment with Thompson Sampling

The Bayesian framework described in Chapter 2 readily lends itself to be used in Bayesian active learning algorithms, by considering model assessment as a multi-armed bandit problem where each group g corresponds to an arm or a bandit. Next we use Thompson sampling to actively assess performance of black-box classifiers.

There are two key building blocks in the Bayesian assessment framework: (i) the assessment model's current belief (prior or posterior distribution) for the metric of interest $\theta_g \sim p(\theta_g)$, and (ii) a generative model (likelihood) of the labeling *outcome* $z \sim q_\theta(z|g), \forall g$. Instead of labeling randomly sampled data points from a pool of unlabeled data, we propose instead to actively select data points to be labeled by iterating between:

1. *labeling*: actively select a group $\hat{g}$ based on the assessment algorithms current belief about θ_g, randomly select a data point $\mathbf{x}_i \sim \mathcal{R}_{\hat{g}}$ and then query its label;
2. *assessment*: update the assessment model given the outcome z_i. This active selection approach requires defining a **reward function** $r(z|g)$ for the revealed outcome z for the g-th group.

For example, if the assessment task is to generate low variance estimates of groupwise accuracy, $r(z|g)$ can be formulated as the reduction in uncertainty about θ_g, given an outcome z, to guide the labeling process.

Our goal in this chapter is to demonstrate the utility of active assessment in general for performance assessment rather than comparing different active selection methods. With this in mind, we focus in particular on using Thompson sampling as our main active selection method since we found it to be more reliable in terms of reliability and efficiency compared to other methods such as epsilon-greedy and upper-confidence bound (UCB) approaches. We

Algorithm 1 Thompson Sampling for Active Assessment(p, q, r, M)

```
 1: Initialize the priors on metrics $\{p_0(\theta_1), \ldots, p_0(\theta_g)\}$
 2: for $i = 1, 2, \cdots$ do
 3:     # Sample parameters for the metrics $\theta$
 4:     $\widetilde{\theta}_g \sim p_{i-1}(\theta_g), g = 1, \ldots, G$
 5:     # Select a group $g$ (or arm) by maximizing expected reward
 6:     $\hat{g} \leftarrow \arg\max_g \mathbb{E}_{q_{\widetilde{\theta}}}[r(z|g)]$
 7:
 8:     # Randomly select an input data point from $\hat{g}$-th group
 9:     $\mathbf{x}_i \sim \mathcal{R}_{\hat{g}}$
10:     # Compute the predicted label of the input data point
11:     $\hat{y}_i(\mathbf{x}_i) = \arg\max_k p_M(y = k|\mathbf{x}_i)$
12:     # Query to get the true label (pull arm $\hat{g}$) and compute label output $z_i$
13:     $z_i \leftarrow f(y_i, \hat{y}_i(\mathbf{x}_i))$
14:
15:     # Update parameters of the $\hat{g}$-th group
16:     $p_i(\theta_{\hat{g}}) \propto p_{i-1}(\theta_{\hat{g}}) q_\theta(z_i|\hat{g})$
17: end for
```

Figure 3.4: An outline of the algorithm for active Bayesian assessment using multi-arm bandit Thompson sampling with arms corresponding to groups g.

include this additional discussion in Section 3.9.1.

Algorithm 1 describes a general active assessment algorithm based on Thompson sampling. At each step i, a set of metrics $\theta_g, 1 \ldots, G$ are sampled from the algorithm's current belief, i.e., $\widetilde{\theta}_g \sim p_{i-1}(\theta_g)$ (line 4). As an example, when assessing groupwise accuracy, $p_{i-1}(\theta_g)$ represents the algorithm's belief (e.g., in the form of a posterior Beta distribution) about the accuracy for group g given $i - 1$ labeled examples observed so far. Conditioned on the sampled θ values, the algorithm then selects the group $\hat{g}$ that maximizes the expected reward $\hat{g} = \arg\max_g \mathbb{E}_{q_{\widetilde{\theta}}}[r(z|g)]$ (line 6) where $r(z|g)$ is task-specific. The algorithm then draws an input datapoint $\mathbf{x}_i$ randomly from $\mathcal{R}_{\hat{g}}$, and uses the model M to generate a predicted label $\hat{y}_i$. The Oracle is then queried (equivalent to “pulling arm $\hat{g}$" in a bandit setting) to obtain a label outcome z_i and the algorithm's belief is updated (line 13) to update the posterior for $\theta_{\hat{g}}$, where $z \sim q_{\widetilde{\theta}}(z|\hat{g})$ is the likelihood for outcome z.

Note that this algorithm implicitly assumes that the θ_g's are independent (by modeling belief about θ_g's independently rather than jointly). In some situations there may be additional information across groups g (e.g., hierarchical structure) that could be leveraged (e.g., via contextual bandits) to improve inference but we leave this for future work.

3.4 Experimental Settings

We conduct a series of experiments across datasets, models, metrics, and assessment tasks, to systematically compare three different assessment methods:

- non-active sampling with uninformative priors (UPrior);
- non-active sampling with informative priors (IPrior);
- active Thompson sampling with informative priors (IPrior+TS).

Note that the non-active UPrior method is equivalent to standard frequentist estimation with random sampling with weak additive smoothing. We use UPrior instead of a pure frequentist method to avoid numerical issues in very low data regimes. We leave out the results of active Thompson sampling with uninformative priors (UPrior+TS) to Section 3.9.

Before running each experiment (i.e., obtaining any labeled data), unlabeled data points $\mathbf{x}_i$ from the test set were assigned to groups (such as predicted classes or score-bins) by each prediction model. Values for p_g (for use in active learning in reward functions and in evaluation of assessment methods) were estimated using the model-based assignments of test datapoints to groups. Ground truth values for θ_g were defined using the full labeled test set for each dataset. Estimates of metrics (as used for example in computing RMSE or ECE) correspond to mean posterior estimates $\hat{\theta}$ for each method.

We set the strengths of both IPrior and UPrior as $\alpha_g + \beta_g = N_0 = 2$ for Beta priors and $\sum \alpha_g = N_0 = 1$ for Dirichlet priors in all experiments, demonstrating the robustness of the settings across a wide variety of contexts. We conduct a Wilcoxon signed-rank test with p=0.05 to determine the statistical significance between the best value and next best. Best-performing values that are statistically significant, across the 3 methods, are indicated in bold in our tables.

Table 3.1: Different (p, q, r) combinations for different assessment tasks. $p(\theta)$ is the distribution of parameters of the assessment model, $q_\theta(z|g)$ is the likelihood function of outcome z for the g-th action, and $r(z|g)$ is the corresponding reward. $\mathcal{L}$ is the set of historical labeled data prior to y. $\widetilde{\theta}$ is the sampled model parameter from its distribution $p(\theta)$, $\hat{\theta}|\{\mathcal{L}, z\}$ is the MPE of θ if the outcome is z.

	Assessment Task	$p(\theta)$	$q_\theta(z\|g)$	$r(z\|g)$
Estimation	Groupwise Accuracy	$\theta_g \sim \text{Beta}(\alpha_g, \beta_g)$	$z \sim \text{Bern}(\theta_g)$	$p_g \cdot (\text{Var}(\hat{\theta}_g\|\mathcal{L}) - \text{Var}(\hat{\theta}_g\|\{\mathcal{L}, z\}))$
	Confusion Matrix$(g = k)$	$\theta_{\cdot k} \sim \text{Dirichlet}(\alpha_{\cdot k})$	$z \sim \text{Multi}(\theta_k)$	$p_k \cdot (\text{Var}(\hat{\theta}_k\|\mathcal{L}) - \text{Var}(\hat{\theta}_k\|\{\mathcal{L}, z\}))$
Identification	Least Accurate Group	$\theta_g \sim \text{Beta}(\alpha_g, \beta_g)$	$z \sim \text{Bern}(\theta_g)$	$-\widetilde{\theta}_g$
	Least Calibrated Group	$\theta_{gb} \sim \text{Beta}(\alpha_{gb}, \beta_{gb})$	$z \sim \text{Bern}(\theta_{gb})$	$\sum_{b=1}^{B} p_{gb} \left\|\widetilde{\theta}_{gb} - s_{gb}\right\|$
	Most Costly Class$(g = k)$	$\theta_{\cdot k} \sim \text{Dirichlet}(\alpha_{\cdot k})$	$z \sim \text{Multi}(\theta_k)$	$\sum_{j=1}^{K} c_{jk}\widetilde{\theta}_{jk}$
Comparison	Accuracy Comparison	$\theta_g \sim \text{Beta}(\alpha_g, \beta_g)$	$z \sim \text{Bern}(\theta_g)$	$\lambda\|\{\mathcal{L}, (g, z)\}$

3.5 Estimation

3.5.1 Active Risk Estimation

The MSE for estimation accuracy for G groups can be written in bias-variance form as $\sum_{g=1}^{G} p_g \big(\mathrm{Bias}^2(\hat{\theta}_g) + \mathrm{Var}(\hat{\theta}_g)\big)$. Given a fixed labeling budget the bias term can be assumed to be small relative to the variance (e.g., see Sawade et al. [2010]), by using relatively weak priors for example. It is straightforward to show that to minimize $\sum_{g=1}^{G} p_g \mathrm{Var}(\hat{\theta}_g)$ the optimal number of labels per group g is proportional to $\sqrt{p_g \theta_g (1-\theta_g)}$, i.e., sample more points from larger groups and from groups where θ_g is furthest from 0 or 1. While the group sizes p_g can be easily estimated from unlabeled data, the θ_g's are unknown, so we can't compute the optimal weights a priori.

Active assessment in this context allows one to minimize MSE (or RMSE) in an adaptive sequential manner. In particular we can do this by defining a reward function $r(z|g) = p_g \cdot (\mathrm{Var}(\hat{\theta}_g|\mathcal{L}) - \mathrm{Var}(\hat{\theta}_g|\{\mathcal{L},z\}))$, where $\mathcal{L}$ is the set of labeled data seen to date, with the goal of selecting examples for labeling to minimize the overall posterior variance at each step. For confusion matrices, a similar argument applies but with multinomial likelihoods and Dirichlet posteriors on vector-valued θ_g's per group (see Table 3.1).

3.5.2 Experiments: Active Risk Estimation

We compared the estimation efficacy of each method as the labeling budget N increases, for classwise accuracy (Table 3.2), confusion matrices (Table 3.3), and ECE (Table 3.4). All reported numbers were obtained by averaging across 1000 independent runs, where a run corresponds to a sequence of sampled $\mathbf{x}_i$ values (and sampled θ_g values for the TS method).

Evaluation We use RMSE of the estimated $\hat{\theta}$ relative to the true θ^* (as computed from the full test set) to measure the estimation error. For Bayesian methods, $\hat{\theta}$ is the MPE of θ's posterior distribution. For frequentist methods, $\hat{\theta}$ is the corresponding point estimation. The estimation error of groupwise accuracy is defined as RMSE $= (\sum_g p_g(\hat{\theta}_g - \theta^*_g)^2)^{\frac{1}{2}}$. For confision matrices, RMSE is defined as RMSE $= (\sum_k p_k(\sum_j(\hat{\theta}_{jk} - \theta^*_{jk})^2)^{\frac{1}{2}}$ where θ_{jk} is the probability that class j is the true class when class k is predicted.

Table 3.2: RMSE of classwise accuracy across 5 datasets. Each RMSE number is the mean across 1000 independent runs.

	N/K	N	UPrior	IPrior	IPrior+TS
			(baseline)	(our work)	(our work)
CIFAR-100	2	200	30.7	**15.0**	15.3
	5	500	20.5	**13.6**	13.8
	10	1000	13.3	**10.9**	11.4
ImageNet	2	2000	29.4	13.2	**13.2**
	5	5000	18.8	12.1	**11.6**
	10	10000	11.8	9.5	**9.4**
SVHN	2	20	13.7	5.1	**3.4**
	5	50	7.7	5.1	**3.4**
	10	100	5.4	4.7	**3.1**
20 Newsgroups	2	40	23.9	12.3	**11.7**
	5	100	15.3	10.8	**10.3**
	10	200	10.4	**8.7**	8.8
DBpedia	2	28	14.9	2.0	**1.5**
	5	70	3.5	2.3	**1.2**
	10	140	2.6	2.1	**1.1**

Table 3.2 shows the mean RMSE of the estimates of classwise accuracy for the 3 methods on the 5 datasets. The results demonstrate that informative priors and active sampling have significantly lower RMSE than the baseline, e.g., reducing RMSE by a factor of 2 or more in the low-data regime of $N/K = 2$. Active sampling (IPrior+TS) improves on the IPrior method in 11 of the 15 results, but the gains are typically small. For other metrics and tasks below we will see much greater gains from using active sampling.

Table 3.3 reports the mean RMSE across runs of estimates of confusion matrix entries for 4

Table 3.3: Mean relative RMSE for confusion matrix estimation. Same setup as Table 3.2.

	N/K	N	UPrior	IPrior	IPrior+TS
			(baseline)	(our work)	(our work)
CIFAR-100	2	200	1.463	0.077	**0.025**
	5	500	0.071	0.012	**0.004**
	10	1000	0.001	0.002	**0.001**
SVHN	2	20	92.823	0.100	**0.045**
	5	50	11.752	0.022	**0.010**
	10	100	0.946	0.005	**0.002**
20 Newsgroups	2	40	3.405	0.018	**0.005**
	5	100	0.188	0.004	**0.001**
	10	200	0.011	0.001	**0.000**
DBpedia	2	28	1307.572	0.144	**0.025**
	5	70	33.617	0.019	**0.003**
	10	140	**0.000**	0.004	0.001

datasets[2]. To help with interpretation, we scaled the errors in the table by a constant θ_0, defined as the RMSE of the confusion matrix estimated with scores from only unlabeled data , i.e. the estimate with IPrior when $N = 0$. Numbers greater than 1 mean that the estimate is worse than using θ_0 (with no labels). The results show that using informed priors (IPrior and IPrior+TS) often produces RMSE values that are orders of magnitude lower than using simple uniform prior (UPrior). Thus, the model scores on the unlabeled test set (used to construct the informative priors) are highly informative for confusion matrix entries, even though the models themselves are (for the most part) miscalibrated. We see in addition that active sampling (IPrior+TS) provides additional significant reductions in RMSE over the IPrior method with no active sampling.

In our ECE experiments samples are grouped into 10 equal-sized bins according to their model scores. Table 3.4 reports the average relative ECE estimation error[3], defined as $(100/R)\sum_{r=1}^{R} |\text{ECE}_N - \hat{\text{ECE}}_r|/\text{ECE}_N$ where ECE_N is the ECE measured on the full test set,

[2] ImageNet is omitted because 50K labeled samples is not sufficient to reliably estimate ground truth for a confusion matrix that contains 1M parameters.

[3] We report error for overall ECE rather than error per score-bin since ECE is of more direct interest and more interpretable.

Table 3.4: Mean percentage estimation error of ECE with bins as groups. Same setup as Table 3.2.

	N/K	N	UPrior	IPrior	IPrior+TS
			(baseline)	(our work)	(our work)
CIFAR-100	2	20	76.7	**26.4**	28.7
	5	50	40.5	**23.4**	26.7
	10	100	25.7	**21.5**	23.2
ImageNet	2	20	198.7	51.8	**36.4**
	5	50	122.0	55.3	**29.6**
	10	100	66.0	40.8	**22.1**
SVHN	2	20	383.6	86.2	**49.7**
	5	50	155.8	93.1	**44.2**
	10	100	108.2	80.6	**36.6**
20 Newsgroups	2	20	54.0	**39.7**	46.1
	5	50	32.8	**28.9**	36.6
	10	100	24.7	**22.3**	28.7
DBpedia	2	20	900.3	118.0	**93.1**
	5	50	249.6	130.5	**74.5**
	10	100	169.1	125.9	**60.9**

and $\mathrm{E\hat{C}E}_r$ is the esimated ECE (using MPE estimates of θ_b's), for a particular method on the r-th run, $r = 1, \ldots, R = 1000$. Both the IPrior and IPrior+TS methods have significantly lower percentage error in general in their ECE estimates compared to the naive UPrior baseline, particularly on the 3 image datasets (CIFAR-100, ImageNet, and SVHN).

3.6 Identification

3.6.1 Best Arm(s) Identification

To identify the best (or worst performing) group, $\hat{g} = \arg\max_g \theta_g$, we can define a reward function using the sampled metrics $\widetilde{\theta}_g$ for each group. For example, to identify the least accurate class, the expected reward of the g-th group is $\mathbb{E}_{q_{\widetilde{\theta}}}[r(z_i)|g] = q_{\widetilde{\theta}}(y = 1)(-\widetilde{\theta}_g) + q_{\widetilde{\theta}}(y = 0)(-\widetilde{\theta}_g) = -\widetilde{\theta}_g$. Similarly, because the reward functions of other identification tasks (Table 3.1) are independent of the value of y, when the assessment tasks are to identify the group with the highest ECE or misclassification cost, maximization of the reward function corresponds to selecting the group with the greatest sampled ECE or misclassification cost.

To extend this approach to identification of the best-m arms[4], instead of selecting the arm with the greatest expected reward, we pull the top-m-ranked arms at each step, i.e. we query the true labels of m samples, with each sample $\mathbf{x}$ randomly drawn from each of the top m ranked groups. This method can be seen as an application of the general best-m arms identification method proposed by Komiyama et al. [2015] for the problem of extreme arms identification. They proposed the multiple-play Thompson sampling (MP-TS) algorithm, and proved that MP-TS has the optimal regret upper bound when the reward is binary. When $m = 1$, MP-TS is equivalent to TS. In our experiments, we use TS for best arm identification and MP-TS for top-m arms identification, and refer to both of the methods as TS in this chapter for simplicity.

We also experimented with a modified version of Thompson sampling (TS) called top-two Thompson sampling (TTTS) [Russo, 2016] but found that that TTTS and TS gave very similar results—so we just focus on TS in the results presented in this section. We include the additional discussion in Section 3.9.2. In the Appendices, we describe the sampling process to identify the least accurate arm(s) with TS (Algorithm 2), TTTS (Algorithm 3) and MP-TS (Algorithm 4).

3.6.2 Experiments: Identification of Extreme Classes

We compare the methods for identification of the top-m classes with the lowest accuracy (Table 3.5), the highest ECE (Table 3.6) and the highest misclassification cost (Figure 3.6).

[4] This is typically referred to as best-k arms identification in the literature. We use the symbol m to avoid overloading k. Best arm identification is a special case of Best-m identification when $m = 1$.

Evaluation For our identification experiments, for a particular metric and choice of groups, we conducted 1000 different sequential runs. For each run, after each labeled sample, we rank the estimates $\hat{\theta}_g$ obtained from each of the 3 methods, and compute the mean-reciprocal-rank (MRR) relative to the true top-m ranked groups (as computed from the full test set). The MRR of the predicted top-m classes is defined as $MRR = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{\text{rank}_i}$ where rank_i is the predicted rank of the ith best class. Following standard practice, other classes in the best-m are ignored when computing rank so that $MRR = 1$ if the predicted top-m classes match ground truth. We set $m = 10$ for CIFAR-100 and ImageNet, and $m = 3$ for the other datasets.

Table 3.5: Percentage of labeled samples needed to identify the least accurate top-1 and top-m predicted classes across 5 datasets.

Dataset	Top m	UPrior (baseline)	IPrior (our work)	IPrior+TS (our work)
CIFAR-100	1	81.1	83.4	**24.9**
	10	99.8	99.8	**55.1**
ImageNet	1	96.9	94.7	**9.3**
	10	99.6	98.5	**17.1**
SVHN	1	90.5	89.8	**82.8**
	3	100.0	100.0	**96.0**
20 Newsgroups	1	53.9	55.4	**16.9**
	3	92.0	92.5	**42.5**
DBpedia	1	8.0	**7.6**	11.6
	3	91.9	90.2	**57.1**

Table 3.6: Percentage of labeled samples needed to identify the least calibrated top-1 and top-m predicted classes. Same setup as Table 3.5.

	ECE, Top 1		ECE, Top m	
Dataset	IPrior	IPrior+TS	IPrior	IPrior+TS
CIFAR-100	88.0	**43.0**	90.0	**59.0**
ImageNet	89.6	**31.0**	90.0	**41.2**
SVHN	58.8	**40.7**	88.4	**77.6**
20 Newsgroups	69.0	**27.9**	90.3	**50.5**
DBpedia	27.9	**8.1**	89.1	**55.6**

Table 3.5 shows the mean percentage of labeled test set examples needed to correctly identify the target classes where “identify" means the minimum number of labeled examples required

so that the MRR is greater than 0.99. The percentage is computed relative to the full test set. For all 5 datasets the active method (IPrior+TS) clearly outperforms the non-active methods, with large gains in particular for cases where the number of classes K is large (CIFAR-100 and Imagenet).

Similar gains are obtained in identifying the least calibrated classes. Table 3.6 shows that the improvement in efficiency is particularly significant when the classwise calibration performance has large variance across the classes (as shown in Figure 2.7), e.g., CIFAR-100, ImageNet and 20 Newsgroups.

