Deep Bayesian Active Learning via Equivalence Class Annealing

Renyu Zhang, Aly A. Khan, Robert L. Grossman, Yuxin Chen

THE UNIVERSITY OF CHICAGO

1 Summary

- **BALanCe**: an efficient deep Bayesian active learning framework motivated by a decision-theoretic selection criterion
- **Batch-BALanCe:** batch-mode version

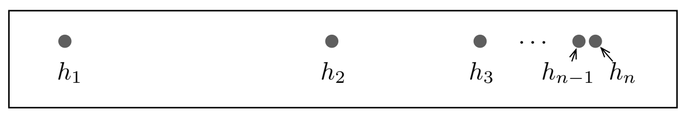
2 Motivation

Numerical example

The Most Information Selection criterion uses mutual information between the predicted label and model parameters as the acquisition function:

$$\Delta_{\text{BALD}}(x \mid \mathscr{D}_{\text{train}}) \triangleq \mathbb{I}(y; \omega \mid x, \mathscr{D}_{\text{train}})$$

MIS/BALD can be ineffective

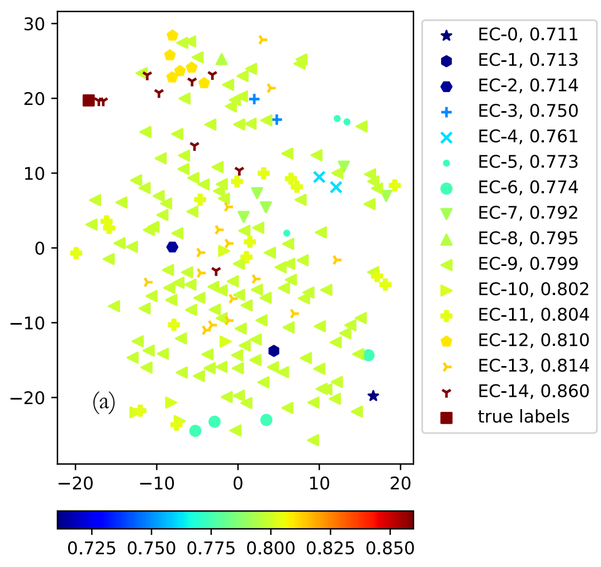


The hypothesis class $\mathscr{H} = \{h_1, \ldots, h_n\}$ is structured such that

$$d_{\text{H}}(h_i, h_j) = \begin{cases} 2^{1-i} - 2^{1-j} & \text{if } i < j, \\ 2^{1-j} - 2^{1-i} & \text{o.w.} \end{cases}$$

- MIS/BALD on average require $O(\log n)$.
- A "smarter" policy could query examples to sequentially check the consistency of $h_1, h_2, \ldots, h_n$ util all remaining hypotheses are within distance σ. It requires $\log(1/\sigma)$.

Empirical validation



Samples from posterior BNN via MC dropout; embedding is generated by applying t-SNE on the disagreement between hypotheses;

Colorbar indicates the (approximate) test accuracy of the sampled neural networks on the MNIST dataset.

3 Notations

- Labeled dataset $\mathscr{D}_{\text{train}}$
- Unlabeled dataset $\mathscr{D}_{\text{pool}}$ drawn i.i.d. from some underlying data distribution.
- A set of hypotheses $\mathscr{H} = \{h_1, \ldots, h_n\}$;
- In this work, we consider BNN hypothesis class with parameters $\omega \sim \mathrm{p}(\omega \mid \mathscr{D}_{\text{train}})$.

4 Problem statement

Assume

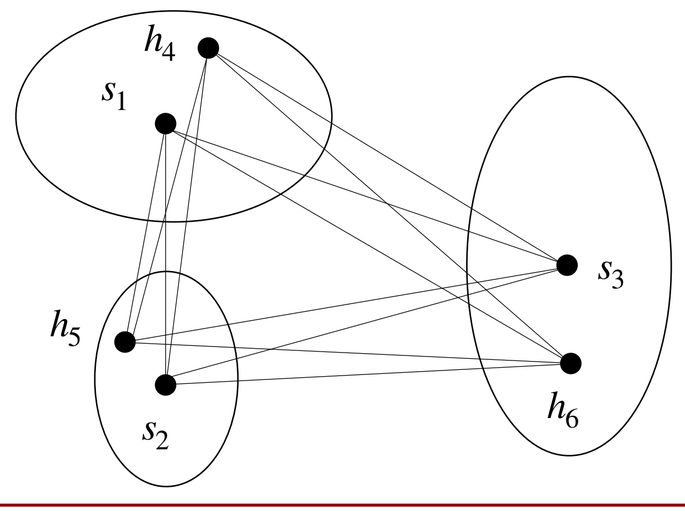
labeling each query x incurs a unit cost.

