# Bayesian Evaluation of Black-Box Classifiers

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

Departments of [1]Computer Science and [2]Cognitive Science, University of California, Irvine

## Introduction

**Goal:** Evaluate blackbox classifiers online in new environments after they have been trained.

Predicted as Tiger with p(y|x) = 0.99

Predicted as Television with p(y|x) = 0.99

- Neural network models are being widely deployed as blackbox classifiers.
- It has been recognized that deep neural networks can be miscalibrated.[1]
- We propose a Bayesian framework for assessing performance characteristics of black-box classifiers, which enables third parties to infer on quantities such as accuracy and calibration bias, as well as measure uncertainty in their estimates.
- We use our framework to design efficient labeling methods which quickly identify weaknesses of blackbox classifiers.

## Approach

- For a blackbox classifier $M$ and input $x$, denote normalized output as $p_M(k|x), k = 1,2,\cdots,K$
- Predicted label on $x$ made by $M$ is $\hat{y}_M = \arg\max_k p_M(k|x)$

- Local score(**confidence**): for a given $x$, score the model assigned to the predicted label
  $$S_M(x) = p_M(\hat{y}_M|x)$$ Model's own assessment of accuracy at x

- Local **accuracy:** for a given $x$, **probability** that predicted label is the same as true label $y$
  True accuracy, need to be evaluated with true label y
  $$A_M(x) = p(y = \hat{y}_M|x)$$

- Empirical estimation of accuracy needs **labeled data.** When getting true label is expensive, estimating accuracy can be costy.

- **Accuracy over a region:** expectation of local accuracy over region $R$:
  $$A_M(R) = E_{p(x|x\in R)}A_M(x) = \int_R p(y = \hat{y}_M|x)p(x|R)dx$$

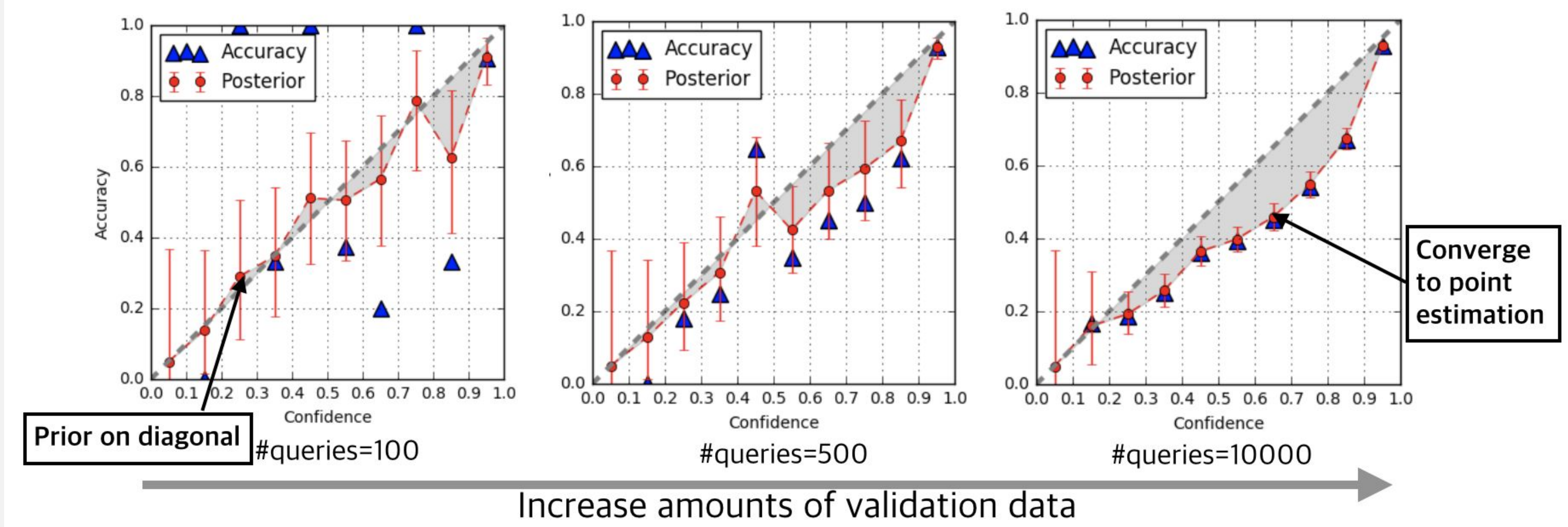
- Accuracy $A_M(R) \in [0,1]$ is an unknown Bernoulli parameter.
- **Generative process:**
  - Accuracy over region $R$: $A_M(R) \sim Beta(a,b)$
  - For $i = 1,2,\cdots,N$:
    - $x_i \sim p_R(x)$ and model makes prediction on it $\hat{y}_i$
    - Query true label: $1(y_i = \hat{y}_i) \sim Bern(A_M(R))$
- Posterior of accuracy gets updated in closed form as more labels get revealed.