Misclassification Cost Matrices To assess misclassification cost of the models, we experimented with 2 different cost matrices on the CIFAR-100 dataset:

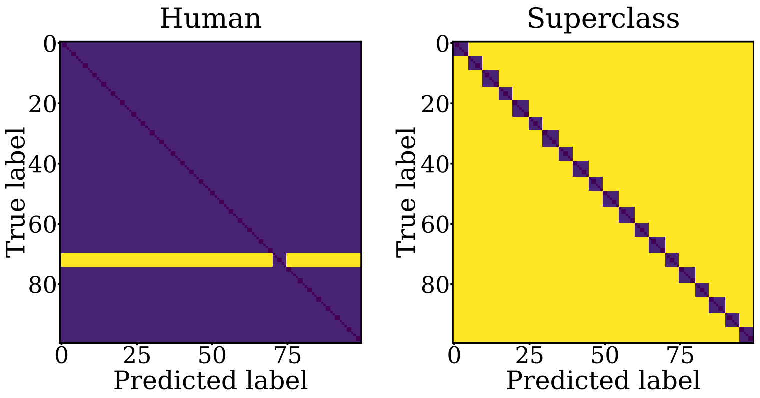


Figure 3.5: Cost matrices used in our experiments. (left): human, (right): superclass.

- **Human**: the cost of misclassifying a person (e.g., predicting *tree* when the true class is a *woman*, *boy* etc.) is more expensive than other mistakes.
- **Superclass**: the cost of confusing a class with another superclass (e.g., a *vehicle* with a *fish*) is more expensive than the cost of mistaking labels within the same superclass (e.g., confusing *shark* with *trout*).

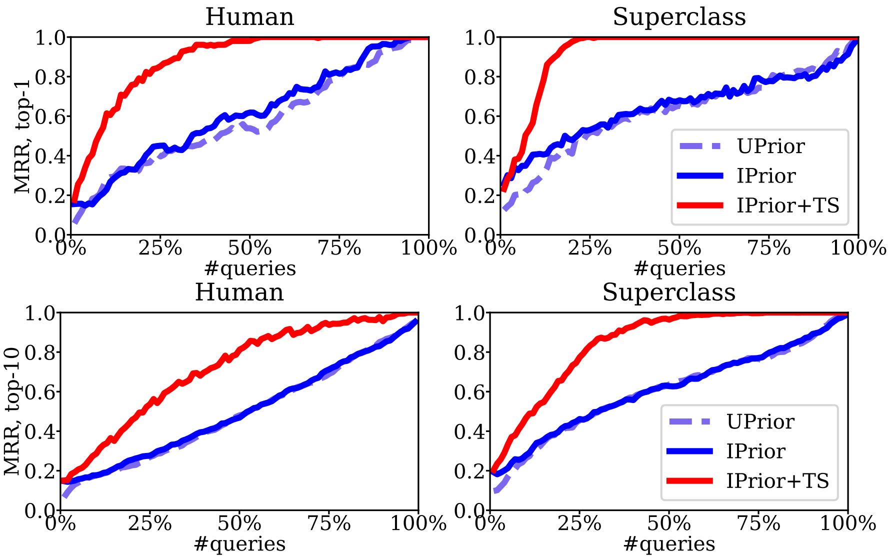


Figure 3.6: MRR of 3 assessment methods for identifying the top 1 (top) and top 10 (bottom) highest-cost predicted classes, with 2 different cost matrices (right and left), averaged over 100 trials. See text for details.

We set the cost of expensive mistakes to be 10x the cost of other mistakes. In Figure 3.5, we plot the two cost matrices.

Figure 3.6 compares our 3 assessment methods for identifying the predicted classes with highest expected cost, using data from CIFAR-100, with two different (synthetic) cost matrices. In this plot the x-axis is the number of labels (queries) and the y-value is the average (over all runs) of the MRR conditioned on the number of labels. The curves show the MRR as a function of the number of labels (on average, over 100 runs) for each of the 3 assessment methods. The active assessment (IPrior+TS) is clearly much more efficient at identifying the highest cost classes than the two non-active methods. The gains from active assessment were also robust to different settings of the relative costs of mistakes (details in Section 3.9.5).

3.7 Comparison

3.7.1 Active Comparison with ROPE

As discussed in Section 2.3.6, we can compare the performance of two groups θ_{g_1} and θ_{g_2} using the "rope" (region of practical equivalence) method. The goal of Bayesian performance comparison is to estimate (η, λ), where η is the region in which Δ has the highest cumulative density, and $\lambda = \max(\mu) \in [0, 1]$ represents the confidence of the assessment.

Using Thompson sampling to actively select labels from g_1 and g_2, at the i-th step, when we get a z_i for a data point from the g-th group, we update the Beta posterior of θ_g. The resulting decrease in uncertainty about θ_g depends on the realization of the binary variable z_i and the current distribution of θ_g. We use λ to measure the amount of evidence we gathered from the labeled data from both of the groups. Then we can select the group in a greedy manner that has the greater expected increase $\mathbb{E}_{q_{\tilde{\theta}}}[\lambda|\{\mathcal{L}, (g, z)] - \mathbb{E}_{q_{\tilde{\theta}}}[\lambda|\mathcal{L}]$, which is equivalent to selecting the arm with the largest $\mathbb{E}_{q_{\tilde{\theta}}}[\lambda|\{\mathcal{L}, (g, z)]$. This approach of *maximal expected model change strategy* has also been used in prior work in active learning for other applications [Freytag et al., 2014, Vezhnevets et al., 2012].

3.7.2 Experiments: Comparison of Groupwise Accuracy

Evaluation We compare the results of rope assessment (η, λ) with the ground truth values (η^*, λ^*). The assessment is considered as a success if (1) the direction of difference is correctly identified $\eta = \eta^*$ and (2) the estimation error of cumulative density is sufficiently small $|\lambda - \lambda^*|/\lambda^* < 0.05$. In the experiments, we evaluate η and λ with the 10,000 Monte Carlo samples from current estimation of Δ after every 10 labeled samples. We set $\epsilon = 0.05$ in "rope", i.e. the performance for two groups are considered to be "practically equivelent" when

$|\theta_{g_1} - \theta_{g_2}| < 0.05$.

Table 3.7: Average number of labels across all pairs of classes required to estimate λ for randomly selected pairs of groups.

	UPrior	IPrior	IPrior+TS
CIFAR-100, Superclass	203.5	129.0	**121.9**
SVHN	391.1	205.2	172.0
20 Newsgroups	197.3	157.4	**136.1**
DBpedia	217.5	4.3	**2.8**

For comparison experiments, Table 3.7 shows the results for the number of labeled data points required by each method to reliably assess the accuracy difference of two predicted classes, averaged over independent runs for all pairwise combinations of classes.[5] The results show that actively allocating a labeling budget and informative priors always improves label efficiency over uniform priors with no active assessment. In addition, active sampling (IPrior+TS) shows a systematic reduction of 5% to 35% in the mean number of labels required across datasets, over non-active sampling (IPrior).

3.8 Discussion

As we demonstrated in the last three sections, our results clearly demonstrate that the active Bayesian assessment framework is significantly more label-efficient and accurate across a wide array of assessment tasks. Overall, we find that IPrior+TS outperforms IPrior, and that IPrior is more effective than UPrior. As we discussed in Section 2.4, even though the model is not well-calibrated there is nonetheless valuable information about confusion probabilities available from the model's estimates of class-conditional probabilities. Results are statistically significant in all rows in all tables, except for SVHN results in Table 3.7.

[5] ImageNet is left out from this set of experiments because there are only 50 samples per predicted class. For CIFAR-100, instead of comparing performance among 100 predicted classes, each of which only contains 100 samples. We make the comparison among 20 superclasses instead.

3.9 Additional Experimental Results

In this section, we include the additional experimental results, including (1) comparisons with alternative active learning algorithms, (2) comparisons between Thompson sampling and top-two Thompson sampling for best arm identification, (3) comparisons between IPrior+TS and UPrior+TS, (4) sensitivity analysis for hyperparameters, and (5) sensitivity analysis for cost matrix values.

3.9.1 Comparisons with Alternative Active Learning Algorithms

There are a variety of other active learning approaches, such as epsilon greedy and Bayesian upper-confidence bound(UCB) methods, that could also be used as alternatives to Thompson sampling.

- Epsilon-greedy: with probability $1 - \epsilon$ the arm currently with the greatest expected reward is selected; with probability ϵ the arm is randomly selected. We set ϵ as 0.1 in our experiments.
- Bayesian upper-confidence bound (UCB): the arm with the greatest upper confidence bound is selected at each step. In our experiments we use the 97.5% quantile, estimated from 10,000 Monte Carlo samples, as the upper confidence bound.

We compare epsilon greedy, Bayesian UCB and Thompson sampling (TS) on the tasks to identify the least accurate and the top-m least accurate predicted classes across five datasets.

Figure 3.7 plots the curves of MRR obtained with three methods as the number of queries increase. We use the uninformative prior with prior strength 2 for all three algorithms. The results show that the MRR curves of Thompson sampling always converge faster than the

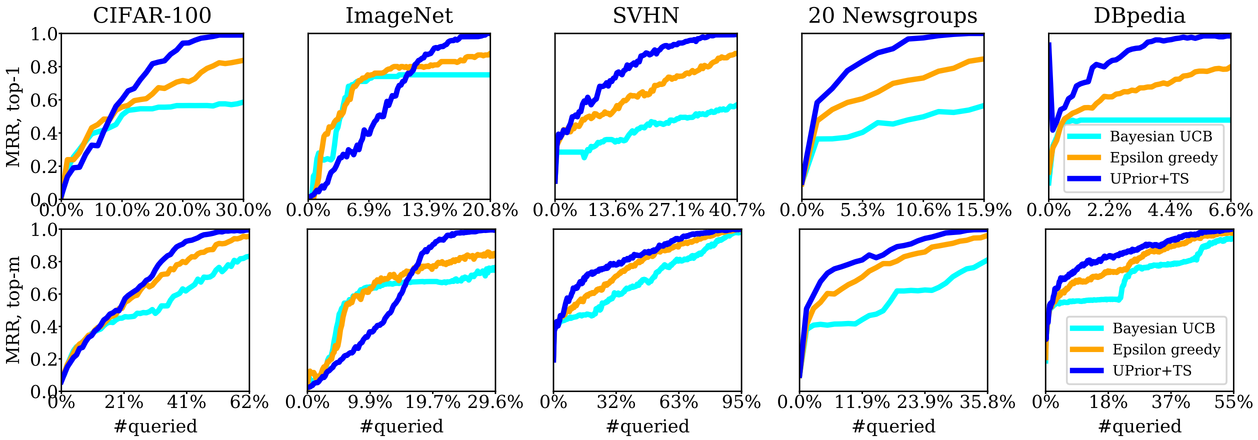


Figure 3.7: Mean reciprocal rank (MRR) of the classes with the estimated lowest classwise accuracy with the strength of the prior set as 2, comparing Thompson sampling (TS) with epsilon greedy and Bayesian UCB, across five datasets. The y-axis is the average MRR over 1000 runs for the percentage of queries, relative to the full test set, as indicated on the x-axis. In the upper row $m = 1$, and in the lower row $m = 10$ for CIFAR-100 and ImageNet, and $m = 3$ for the other datasets.

curves of epsilon greedy and Bayesian UCB, indicating that Thompson sampling is broadly more reliable and more consistent in terms of efficiency for these tasks.

3.9.2 Comparisons between Thompson Sampling and Top-Two Thompson Sampling for Best Arm Identification

For best identification problems, apart from Thompson sampling, we also experimented with a modified version of Thompson sampling called top-two Thompson sampling (TTTS) which has theoretical advantages for identifying the best arm in a pure exploration mode [Russo, 2016].

- Top-two Thompson sampling (TTTS) is a modified version of TS that is tailored for best arm identification, and has some theoretical advantages over TS. Compared to TS, this algorithm adds a re-sampling process to encourage more exploration. At each step, with probability $1 - \beta$ the algorithm selects the class I which has the highest expected reward; in order to encourage more exploration, with probability β the algorithm re-samples until a different class $J \neq I$ has the highest expected reward. β is a tuning parameter. When $\beta = 0$, there is no re-sampling in TTTS and it is reduced to TS.

Figure 3.8 compares TS and TTTS for identifying the least accurate class for CIFAR-100. The results show that two methods are equally efficient across 5 datasets. For TTTS, we set the probability for re-sampling to $\beta = 0.5$ as recommended in Russo [2016]. We found that for the problems and datasets we investigated in this section that TS and TTTS gave very similar performance.

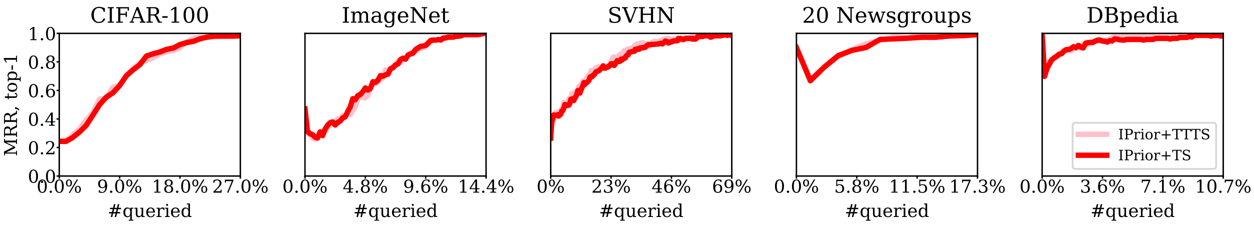


Figure 3.8: Mean reciprocal rank (MRR) of the class with the estimated lowest classwise accuracy with the strength of the prior set as $\alpha + \beta = 2$, comparing TS and TTTS, across five datasets. The y-axis is the average MRR over 1000 runs for the percentage of queries, relative to the full test set, as indicated on the x-axis.

3.9.3 Comparisons Between IPrior+TS and UPrior+TS

We use the comparison between UPrior+TS and IPrior+TS to demonstrate the influence of informative priors when samples are actively labeled for identifying the least accurate top-1 or top-m predicted classes. We set the strength of both the informative prior and the uninformative prior as 2.

The results in Figure 3.9 illustrate that the informative prior can be helpful when the prior captures the relative ordering of classwise accuracy well (e.g., ImageNet), but less helpful when the difference in classwise accuracy across classes is small and the classwise ordering

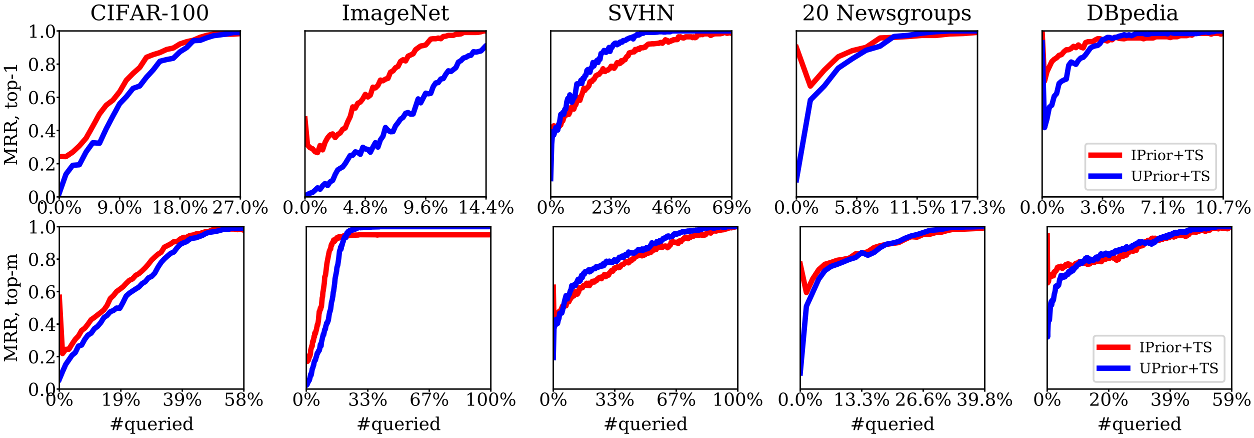
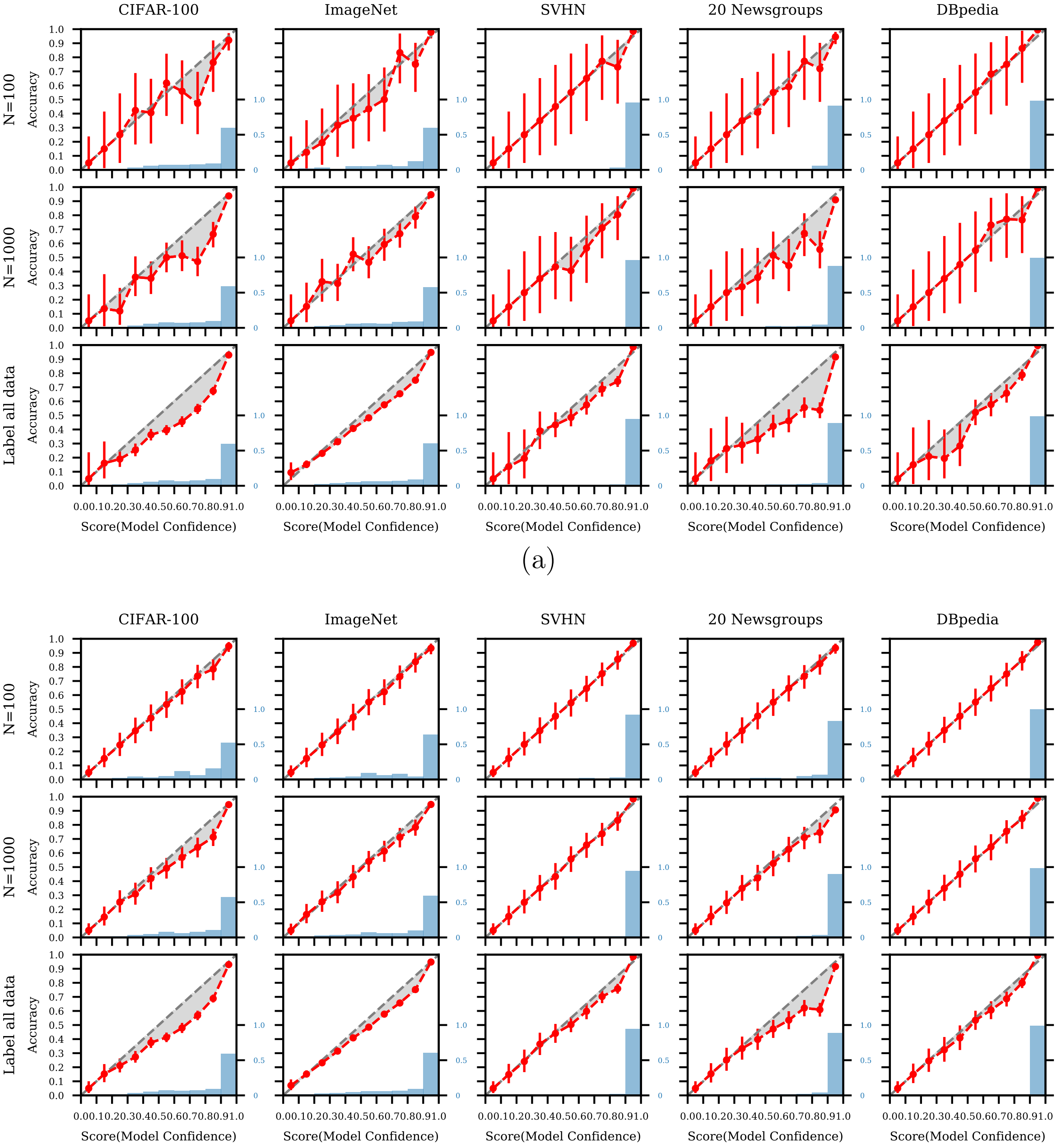


Figure 3.9: Comparison of the effect of informative (red) and uninformative (blue) priors on identifying the least accurate predicted class with Thompson sampling across 5 datasets. The y-axis is the average MRR over 1000 runs for the percentage of queries, relative to the full test set, as indicated on the x-axis. In the upper row $m = 1$, and in the lower row $m = 10$ for CIFAR-100 and ImageNet, and $m = 3$ for the other datasets.

reflected in the “self-assessment prior" is more likely to be in error (e.g., SVHN, as shown in Figure 2.7.). In general, across the different estimation tasks, we found that when using active assessment (TS) informative priors (rather than uninformative priors) generally improved performance and rarely hurt it.

