Bayesian Evaluation of Black-Box Classifiers

Disi Ji [*1] **Robert Logan** [*1] **Padhraic Smyth** [1] **Mark Steyvers** [2]

Abstract

There is an increasing need for accurate quantitative assessment of the performance of prediction models (such as deep neural networks), out-of-sample, e.g., in new environments after they have been trained. In this context we propose a Bayesian framework for assessing performance characteristics of black-box classifiers, performing inference on quantities such as accuracy and calibration bias. We demonstrate the approach using three deep neural networks applied to large real-world data sets, performing inference and active learning to assess class-specific performance.

1. Introduction

Deep learning models are now being applied to a variety of practical problems ranging from diagnosis of medical images (Kermany et al., 2018) to autonomous driving (Du et al., 2017). As a result, software systems with embedded machine learning components are likely to become relatively commonplace in the future. Many of these machine learning predictors will be, in effect, black-boxes from the perspective of the humans that are using them. For example, predictive models can be developed remotely by some commercial entity and the models may be hosted as a service in the cloud (Sanyal et al., 2018). For a variety of reasons (legal, cost, industry competition), the human user will often have no access to the detailed workings of the model, how the model was trained, or the data the model was trained on.

In this context it is increasingly important to develop techniques that can provide accurate and robust assessments of the quality of a model's predictions. However, it is well-known that the "self-confidence" estimates provided by machine learning predictors can often be quite unreliable and miscalibrated (Zadrozny & Elkan, 2002; Kull et al., 2017). In particular, complex models such as deep networks with high-dimensional inputs $\mathbf{x}$ (such as images or text) can be significantly overconfident in practice (Gal & Ghahramani, 2016; Guo et al., 2017; Lakshminarayanan et al., 2017; Kuleshov et al., 2018; Keren et al., 2018).

Independent assessment of accuracy and confidence for a predictor (rather than self-reported confidence or recalibration) will likely become increasingly important in the future. By independent we mean assessment that is carried out independently from training procedures, perhaps by individuals/organizations not involved in training the model, in a manner similar to the assessments of commercial products carried out by regulatory agencies. Reasons for independent assessment include legal requirements that may mandate independent assessment of models, the need for building trust on the part of a human consumer of model predictions, or situations where the predictor is being used in an environment $p(\mathbf{x}, y)$ which is different to the joint distribution characterizing the training environment.

In this paper we discuss results on the development of Bayesian approaches for independent assessment of the quality of black-box predictors, focusing on accuracy and calibration bias for classification models. We develop a number of concepts that underlie our proposed framework and illustrate the potential approach using image and text classification datasets. We view our paper as preliminary work intended to spur further discussion and interest among attendees of this workshop.

2. Related Work

While there is plenty of prior work in machine learning on calibration and recalibration methods for classification models, there is relatively little work on quantifying uncertainty in this context. Goutte & Gaussier (2005) proposed the use of Bayesian estimation of precision and recall in an information retrieval context—our work extends the Bayesian framework to a broader perspective on classifier accuracy and calibration. More recently Vaicenavicius et al. (2019) proposed a general framework for evaluating calibration for classification models, including the use of the bootstrap to obtain confidence intervals on the degree of miscalibration of a model. Their approach builds on earlier work that also proposed the bootstrap in a calibration context (Bröcker & Smith, 2007). The primary contributions of our paper are to

[*]Equal contribution [1]Department of Computer Science, University of California, Irvine, CA, USA [2]Department of Cognitive Science, University of California, Irvine, CA, USA. Correspondence to: Disi Ji <disij@uci.edu>.

Presented at the ICML 2019 Workshop on Uncertainty and Robustness in Deep Learning.

provide a Bayesian alternative to these frequentist perspectives and to develop generalizations of notions of classifier accuracy and calibration in this context.

3. Framework and Notation

Consider a classification problem with a feature space $\mathbf{x}$ and a class label $y \in \{1, \ldots, K\}$, e.g., classifying image pixels $\mathbf{x}$ into one of K classes. We have a prediction model M that has already been trained on training data and that makes predictions of y given a feature vector $\mathbf{x}$. We assume that the model can also produce numerical scores per class reflecting its confidence, typically in the form of a set of class probabilities $p_M(y = k|\mathbf{x}), k = 1, \ldots, K$.