- By **partitioning** the data space $D$ into disjoint subsets
  - finer grained estimation of model characteristics can be conducted on the each subset to have more comprehensive assessment of model performance in environment $p(x,y)$:
    $$A_M(R_k) = E_{p(x|x\in R_k)}A_M(x) = \int_{R_k} p(y = \hat{y}_M|x)p(x|k)dx$$
  - Denote each $A_M(R_k) = \theta_k$, model accuracy over each region independently with
    $$\theta_k \sim Beta(a_k, b_k)$$
  - Examples of different partitions:
    - For modeling reliability diagram: $R_k = \{x|S_M(x) \in [\frac{k}{10}, \frac{k+1}{10})\}$
    - For modeling classwise accuracy: $R_k = \{x|\hat{y}_M = k\}$

- Local **calibration error**: for a given $x$, the difference between local accuracy and confidence
  $$CE_M(x) = \Delta\left(S_M(x) - A_M(x)\right)$$

- A model at $x$ is
  - **calibrated** if $S_M(x) = A_M(x)$
  - **overconfident** if $S_M(x) > A_M(x)$
- **Calibration bias:** $\Delta(a,b) = a - b$

## Bayesian Reliability Diagram




## Classwise Accuracy and Calibration Bias

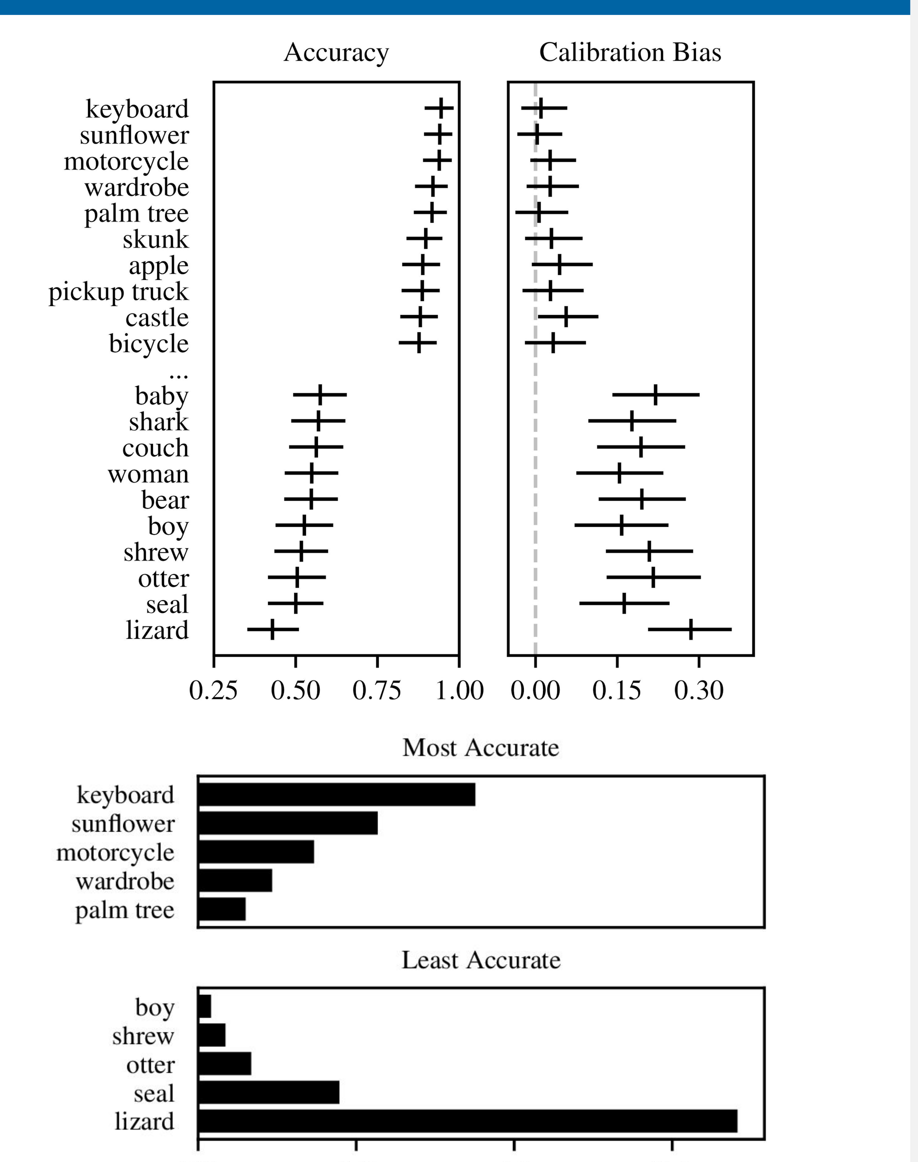
**Model**: ResNet-110[2]
**Dataset**: CIFAR-100[3]