3.9.4 Sensitivity Analysis for Hyperparameters

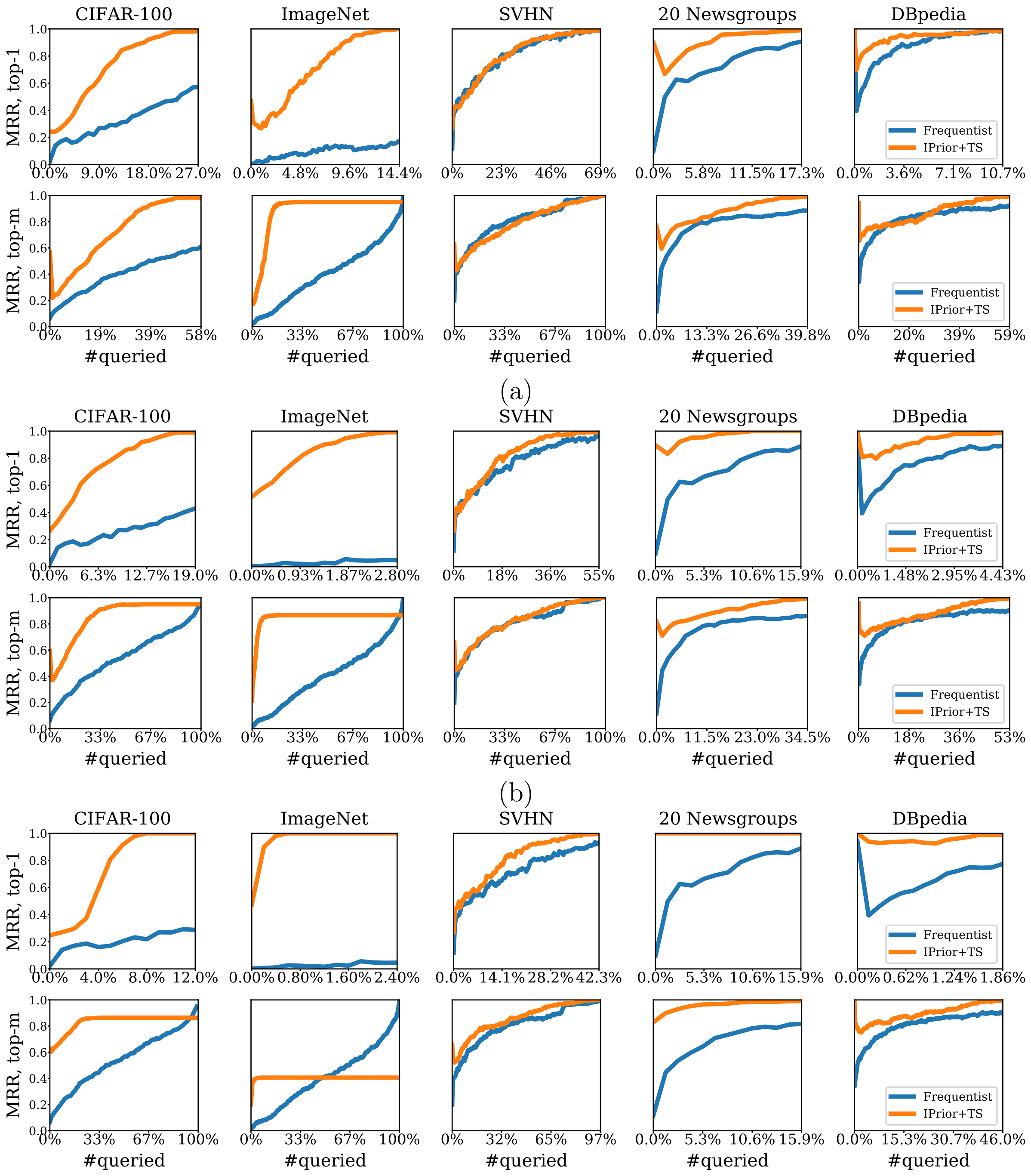
In Figure 3.10, we show Bayesian reliability diagrams for five datasets as the strength of the prior increases from 10 to 100. As the strength of the prior increases, it takes more labeled data to overcome the prior belief that the model is calibrated. In Figure 3.11, we show MRR of the m lowest accurate predicted classes as the strength of the prior increases from 2 to 10 to 100. And in Figure 3.12, we show MRR of the m least calibrated predicted classes as the strength of the prior increase from 2 to 5 and 10. From these plots, the proposed approach appears to be relatively robust to the prior strength.



(a)

(b)

Figure 3.10: Bayesian reliability diagrams for five datasets (columns) estimated using varying amounts of test data (rows) with prior strength ($\alpha_b + \beta_b$ for each bin) set to be (a) 10 and (b) 100 respectively. The red circles plot the posterior mean for θ_b under our Bayesian approach. Red bars display 95% credible intervals. Shaded gray areas indicate the estimated magnitudes of the calibration errors, relative to the Bayesian estimates. The blue histogram shows the distribution of the scores for N randomly drawn samples.

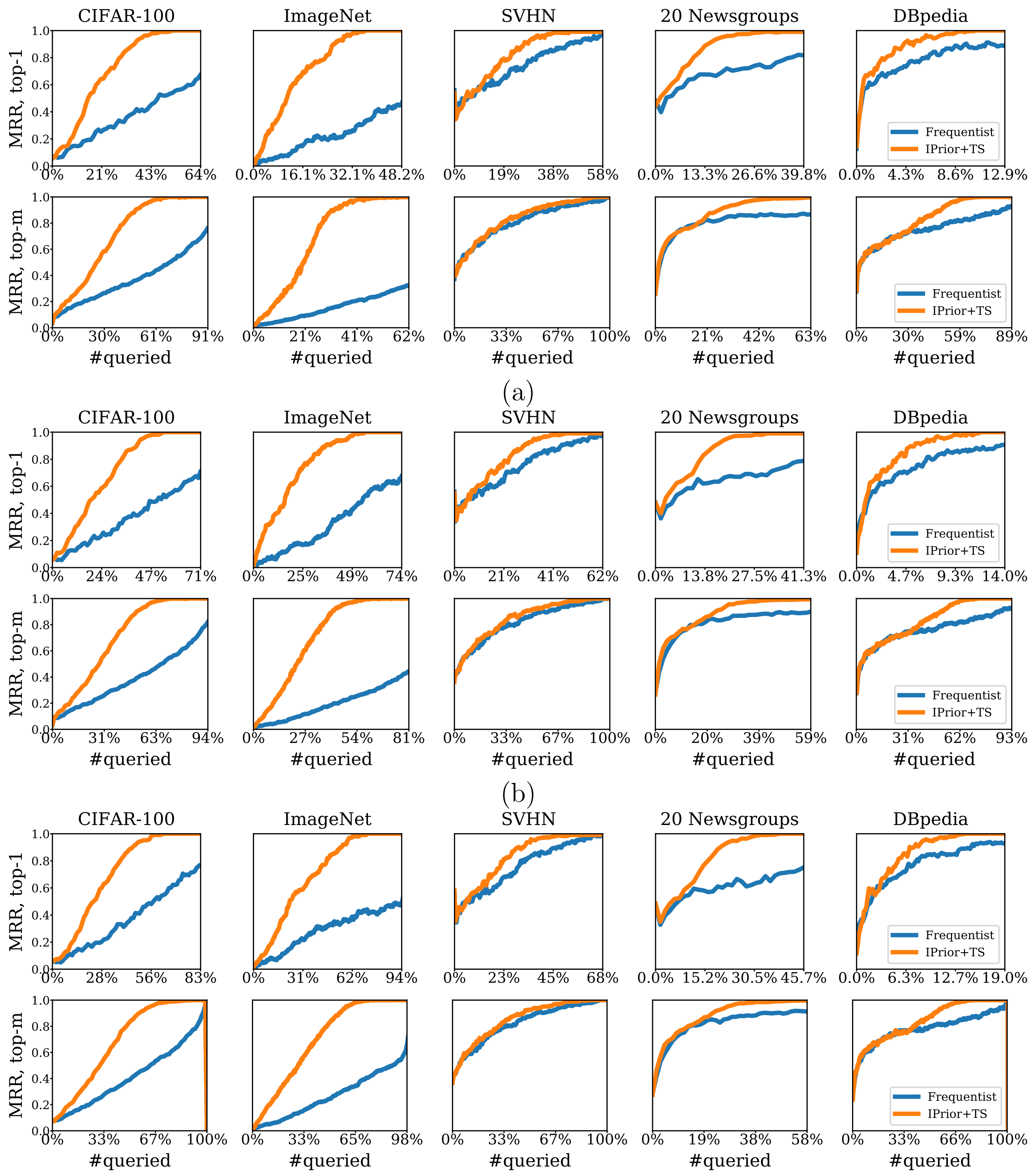


(a)

(b)

(c)

Figure 3.11: Mean reciprocal rank (MRR) of the m classes with the estimated lowest classwise accuracy as the strength of the prior varies from (a) 2 to (b) 10 and (c) 100, comparing active learning (with Thompson sampling (IPrior+TS)) with no active learning(Frequentist), across five datasets. The y-axis is the average MRR over 1000 runs for the percentage of queries, relative to the full test set, as indicated on the x-axis. For each of (a), (b) and (c), in the upper row $m = 1$, and in the lower row $m = 10$ for CIFAR-100 and ImageNet, and $m = 3$ for the other datasets.



(a)

(b)

(c)

Figure 3.12: Mean reciprocal rank (MRR) of the m classes with the estimated highest classwise ECE as the strength of the prior varies from (a) 2 to (b) 5 and (c) 10, comparing active learning (with Thompson sampling (IPrior+TS)) with no active learning(Frequentist), across five datasets. The y-axis is the average MRR over 1000 runs for the percentage of queries, relative to the full test set, as indicated on the x-axis. For each of (a), (b) and (c), in the upper row $m = 1$, and in the lower row $m = 10$ for CIFAR-100 and ImageNet, and $m = 3$ for the other datasets.

3.9.5 Sensitivity Analysis for Cost Matrix Values

We also investigated the sensitivity of varying the relative cost of mistakes in our cost experiments. Results are provided in Table 3.8. A pseudocount of 1 is used in the Dirichlet priors for Bayesian models. We consistently observe that active assessment with an informative prior (IPrior+TS) performs the best, followed by non-active assessment with an informative prior (IPrior) and finally non-active assessment with an uninformative prior (UPrior).

Table 3.8: Number of queries required by different methods to achieve a 0.99 mean reciprocal rank(MRR) identifying the class with highest classwise expected cost. The cost types are "Human" (left) and "Superclass" (right).

"Human"					"Superclass"				
Cost	Top m	UPrior	IPrior	IPrior+TS	Cost	Top m	UPrior	IPrior	IPrior+TS
1	1	9.6K	9.4K	**5.0K**	1	1	9.9K	10.0K	**2.2K**
	10	10.0K	10.0K	**9.4K**		10	9.8K	9.9K	**5.9K**
2	1	9.3K	9.3K	**4.4K**	2	1	10.0K	10.0K	**2.2K**
	10	9.8K	10.0K	**8.4K**		10	9.9K	9.9K	**5.2K**
5	1	9.5K	9.7K	**4.5K**	5	1	9.9K	10.0K	**1.8K**
	10	9.6K	10.0K	**7.9K**		10	9.9K	9.9K	**5.3K**
10	1	9.3K	9.1K	**2.2K**	10	1	10.0K	9.8K	**1.4K**
	10	9.6K	9.7K	**7.4K**		10	9.9K	9.9K	**4.0K**

3.10 Related Work

There has been a limited amount of previous work on non-active methods for label-efficient performance assessment. For non-active label-efficient risk estimation, Sawade et al. [2010] and Kumar and Raj [2018] use importance sampling and stratified sampling respectively to allocate labeling resources among different groups. In the information retrieval literature, label efficiency of model assessment has mainly been studied from the perspective of leveraging incomplte judgements, instead of using actively labeled data [Aslam et al., 2006, Yilmaz and Aslam, 2006, Moffat et al., 2007]. A significant difference between our work and this prior work on label-efficient assessment is our framing of the assessment problem as an MAB problem and addressing it via Bayesian active assessment.

While there is a large literature on active learning and multi-armed bandits (MAB) in general, e.g., Settles [2012], Russo et al. [2018], Lattimore and Szepesvári [2020], our work is the first that applies ideas from Bayesian active learning to general classifier assessment, building on MAB-inspired, pool-based active learning algorithms for data selection [Thompson, 1933, Komiyama et al., 2015, Russo, 2016].

There are also non-Bayesian active learning methods for model assessment. Nguyen et al. [2018] selects samples for estimating visual recognition performance of an algorithm on a fixed test set, by individual accuracy of data points as latent variables to estimate accuracy and precision of recognition systems with large-scale noisy labels, for applications like multi-label tags and instance segmentation. Similar ideas have also been explored in the information retrieval literature. Sabharwal and Sedghi [2017] use system output ranking to select samples to estimate PR curve of the information retrieval system with error bound; Li and Kanoulas [2017] and Rahman et al. [2020] select test items to label and estimate performance on unlabeled data; Voorhees [2018] and Rahman et al. [2019] use ideas from multi-armed bandits to construct test datasets. However, this line of prior work is significantly narrower in scope in terms of performance metrics and tasks compared to the more general approach we propose here.

3.11 Conclusions

In this chapter, we developed active assessment methods to improve label-efficiency of black-box classifiers assessment using techniques from Bayesian active learning. Our primary contributions are:

- We proposed a general framework for active Bayesian assessment for an array of fundamental tasks including (1) estimation of model performance; (2) identification of model deficiencies; (3) performance comparison between groups;

- We developed a set of Thompson sampling algorithms for label-efficient active assessment;
- We demonstrated that our proposed approaches need significantly fewer labels than baselines, via a series of experiments assessing the performance of modern neural classifiers (e.g., ResNet and BERT) on several standard image and text classification datasets.

There are a number of interesting directions for future work, such as Bayesian estimation of continuous functions related to accuracy and calibration (rather than over regions). The framework can also be extended to assess a particular model operating in multiple environments using a Bayesian hierarchical approach, or to comparatively assess multiple models operating in the same environment.

A related direction is to consider environments where humans are in the loop where, given a constraint on the number of problems that can be allocated to humans, the goal is to identify for which types of prediction problems human accuracy will most likely exceed model accuracy.

The techniques we use in this chapter can in principle be replaced by any Bayesian active learning algorithms designed for MAB problems—determining optimal active learning approaches for model assessment problems is also an interesting avenue for future research.

Chapter 4

Assess Fairness with Unlabeled Data and Bayesian Inference

In this chapter, we investigate the problem of reliably assessing group fairness when labeled examples are few but unlabeled examples are plentiful. We propose a general Bayesian framework that can augment labeled data with unlabeled data to produce more accurate and lower-variance estimates compared to methods based on labeled data alone. Our approach estimates calibrated scores for unlabeled examples in each group using a hierarchical latent variable model conditioned on labeled examples. This in turn allows for inference of posterior distributions with associated notions of uncertainty for a variety of group fairness metrics. We demonstrate that our approach leads to significant and consistent reductions in estimation error across multiple well-known fairness datasets, sensitive attributes, and predictive models. The results show the benefits of using both unlabeled data and Bayesian inference in terms of assessing whether a prediction model is fair or not.

4.1 Introduction

Machine learning models are increasingly used to make important decisions about individuals. At the same time it has become apparent that these models are susceptible to producing systematically biased decisions with respect to sensitive attributes such as gender, ethnicity, and age [Angwin et al., 2017, Berk et al., 2018, Corbett-Davies and Goel, 2018, Chen et al., 2019, Beutel et al., 2019]. This has led to a significant amount of recent work in machine learning addressing these issues, including research on both (i) definitions of fairness in a machine learning context (e.g., Dwork et al. [2012], Chouldechova [2017]), and (ii) design of fairness-aware learning algorithms that can mitigate issues such as algorithmic bias (e.g., Calders and Verwer [2010], Kamishima et al. [2012], Feldman et al. [2015], Zafar et al. [2017], Chzhen et al. [2019]).

In this chapter we focus on an important yet under-studied aspect of the fairness problem: reliably assessing how fair a blackbox model is, given limited labeled data. In particular, we focus on assessment of group fairness of binary classifiers. Group fairness is measured with respect to parity in prediction performance between different demographic groups. Examples include differences in performance for metrics such as true positive rates and false positive rates (also known as equalized odds [Hardt et al., 2016]), accuracy [Chouldechova, 2017], false discovery/omission rates [Zafar et al., 2017], and calibration and balance [Kleinberg et al., 2016].

Despite the simplicity of these definitions, a significant challenge in assessment of group fairness is high variance in estimates of these metrics based on small amounts of labeled data. To illustrate this point, Figure 4.1 shows frequency-based estimates of group differences in true positive rates (TPRs) for four real-world datasets. The boxplots clearly show the high variability for the estimated TPRs relative to the true TPRs (shown in red) as a function of the number of labeled examples n_L. In many cases the estimates are two or three or more

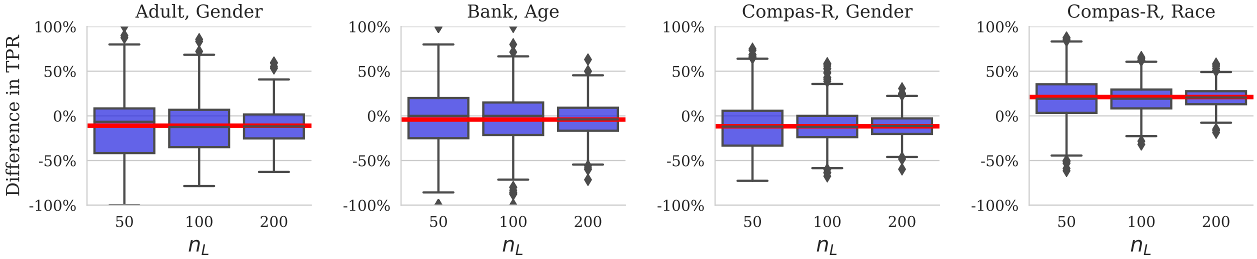


Figure 4.1: Boxplots of frequency-based estimates of the difference in true positive rate (TPR) for four fairness datasets and binary sensitive attributes, across 1000 randomly sampled sets of labeled test examples of size $n_L = 50, 100, 200$. The horizontal red line is the TPR difference computed on the full test dataset.

times larger than the true difference. In addition, a relatively large percentage of the estimates have the opposite sign of the true difference, potentially leading to mistaken conclusions.

The variance of these estimates decreases relatively slowly, e.g., at a rate of approximately $\frac{1}{n}$ for group differences in accuracy where n is the number of labels in the smaller of the two groups[1]. Imbalances in label distributions can further compound the problem, for example for estimation of group differences in TPR or FPR. For example, consider a simple simulation with two groups, where the underrepresented group makes up 20% of the whole dataset, groupwise positive rates $P(y = 1)$ are both 20%, and the true groupwise TPRs are 95% and 90%. In Figure 4.2, we show in this simulation that a large number n_L of labeled examples (at least 96,000) is needed to ensure there is a 95% chance that our estimate of the true TPR difference (which is 0.05) lies in the range [0.04, 0.06]. Yet for real-world datasets used in the fairness literature (e.g., Friedler et al. [2019]; see also Table 4.1 later in the chapter), test set sizes are often much smaller than this, and it is not uncommon for the group and label distributions to be even more imbalanced.

The real-world and synthetic examples above show that frequentist assessment of group fairness is unreliable unless the labeled dataset is unrealistically large. Acquiring large amounts of labeled data can be difficult and time-consuming, particularly for the types of

[1] Stratified sampling by group could help with this issue (e.g., see Sawade et al. [2010]), but stratification might not always be possible in practice, and the total variance will still converge slowly overall.

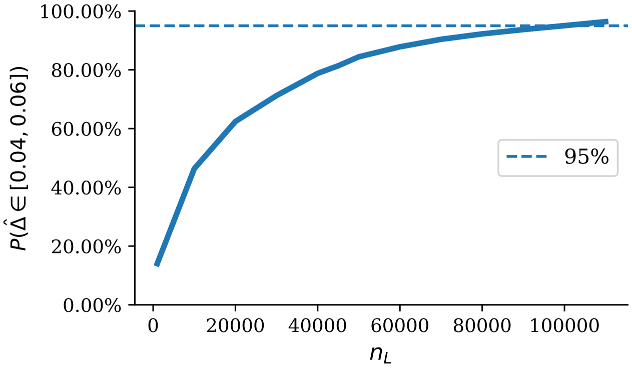


Figure 4.2: Percentage of 10000 independent simulations whose estimates of Δ TPR are in the range $[0.04, 0.06]$, as a function of the number of labeled examples n_L.

applications where fairness is important, such as decision-making in medical or criminal justice contexts [Angwin et al., 2017, Berk et al., 2018, Rajkomar et al., 2018]. This is in contrast to applications such as image classification where approaches like Mechanical Turk can be used to readily generate large amounts of labeled data.

To address this problem, we propose to augment labeled data with unlabeled data to generate more accurate and lower-variance estimates compared to methods based on labeled data alone.

4.2 Notation and Problem Statement

We use the same set of notation as the previous chapters: Consider a trained binary classification model M, with inputs $\mathbf{x}$ and class labels $y \in \{0, 1\}$. The model produces scores[2] $s(\mathbf{x}) = p_M(y = 1|\mathbf{x}) \in [0, 1]$, where p_M denotes the fact that this is the model's estimate of the probability that $y = 1$ conditioned on $\mathbf{x}$. When there is no ambiguity, we use s instead of $s(\mathbf{x})$ to implicitly represent the model score as a deterministic function of $\mathbf{x}$. Under 0-1 loss the model predicts $\hat{y} = 1$ if $s \geq 0.5$ and $\hat{y} = 0$ otherwise. The marginal

[2] Note that the term "score" is sometimes defined differently in the calibration literature as the maximum class probability for the model. Both definitions are equivalent mathematically for binary classification.

accuracy of the classifier is $p(\hat{y} = y)$ and the accuracy conditioned on a particular value of the score s is $p(\hat{y} = y|s)$. A classifier is calibrated if $p(\hat{y} = y)|s) = s$, e.g., if whenever the model produces a score of $s = 0.9$ then its prediction is correct 90% of the time.

For group fairness we are interested in potential differences in performance metrics with respect to a sensitive attribute (such as gender or race) whose values g correspond to different groups, $g \in \{0, 1, \ldots, G-1\}$. We will use θ_g to denote a particular metric of interest, such as accuracy, TPR, FPR, etc. for group g. We focus on group differences for two groups, defined as $\Delta = \theta_1 - \theta_0$, e.g., the difference in a model's predictive accuracy between females and males, $\Delta = p(\hat{y} = y|g = 1) - p(\hat{y} = y|g = 0)$.