We focus on the problem of assessing the performance of the model M on data drawn from some unknown distribution $p(\mathbf{x}, y)$. We are interested in the situation where the model M is a black-box, where 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. While our assessment framework below is Bayesian, the prediction scores could be coming from any black-box model (including either Bayesian or non-Bayesian predictive models).

Let $\hat{y}_M = \arg\max_k p_M(y = k|\mathbf{x})$ denote the model M's label prediction when the input is $\mathbf{x}$, assuming the prediction model implements a deterministic mapping[1] from the input space $\mathbf{x}$ to $\hat{y}_M$. In this manner the input space $\mathbf{x}$ is partitioned by the model into K decision regions $\mathrm{R}_1, \ldots, \mathrm{R}_K$, with K corresponding conditional densities $p(\mathbf{x}|\mathbf{x} \in \mathrm{R}_k)$, which are the (normalized) densities of $\mathbf{x}$ conditioned on class k being predicted by the model M.

We define $s_M(\mathbf{x}) = p_M(y = \hat{y}_M|\mathbf{x}) = \max_k p_M(y = k|\mathbf{x})$ as the **score** of the model as a function of $\mathbf{x}$, i.e., the class probability that the model produces for its prediction $\hat{y}_M$ given input $\mathbf{x}$. Under this notation, an expression such as $E_{p(\mathbf{x}|\mathrm{R}_k)}[s_M(\mathbf{x})]$ can be interpreted as the expected value of the model's score given that it predicts class k, averaging over the $\mathbf{x}$ values in decision region R_k, i.e., $E_{p(\mathbf{x}|\mathrm{R}_k)}[s_M(\mathbf{x})] = \int_{\mathrm{R}_k} s_M(\mathbf{x}) p(\mathbf{x}|\mathbf{x} \in \mathrm{R}_k) d\mathbf{x}$.

4. Local and Classwise Accuracy and Calibration Error

The **true local accuracy** of a model can be defined (in theory at least) at any input point $\mathbf{x}$ as $A_M(\mathbf{x}) = p(y = \hat{y}_M|\mathbf{x})$, i.e., the true probability that the prediction $\hat{y}_M$ matches an observed y conditioned on $\mathbf{x}$. This will be different (typically, in practice) to $s_M(\mathbf{x}) = p_M(y = \hat{y}_M|\mathbf{x})$ which is the model's own assessment of its accuracy at $\mathbf{x}$.

[1] Assume for convenience that ties in the model's class scores $p_M(y = k|\mathbf{x})$ are resolved deterministically.

Thus, for example, the expected accuracy of the model over the whole $\mathbf{x}$ space is $E_{p(\mathbf{x})}[p(y = \hat{y}_M|\mathbf{x})] = \int p(y = \hat{y}_M|\mathbf{x}) p(\mathbf{x}) d\mathbf{x}$. In machine learning this overall accuracy is typically estimated empirically on a test data set by drawing S samples randomly from $p(\mathbf{x}, y)$ and computing $\frac{1}{S}\sum_{i=1}^{S} I(y^{(i)}, \hat{y}_M^{(i)})$ where $\hat{y}_M^{(i)}$ is the class predicted by M given $\mathbf{x}^{(i)}$.

We can also define the **local calibration error** $CE_M(\mathbf{x})$ of a prediction model as a function of $\mathbf{x}$ (see also Vaicenavicius et al. (2019)). $CE_M(\mathbf{x})$ is the difference between the true local accuracy $A_M(\mathbf{x})$ of a model and the model's own estimate of its accuracy at $\mathbf{x}$, $s_M(\mathbf{x})$, i.e., $CE_M(\mathbf{x}) = \Delta\big(s_M(\mathbf{x}), A_M(\mathbf{x})\big) = \Delta\big(\ p_M(y = \hat{y}_M|\mathbf{x}),\ p(y = \hat{y}_M|\mathbf{x})\ \big)$, where $\Delta(a, b)$ is some error measure such as absolute error.

We can marginalize over $\mathbf{x}$ to compute the expected accuracy or calibration error per **predicted class** k in region R_k, expressed as conditional expectations $A_{M,k} = E_{p(\mathbf{x}|\mathrm{R}_k)}[A_M(\mathbf{x})]$ and $CE_{M,k} = E_{p(\mathbf{x}|\mathrm{R}_k)}[CE_M(\mathbf{x})]$, respectively, conditioned on class k being the predicted class[2]. For a visual illustration of these ideas for a toy one-dimensional problem see Appendix A in the Supplement.