**Experiments**

- Measure posterior accuracy and calibration bias of model predictions on test set, where $R_k$ : class $k$ predicted by model
- Draw samples from posterior to form Monte-Carlo estimates of most and least accurate classes

**Observations**

- Calibration bias mostly affects inaccurate classes.



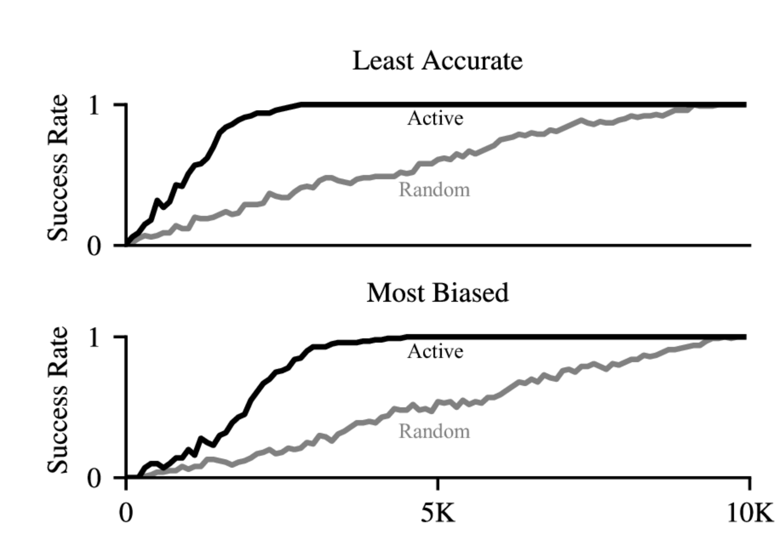

## Active Learning to Find Extreme Classes

**Idea**

Use Thompson sampling-based approach to efficiently determine most accurate/biased classes.

**Algorithm**

- Sample accuracies/biases from posterior.
- Determine least accurate/most biased class according to sample.
- Obtain label for a data point with least accurate/most biased class.
- Update posteriors.




**Success rates** of Thompson sampling vs. random selection strategy as a function of the number of queries submitted to the oracle. Averaged over 100 runs.

## Conclusion

- Bayesian methods show promise for blackbox model assessment, allowing for uncertainty quantification in estimates of calibration and accuracy
- We also show how our framework can be used to quickly identify potential issues in a deployed model (e.g., least calibrated class predictions)

## References

1. Guo, Chuan, et al. "On calibration of modern neural networks." *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017.
2. 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 ecognition*, pp. 770–778, 2016.
3. Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

For any questions, email: **disij@uci.edu**