We assume in general that the available data consists of both n_L labeled examples and n_U unlabeled examples, where $n_L \ll n_U$, which is a common situation in practice where far more unlabeled data is available than labeled data. For the unlabeled examples, we do not have access to the true labels y_j but we do have the scores $s_j = p_M(y_j = 1|\mathbf{x}_j)$, $j = 1, \ldots, n_U$. For the labeled examples, we have the true labels y_i as well as the scores s_i, $i = 1, \ldots, n_L$. The examples (inputs $\mathbf{x}$, scores s, and labels y if available) are sampled IID from an underlying joint distribution $p(\mathbf{x}, y)$ (or equivalently $p(s, y)$ given that s is a deterministic function via M of $\mathbf{x}$), where this underlying distribution represents the population we wish to evaluate fairness with respect to. Note that in practice $p(\mathbf{x}, y)$ might very well not be the same distribution the model was trained on. For unlabeled data D_u the corresponding distributions are $p(\mathbf{x})$ or $p(s)$.

4.3 Beta-Binomial Estimation with Labeled Data

Consider initially the case with only labeled data D_L (i.e., $n_U = 0$) and for simplicity let the metric of interest Δ be group difference in classification accuracy. In Section 2.3.6 we discussed Bayesian assessment of model performance between two groups. In this section, we use the same Beta-Binomial model to assess group fairness, which is defined as the difference of model performance between two groups.

Let $I_i = I_{\hat{y}_i = y_i}, 1 \leq i \leq n_L$, be a binary indicator of whether each labeled example i was classified correctly or not by the model. The binomial likelihood for group accuracy $\theta_g, g = 0, 1$, treats the I_i's as conditionally independent draws from a true unknown accuracy θ_g, $I_i \sim \text{Bernoulli}(\theta_g)$. As in earlier chapters in this thesis, we can perform Bayesian inference on the θ_g's by specifying conjugate $\text{Beta}(\alpha_g, \beta_g)$ priors for each θ_g, combining these priors with the binomial likelihoods, and obtaining posterior densities in the form of the beta densities on each θ_g.

From here we can get a posterior density on the group difference in accuracy, $p(\Delta|D_L)$ where $\Delta = \theta_1 - \theta_0$. Since the density for the difference of two beta-distributed quantities (the θ's) is not in general in closed form, we use posterior simulation (e.g., Gelman et al. [2013]) to obtain posterior samples of Δ by sampling θ's from their posterior densities and taking the difference. For metrics such as TPR we place beta priors on conditional quantities such as $\theta_g = p(\hat{y} = 1|y = 1, g)$. In all of the results in the chapter we use weak uninformative priors for θ_g with $\alpha_g = \beta_g = 1$. This general idea of using Bayesian inference on classifier-related metrics has been noted before for metrics such marginal accuracy [Benavoli et al., 2017], TPR [Johnson et al., 2019], and precision-recall [Goutte and Gaussier, 2005], but has not been developed or evaluated in the context of fairness assessment.

This beta-binomial approach above provides a useful, simple, and practical tool for understanding and visualizing uncertainty about fairness-related metrics, conditioned on a set of n_L labeled examples. However, with weak uninformative priors, the posterior density for Δ will be relatively wide unless n_L is very large, analogous to the high empirical variance for frequentist point estimates in Figure 4.1. As with the frequentist variance, the width of the posterior density on Δ will decrease relatively slowly at a rate of approximately $\frac{1}{n_L}$. This motivates the main goal of the chapter: can we combine unlabeled examples with labeled examples to make more accurate inferences about fairness metrics?

4.4 Leveraging Unlabeled Data with a Bayesian Calibration Model

Consider the situation where we have n_U unlabeled examples, in addition to the n_L labeled ones. For each unlabeled example $j = 1, \ldots, n_U$ we can use the model M to generate a score, $s_j = p_M(y_j = 1|\mathbf{x}_j)$. If the model M is perfectly calibrated then the model's score is the true probability that $y = 1$, i.e., we have $s_j = p_M(y_j = 1|s_j)$ and the accuracy equals s_j if $s_j \geq 0.5$ and $1 - s_j$ otherwise. Therefore, in the perfectly calibrated case, we could empirically estimate accuracy per group for the unlabeled data using scores via

$$\hat{\theta}_g = (1/n_{U,g}) \sum_{j \in g} s_j I(s_j \geq 0.5) + (1 - s_j) I(s_j < 0.5) \tag{4.1}$$

where $n_{U,g}$ is the number of unlabeled examples that belong to group g. Metrics other than accuracy could also be estimated per group in a similar fashion.

In practice, however, many classification models, particularly complex ones such as deep learning models, can be significantly miscalibrated (see, e.g., Guo et al. [2017], Kull et al. [2017], Kumar et al. [2019], Ovadia et al. [2019]) and using the uncalibrated scores in such situations will lead to biased estimates of the true accuracy per group. The key idea of our approach is to use the labeled data to learn how to calibrate the scores such that the calibrated scores can contribute to less biased estimates of accuracy. Let

$$z_j = E[I(\hat{y}_j = y_j)] = p(y_j = \hat{y}_j|s_j) \tag{4.2}$$

be the true (unknown) accuracy of the model given score s_j. We treat each $z_j, j = 1, \ldots, n_U$ as a latent variable per example. The high-level steps of the approach are as follows:

- We use the n_L labeled examples to estimate groupwise calibration functions with parameters ϕ_g, that transform the (potentially) uncalibrated scores s of the model to calibrated scores. More specifically, we perform Bayesian inference (see Section 4.5 below) to obtain posterior samples from $p(\phi_g|D_L)$ for the groupwise calibration parameters ϕ_g.
- We then obtain posterior samples from $p_{\phi_g}(z_j|D_L, s_j)$ for each unlabeled example $j = 1, \ldots, n_U$, conditioned on posterior samples of the ϕ_g's.
- Finally, we use posterior samples from the z_j's, combined with the labeled data, to generate estimates of the groupwise metrics θ_g and the difference in metrics Δ.

We can compute a posterior sample for θ_g^t, given each set of posterior samples for ϕ_g^t and $z_1^t, \ldots, z_{n_U}^t$, by combining estimates of accuracies for the unlabeled examples with the observed outcomes for the labeled instances:

$$\theta_g^t = \frac{1}{n_{L,g} + n_{U,g}} \left(\sum_{i:i \in g} I(\hat{y}_i = y_i) + \sum_{j:j \in g} z_j^t \right) \tag{4.3}$$

where $t = 1, ..., T$ is an index over T MCMC samples. These posterior samples in turn can be used to generate an empirical posterior distribution $\{\Delta^1, \ldots, \Delta^T\}$ for Δ, where $\Delta^t = \theta_1^t - \theta_0^t$. Mean posterior estimates can be obtained by averaging over samples, i.e. $\hat{\Delta} = (1/T) \sum_t^T \Delta^t$. Even with very small amounts of labeled data (e.g., $n_L = 10$) we will demonstrate later in the chapter that we can make much more accurate inferences about fairness metrics via this Bayesian calibration approach, compared to using only the labeled data directly.

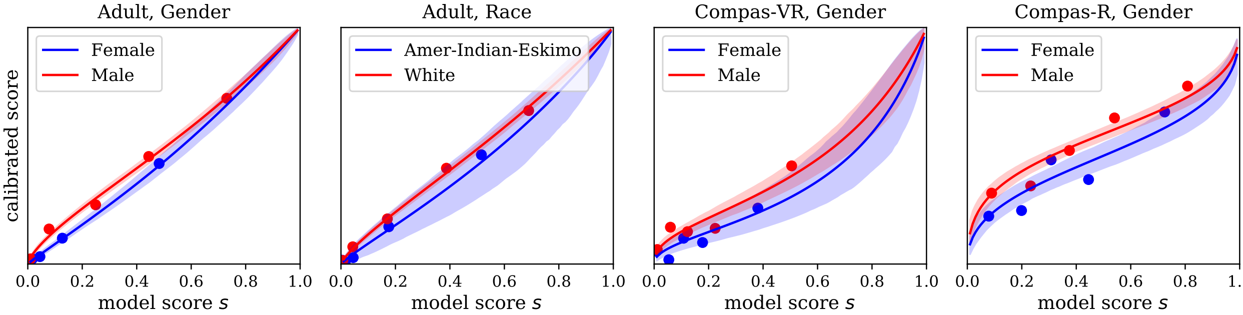


Figure 4.3: Hierarchical Bayesian calibration of two demographic groups across four dataset-group pairs, with posterior means and 95% credible intervals per group. The x-axis is the model score s for class $y = 1$, and the y-axis is the calibrated score. Instances in each group are binned into 5 equal-sized bins by model score, and blue and red points show the fraction of positive samples per group for each bin.

4.5 Hierarchical Bayesian Calibration

Bayesian calibration is a key step in our approach above. We describe Bayesian inference below for the beta calibration model specifically [Kull et al., 2017] but other calibration models could also be used. The beta calibration model maps a score from a binary classifier with scores $s = p_M(y = 1|\mathbf{x}) \in [0, 1]$ to a recalibrated score according to:

$$f(s; a, b, c) = \frac{1}{1 + e^{-c-a\log s+b\log(1-s)}} \tag{4.4}$$

where a, b, and c are calibration parameters with $a, b \geq 0$. This model can capture a wide variety of miscalibration patterns, producing the identity mapping if s is already calibrated when $a = b = 1, c = 0$. Special cases of this model are the linear-log-odds (LLO) calibration model [Turner et al., 2014] when $a = b$, and temperature scaling [Guo et al., 2017] when $a = b, c = 0$.

In our hierarchical Bayesian extension of the beta calibration model, we assume that each group (e.g., female, male) is associated with its own set of calibration parameters $\phi_g = \{a_g, b_g, c_g\}$ and therefore each group can be miscalibrated in different ways (e.g., see Figure 4.3). To apply this model to the observed data, we assume that the true labels for the observed

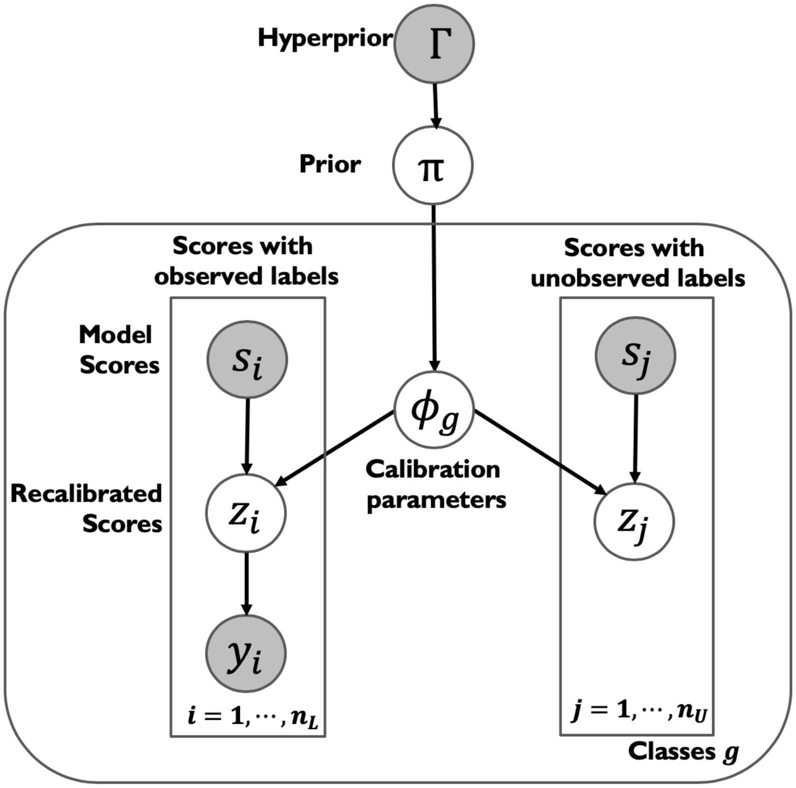


Figure 4.4: Graphical model for hierarchical Beta calibration as described in Section 2.4 of the main chapter. Γ is the hyperprior on π, representing the fixed parameters for the normal and truncated normal hyperpriors described in Section 2.4 in the main chapter.

instances are sampled according to:

$$y_i \sim \text{Bernoulli}\big(f(s_i; a_{g_i}, b_{g_i}, c_{g_i})\big) \tag{4.5}$$

where g_i is the group associated with instance i, $1 \leq i \leq n_L$. For any unlabeled example $j = 1, \ldots, n_U$, conditioned on calibration parameters ϕ_{g_j} for the group for j, we can compute the latent variable $z_j = f(s_j; \ldots)I(s_j \geq 0.5) + (1 - f(s_j; \ldots))I(s_j < 0.5)$, i.e., the calibrated probability that the model's prediction on instance j is correct.

We assume that the parameters from each individual group are sampled from a shared distribution:

$$\log a_g \sim \mathrm{N}(\mu_a, \sigma_a) \tag{4.6}$$

$$\log b_g \sim \mathrm{N}(\mu_b, \sigma_b) \tag{4.7}$$

$$c_g \sim \mathrm{N}(\mu_c, \sigma_c) \tag{4.8}$$

where $\pi = \{\mu_a, \sigma_a, \mu_b, \sigma_b, \mu_c, \sigma_c\}$ is the set of hyperparameters of the shared distributions. We complete the hierarchical model by placing the following priors on the hyperparameters (TN is the truncated normal distribution):

$$\mu_a \sim \mathrm{N}(0, .4), \sigma_a \sim \mathrm{TN}(0, .15) \tag{4.9}$$

$$\mu_b \sim \mathrm{N}(0, .4), \sigma_b \sim \mathrm{TN}(0, .15) \tag{4.10}$$

$$\mu_c \sim \mathrm{N}(0, 2), \sigma_c \sim \mathrm{TN}(0, .75) \tag{4.11}$$

These priors were chosen to place reasonable bounds on the calibration parameters and allow for diverse patterns of miscalibration (e.g., both over and under-confidence or a model) to be expressed a priori. We use exactly these same prior settings in all our experiments across all datasets, all groups, and all labeled and unlabeled dataset sizes, demonstrating the robustness of these settings across a wide variety of contexts. In Section 4.9.4 we conduct sensitivity analysis and show that the method is robust to settings of the priors.

The model was implemented as a graphical model (see Figure 4.4) in JAGS, a common tool for Bayesian inference with Markov chain Monte Carlo [Plummer, 2003]. All of the results in this paper are based on 4 chains, with 1500 burn-in iterations and 200 samples per chain, resulting in $T = 800$ sample overall. These hyperparameters were determined based on a few simulation runs across datasets, checking visually for lack of auto-correlation, with convergence assessed using the standard measure of within-to-between-chain variability. Although MCMC can sometimes be slow for high-dimensional problems, with 100 labeled data points (for example) and 10k unlabeled data points the sampling procedure takes about 30 seconds (using non-optimized Python/JAGS code on a standard desktop computer) demonstrating the practicality of this procedure.

4.5.1 Theoretical Considerations:

Lemma 4.5.1 below relates potential error in the calibration mapping (e.g., due to misspecification of the parametric form of the calibration function $f(s; \ldots)$) to error in the estimate of Δ itself.

Lemma 4.5.1. *Given a prediction model M and score distribution $P(s)$, let $f_g(s; \phi_g) : [0,1] \to [0,1]$ denote the calibration model for group g; let $f^*_g(s) : [0,1] \to [0,1]$ be the optimal calibration function which maps $s = P_M(\hat{y} = 1|g)$ to $P(y = 1|g)$; and Δ^* is the true value of the metric. Then the absolute error of the expected estimate w.r.t. ϕ can be bounded as: $|\mathbb{E}_\phi \Delta - \Delta^*| \leq \|\bar{f}_0 - f^*_0\|_1 + \|\bar{f}_1 - f^*_1\|_1$, where $\bar{f}_g(s) = \mathbb{E}_{\phi_g} f_g(s; \phi_g), \forall s \in [0,1]$, and $\| \cdot \|_1$ is the expected $L1$ distance w.r.t. $P(s|g)$.*

Proof.

$$\begin{aligned}
|\mathbb{E}_\phi \Delta - \Delta^*| &= |(\mathbb{E}_{\phi_1}\theta_1 - \mathbb{E}_{\phi_0}\theta_0) - (\theta^*_1 - \theta^*_0)| \\
&\leq |\mathbb{E}_{\phi_0}\theta_0 - \theta^*_0| + |\mathbb{E}_{\phi_1}\theta_1 - \theta^*_1| && \text{(triangle inequality)} \\
&= \|\bar{f}_0 - f^*_0\|_1 + \|\bar{f}_1 - f^*_1\|_1 && \text{(Lemma 4.5.2)}
\end{aligned}$$

□

Lemma 4.5.2. *Given a prediction model M and score distribution $P(s)$, let $f_g(s; \phi_g) : [0,1] \to [0,1]$ denote the calibration model for group g; let $f^*_g(s) : [0,1] \to [0,1]$ be the optimal calibration function which maps $s = P_M(\hat{y} = 1|g)$ to $P(y = 1|g)$; and θ^* is the true value of the accuracy. Then the absolute value of expected estimation error w.r.t. ϕ can be bounded* *as:* $|\mathbb{E}_{\phi}\theta_g - \theta^*_g| \leq \|\bar{f}_g - f^*_g\|_1$, *where* $\bar{f}_g(s) = \mathbb{E}_{\phi_g} f_g(s;\phi_g), \forall s \in [0,1]$, *and* $\|\cdot\|_1$ *is the expected* L^1 *distance w.r.t.* $P(s|g)$*.*

Proof.

$$\begin{aligned}
\theta^*_g &= P(y=0,\hat{y}=0|g) + P(y=1,\hat{y}=1|g) \\
&= \int_{s<0.5} P(y=0|s)P(s|g)ds + \int_{s>=0.5} P(y=1|s)P(s|g)ds \\
&= \int_{s<0.5} (1-f^*(s))P(s|g)ds + \int_{s>=0.5} f^*(s)P(s|g)ds
\end{aligned}$$

Similarly, our method makes prediction about groupwise accuracy with calibrated scores given P(ϕ):

$$\begin{aligned}
\mathbb{E}_{\phi_g}\theta_g &= \mathbb{E}_{\phi_g} \int_{s<0.5} (1-f_g(s;\phi))P(s|g)ds + \int_{s\geq 0.5} f_g(s;\phi)P(s|g)ds \\
&= \int_{s<0.5} (1-\mathbb{E}_{\phi} f_g(s;\phi))P(s|g)ds + \int_{s>=0.5} \mathbb{E}_{\phi} f_g(s;\phi)P(s|g)ds \\
&= \int_{s<0.5} (1-\bar{f}_g(s))P(s|g)ds + \int_{s>=0.5} \bar{f}_g(s)P(s|g)ds
\end{aligned}$$

Then the absolute estimation bias of estimator $\mathbb{E}_{\phi\in\Phi}\theta_{\phi}$ is:

$$\begin{aligned}
|\mathbb{E}_{\phi}\theta_g - \theta^*_g| &= |\int_{s<0.5} (\bar{f}(s) - f^*(s))P(s|g)ds + \int_{s>=0.5} (f^*(s) - \bar{f}(s))P(s|g)ds| \\
&\leq \int_{s<0.5} |\bar{f}(s) - f^*(s)|P(s|g)ds + \int_{s>=0.5} |f^*(s) - \bar{f}(s)|P(s|g)ds \\
&= \int_{s} |\bar{f}(s) - f^*(s)|P(s|g)ds \\
&= \|\bar{f} - f^*\|_1
\end{aligned}$$

□

Thus, reductions in the L1 calibration error directly reduce an upper bound on the L1 error in estimating Δ. The results in Figure 4.3 suggest that even with the relatively simple parametric beta calibration method, the error in calibration (difference between the fitted calibration functions) (blue and red curves) and the empirical data (blue and red dots) is quite low across all 4 datasets. The possibility of using more flexible calibration functions is an interesting direction for future work.

4.6 Datasets, Classification Models

One of the main goals of our experiments is to assess the accuracy of different estimation methods, using relatively limited amounts of labeled data, relative to the true value of the metric. By “true value" we mean the value we could measure on an infinitely large test sample. Since such a sample is not available, we use as a proxy the value of metric computed on all of the test set for each dataset in our experiments.

Datasets We performed experiments with six different real-world datasets. Specifically, we use the Adult [Dua and Graff, 2017], German Credit [Dua and Graff, 2017], Ricci [Supreme Court of the United States, 2009] and Compas datasets (for recidivism and violent recidivism) [Angwin et al., 2017] all used in Friedler et al. [2019], as well as the Bank Telemarketing dataset [Moro et al., 2014].

For all datasets, we preprocessed the data using the code from Friedler et al. [2019] [3]. We removed all instances that have missing data, and represented categorical variables with one-hot encoding. As in Friedler et al. [2019], for all datasets except Adult we randomly sampled 2/3 of the data for training and use the remaining 1/3 for test. For the Adult data we re-split the training set of the original data into train and test as in Friedler et al. [2019].

[3] `https://github.com/algofairness/fairness-comparison/blob/master/fairness/preprocess.py`

Table 4.1: Datasets used in this chapter. G is the sensitive attribute, $p(g = 0)$ is the probability of the privileged group, and $p(y = 1)$ is the probability of the positive label. The privileged groups $g = 0$ are gender: male, age: senior or adult, and race: white or Caucasian.

Dataset	Test Size	G	$p(g = 0)$	$p(y = 1)$
Adult	10054	gender, race	0.68, 0.86	0.25
Bank	13730	age	0.45	0.11
German	334	age, gender	0.79, 0.37	0.17
Compas-R	2056	gender, race	0.7, 0.85	0.69
Compas-VR	1337	gender, race	0.8, 0.34	0.47
Ricci	40	race	0.65	0.50

Summary statistics (e.g., test set size) are provided in Table 4.1. Below we provide additional details about these datasets in terms of background, and relevant attributes.