Assessing performance measures per predicted class, such as calibration error or accuracy, can be useful in practice to a user of a model M, for example in situations when the model's predictions and probabilities are being used in a downstream application to make critical decisions with different costs (such as autonomous driving or medical diagnosis). Also of interest is the situation where the test environment $p(\mathbf{x}, y)$ may differ significantly to the environment the model was trained on and where robust performance measures per class can provide a decision-maker with an important summary of how a model will perform on different predicted classes.

As an aside we note that calibration error is often expressed in the literature as a function of the model's score, in the form $CE(s_M)$, providing the basis for reliability diagrams with the score on the x-axis and the model's accuracy on the y-axis (e.g., Guo et al. (2017)). The Bayesian framework below can be extended to Bayesian estimation of such functions, either bin-based or continuous—here we focus just on per-class performance measures.

5. Bayesian Estimation

To estimate the per-class accuracy $A_k = E_{p(\mathbf{x}|\mathrm{R}_k)}[A_M(\mathbf{x})]$ from data, we can empirically approximate the integral

[2] To get the accuracy or calibration error per **true class** k the expectations can be defined with respect to $p(\mathbf{x}|y = k)$ rather than $p(\mathbf{x}|\hat{y}_M = k) = p(\mathbf{x}|\mathrm{R}_k)$.

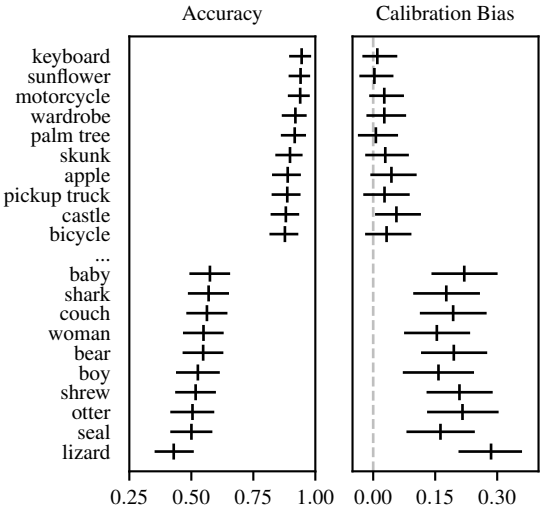

Figure 1. Mean posterior estimate and 95% credible intervals for the classwise accuracy and calibration bias of Resnet-110 predictions on CIFAR-100. Overlapping bars indicate uncertainty about the difference between classes. Generally, differences are significant between classes in the top/bottom cohorts, but insignificant within the cohorts.

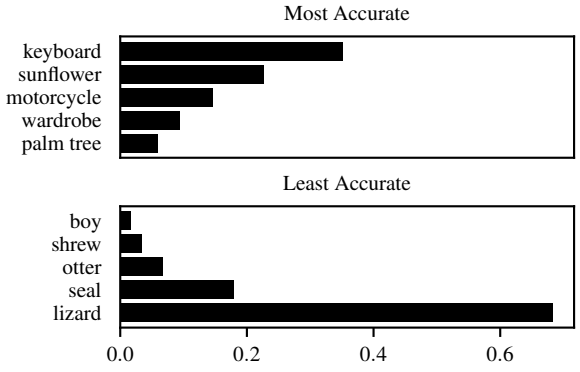

Figure 2. Posterior probabilities of the most and least accurate predictions on CIFAR-100. The most accurate predicted class is somewhat uncertain, while the least accurate predicted class is very likely *lizard*.

by sampling $\mathbf{x}, y$ pairs from the conditional distribution $p(\mathbf{x}, y|\mathbf{x} \in \mathrm{R}_k)$. One can treat A_k as an unknown Bernoulli parameter, with draws $(\mathbf{x}^{(i)}, y^{(i)})$, conditioned on $\mathbf{x} \in \mathrm{R}_k$, leading to binary outcomes $I(y^{(i)}, \hat{y}_M^{(i)}) \in \{0, 1\}$, with a frequency-based (maximum likelihood) estimate: $\hat{A}_{M,k} = \frac{1}{S}\sum_{i=1}^{S} A_M(\mathbf{x}^{(i)}) = \frac{1}{S}\sum_{i=1}^{S} I(y^{(i)}, \hat{y}_M^{(i)})$.