- Adult: The Adult dataset[4] from the UCI Repository of Machine Learning Databases is based on 1994 U.S. census income data. This dataset consists of 14 demographic attributes for individuals. Instances are labeled according to whether their income exceeds \$50,000 per year. In our experiments, “race" and “gender" are considered sensitive attributes. Instances are grouped into “Amer-Indian-Inuit," “Asian-Pac-Islander," “Black," “Other" and “White" by race, and “Female" and “Male" by gender. “White" and “Male" are the privileged groups.

- Bank: The Bank dataset[5] contains information about individual collected from a Portuguese banking institution. There are 20 attributes for each individuals, including marital status, education, and type of job. The sensitive attribute we use is “age," binarized by whether a individual's age is above 40 or not. The senior group is considered to be privileged. Instances are labeled by whether the individual has subscribed to a term deposit account or not.

- German: The German Credit dataset[6] from the UCI Repository of Machine Learning

[4] `https://archive.ics.uci.edu/ml/machine-learning-databases/adult`

[5] `http://archive.ics.uci.edu/ml/datasets/Bank+Marketing`

[6] `https://archive.ics.uci.edu/ml/machine-learning-databases/statlog/german`

Databases describes individuals with 20 attributes including type of housing, credit history status, and employment status. Each instance is labeled as being a good or bad credit risk. The sensitive attributes used are “gender" and “age" (age at least 25 years old) and the privileged groups are defined as “male" and “adult."

- Compas-R: The ProPublica dataset[7] contains information about the use of the COMPAS (Correctional Offender Management Profiling for Alternative Sanctions) risk assessment tool applied to 6,167 individuals in Broward County, Florida. Each individual is labeled by whether they were rearrested within two years after the first arrest. Sensitive attributees are “gender" and “race." By “gender", individuals are grouped into “Male" and “Female"; by “race", individuals are grouped into “Caucasian." “Asian," “Native-American," “African-American," “Hispanic" and “Others." The privileged groups are defined to be “Male" and “Caucasian."

- Compas-VR: This is the violent recidivism version[8] of the ProPublica data (Compas-R above), where the predicted outcome is a re-arrest for a violent crime.

- Ricci: The Ricci dataset[9] is from the case of Ricci v. DeStefano from the Supreme Court of the United States (2009). It contains 118 instances and 5 attributes, including the sensitive attribute “race." The privileged group was defined to be “White." Each instance is labeled by a promotion decision for each individual.

Classification Models We used the following classification models in our experiments: logistic regression, multi-layer perceptron (MLP) with a single hidden layer of size 10, random forests (the number of trees in the forest is set to 100), Gaussian Naive Bayes. The models were trained using standard default parameter settings and using the code provided by Friedler et al. [2019]. Predictions from the trained models were generated on the test data. Sensitive

[7] https://github.com/propublica/compas-analysis

[8] https://github.com/propublica/compas-analysis

[9] https://ww2.amstat.org/publications/jse/v18n3/RicciData.csv

attributes were not included as inputs to the models during training or test.

4.7 Illustrative Results

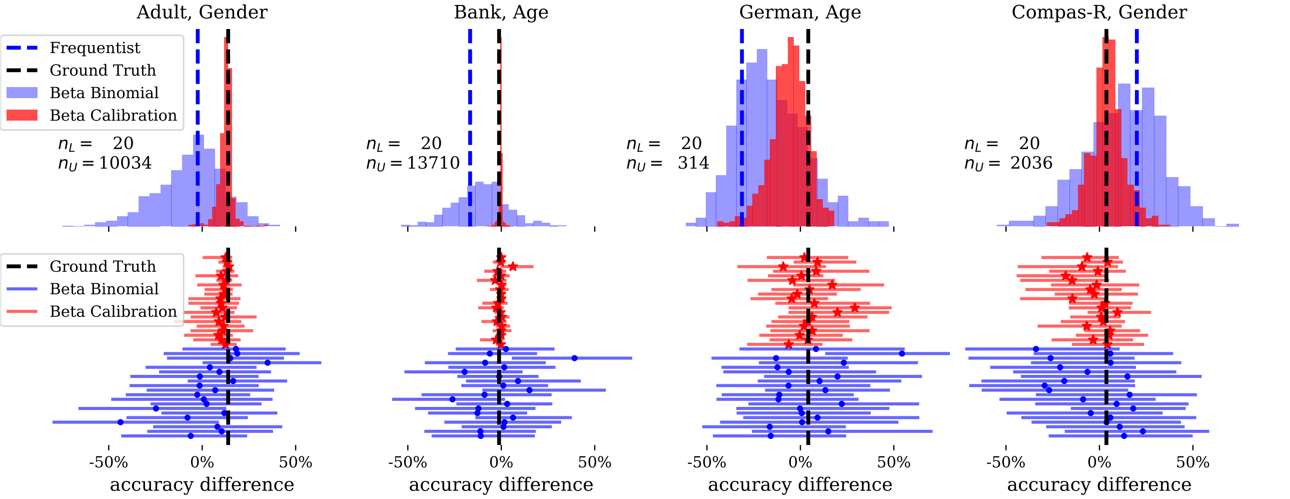


Figure 4.5: Posterior density (samples) and frequentist estimates (dotted vertical blue lines) for the difference in group accuracy Δ for 4 datasets with $n_L = 20$ random labeled examples for both the BB (Beta-binomial) and BC (Bayesian calibration) methods. Ground truth is a vertical black line. The underlying model is an MLP. The 20 examples were randomly sampled 20 different times. Upper plots show the histograms of posterior samples for the first sample, lower plots show the 95% posterior credible intervals for all 20 runs, where the x-axis is Δ.

To illustrate our approach we compare the results of the frequentist, beta-binomial (BB), and Bayesian calibration (BC) approaches for assessing group differences in accuracy across 4 datasets, for a multi-layer perceptron (MLP) binary classifier. We ran the methods on 20 runs of randomly sampled sets of $n_L = 20$ labeled examples. The BC method was given access to the remaining n_U unlabeled test examples minus the 20 labeled examples for each run, as described in Table 4.1. We define ground truth as the frequentist Δ value computed on all the labeled data in the test set.

Figure 4.5 shows the results across the 4 datasets. The top figure corresponds to the first run out of 20 runs, showing the histogram of 800 posterior samples from the BB (blue) and BC (red) methods. The lower row of plots summarizes the results for all 20 runs, showing the 95% posterior credible intervals (CIs) (red and blue horizontal lines for BC and BB respectively) along with posterior means (red and blue marks).

Because of the relatively weak prior (Beta(1,1) on group accuracy) the posterior means of the BB samples tend to be relatively close to the frequentist estimate (light and dark blue respectively) on each run and both can be relatively far away from ground truth value for Δ (in black). Although the BB method is an improvement over being frequentist, in that it provides posterior uncertainty about Δ, it nonetheless has high variance (locations of the posterior means) as well as high posterior uncertainty (relatively wide CIs). The BC method in contrast, by using the unlabeled data in addition to the labeled data, produces posterior estimates where the mean tends to be much closer to ground truth than BC.

The posterior information about Δ can be used to provide users with a summary report that includes information about the direction of potential bias (e.g., $p(\Delta > 0|D_L, D_U)$, the degree of bias (e.g., via the MPE $\hat{\Delta}$), 95% posterior CIs on Δ, and the probability that the model is “practically fair" (assessed via $p(|\Delta| < \epsilon|D_L, D_U)$, e.g., see Section 2.3.6 and Benavoli et al. [2017]). For example with BC, given the observed data, practitioners can conclude from the information in the upper row of Figure 4.5, and with $\epsilon = 0.02$, that there is a 0.99 probability for the Adult data that the classifier is more accurate for females than males; and with probability 0.87 that the classifier is practically fair with respect to accuracy for junior and senior individuals in the Bank data.

4.8 Experiments and Results

4.8.1 Assessment of Δ Accuracy, TPR and FPR

In this section we systematically evaluate the quality of different estimation approaches by repeating the same type of experiment as in Section 4.7 and Figure 4.5 across different fairness metrics and different amounts of labeled data n_L.

In particular, for each value of n_L we randomly sample sets of labeled datasets of size n_L, generate point estimates of a metric Δ of interest for each labeled dataset for each of the BB and BC estimation methods, and compute the mean absolute error (MAE) between the point estimates and the true value (computed on the full labeled test set). The frequency-based estimates are not shown for clarity—they are almost always worse than both BB and BC.

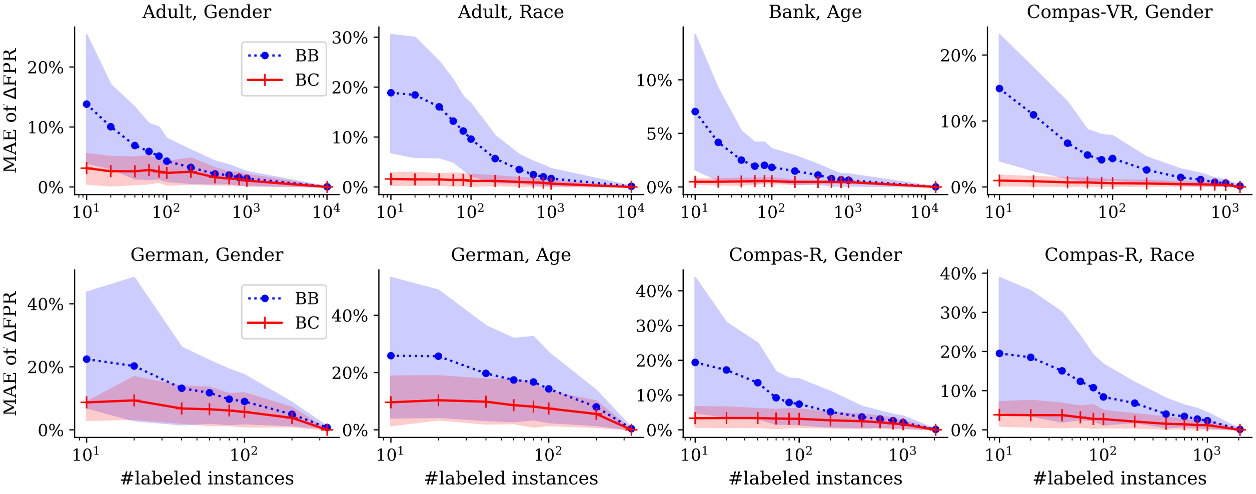


Figure 4.6: Mean absolute error (MAE) of the difference between algorithm estimates and ground truth for group difference in FPR, as a function of number of labeled instances, for 8 different dataset-group pairs. Shading indicates 95% error bars for each method.

As an example, Figure 4.6 illustrates the quality of estimation where Δ is the FPR group difference Δ for the MLP classification model, evaluated across 8 different dataset-group pairs. Each y-value is the average of 100 different randomly sampled sets of n_L instances, where n_L is the corresponding x-axis value. The BC method dominates BB across all datasets indicating that the calibrated scores are very effective at improving the accuracy in estimating group FPR. This is particularly true for small amounts of labeled data (e.g., up to $n_L = 100$) where the BB Method can be highly inaccurate, e.g., MAEs on the order of 10 or 20% when the true value of Δ is often less than 10%.

Table 4.2: **MAE for Δ Accuracy Estimates**, with $n_L = 10$, across 100 runs of labeled samples, for 4 different trained models (groups of columns) and 10 different dataset-group combinations (rows). Lowest error rate per row-col group in bold if the difference among methods are statistically significant under Wilcoxon signed-rank test (p=0.05). Estimation methods are Freq (Frequentist), BB, and BC. Freq and BB use only labeled samples, BC uses both labeled samples and unlabeled data. Trained models are multi-layer perceptron, logistic regression, random forests, and Gaussian naive Bayes.

	Multi-layer Perceptron			Logistic Regression			Random Forest			Gaussian Naive Bayes		
Dataset, Attribute	Freq	BB	BC	Freq	BB	BC	Freq	BB	BC	Freq	BB	BC
Adult, Race	16.5	18.5	**3.9**	16.4	18.7	**2.9**	16.5	18.2	**3.2**	17.6	18.9	**3.6**
Adult, Gender	19.7	17.4	**5.1**	19.1	16.1	**2.2**	17.7	17.4	**4.8**	19.7	16.2	**5.4**
Bank, Age	15.9	13.9	**2.5**	13.9	13.0	**1.4**	11.8	11.1	**1.0**	15.5	13.7	**1.7**
German, Age	34.6	19.8	**5.0**	37.1	21.2	**8.7**	33.6	18.7	**8.2**	36.6	20.4	**11.5**
German, Gender	30.7	21.6	**8.2**	25.6	17.4	**6.3**	27.7	19.3	**8.6**	30.0	20.1	**6.5**
Compas-R, Race	31.5	21.0	**4.2**	31.7	20.4	**4.8**	29.3	20.3	**2.4**	33.5	23.2	**8.4**
Compas-R, Gender	33.7	21.6	**5.0**	34.3	21.9	**3.8**	36.3	23.3	**4.4**	40.5	25.5	**13.7**
Compas-VR, Race	18.7	17.1	**4.0**	18.5	15.6	**4.4**	18.2	15.8	**2.4**	26.6	19.8	**6.5**
Compas-VR, Gender	20.6	16.9	**5.4**	19.9	16.6	**5.3**	22.3	19.0	**6.3**	31.3	21.5	**9.8**
Ricci, Race	23.5	17.7	**14.6**	14.6	14.6	**7.9**	6.3	12.2	**2.1**	8.9	13.1	**1.6**

In Appendix B.1 we show that the trend of results shown in Figure 4.6, namely that BC produces significantly more accurate estimates of group fairness metrics Δ, is replicated across all 4 classification models that we investigated, across FPR, TPR and Accuracy metrics, and across all datasets. To summarize the full set of results we show a subset in tabular form, across all 4 classification models and 10 dataset-group pairs, with n_L fixed: Table 4.2 for Accuracy with $n_L = 10$ and Table 4.3 for TPR with $n_L = 200$. (We used larger n_L values for TPR and FPR than for accuracy in the results above since TPR and FPR depend on estimating conditional probabilities that can have zero supporting counts in

Table 4.3: **MAE for Δ TPR Estimates**, with $n_L = 200$. Same setup as for Table 4.2. Compas-VR race and Ricci race are not included since there are no positive instances for some groups, and some entries under Freq cannot be estimated for the same reason.

	Multi-layer Perceptron			Logistic Regression			Random Forest			Gaussian Naive Bayes		
Dataset, Attribute	Freq	BB	BC	Freq	BB	BC	Freq	BB	BC	Freq	BB	BC
Adult, Race	—	12.5	**5.8**	—	14.7	**7.0**	—	14.3	**4.6**	—	14.6	**3.0**
Adult, Gender	16.3	14.3	**4.3**	15.8	14.0	**4.6**	16.1	14.2	**7.3**	15.0	13.4	11.5
Bank, Age	16.8	15.0	**4.8**	17.7	15.9	**4.2**	16.6	14.9	**3.1**	17.3	15.7	**2.3**
German, Age	4.7	4.7	**3.0**	5.6	5.4	**2.6**	5.1	5.1	**3.1**	6.8	6.5	**2.8**
German, Gender	**0.7**	1.0	1.6	3.3	3.3	**2.1**	3.1	3.2	**2.1**	4.8	4.7	**2.2**
Compas-R, Race	—	7.6	**2.5**	—	7.9	**2.6**	—	9.2	**2.1**	—	4.5	**2.0**
Compas-R, Gender	10.0	9.5	**1.9**	10.0	9.4	**1.8**	11.3	10.7	**2.6**	5.6	5.5	**0.3**
Compas-VR, Gender	14.9	12.2	**2.9**	8.9	10.7	**2.0**	14.6	10.5	7.2	12.5	10.0	**1.3**

the labeled data, causing a problem for frequentist estimators). The results above and in Appendix B.1 demonstrate the significant gains in accuracy that can be achieved with the proposed approach.

4.8.2 Discussion

For concreteness we demonstrated our results with three popular fairness metrics (Δ accuracy, TPR, and FPR) in the chapter. However, we can directly extend this approach to handle metrics such as calibration and balance [Kleinberg et al., 2016] as well as ratio-based metrics. In particular, by predicting the distribution of class labels y with the calibrated model scores, any fairness metric that can be defined as a deterministic function of calibrated model scores s, labels y and groups g can leverage unlabeled data to reduce variance using our proposed method.

Consideration of the bias-variance properties of the different methods reveals a fundamental trade-off. The labeled data contribute no bias to the estimate but can have high variance for small n_L, whereas the unlabeled data (via their calibrated scores) contribute little variance but can have a persistent bias due to potential misspecification in the parametric calibration model. An open question, that is beyond the scope of this work, is how to balance this bias-variance trade-off in a more adaptive fashion as a function of n_L and n_U, to further improve the accuracy of estimates of fairness metrics for arbitrary datasets. One potential option would be to a more flexible calibration method (e.g., Gaussian process calibration as proposed in Wenger et al. [2020]). Another option would be to automatically quantify the calibration bias and trade-off the contributions of labeled and unlabeled data accordingly in estimating θ_g's and Δ.

In the next section, we provide additional results about:

- evaluation of the calibration coverage of the posterior credible intervals generated by different methods (Section 4.9.1);
- comparisons with an alternative calibration model, i.e. LLO calibration (Section 4.9.2);
- ablation study by comparing with non-hierarchical Bayesian calibration (Section 4.9.3);
- sensitivity analysis for the calibration priors (Section 4.9.4).

4.9 Additional Results

4.9.1 Calibration Coverage of Posterior Credible Intervals

We can generate posterior credible intervals on Δ (as shown in red in Figure 4.5) for both the BB and BC methods by computing upper and lower percentiles from posterior samples for Δ. Below in Table 4.4 we show the coverage of 95% credible intervals for both the BB (beta-bernoulli) and BC (Bayesian-calibration) methods, for the multi-layer perceptron model. Coverage is defined as the percentage of credible intervals (across multiple different labeled datasets of size n_L) that contain the true value: a perfectly calibrated 95% credible interval would have 95% coverage.

Table 4.4 shows that while the coverage for both methods is generally not far from 95% there is room for improvement. For example, for small values of n_L the coverage of both methods is often too high (above 95%), with some evidence of coverage decreasing as n_L increasing. Generating accurate posterior credible intervals is a known issue in Bayesian analysis in the presence of model misspecification (e.g., Syring and Martin [2019]) and is an interesting direction for future work on Bayesian analysis of fairness metrics.

Table 4.4: **Calibration Coverage of Posterior Credible Intervals Comparison**, across 1000 runs of labeled samples of different sizes n_L for 10 different dataset-group combinations (rows). Estimation methods are BC (Bayesian-Calibration) and BB (beta-bernoulli). Trained model is multi-layer perceptron.

	$n_L = 10$		$n_L = 20$		$n_L = 40$		$n_L = 100$	
Group	BC	BB	BC	BB	BC	BB	BC	BB
Adult, Race	99.9	97.7	98.6	93.5	96.2	93.2	92.3	95.3
Adult, Gender	100.0	96.4	99.7	95.5	99.2	94.9	96.8	95.5
Bank, Age	99.4	98.7	98.8	98.5	98.0	96.4	93.7	95.3
German, age	99.9	98.8	99.6	98.1	99.0	98.3	96.9	98.3
German, Gender	99.1	97.4	99.1	97.4	97.7	96.4	94.6	97.8
Compas-R, Race	99.3	98.8	99.4	97.2	99.1	96.7	99.3	96.6
Compas-R, Gender	99.3	97.7	99.3	97.0	98.6	95.9	97.6	96.5
Compas-VR, Race	99.6	100.0	98.6	97.8	97.9	95.2	97.5	93.1
Compas-VR, Gender	96.3	97.2	94.3	96.5	95.4	96.1	95.8	97.1
Ricci, Race	93.2	99.7	91.4	99.7	—	—	—	—

4.9.2 Error Results with LLO Calibration

Our hierarchical Bayesian calibration approach can be adapted to use other parametric calibration methods. In addition to the beta calibration method described in the main paper, we also experimented with LLO (linear in log odds) calibration.

Table 4.5 shows a direct comparison of the mean absolute error (MAE) rate for estimation of differences in accuracy between groups (same setup as Tables 4.2 and 4.3 in terms of how MAE is computed). The results show that in general the MAE of the two calibration methods tends to be very similar (relative to the size of the BB and frequentist MAEs) across different dataset-attribute combinations, different prediction models, and different n_L values.

4.9.3 Error Results with Non-Hierarchical Bayesian Calibration

In our hierarchical Bayesian calibration model we allows different groups to share statistical strength via a hierarchical structure. In this section, we compare our proposed Bayesian calibration model (BC) that uses this hierarchy with a non-hierarchical Bayesian calibration (NHBC) approach.

Table 4.6 compares the mean absolute error (MAE) rate for both approaches in estimating differences in accuracy between groups. The results show that (1) both BC and NHBC significantly improve MAE compared to BB; (2) BC and NHBC are comparable in most cases, but with the hierarchical structure the BC method avoids occasional catastrophic errors that NHBC can make, e.g. when assessing Δ Accuracy of a Gaussian Naive Bayes model on Compas-R Gender and Compas-VR Gender.

4.9.4 Sensitivity Analysis for Calibration Priors

As discussed in Section 4.5, in our experiments we set the hyperparameters as

$$\begin{aligned}
\mu_a &\sim \mathrm{N}(0, .4), \sigma_a \sim \mathrm{TN}(0, .15)\\
\mu_b &\sim \mathrm{N}(0, .4), \sigma_b \sim \mathrm{TN}(0, .15)\\
\mu_c &\sim \mathrm{N}(0, 2), \sigma_c \sim \mathrm{TN}(0, .75)
\end{aligned}$$

Since we assumed that the parameters from each individual group are sampled from a shared distribution: $\log a_g \sim \mathrm{N}(\mu_a, \sigma_a), \log b_g \sim \mathrm{N}(\mu_b, \sigma_b), c_g \sim \mathrm{N}(\mu_c, \sigma_c)$, these prior distributions encode a weak prior belief that the model scores are calibrated by placing the mode of a_g, b_g and c_g at 1, 1, and 0 respectively. We used exactly these prior settings in all our experiments across all datasets, all groups, and all labeled and unlabeled dataset sizes, which already demonstrates to a certain extent the robustness of these settings.

In this section we describe the results of a sensitivity analysis with respect to the variances in the prior discussed in Section 4.5. We evaluate our proposed methodology over a range of settings for the variances, multiplying the default values with different values of α, i.e.

$$\begin{aligned}
\mu_a &\sim \mathrm{N}(0, .4\alpha), \sigma_a \sim \mathrm{TN}(0, .15\alpha) \\
\mu_b &\sim \mathrm{N}(0, .4\alpha), \sigma_b \sim \mathrm{TN}(0, .15\alpha) \\
\mu_c &\sim \mathrm{N}(0, 2\alpha), \sigma_c \sim \mathrm{TN}(0, .75\alpha)
\end{aligned}$$

with α ranging from 0.1 to 10. We reran our analysis, using the different variance settings, for the specific case of estimating the change Δ in accuracy estimates for the Adult dataset grouped by the attribute “race," for each of the four classification models in our study and with different amounts of labeled data.

Table 4.7 shows the resulting MAE values as α is varied. The results demonstrate that the Bayesian calibration (BC) model is robust to the settings of prior variances. Specifically, as α varies from 0.1 to 10 the MAE values with BC are almost always smaller than the ones obtained with BB, and there is a broad range of values α where the MAE values are close to their minimum The results also show that the BC method has less sensitivity to α when the number of labeled examples n_L is large, e.g. $n_L = 1000$.

Table 4.5: **Comparing LLO and BC:** MAE for Δ accuracy estimates of LLO and BC, with different n_L. Mean absolute error between estimates and true Δ across 100 runs of labeled samples of different sizes n_L for different trained models (groups of columns) and 10 different dataset-group combinations (groups of rows). Estimation methods are BC (Bayesian-Calibration) and LLO (Linear in Log Odds Calibration). Both methods use both labeled samples and unlabeled data.

		Multi-layer Perceptron		Logistic Regression		Random Forest		Gaussian Naive Bayes	
Group	n	BC	LLO	BC	LLO	BC	LLO	BC	LLO
Adult	10	3.9	3.8	2.9	2.8	3.2	3.2	3.6	3.5
Race	100	3.5	3.4	3.2	3.1	3.1	2.9	2.8	2.4
	1000	1.6	2.3	1.7	2.0	1.4	1.5	1.4	1.6
Adult	10	5.1	5.1	2.2	2.3	4.8	4.7	5.4	5.0
Gender	100	4.4	4.3	1.9	2.0	4.1	3.7	2.7	2.7
	1000	1.6	2.2	1.1	1.0	2.0	1.5	1.1	1.1
Bank	10	2.5	2.3	1.4	1.2	1.0	0.9	1.7	1.7
Age	100	2.0	2.0	1.2	1.2	0.9	0.9	1.1	1.2
	1000	1.1	1.2	0.7	0.7	0.5	0.5	0.8	0.9
German	10	5.0	4.6	8.7	8.0	8.2	7.5	11.5	10.7
age	100	3.9	4.1	3.8	4.7	4.3	4.0	4.2	6.0
	200	3.1	3.9	3.3	4.2	3.3	3.1	3.5	6.0
German	10	8.2	6.4	6.3	5.0	8.6	6.9	6.5	5.3
Gender	100	5.4	5.1	3.7	3.6	4.8	4.5	2.8	3.1
	200	3.0	3.4	2.9	2.8	2.9	3.1	2.2	2.9
Compas-R	10	4.2	4.6	4.8	5.2	2.4	2.5	8.4	8.2
Race	100	2.8	4.4	3.4	4.8	1.8	1.4	6.0	5.6
	1000	1.6	5.0	1.6	4.4	1.2	1.1	1.8	2.9
Compas-R	10	5.0	4.3	3.8	3.9	4.4	4.1	13.7	13.0
Gender	100	3.3	2.7	2.6	2.3	2.7	2.8	8.0	7.4
	1000	1.4	2.1	1.3	1.3	1.4	3.0	1.8	2.4
Compas-VR	10	4.0	3.9	4.4	4.7	2.4	2.9	6.5	6.4
Race	100	3.1	2.8	3.4	3.3	2.0	2.1	3.7	3.6
	1000	0.8	1.5	0.8	0.8	0.8	2.5	0.9	1.8
Compas-VR	10	5.4	4.8	5.3	5.2	6.3	8.2	9.8	9.0
Gender	100	3.4	3.0	3.1	3.3	4.4	5.4	4.5	4.2
	1000	0.9	1.2	0.9	1.5	1.0	1.7	0.9	0.9
Ricci	10	14.6	14.2	7.9	8.1	2.1	2.0	1.6	2.1
Race	20	9.8	13.6	7.1	6.6	1.5	1.6	2.1	2.5
	30	6.5	12.1	4.6	4.2	1.1	1.4	2.0	2.3

Table 4.6: **Comparing Hierarchical and Non-hierarchical Bayesian Calibration**, MAE for Δ accuracy estimates, with different n_L. Mean absolute error between estimates and true Δ across 100 runs of labeled samples of different sizes n_L for different trained models (groups of columns) and 10 different dataset-group combinations (groups of rows). Estimation methods are BB (beta-binomial), and NHBC (non-hierarchical Bayesian calibration), BC (Bayesian calibration). BB uses only labeled samples, NHBC and BC use both labeled samples and unlabeled data.

		Multi-layer Perceptron			Logistic Regression			Random Forest			Gaussian Naive Bayes		
Group	n	BB	NHBC	BC	BB	NHBC	BC	BB	NHBC	BC	BB	NHBC	BC
Adult Race	10	18.4	**3.2**	3.9	18.8	**2.7**	2.9	18.1	**2.8**	3.2	18.9	4.5	**3.6**
	20	16.1	**3.3**	4.4	16.7	**2.9**	3.4	16.3	**3.0**	3.7	16.8	4.1	**3.7**
	40	13.1	**2.8**	4.5	14.0	**2.9**	3.7	14.4	**2.9**	3.8	14.4	3.7	**3.3**
	100	8.6	**2.7**	3.5	9.2	**3.0**	3.2	9.0	**2.6**	3.1	9.6	**2.4**	2.8
	1000	2.5	**1.4**	1.6	2.3	2.1	**1.7**	2.1	**0.7**	1.4	2.3	1.8	**1.4**
Adult Gender	10	17.4	**4.1**	5.1	16.3	2.6	**2.2**	17.3	5.3	**4.8**	16.3	7.2	**5.4**
	20	12.9	**4.4**	5.1	12.2	2.6	**2.2**	12.4	5.3	**4.9**	11.6	6.7	**4.5**
	40	9.0	**4.1**	4.9	9.2	2.5	**2.1**	9.6	5.1	**4.5**	9.7	6.3	**3.9**
	100	5.4	**3.1**	4.4	5.5	2.0	**2.0**	5.9	4.7	**4.1**	6.0	4.8	**2.7**
	1000	1.9	**1.4**	1.6	1.7	**1.0**	1.1	**1.5**	1.8	2.0	1.5	**0.9**	1.0
Bank Age	10	14.0	**1.7**	2.5	12.8	1.5	**1.4**	11.2	1.1	**1.0**	13.7	**1.4**	1.7
	20	11.6	**2.3**	2.9	10.9	1.9	**1.7**	8.8	1.4	**1.2**	10.3	**1.6**	1.7
	40	8.0	**2.3**	2.6	7.3	1.7	**1.4**	6.5	1.5	**1.1**	7.5	1.7	**1.5**
	100	4.3	2.2	**2.0**	4.3	1.4	**1.2**	4.2	1.2	**0.9**	4.9	1.3	**1.1**
	1000	1.5	1.2	**1.1**	1.6	0.8	**0.7**	1.4	0.6	**0.5**	1.7	**0.7**	0.8
German age	10	19.7	5.6	**5.0**	21.3	10.3	**8.7**	19.1	**8.2**	8.2	20.4	14.2	**11.5**
	20	18.1	6.0	**4.4**	18.6	6.7	**6.4**	16.7	**7.0**	7.0	18.8	9.9	**9.0**
	40	15.9	6.7	**4.8**	15.0	5.6	**4.9**	11.7	6.6	**5.8**	14.9	**6.4**	6.9
	100	7.9	5.8	**3.9**	7.5	5.5	**3.8**	8.2	6.5	**4.3**	9.1	4.4	**4.2**
	200	4.2	3.7	**3.1**	4.4	4.1	**3.3**	4.7	4.1	**3.3**	4.7	3.8	**3.5**
German Gender	10	21.5	10.5	**8.2**	17.6	7.0	**6.3**	19.4	**8.5**	8.6	20.0	**5.9**	6.5
	20	16.2	10.0	**7.8**	13.2	7.1	**5.1**	14.1	8.4	**7.8**	15.4	5.9	**4.9**
	40	11.6	9.2	**6.6**	11.4	8.4	**4.5**	11.1	7.7	**5.9**	11.1	6.1	**3.8**
	100	7.1	6.5	**5.4**	6.9	6.6	**3.7**	7.0	6.1	**4.8**	5.9	6.4	**2.8**
	200	3.2	3.3	**3.0**	4.0	4.0	**2.9**	3.6	3.4	**2.9**	4.0	4.0	**2.2**
Compas-R Race	10	21.1	**2.9**	4.2	20.7	**4.0**	4.8	20.3	**1.4**	2.4	23.1	**6.6**	8.4
	20	14.8	**2.8**	3.3	15.2	3.9	**3.8**	15.8	**2.0**	2.5	16.6	**7.8**	8.0
	40	11.7	**3.0**	3.0	12.1	3.9	**3.6**	11.6	**2.0**	2.0	10.9	9.9	**8.1**
	100	6.8	2.9	**2.8**	7.4	3.7	**3.4**	8.5	2.1	**1.8**	7.9	7.7	**6.0**
	1000	2.0	**1.5**	1.6	1.9	**1.6**	1.7	1.9	1.3	**1.2**	1.9	1.9	**1.8**
Compas-R Gender	10	21.3	**3.8**	5.0	22.0	**3.4**	3.8	23.4	**3.5**	4.4	25.4	19.1	**13.7**
	20	18.5	**3.8**	5.1	18.4	**3.3**	4.0	17.4	**3.3**	4.6	21.4	23.8	**12.3**
	40	12.2	**3.4**	4.0	13.0	**3.0**	3.3	13.7	**2.8**	3.6	15.0	23.8	**9.5**
	100	8.8	**3.2**	3.3	9.1	2.7	**2.6**	8.5	**2.1**	2.7	9.8	15.5	**8.0**
	1000	2.0	1.7	**1.4**	2.2	1.4	**1.3**	2.4	1.6	**1.4**	1.9	1.9	**1.8**
Compas-VR Race	10	17.4	4.0	**4.0**	15.6	**4.4**	4.4	15.7	2.6	**2.4**	19.7	**6.1**	6.5
	20	13.5	4.7	**4.3**	13.7	5.0	**4.8**	13.6	3.3	**2.9**	15.9	10.7	**6.5**
	40	9.6	4.5	**3.8**	9.6	4.5	**3.9**	9.9	3.1	**2.4**	11.1	8.8	**5.5**
	100	5.6	3.6	**3.1**	5.2	3.8	**3.4**	6.2	2.6	**2.0**	6.6	6.8	**3.7**
	1000	0.9	0.8	**0.8**	0.9	**0.8**	0.8	0.9	0.8	**0.8**	1.1	1.2	**0.9**
Compas-VR Gender	10	17.2	5.6	**5.4**	16.8	5.7	**5.3**	19.0	**5.8**	6.3	21.3	18.9	**9.8**
	20	13.3	5.4	**5.1**	14.1	5.4	**4.9**	14.0	**5.7**	6.2	16.0	28.2	**8.7**
	40	9.3	5.1	**4.7**	9.7	4.9	**4.5**	10.5	**5.3**	5.7	12.4	30.9	**6.9**
	100	6.4	3.7	**3.4**	5.9	3.5	**3.1**	6.3	**4.2**	4.4	7.1	18.5	**4.5**
	1000	1.0	**0.8**	0.9	1.0	**0.9**	0.9	0.9	**0.9**	1.0	1.4	0.9	**0.9**
Ricci Race	10	17.7	16.1	**14.6**	14.4	**7.5**	7.9	12.2	**1.9**	2.1	13.1	1.7	**1.6**
	20	11.2	11.8	**9.8**	9.3	7.2	**7.1**	8.5	**1.5**	1.5	9.5	**2.0**	2.1
	30	7.4	7.7	**6.5**	5.8	5.1	**4.6**	6.0	1.1	**1.1**	6.4	**1.9**	2.0

Table 4.7: **Prior Sensitivity Results:** MAE for Δ accuracy estimates of the adult data grouped by attribute “race," with different values of n_L. Shown are mean absolute error (MAE) values between estimates and true Δ across 100 runs of labeled samples of different sizes n_L for different trained models (groups of columns). Estimation methods are BB (beta-binomial) and BC (Bayesian-calibration) with different values of α (rows). BB uses only labeled samples, and BC use both labeled samples and unlabeled data.

	Multi-layer Perceptron			Logistic Regression			Random Forest			Gaussian Naive Bayes		
Method	10	100	1000	10	100	1000	10	100	1000	10	100	1000
BB	18.52	8.48	2.46	18.74	9.14	2.30	18.24	9.00	2.12	18.88	9.54	2.32
BC, α=0.1	2.63	2.60	2.27	2.46	2.49	2.13	2.87	2.84	2.43	4.67	4.51	0.78
BC, α=0.2	2.63	2.56	2.08	2.46	2.51	2.06	2.85	2.83	2.09	4.63	3.95	0.82
BC, α=0.3	2.60	2.52	1.88	2.42	2.51	1.95	2.85	2.79	1.86	4.44	3.36	0.97
BC, α=0.4	2.49	2.46	1.74	2.41	2.57	1.90	2.74	2.82	1.70	4.25	3.06	1.11
BC, α=0.5	2.49	2.38	1.71	2.44	2.60	1.82	2.82	2.77	1.65	4.01	2.86	1.43
BC, α=0.6	2.47	2.37	1.62	2.55	2.62	1.75	2.82	2.88	1.60	3.81	2.79	1.46
BC, α=0.7	2.61	2.48	1.51	2.36	2.63	1.70	2.90	2.86	1.54	3.54	2.80	1.50
BC, α=0.8	2.86	2.30	1.47	2.52	2.73	1.63	2.87	2.86	1.46	3.51	2.77	1.60
BC, α=0.9	2.93	2.27	1.43	2.44	2.82	1.64	2.87	2.90	1.46	3.14	2.91	1.58
BC, α=1.0	3.05	2.31	1.50	2.71	2.74	1.57	2.99	2.96	1.42	3.31	2.85	1.68
BC, α=1.1	3.14	2.37	1.45	2.65	2.86	1.55	2.90	3.10	1.40	3.25	3.03	1.65
BC, α=1.2	3.11	2.19	1.49	2.73	2.80	1.52	3.27	3.01	1.39	3.20	3.03	1.68
BC, α=1.3	3.48	2.30	1.51	2.91	2.94	1.54	3.11	3.21	1.39	3.15	2.96	1.71
BC, α=1.4	3.76	2.28	1.47	3.17	3.01	1.51	3.26	3.21	1.30	3.48	3.21	1.75
BC, α=1.5	3.67	2.20	1.49	3.12	2.94	1.51	3.46	3.05	1.34	3.23	3.19	1.66
BC, α=1.6	4.06	2.24	1.45	3.26	2.93	1.47	3.56	3.13	1.33	3.48	3.17	1.69
BC, α=1.7	4.02	2.27	1.46	3.46	3.15	1.46	3.75	3.10	1.27	3.43	3.19	1.74
BC, α=1.8	4.35	2.14	1.42	3.36	3.09	1.50	3.76	3.26	1.29	3.67	3.22	1.81
BC, α=1.9	4.35	2.30	1.48	3.48	2.94	1.42	3.54	3.30	1.28	3.82	3.35	1.84
BC, α=2.0	4.69	2.16	1.44	3.87	2.99	1.54	3.91	3.46	1.21	3.83	3.18	1.81
BC, α=5.0	8.11	2.54	1.63	6.31	3.32	1.53	5.32	4.13	1.31	5.25	3.82	2.13
BC, α=10.0	10.39	2.63	1.63	7.18	3.83	1.70	7.19	4.41	1.42	6.32	4.08	2.33

4.10 Related Work

Our Bayesian calibration approach builds on the work of Turner et al. [2014] who used hierarchical Bayesian methods for calibration of human judgement data using the LLO calibration model. Other Bayesian approaches to classifier calibration include marginalizing over binned model scores [Naeini et al., 2015] and calibration based on Gaussian processes [Wenger et al., 2020]. The Bayesian framework of Welinder et al. [2013] in particular is close in spirit to our work in that unlabeled examples are used to improve calibration, but differs in that a generative mixture model is used for modeling of scores rather than direct calibration. None of this prior work on Bayesian calibration addresses fairness assessment and none (apart from Welinder et al. [2013]) leverages unlabeled data.

There has also been work on uncertainty-aware assessment of classifier performance such as the use of Bayesian inference for classifier-related metrics such as marginal accuracy [Benavoli et al., 2017] and precision-recall [Goutte and Gaussier, 2005]. Although these approaches share similarities with our work, they do not make use of unlabeled data. In contrast, the Bayesian evaluation methods proposed by Johnson et al. [2019] can use unlabeled data but makes strong prior assumptions that are specific to the application domain of diagnostic testing. More broadly, other general approaches have been proposed for label-efficient classifier assessment including stratified sampling [Sawade et al., 2010], importance sampling [Kumar and Raj, 2018], and active assessment with Thompson sampling [Ji et al., 2020]. All of these ideas could in principle be used in conjunction with our approach to further reduce estimation error.

In the literature on algorithmic fairness there has been little prior work on uncertainty-aware assessment of fairness metrics—one exception is the proposed use of frequentist confidence interval methods for groupwise fairness in Besse et al. [2018].

Dimitrakakis et al. [2019] proposed a framework called “Bayesian fairness," but focused on decision-theoretic aspects of the problem rather than estimation of metrics. Foulds et al. [2020] developed Bayesian approches for for parametric smoothing across groups to improve the quality of estimation of intersectional fairness metrics. However, none of this work makes use of unlabeled data to improve fairness assessment. And while there is prior work in fairness on leveraging unlabeled data [Chzhen et al., 2019, Noroozi et al., 2019, Wick et al., 2019, Zhang et al., 2020], the goal of that work has been to produce classifiers that are fair, rather than to assess the fairness of existing classifiers.

Finally, there is recent concurrent work from a frequentist perspective that uses Bernstein inequalities and knowledge of group proportions to upper bound the probability that the difference between the frequentist estimate of Δ and the true Δ exceeds some value [Ethayarajh, 2020]. While this work differs from our approach in that it does not explore the use of unlabeled data, the same broad conclusion is reached, namely that there can be high uncertainty in empirical estimates of groupwise fairness metrics, given the typical sizes of datasets used in machine learning.

4.11 Conclusions

To answer to the question “can I trust my fairness metric," we have stressed the importance of being aware of uncertainty in fairness assessment, especially when test sizes are relatively small (as is often the case in practice). To address this issue we propose a framework for combining labeled and unlabeled data to reduce estimation variance, using Bayesian calibration of model scores on unlabeled data. The results demonstrate that the proposed method can systematically produce significantly more accurate estimates of fairness metrics, when compared to only using labeled data, across multiple different classification models, datasets, and sensitive attributes. The framework is straightforward to apply in practice and easy to extend to problems such as intersectional fairness (where estimation uncertainty is likely a significant issue) and to evaluation of fairness-aware algorithms.

In particular, the three primary contributions are

- We proposed a comprehensive Bayesian treatment of fairness assessment that provides uncertainty about estimates of group fairness metrics;
- We developed a new hierarchical Bayesian methodology that leverages information from both unlabeled and labeled examples;
- We demonstrated with systematic large-scale experiments across multiple datasets and models that using unlabeled data can reduce estimation error significantly.

Bibliography

Shipra Agrawal and Navin Goyal. Analysis of Thompson sampling for the multi-armed bandit problem. In *Conference on Learning Theory*, pages 39.1–39.26, 2012.

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. How we analyzed the COMPAS recidivism algorithm. *URL https://www. propublica. org/article/how-we-analyzed-the-compas-recidivism-algorithm*, 2017.

Javed A Aslam, Virgil Pavlu, and Emine Yilmaz. A statistical method for system evaluation using incomplete judgments. In *Proceedings of the ACM Conference on Research and Development in Information retrieval*, pages 541–548, 2006.

Alessio Benavoli, Giorgio Corani, Janez Demšar, and Marco Zaffalon. Time for a change: a tutorial for comparing multiple classifiers through Bayesian analysis. *The Journal of Machine Learning Research*, 18(1):2653–2688, 2017.

Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research*, 2018.

Philippe Besse, Eustasio del Barrio, Paula Gordaliza, and Jean-Michel Loubes. Confidence intervals for testing disparate impact in fair learning. *arXiv preprint arXiv:1807.06362*, 2018.

Alex Beutel, Jilin Chen, Tulsee Doshi, Hai Qian, Allison Woodruff, Christine Luu, Pierre Kreitmann, Jonathan Bischof, and Ed H Chi. Putting fairness principles into practice: Challenges, metrics, and improvements. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 453–459, 2019.

Glenn W Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

Jochen Bröcker and Leonard A Smith. Increasing the reliability of reliability diagrams. *Weather and Forecasting*, 22(3):651–661, 2007.

Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11621–11631, 2020.

Toon Calders and Sicco Verwer. Three naive Bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, 21(2):277–292, 2010.

Olivier Chapelle and Lihong Li. An empirical evaluation of Thompson sampling. In *Advances in Neural Information Processing Systems*, pages 2249–2257, 2011.

Irene Y Chen, Peter Szolovits, and Marzyeh Ghassemi. Can AI help reduce disparities in general medical and mental health care? *AMA Journal of Ethics*, 21(2):167–179, 2019.

Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data*, 5(2):153–163, 2017.

Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, Luca Oneto, and Massimiliano Pontil. Leveraging labeled and unlabeled data for consistent fair binary classification. In *Advances in Neural Information Processing Systems*, pages 12739–12750, 2019.

Sam Corbett-Davies and Sharad Goel. The measure and mismeasure of fairness: A critical review of fair machine learning. *arXiv preprint arXiv:1808.00023*, 2018.

Morris H DeGroot and Stephen E Fienberg. The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 32(1-2):12–22, 1983.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT 2019*, volume 1, pages 4171–4186, 2019.

Christos Dimitrakakis, Yang Liu, David C Parkes, and Goran Radanovic. Bayesian fairness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 509–516, 2019.

Xianzhi Du, Mostafa El-Khamy, Jungwon Lee, and Larry Davis. Fused DNN: A deep neural network fusion approach to fast and robust pedestrian detection. In *Winter Conference on Applications of Computer Vision*, pages 953–961, 2017.

Dheeru Dua and Casey Graff. UCI Machine Learning Repository. 2017.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pages 214–226, 2012.

Kawin Ethayarajh. Is your classifier actually biased? Measuring fairness under uncertainty with Bernstein bounds. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 2914–2919, 2020.

Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 259–268, 2015.

James R Foulds, Rashidul Islam, Kamrun Naher Keya, and Shimei Pan. Bayesian modeling of intersectional fairness: The variance of bias. In *Proceedings of the SIAM International Conference on Data Mining*, pages 424–432, 2020.

Alexander Freytag, Erik Rodner, and Joachim Denzler. Selecting influential examples: active learning with expected model output changes. In *Proceedings of the European Conference on Computer Vision*, pages 562–577. Springer, 2014.

Sorelle A Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P Hamilton, and Derek Roth. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 329–338, 2019.

Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the International Conference on Machine Learning*, pages 1050–1059, 2016.

Andrew Gelman, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian Data Analysis*. CRC press, 2013.

John C Gittins and David M Jones. A dynamic allocation index for the discounted multiarmed bandit problem. *Biometrika*, 66(3):561–565, 1979.

Cyril Goutte and Eric Gaussier. A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In *Proceedings of the European Conference on Information Retrieval*, pages 345–359, 2005.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1321–1330, 2017.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems*, pages 3315–3323, 2016.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition*, pages 770–778, 2016.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *ICLR*, 2019.

Harold Jeffreys. An invariant form for the prior probability in estimation problems. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 186(1007): 453–461, 1946.

Disi Ji, Robert L Logan IV, Padhraic Smyth, and Mark Steyvers. Active Bayesian assessment for black-box classifiers. *arXiv preprint arXiv:2002.06532*, 2020.

Wesley O Johnson, Geoff Jones, and Ian A Gardner. Gold standards are out and Bayes is in: Implementing the cure for imperfect reference tests in diagnostic accuracy studies. *Preventive Veterinary Medicine*, 167:113–127, 2019.

Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 35–50. Springer, 2012.

Daniel S Kermany, Michael Goldbaum, Wenjia Cai, Carolina CS Valentim, Huiying Liang, Sally L Baxter, Alex McKeown, Ge Yang, Xiaokang Wu, Fangbing Yan, et al. Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell*, 172(5): 1122–1131, 2018.

Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*, 2016.

Junpei Komiyama, Junya Honda, and Hiroshi Nakagawa. Optimal regret analysis of Thompson sampling in stochastic multi-armed bandit problem with multiple plays. In *Proceedings of the International Conference on Machine Learning*, pages 1152–1161, 2015.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

Meelis Kull, Telmo Silva Filho, and Peter Flach. Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 623–631, 2017.

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. In *Advances in Neural Information Processing Systems*, pages 12295–12305, 2019.

Ananya Kumar, Percy S Liang, and Tengyu Ma. Verified uncertainty calibration. In *Advances in Neural Information Processing Systems*, pages 3787–3798, 2019.

Anurag Kumar and Bhiksha Raj. Classifier risk estimation under limited labeling resources. In *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 3–15. Springer, 2018.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, pages 6402–6413, 2017.

Ken Lang. Newsweeder: Learning to filter netnews. In *Proceedings of the International Conference on Machine Learning*, pages 331–339. Elsevier, 1995.

Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms.* Cambridge University Press, 2020.

Dan Li and Evangelos Kanoulas. Active sampling for large-scale information retrieval evaluation. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 49–58, 2017.

Zachary C Lipton, Yu-Xiang Wang, and Alex Smola. Detecting and correcting for label shift with black box predictors. In *Proceedings of the International Conference on Machine Learning*, pages 3128–3136, 2018.

Jun S Liu. *Monte Carlo Strategies in Scientific Computing*. Springer Science & Business Media, 2008.

E Clare Marshall and David J Spiegelhalter. League tables of in vitro fertilisation clinics: how confident can we be about the rankings. *British Medical Journal*, 316:1701–1704, 1998.

Alistair Moffat, William Webber, and Justin Zobel. Strategic system comparisons via targeted relevance judgments. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 375–382, 2007.

Sérgio Moro, Paulo Cortez, and Paulo Rita. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31, 2014.

Allan H Murphy and Robert L Winkler. Reliability of subjective probability forecasts of precipitation and temperature. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 26(1):41–47, 1977.

Mahdi Pakdaman Naeini, Gregory F Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using Bayesian binning. In *Proceedings of the AAAI Conference*, pages 2901–2907, 2015.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

Phuc Nguyen, Deva Ramanan, and Charless Fowlkes. Active testing: an efficient and robust framework for estimating accuracy. In *Proceedings of the International Conference on Machine Learning*, pages 3759–3768, 2018.

Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the International Conference on Machine Learning*, pages 625–632, 2005.

Jeremy Nixon, Michael W. Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. Measuring calibration in deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 38–41, June 2019.

Vahid Noroozi, Sara Bahaadini, Samira Sheikhi, Nooshin Mojab, and Philip Yu. Leveraging semi-supervised learning for fairness using neural networks. In *Proceedings of the IEEE International Conference On Machine Learning And Applications*, pages 50–55, 2019.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems*, pages 13991–14002, 2019.

Martyn Plummer. JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling. In *Proceedings of the International Workshop on Distributed Statistical Computing*, 2003.

Maithra Raghu, Katy Blumer, Greg Corrado, Jon Kleinberg, Ziad Obermeyer, and Sendhil Mullainathan. The algorithmic automation problem: prediction, triage, and human effort. *arXiv preprint arXiv:1903.12220*, 2019.

Md Mustafizur Rahman, Mucahid Kutlu, and Matthew Lease. Constructing test collections using multi-armed bandits and active learning. In *Proceedings of the World Wide Web Conference*, pages 3158–3164, 2019.

Md Mustafizur Rahman, Mucahid Kutlu, Tamer Elsayed, and Matthew Lease. Efficient test collection construction via active learning. In *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval*, pages 177–184, 2020.

Alvin Rajkomar, Michaela Hardt, Michael D Howell, Greg Corrado, and Marshall H Chin. Ensuring fairness in machine learning to advance health equity. *Annals of Internal Medicine*, 169(12):866–872, 2018.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *Proceedings of the International Conference on Machine Learning*, pages 5389–5400, 2019.

Herbert Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

Daniel Russo. Simple Bayesian algorithms for best arm identification. In *Proceedings of the Conference on Learning Theory*, pages 1417–1418, 2016.

Daniel J Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, Zheng Wen, et al. A tutorial on Thompson sampling. *Foundations and Trends in Machine Learning*, 11(1):1–96, 2018.

Ashish Sabharwal and Hanie Sedghi. How good are my predictions? Efficiently approximating precision-recall curves for massive datasets. In *UAI*, 2017.

Amartya Sanyal, Matt J Kusner, Adrià Gascón, and Varun Kanade. TAPAS: Tricks to accelerate (encrypted) prediction as a service. In *Proceedings of the International Conference on Machine Learning*, volume 80, pages 4490–4499. PMLR, 2018.

Christoph Sawade, Niels Landwehr, Steffen Bickel, and Tobias Scheffer. Active risk estimation. In *Proceedings of the 27th International Conference on Machine Learning*, pages 951–958, 2010.

Steven L Scott. A modern bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*, 26(6):639–658, 2010.

Burr Settles. *Active Learning.* Synthesis Lectures on AI and ML. Morgan Claypool, 2012.

Malcolm Strens. A Bayesian framework for reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, volume 2000, pages 943–950, 2000.

Supreme Court of the United States. Ricci v. Destefano. *557 U.S. 557, 174.*, page 2658, 2009.

Nicholas Syring and Ryan Martin. Calibrating general posterior credible regions. *Biometrika*, 106(2):479–486, 2019.

William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

William R Thompson. On the theory of apportionment. *American Journal of Mathematics*, 57(2):450–456, 1935.

Brandon M Turner, Mark Steyvers, Edgar C Merkle, David V Budescu, and Thomas S Wallsten. Forecast aggregation via recalibration. *Machine Learning*, 95(3):261–289, 2014.

Juozas Vaicenavicius, David Widmann, Carl Andersson, Fredrik Lindsten, Jacob Roll, and Thomas Schön. Evaluating model calibration in classification. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 3459–3467, 2019.

Alexander Vezhnevets, Joachim M Buhmann, and Vittorio Ferrari. Active learning for semantic segmentation with expected change. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3162–3169. IEEE, 2012.

Ellen M Voorhees. On building fair and reusable test collections using bandit techniques. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 407–416, 2018.

Peter Welinder, Max Welling, and Pietro Perona. A lazy man's approach to benchmarking: Semisupervised classifier evaluation and recalibration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3262–3269, 2013.

Jonathan Wenger, Hedvig Kjellström, and Rudolph Triebel. Non-parametric calibration for classification. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 178–190. PMLR, 2020.

Michael Wick, Swetasudha Panda, and Jean-Baptiste Tristan. Unlocking fairness: a trade-off revisited. In *Advances in Neural Information Processing Systems*, pages 8780–8789, 2019.

Jeremy Wyatt. *Exploration and Inference in Learning from Reinforcement.* PhD thesis, 1998.

Yuanshun Yao, Zhujun Xiao, Bolun Wang, Bimal Viswanath, Haitao Zheng, and Ben Y Zhao. Complexity vs. performance: empirical analysis of machine learning as a service. In *Internet Measurement Conference*, pages 384–397, 2017.

Xin Yi, Ekta Walia, and Paul Babyn. Generative adversarial network in medical imaging: A review. *Medical Image Analysis*, 58:101552, 2019.

Emine Yilmaz and Javed A Aslam. Estimating average precision with incomplete and imperfect judgments. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, pages 102–111, 2006.

Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining*, pages 694–699, 2002.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1171–1180, 2017.

Tao Zhang, Jing Li, Mengde Han, Wanlei Zhou, Philip Yu, et al. Fairness in semi-supervised learning: Unlabeled data help to reduce discrimination. *IEEE Transactions on Knowledge and Data Engineering*, 2020.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, pages 649–657, 2015.

Appendix A

Supplemental Material for Chapter 3

A.1 Different Multi-Armed Bandit Algorithms for Best-Arm(s) Identification

In Figure A.1 and A.2 we provide the algorithm for identifying the least accurate class with Thompson sampling(TS) and Top-two Thompson sampling(TTTS). In Figure A.3 we provide the algorithm for identifying the least accurate m classes with multiple-play Thompson sampling(MP-TS).

Algorithm 2 Thompson Sampling (TS) Strategy

```
**Input:** prior hyperparameters $\alpha$, $\beta$
initialize $n_{k,0} = n_{k,1} = 0$ for $k = 1$ **to** $K$
**repeat**
    # Sample accuracy for each predicted class
    **for** $k = 1$ **to** $K$ **do**
        $\widetilde{\theta}_k \sim \mathrm{Beta}(\alpha + n_{k,0}, \beta + n_{k,1})$
    **end for**
    # Select a class $k$ with the lowest sampled accuracy
    $\hat{k} = \arg\min_k \widetilde{\theta}_{1:K}$
    # Randomly select an input data point from the $\hat{k}$-th class and compute its predicted label
    $\mathbf{x}_i \sim \mathcal{R}_{\hat{k}}$
    $\hat{y}_i = \arg\max_k p_M(y = k|\mathbf{x}_i)$
    # Update parameters of the $\hat{k}$-th metric
    **if** $\hat{y}_i = \hat{k}$ **then**
        $n_{\hat{k},0} \leftarrow n_{\hat{k},0} + 1$
    **else**
        $n_{\hat{k},1} \leftarrow n_{\hat{k},1} + 1$
    **end if**
**until** all data labeled
```

Figure A.1: Thompson Sampling (TS) for identifying the least accurate class.

Algorithm 3 Top Two Thompson Sampling (TTTS) Strategy

```
Input: prior hyperparameters $\alpha$, $\beta$
initialize $n_{k,0} = n_{k,1} = 0$ for $k = 1$ to $K$
repeat
    # Sample accuracy for each predicted class
    for $k = 1$ to $K$ do
        $\widetilde{\theta}_k \sim \text{Beta}(\alpha + n_{k,0}, \beta + n_{k,1})$
    end for
    # Select a class $k$ with the lowest sampled accuracy
    $I = \arg\min_k \widetilde{\theta}_{1:K}$
    # Decide whether to re-sample
    $B \sim \text{Bernoulli}(\beta)$
    if $B = 1$ then
        # If not re-sample, select $I$
        $\hat{k} = I$
    else
        # If re-sample, keep sampling until a different arm $J$ is selected
        repeat
            for $k = 1$ to $K$ do
                $\widetilde{\theta}_k \sim \text{Beta}(\alpha + n_{k,0}, \beta + n_{k,1})$
            end for
            $J = \arg\min_k \widetilde{\theta}_{1:K}$
        until $J \neq I$
        $\hat{k} = J$
    end if
    # Randomly select an input data point from the $\hat{k}$-th class and compute its predicted label
    $\mathbf{x}_i \sim \mathcal{R}_{\hat{k}}$
    $\hat{y}_i = \arg\max_k p_M(y = k|\mathbf{x}_i)$
    # Update parameters of the $\hat{k}$-th metric
    if $\hat{y}_i = \hat{k}$ then
        $n_{\hat{k},0} \leftarrow n_{\hat{k},0} + 1$
    else
        $n_{\hat{k},1} \leftarrow n_{\hat{k},1} + 1$
    end if
until all data labeled
```

Figure A.2: Top Two Thompson Sampling (TTTS) for identifying the least accurate class.

Algorithm 4 Multiple-play Thompson sampling (MP-TS) Strategy

```
**Input:** prior hyperparameters $\alpha$, $\beta$
initialize $n_{k,0} = n_{k,1} = 0$ for $k = 1$ **to** $K$
**repeat**
    # Sample accuracy for each predicted class
    **for** $k = 1$ **to** $K$ **do**
        $\widetilde{\theta}_k \sim \text{Beta}(\alpha + n_{k,0}, \beta + n_{k,1})$
    **end for**
    # Select a set of $m$ classes with the lowest sampled accuracies
    $I^* =$ top-m arms ranked by $\widetilde{\theta}_k$.
    **for** $\hat{k} \in I^*$ **do**
        # Randomly select an input data point from the $\hat{k}$-th class and compute its predicted label
        $\mathbf{x}_i \sim \mathcal{R}_{\hat{k}}$
        $\hat{y}_i = \arg\max_k p_M(y = k|\mathbf{x}_i)$
        # Update parameters of the $\hat{k}$-th metric
        **if** $\hat{y}_i = \hat{k}$ **then**
            $n_{\hat{k},0} \leftarrow n_{\hat{k},0} + 1$
        **else**
            $n_{\hat{k},1} \leftarrow n_{\hat{k},1} + 1$
        **end if**
    **end for**
**until** all data labeled
```

Figure A.3: Multiple-play Thompson Sampling (MP-TS) for identifying the least accurate m classes.

Appendix B

Supplemental Material for Chapter 4

B.1 Complete Experimental Results

In Figure 4 and in Tables 2 and 3 in the main paper we reported summary results of systematic comparisons between the frequentist method, the Beta-Binomial model (BB) method, and the Bayesian calibration (BC) method, in terms of the mean absolute estimation error as a function of the number of labeled examples n_L.

In this section we provide complete tables and graphs for these results. In the tables the lowest error rate per row-column group is in bold if the difference among methods is statistically significant under a Wilcoxon signed-rank test (p=0.05). As in the results in the main paper, the results below demonstrate that BC produces significantly more accurate estimates of group fairness metrics Δ than the BB or frequentist estimates, across all 4 classification models that we investigated, across FPR, TPR and Accuracy metrics, and across all datasets[1]

[1]—In Tables B.2 and B.3 there are entries where the frequentist estimates of TPR or FPR do not exist.

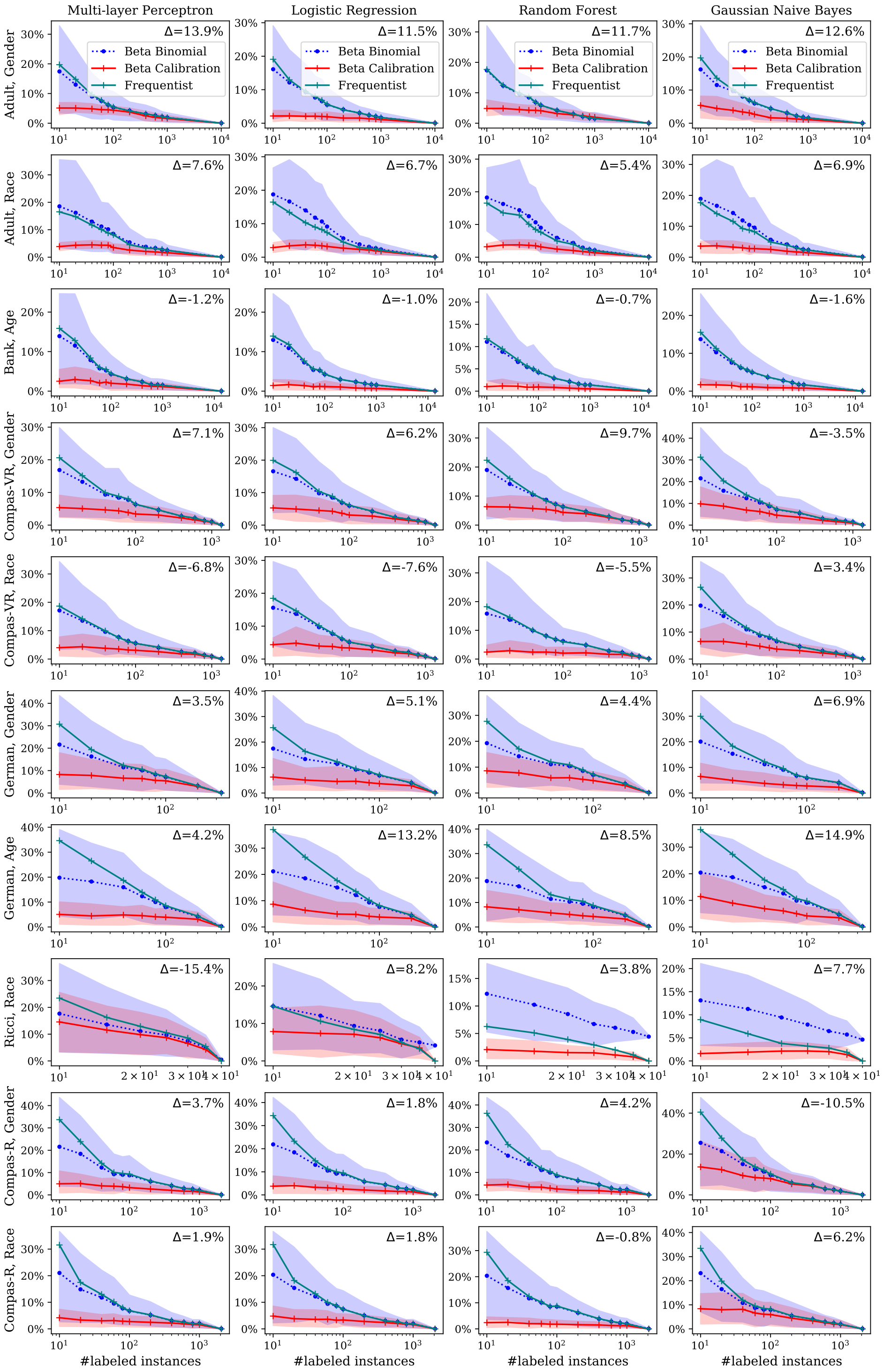


Figure B.1: **MAE for Accuracy:** Mean absolute error (MAE) of the difference between algorithm estimates and ground truth for group difference in accuracy across 100 runs, as a function of number of labeled instances, for 10 different dataset-group pairs and 4 classifiers. Shading indicates 95% error bars for each method (not shown for the frequentist curve to avoid overplotting). Upper right corner shows the ground truth Δ between the unprivileged group and the privileged group.