It is natural to consider Bayesian estimation in this context, especially in situations where there is not a large amount of labeled data available for evaluation and/or where K is large, allowing for uncertainty in our inferences about quantities such as $A_{M,k}$. In particular, we can put a Beta prior $Beta(\alpha, \beta)$ on $A_{M,k}$, model the draws $I(y^{(i)}, \hat{y}_M^{(i)})$ with a binomial likelihood, and produce Beta posteriors for each $A_{M,k}, k = 1, \ldots, K$.

For calibration, we focus for simplicity just on the **calibration bias** $\hat{CB}_{M,k}$ per predicted class, where the error Δ is defined as $s_M(\mathbf{x}) - A_M(\mathbf{x})$. This leads to $CB_{M,k} = E_{p(\mathbf{x}|\mathrm{R}_k)}[s_M(\mathbf{x})] - A_{M,k}$. We assume that the uncertainty in estimating $CB_{M,k}$ will be dominated by the second term $A_{M,k}$ since this term requires labeled examples, whereas the first term (the average score for the model when class k is predicted) can be estimated from unlabeled examples alone and uncertainty about this term can be driven to zero with enough unlabeled data. In our experiments below we use a Beta/binomial prior/likelihood for $A_{M,k}$ and ignore uncertainty in $s_M(\mathbf{x})$.

6. Illustrative Results

6.1. Experimental Setup

In this section we apply the framework presented in the previous section to evaluate the classwise accuracy and calibration bias of a deep residual neural network (ResNet) (He et al., 2016) on CIFAR-100 (Krizhevsky & Hinton, 2009), a standard benchmark for deep model assessment (Guo et al., 2017; Hendrycks & Dietterich, 2019). The ResNet model we use has 110 layers and 1.7 million parameters.

We also carry out additional experiments on the SVHN (Netzer et al., 2011) dataset, and a medical dialog classification dataset (Tai-Seale & et al., 2016). For the sake of brevity, the details and results of these experiments are included in the Supplementary Materials.

6.2. Classwise Accuracy and Calibration Bias

For illustration, we apply the beta-binomial model discussed in Section 5 to measure classwise accuracies and calibration bias on CIFAR-100 using the entire test dataset of 10k examples. Mean posterior estimates (MPE) and 95% credible intervals for the ten most and least accurate classes (according to MPE) are plotted in Figure 1. We observe (by examining overlapping credible intervals) that it is uncertain which class is truly most accurate, whereas it is highly likely that model performance is much better on the top ten classes than it is on the bottom ten. We draw similar conclusions for calibration bias, which is not surprising given that classwise accuracy and calibration bias are highly correlated.

Interestingly, we also observe that there is possibly no calibration bias on the most accurate classes (i.e., the posterior probability density for their calibration bias is centered near 0), while there is clear evidence that the least accurate

Algorithm 1 Thompson Sampling Strategy

Input: prior hyperparameters α, β
initialize $n_{k,0} = n_{k,1} = 0$ for $k = 1$ **to** K
repeat
  for $k = 1$ **to** K **do**
    $\hat{A}_k \sim \text{Beta}(\alpha + n_{k,0}, \beta + n_{k,1})$
  end for
  $k^* = \arg\min_k \hat{A}_{1:K}$
  select data point $(x, \hat{y} = k^*)$
  query oracle for true label y
  if $y = k^*$ **then**
    $n_{k,0} \leftarrow n_{k,0} + 1$
  else
    $n_{k,1} \leftarrow n_{k,1} + 1$
  end if
until all data labeled

classes are biased. This implies that recalibration may only be necessary for certain predicted classes, an observation which is also supported by recent results by Vaicenavicius et al. (2019).

An additional benefit of the Bayesian framework is that we can draw samples from the posterior to infer other statistics of interest from the posterior distribution of calibration measures. For instance, we can estimate the probability that *lizard* is the least accurate predicted class by sampling $\hat{A}_k$'s (from their respective posterior Beta densities) for each of the classes and then measuring whether $\hat{A}_{lizard}$ 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 this model. The posterior probabilities for other classes are provided in Figure 2, along with results for estimating which class has the highest classwise accuracy.

6.3. Active Learning to Find Extreme Classes

Above we assumed we have access to ground truth labels for all of the test data for model assessment. This is unrealistic in many real world scenarios as obtaining ground truth can be costly (one of the main reasons black-box models are used to begin with). Accordingly, it is desirable for model assessment to be performed online in a data efficient manner.

The Bayesian framework we presented is amenable to designing active learning strategies for data selection that achieve this goal. As an example, suppose we want to determine the class with the worst classwise accuracy. To do this we can utilize the beta-binomial model from the previous section in conjunction with Thompson sampling (Thompson, 1933), a method typically employed to solve the multi-armed bandit problem. A detailed description of the strategy is given in Algorithm 1.

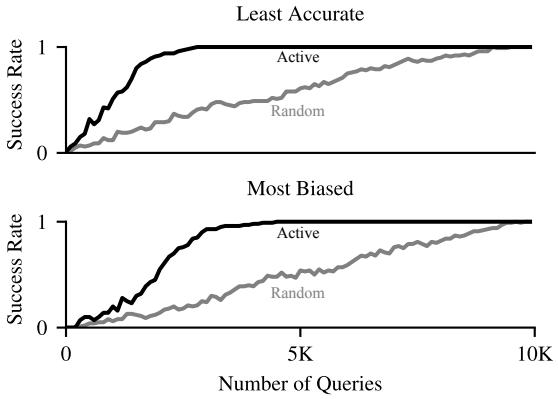

Figure 3. Success rates of active learning vs. a random selection strategy for determining the least accurate and most biased predicted classes on CIFAR-100, collected over 100 runs. The active learning strategy is able to identify the least accurate class (*lizard*) much quicker than using random selection.

Aggregate results for 100 independent trials of this strategy on CIFAR-100 are given in Figure 3. The x-axis measures the number of queries made to the oracle to obtain ground truth labels. The y-axis measures the fraction of runs where the correct class was identified as the having the lowest expected accuracy according to MPE. As a baseline we also include results for when data points are selected at random.

Our results demonstrate that the active learning approach is much more effective at identifying the least likely class. In all of the trials, the correct class is identified within 3000 queries. In contrast, the random selection strategy sometimes labels almost all of the data before identifying the correct class.

7. Conclusions

This abstract describes a Bayesian framework for assessing performance metrics of black-box classifiers, focusing in particular on classification accuracy and calibration bias. We illustrated a number of different ways that the framework can be used to understand performance aspects of three 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.

Acknowledgements

This work was supported in part by PCORI under award ME-1602-34167, by the NASA MIRO program under award NNX15AQ06A, by NSF under awards NSF-1839336 and NSF-1633631, by NIH under award U01TR001801-01, and by funding support from Adobe, eBay and Cylance.

References

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

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 EMNLP Conference*, pp. 1724–1734. ACM, 2014.

Du, X., El-Khamy, M., Lee, J., and Davis, L. Fused DNN: A deep neural network fusion approach to fast and robust pedestrian detection. In *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*, pp. 953–961. IEEE, 2017.

Gal, Y. and Ghahramani, Z. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, pp. 1050–1059, 2016.

Goutte, C. and Gaussier, E. A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In *European Conference on Information Retrieval*, pp. 345–359. Springer, 2005.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1321–1330. JMLR.org, 2017.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

Keren, G., Cummins, N., and Schuller, B. Calibrated prediction intervals for neural network regressors. *arXiv preprint arXiv:1803.09546*, 2018.

Kermany, D. S., Goldbaum, M., Cai, W., Valentim, C. C., Liang, H., Baxter, S. L., McKeown, A., Yang, G., Wu, X., Yan, F., et al. Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell*, 172(5): 1122–1131, 2018.

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

Kuleshov, V., Fenner, N., and Ermon, S. Accurate uncertainties for deep learning using calibrated regression. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pp. 2801–2809, 2018.

Kull, M., Filho, T. S., and Flach, P. Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. In Singh, A. and Zhu, J. (eds.), *Proceedings of the 20th International Conference on AI and Statistics*, volume 54 of *PMLR*, pp. 623–631, 2017.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, pp. 6402–6413, 2017.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

Sanyal, A., Kusner, M. J., Gascón, A., and Kanade, V. TAPAS: Tricks to accelerate (encrypted) prediction as a service. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 80, pp. 4490–4499. PMLR, 2018.

Tai-Seale, M. and et al. Periodic health examinations and missed opportunities among patients likely needing mental health care. *The American Journal of Managed Care*, 22(10):e350e357, Oct 2016.