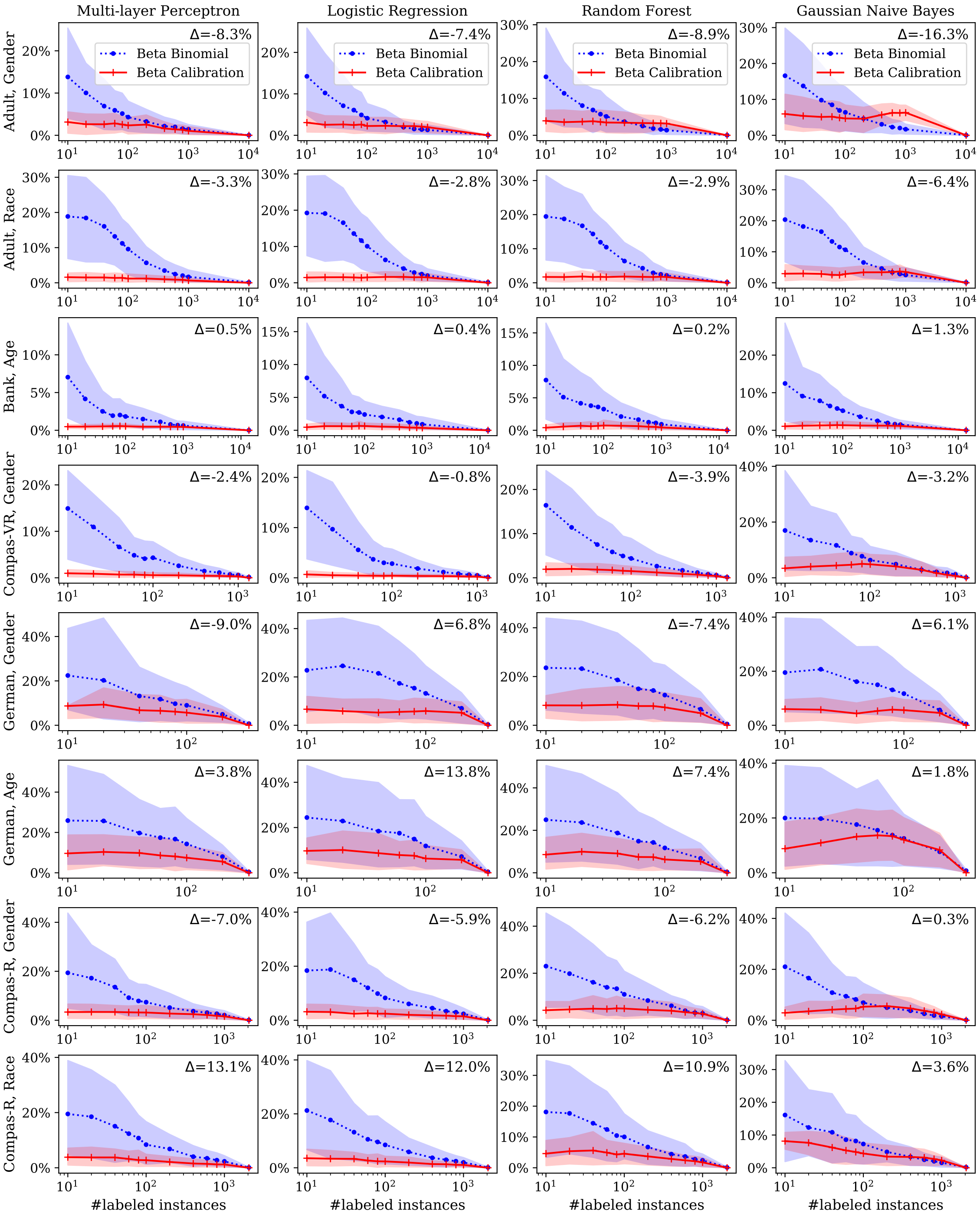


Figure B.2: **MAE for TPR:** Mean absolute error (MAE) of the difference between algorithm estimates and ground truth for group difference in TPR across 100 runs. Compas-VR race and Ricci race are not included since there are no positive instances for some groups. Same setup as Figure B.1.

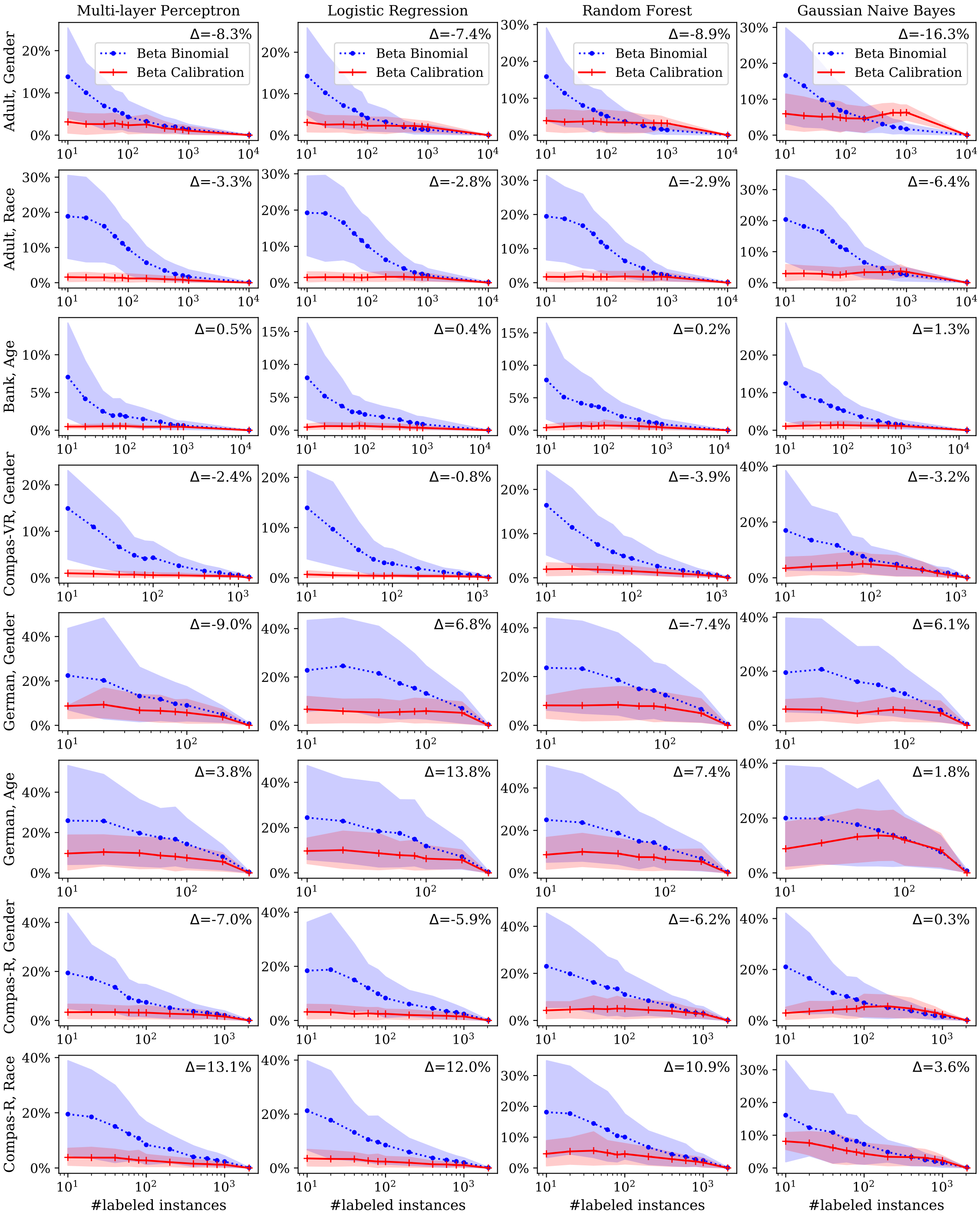


Figure B.3: **MAE for FPR:** Mean absolute error (MAE) of the difference between algorithm estimates and ground truth for groupwise difference in FPR across 100 runs. Same setup as Figure B.2.

Table B.1: **MAE for Δ Accuracy Estimates**, with different n_L. Mean absolute error between estimates and true Δ across 100 runs of labeled samples of different sizes n_L for different trained models (groups of columns) and 10 different dataset-group combinations (groups of rows). Estimation methods are Freq (Frequentist), BB (Beta-Binomial), and BC (Bayesian-Calibration). Freq and BB use only labeled samples, BC uses both labeled samples and unlabeled data. Trained models are Multilayer Perceptron, Logistic Regression, Random Forests, and Gaussian NaiveBayes.

		Multi-layer Perceptron			Logistic Regression			Random Forest			Gaussian Naive Bayes		
Group	n	Freq	BB	BC	Freq	BB	BC	Freq	BB	BC	Freq	BB	BC
Adult	10	16.5	18.5	**3.9**	16.4	18.7	**2.9**	16.5	18.2	**3.2**	17.6	18.9	**3.6**
Race	100	8.2	8.5	**3.5**	7.3	9.1	**3.2**	7.6	9.0	**3.1**	8.2	9.5	**2.8**
	1000	2.5	2.5	**1.6**	2.1	2.3	**1.7**	2.0	2.1	**1.4**	2.3	2.3	**1.4**
Adult	10	19.7	17.4	**5.1**	19.1	16.1	**2.2**	17.7	17.4	**4.8**	19.7	16.2	**5.4**
Gender	100	5.5	5.4	4.4	5.6	5.5	**1.9**	5.9	5.9	**4.1**	6.2	6.0	**2.7**
	1000	1.9	1.9	1.6	1.7	1.7	**1.1**	1.6	**1.5**	2.0	1.6	1.5	**1.1**
Bank	10	15.9	13.9	**2.5**	13.9	13.0	**1.4**	11.8	11.1	**1.0**	15.5	13.7	**1.7**
Age	100	4.4	4.3	**2.0**	4.3	4.3	**1.2**	4.3	4.2	**0.9**	5.0	5.0	**1.1**
	1000	1.5	1.5	**1.1**	1.6	1.6	**0.7**	1.4	1.4	**0.5**	1.7	1.7	**0.8**
German	10	34.6	19.8	**5.0**	37.1	21.2	**8.7**	33.6	18.7	**8.2**	36.6	20.4	**11.5**
age	100	8.5	8.0	**3.9**	8.2	7.6	**3.8**	8.8	8.2	**4.3**	9.7	9.1	**4.2**
	200	4.4	4.2	**3.1**	4.5	4.4	**3.3**	4.9	4.8	**3.3**	4.8	4.7	**3.5**
German	10	30.7	21.6	**8.2**	25.6	17.4	**6.3**	27.7	19.3	**8.6**	30.0	20.1	**6.5**
Gender	100	7.3	7.1	**5.4**	7.1	6.9	**3.7**	7.2	7.0	**4.8**	6.0	5.9	**2.8**
	200	3.2	3.2	3.0	4.0	3.9	**2.9**	3.6	3.5	**2.9**	4.0	4.0	**2.2**
Compas-R	10	31.5	21.0	**4.2**	31.7	20.4	**4.8**	29.3	20.3	**2.4**	33.5	23.2	**8.4**
Race	100	6.8	6.8	**2.8**	7.4	7.4	**3.4**	8.7	8.5	**1.8**	8.2	7.9	**6.0**
	1000	2.0	2.0	**1.6**	1.9	1.9	1.6	1.9	2.0	**1.2**	2.0	1.9	1.8
Compas-R	10	33.7	21.6	**5.0**	34.3	21.9	**3.8**	36.3	23.3	**4.4**	40.5	25.5	**13.7**
Gender	100	9.3	8.8	**3.3**	9.5	9.0	**2.6**	8.8	8.5	**2.7**	10.2	9.7	**8.0**
	1000	2.1	2.0	**1.4**	2.2	2.2	**1.3**	2.4	2.4	**1.4**	1.9	1.9	**1.8**
Compas-VR	10	18.7	17.1	**4.0**	18.5	15.6	**4.4**	18.2	15.8	**2.4**	26.6	19.8	**6.5**
Race	100	5.5	5.6	**3.1**	5.1	5.1	**3.4**	6.0	6.3	**2.0**	6.8	6.6	**3.7**
	1000	0.9	0.9	**0.8**	0.9	0.9	0.8	0.9	0.9	**0.8**	1.1	1.1	**0.9**
Compas-VR	10	20.6	16.9	**5.4**	19.9	16.6	**5.3**	22.3	19.0	**6.3**	31.3	21.5	**9.8**
Gender	100	6.4	6.3	**3.4**	6.1	6.0	**3.1**	6.3	6.3	**4.4**	7.3	7.1	**4.5**
	1000	1.0	1.0	**0.9**	1.0	1.0	**0.9**	0.9	0.9	1.0	1.4	1.4	**0.9**
Ricci	10	23.5	17.7	**14.6**	14.6	14.6	**7.9**	6.3	12.2	**2.1**	8.9	13.1	**1.6**
Race	20	12.9	11.1	9.8	8.4	9.3	**7.1**	3.9	8.5	**1.5**	3.8	9.4	**2.1**
	30	8.5	7.5	**6.5**	4.9	5.7	**4.6**	2.0	6.0	**1.1**	2.8	6.5	**2.0**

Table B.2: **MAE for Δ TPR Estimates**, with different n_L. Mean absolute error between estimates and true Δ across 100 runs of labeled samples of different sizes n_L for different trained models (groups of columns) and 8 different dataset-group combinations (groups of rows). Estimation methods are Freq (Frequentist), BB (Beta-Binomial), and BC (Bayesian-Calibration). Freq and BB use only labeled samples, BC uses both labeled samples and unlabeled data. Trained models are Multilayer Perceptron, Logistic Regression, Random Forests, and Gaussian NaiveBayes.

		Multi-layer Perceptron			Logistic Regression			Random Forest			Gaussian Naive Bayes		
Group	n	Freq	BB	BC	Freq	BB	BC	Freq	BB	BC	Freq	BB	BC
Adult Race	40	—	16.3	**7.0**	—	16.7	**9.3**	—	17.9	**6.2**	—	23.6	**4.5**
	100	—	15.3	**6.4**	—	16.1	**8.4**	—	14.9	**5.6**	—	20.7	**3.9**
	200	—	12.5	**5.8**	—	14.7	**7.0**	—	14.3	**4.6**	—	14.6	**3.0**
Adult Gender	40	—	21.8	**5.5**	—	22.8	**5.8**	—	20.9	**8.4**	—	21.1	**11.7**
	100	—	17.8	**5.1**	—	18.9	**5.7**	—	18.6	**8.4**	—	17.7	**11.4**
	200	16.3	14.3	**4.3**	15.8	14.0	**4.6**	16.1	14.2	**7.3**	15.0	13.4	11.5
Bank Age	40	—	24.2	**6.1**	—	25.4	**3.8**	—	25.2	**2.7**	—	23.0	**3.6**
	100	25.9	20.0	**5.0**	25.7	20.4	**4.0**	20.9	16.6	**2.8**	24.9	19.6	**2.6**
	200	16.8	15.0	**4.8**	17.7	15.9	**4.2**	16.6	14.9	**3.1**	17.3	15.7	**2.3**
German age	40	—	15.0	**3.9**	—	18.4	**3.0**	—	11.3	**3.6**	—	16.7	**6.3**
	100	8.9	8.0	**3.5**	10.7	9.7	**3.1**	8.0	7.1	**3.5**	12.9	11.5	**3.3**
	200	4.7	4.7	**3.0**	5.6	5.4	**2.6**	5.1	5.1	**3.1**	6.8	6.5	**2.8**
German Gender	40	2.6	4.5	**2.3**	11.8	10.0	**2.4**	9.4	8.1	**2.4**	15.0	13.1	**3.8**
	100	1.4	2.1	2.0	6.5	6.3	**2.1**	5.9	5.8	**2.3**	7.7	7.4	**3.1**
	200	**0.7**	1.0	1.6	3.3	3.3	**2.1**	3.1	3.2	**2.1**	4.8	4.7	**2.2**
Compas-R Race	40	—	15.2	**3.4**	—	16.3	**3.4**	—	14.8	**3.2**	—	10.1	**2.2**
	100	—	11.5	**2.9**	—	11.5	**3.1**	—	10.6	**2.5**	—	6.7	**2.1**
	200	—	7.6	**2.5**	—	7.9	**2.6**	—	9.2	**2.1**	—	4.5	**2.0**
Compas-R Gender	40	—	19.3	**2.7**	—	21.8	**2.5**	—	19.3	**3.4**	—	14.0	**0.1**
	100	15.9	13.7	**2.4**	17.6	15.1	**2.1**	14.3	12.5	**3.2**	8.7	8.0	**0.2**
	200	10.0	9.5	**1.9**	10.0	9.4	**1.8**	11.3	10.7	**2.6**	5.6	5.5	**0.3**
Compas-VR Gender	40	—	23.0	**3.8**	—	27.0	**2.2**	—	20.9	**9.0**	—	21.1	**1.2**
	100	—	18.0	**3.2**	—	19.7	**2.1**	—	16.3	**8.1**	—	14.9	**1.2**
	200	14.9	12.2	**2.9**	8.9	10.7	**2.0**	14.6	10.5	7.2	12.5	10.0	**1.3**

Table B.3: **MAE for Δ FPR Estimates**, with different n_L. Same setup as Table B.2.

		Multi-layer Perceptron			Logistic Regression			Random Forest			Gaussian Naive Bayes		
Group	n	Freq	BB	BC	Freq	BB	BC	Freq	BB	BC	Freq	BB	BC
Adult Race	40	—	16.1	**1.5**	—	16.6	**1.5**	—	16.7	**1.8**	—	16.5	**2.9**
	100	—	9.6	**1.2**	—	10.1	**1.5**	—	10.5	**1.7**	—	10.7	**2.8**
	200	—	5.7	**1.2**	—	6.3	**1.6**	—	6.4	**1.8**	—	6.6	**3.4**
Adult Gender	40	7.1	6.9	**2.6**	7.2	7.1	**2.6**	8.3	8.1	**3.7**	10.3	9.8	**5.1**
	100	4.4	4.3	**2.3**	4.3	4.1	**2.2**	5.2	5.1	**3.5**	6.6	6.4	**4.7**
	200	3.2	3.3	**2.5**	3.2	3.2	**2.3**	3.7	3.7	**3.4**	4.7	4.6	**4.6**
Bank Age	40	2.4	2.5	**0.5**	3.6	3.7	**0.6**	4.1	4.2	**0.7**	8.5	7.9	**1.3**
	100	1.9	1.8	**0.5**	2.4	2.4	**0.6**	3.3	3.3	**0.7**	5.3	5.2	**1.3**
	200	1.5	1.5	**0.5**	2.1	2.0	**0.6**	2.1	2.1	**0.7**	3.6	3.6	**1.3**
German age	40	—	19.7	**9.8**	—	18.4	**8.7**	—	18.7	**9.1**	—	17.6	**13.2**
	100	16.6	14.3	**7.4**	13.6	11.9	**6.3**	13.7	11.7	**6.3**	14.9	12.5	**12.0**
	200	8.6	8.0	**5.6**	7.7	7.2	**5.7**	7.2	6.8	**5.4**	8.4	**7.7**	8.3
German Gender	40	15.6	13.2	**6.8**	27.3	21.5	**5.3**	23.1	18.6	**8.4**	20.3	16.1	**4.4**
	100	9.2	9.0	**5.7**	14.4	13.2	**5.9**	13.3	12.4	**7.4**	12.6	11.7	**5.6**
	200	4.9	4.9	**3.8**	7.3	7.0	**5.2**	6.8	6.6	**5.0**	5.9	5.7	**4.6**
Compas-R Race	40	—	15.1	**3.7**	—	13.2	**3.3**	—	14.5	**5.6**	—	10.8	**6.2**
	100	—	8.4	**2.7**	—	8.5	**2.4**	—	10.0	**4.6**	—	7.3	**4.4**
	200	—	6.8	**2.1**	—	5.9	**1.9**	—	6.7	**3.7**	—	4.9	**3.4**
Compas-R Gender	40	—	13.5	**3.4**	—	15.0	**2.4**	—	16.2	**4.9**	—	10.9	**4.2**
	100	7.7	7.4	**3.2**	8.5	8.3	**2.4**	11.5	11.0	**5.0**	7.4	6.9	**5.3**
	200	5.3	5.2	**2.7**	6.1	6.1	**2.0**	8.5	8.4	**4.4**	5.1	**5.0**	5.6
Compas-VR Gender	40	5.6	6.6	**0.7**	3.3	5.6	**0.4**	5.5	7.5	**1.9**	12.8	11.7	**4.4**
	100	4.0	4.3	**0.6**	2.4	2.8	**0.4**	3.9	4.4	**1.5**	6.3	6.3	**4.8**
	200	2.5	2.6	**0.5**	1.8	1.8	**0.4**	2.5	2.6	**1.2**	5.1	4.9	**4.1**