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

Vaicenavicius, J., Widmann, D., Andersson, C., Lindsten, F., Roll, J., and Schön, T. Evaluating model calibration in classification. In *AI Statistics 2019/Proceedings of Machine Learning Research*, volume 89, pp. 3459–3467. PMLR, 2019.

Zadrozny, B. and Elkan, C. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth ACM SIGKDD Conference*, pp. 694–699. ACM, 2002.

	Input	Train / Test	Classes	Balanced
CIFAR-100	Image	50K / 10K	100	✓
SVHN	Image	600K / 100K	10	✓
Dialog	Text	110K / 12K	27	✗

Table 1. Dataset statistics.

Supplemental Materials

Appendix 1: Notation for a One-Dimensional Problem

Figure 4 provides an illustration of the notation used in the paper for a simple two-class problem, $y \in \{1, 2\}$, with a one-dimensional input feature x. The data-generating mechanism for class 1, $p(x|y = 1)$ is assumed to be Normal density $N(\mu = 5; \sigma = 1)$, and for class 2, $p(x|y = 2)$, a Gamma density Gamma$(a = 2, b = 2)$, with both classes assumed to be equally likely, i.e., $p(y_1) = p(y_2) = 0.5$. The optimal decision region for classifying x into class 1 is shaded in gray on all 4 plots.

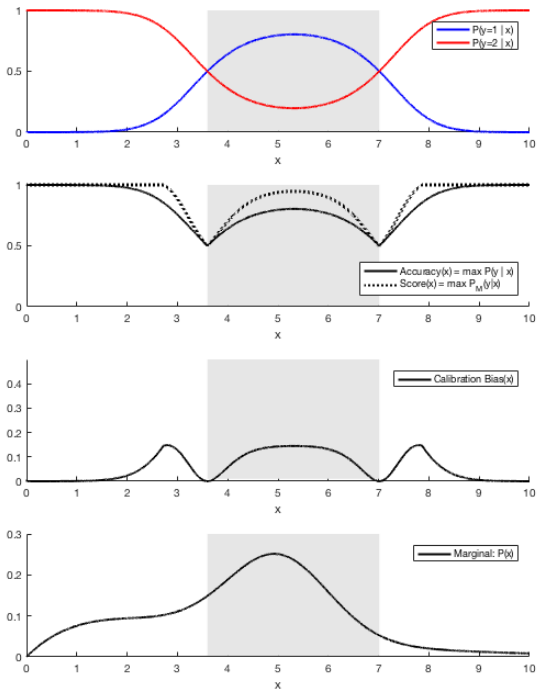

Figure 4. Illustration of notation for a simple classification problem with two classes and one input feature x. The optimal decision region R_1 for class 1 is shown shaded in gray for all 4 panels. Top panel: posterior class probabilities as a function of x. Second panel: Accuracy $A_M(x)$ and the model's score $s_M(x) = \max_k\{P_M(y = k|x)\}$. Third panel: Calibration bias $s_M(x) - A_M(x)$. Bottom panel: marginal density for x.

The top panel shows the true class probabilities for this problem. The second panel shows the maximum of the two true class probabilities as a function of x (solid black line): this is the true accuracy $A_M(x)$. The dotted line in the second panel shows scores $s_M(x)$ from a hypothetical model M, where $s_M(x) = \max_k\{P_M(y = k|x)\}$. For simplicity it is assumed that this model has learned the optimal decision boundaries exactly, but its predictions are over-confident (miscalibrated), i.e., it generates probabilities that tend to be higher than the true accuracy. Note that in practice a user of this model would only have direct access to the information in the dotted line, the model's scores $s_M(x)$, and the true accuracy of the model $A_M(x)$ would be unknown. The third panel shows the Calibration Bias, $s_M(x) - A_M(x)$, as a function of x, where for this simulated problem and hypothesized model the calibration bias ranges between 0 and 1.

From this figure, we can see that to compute the expected accuracy $A_{M,k}$ or calibration bias $CB_{M,k}$, for say class $k = 1$, we need to compute the expected value of $A_M(x)$ or $CB_M(x)$, with respect to $p(x)$, in the decision region R_1 corresponding to $k = 1$, i.e., over the shaded region in the plots, where the model always makes the prediction $k = 1$. As discussed earlier in the paper, this can be carried out empirically by sampling pairs of x's and y's from the region R_1 and comparing the model's predictions for each x with the corresponding sampled y.

This example is only intended to provide some intuition to the reader for a low-dimensional simple situation. In practical problems of interest, e.g., in deep learning models, the input space will often be high-dimensional, and direct estimation or visualization of quantities such as $A_M(\mathbf{x})$ and $CE_M(\mathbf{x})$ as a function of $\mathbf{x}$ is impractical.

Appendix 2: Additional Results

Additional Results on SVHN Data

There are 10 classes in the street view house number (SVHN) data, one for each digit. Mean posterior estimates (MPE) and 95% credible intervals for 10 classes are plotted in Figure 5. Resnet-100 predictions are relatively accurate and well-calibrated in terms of calibration bias on all of 10 classes. The posterior probabilities of the most and least accurate predictions on SVHN are plotted in Figure 6. Although there is obvious overlap of credible intervals between class *nine* and *three* in Figure 5, the model is quite certain that class *two* and *nine* are the most and the least accurate, by ranking samples from the posterior of classwise accuracy. Aggregate results for 100 independent trials of active learning strategy to detect the least accurate class and the most biased class on SVHN are given in Figure 7. Success rate of identifying the correct class in both of the scenarios starts to converge within 2K queries, while ran-

dom strategy requires around twice as much the number of queries. The variances of classwise accuracy and calibration bias in SVHN are relatively low compared to CIFAR-100 data, which makes selecting the least accurate or the most biased class more difficult. This is likely the reason why less significant improvement with active learning is observed, as shown in Figure 7.

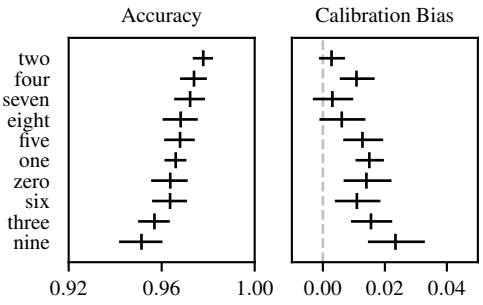

Figure 5. Mean posterior estimate and 95% credible intervals for the classwise accuracy and calibration bias of Resnet-110 predictions on SVHN. Classwise accuracies do not significantly differ. Although there appears to be positive bias for all classes, we cannot reject the hypothesis that there is no bias for the classes *two* and *seven* at the 95% level.

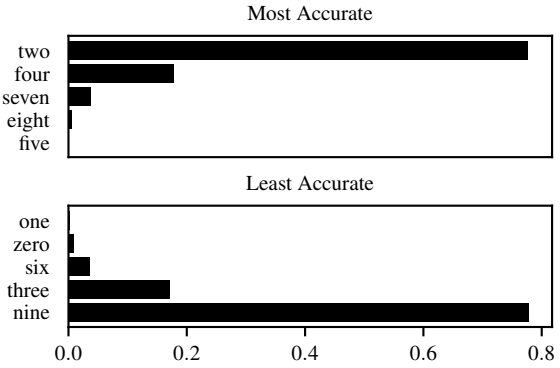

Figure 6. Posterior probabilities of the most and least accurate predictions on SVHN. It is highly probably that most accurate predicted class is *two*, and that the least accurate predicted class is *nine*.

Additional Results on Dialog Data

The medical dialog dataset contains transcripts of conversations between doctors and patients. Each utterance has been labeled into 1 of 27 classes, each class corresponding to a topic being discussed. The class distributions are highly imbalanced (unlike CIFAR and SVHN). Utterance label predictions are made using a heirarchical RNN with GRU (Cho et al., 2014) units. Mean posterior estimates (MPE) and 95% credible intervals for 27 classes are plotted in Figure 8.

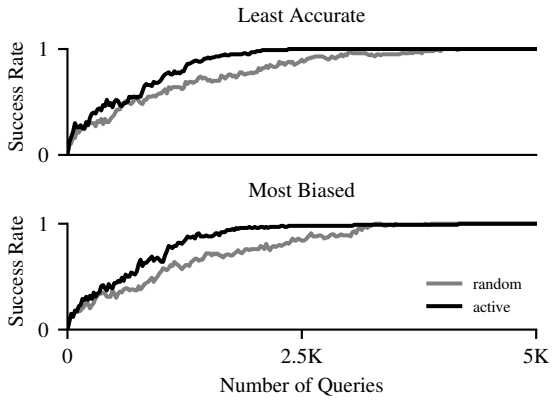

Figure 7. Success rates of active learning vs. a random selection strategy for determining the least accurate and most biased predicted classes on the medical dialog data, collected over 100 runs.

More uncertainty is observed in rare classes, e.g. *OtherAddictions* and *Sex*. The posterior probabilities of the most and least accurate predictions are plotted in Figure 9. It is highly likely that the most accurate class is *PreventiveCare*, while we are less certain about *Diet* as the least accurate class because the 95% credible intervals of *DizzyDentHearVision*, *MDLife*, *GeneralAnxieties* and *Depression* are all in the 95% credible interval span of *Diet*.

Aggregate results for 100 independent trials of active learning strategy are given in Figure 10. The active learning strategy detects the correct class within less than 500 queries, while random selecting requires around 5K queries in the scenario of detecting the least accurate class. The high efficiency of active learning on this problem can likely be explained by the fact that the label distribution is highly imbalanced, so that the random selection policy has less opportunity to explore low accuracy classes that are also relatively rarely predicted.

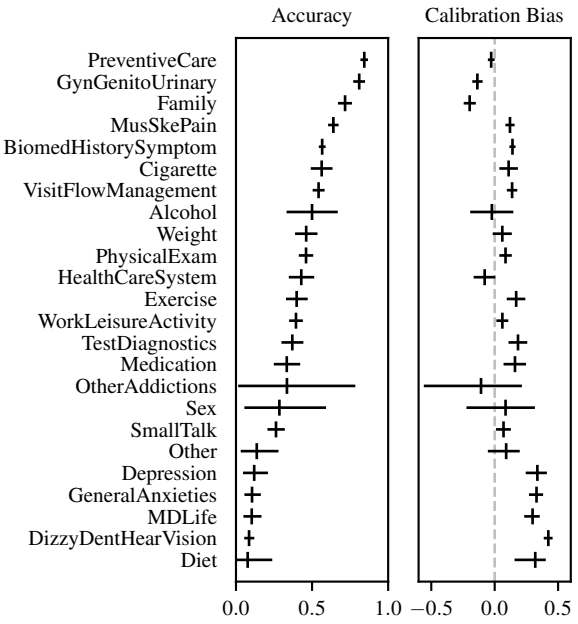

Figure 8. Mean posterior estimate and 95% credible intervals for the classwise accuracy and calibration bias of HGRU predictions on the medical dialogue classification dataset. The imbalance in label distribution results in much more uncertainty for rare labels than common labels.

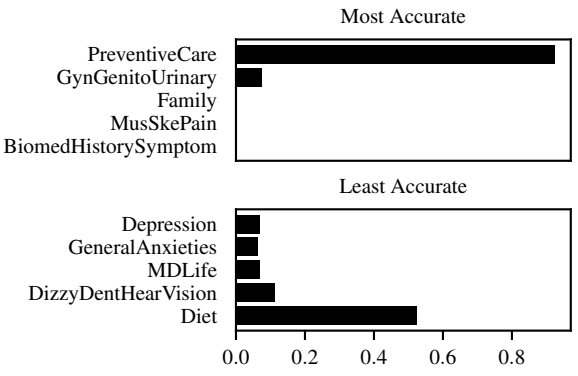

Figure 9. Posterior probabilities of the most and least accurate predictions on SVHN. It is highly probable that most accurate predicted class is *PreventiveCare*, while the least accurate predicted class is *Diet*.

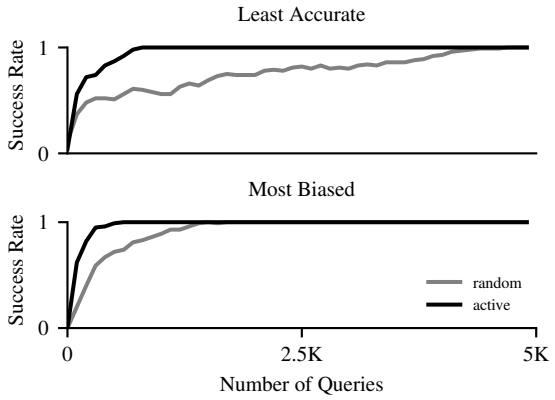

Figure 10. Success rates of active learning vs. a random selection strategy for determining the least accurate and most biased predicted classes on the medical dialogue data, collected over 100 runs.