Want

an active learning algorithm that allows us to find a hypotheses with a target error rate $\sigma \in [0,1]$ while minimizing the total query cost.

5 Equivalence class

Definition 1.
Let $(\mathscr{H}, d)$ be a metric space where $\mathscr{H}$ is a hypothesis class and d is a metric. For a given set $\mathscr{V} \subseteq \mathscr{H}$ and centers $\mathcal{S} = \{s_1, \ldots, s_k\} \subseteq \mathscr{V}$ of size k, let $r^{\mathcal{S}} : \mathscr{V} \to [k]$ be a partition function over $\mathscr{V}$, and $\mathscr{D}_i := \{h \in \mathscr{V} \mid r^{\mathcal{S}}(h) = i\}$, such that $\forall i, j \in [k], r^{\mathcal{S}}(s_i) = i$ and $\forall h \in \mathscr{D}_i, d(h, s_i) \leq d(h, s_j)$. Each $\mathscr{D}_i \subseteq \mathscr{V}$ is called an equivalence class induced by $s_i \in \mathcal{S}$.



6 Our approach

$$\Delta_{\text{BALanCe}}(x \mid \mathscr{D}_{\text{train}}) \triangleq \mathbb{E}_y \mathbb{E}_{\omega, \omega' \sim p(\omega \mid \mathscr{D}_{\text{train}})} \left[1_{d_{\text{H}}(\omega, \omega') > \tau} \cdot \text{WEIGHT} - \text{DISCOUNTED}(y \mid x, \mathscr{D}_{\text{train}}) \right]$$

Threshold adaptively annealed based on validation prediction

Fully sequential setting:

$$\Delta_{\text{BALanCe}}(x \mid \mathscr{D}_{\text{train}}) \approx \sum_{\hat{y}} \sum_{k=1}^{K} \frac{p(\hat{y} \mid \hat{\omega}_k) + p(\hat{y} \mid \hat{\omega}'_k)}{2K} \sum_{k=1}^{K} \frac{1_{d_{\text{H}}(\hat{\omega}_k, \hat{\omega}'_k) > \tau} \cdot \left(1 - \lambda_{\hat{\omega}_k, \hat{y}} \lambda_{\hat{\omega}'_k, \hat{y}}\right)}{K}$$

Batch mode setting:

$$\Delta_{\text{Batch-BALanCe}}(x_{1:b} \mid \mathscr{D}_{\text{train}}) \approx \sum_{\hat{y}_{1:b}} \left(\sum_{k=1}^{K} \frac{p(\hat{y}_{1:b} \mid \hat{\omega}_k) + p(\hat{y}_{1:b} \mid \hat{\omega}'_k)}{2K} \right) \cdot \left[\sum_{k=1}^{K} \frac{1_{d_{\text{H}}(\hat{\omega}_k, \hat{\omega}'_k) > \tau} \cdot \left(1 - \lambda_{\hat{\omega}_k, \hat{y}_{1:b}} \lambda_{\hat{\omega}'_k, \hat{y}_{1:b}}\right)}{K} \right]$$

The BALanCe Algorithm

Anneal τ at each step

Algorithm 1: BALanCe

input: $\mathscr{D}_{\text{pool}}$, $\bar{\mathscr{D}}_{\text{pool}}$, a trained BNN and threshold τ
draw K pairs of MC dropout samples $\{\hat{\omega}_k, \hat{\omega}'_k\}_{k=1}^{K}$ from the trained BNN
for $k \in [K]$ **do**
  Calculate indicator function $\mathbb{I}_{d_{\text{H}}(\hat{\omega}_k, \hat{\omega}'_k) > \tau}$ with prediction on $\mathscr{D}_{\text{pool}}$
for each $x \in \mathscr{D}_{\text{pool}}$ **do**
  $\Delta(x) = 0$
  for each $\hat{y}$ **do**
    $\mathrm{p}(\hat{y}) = \frac{1}{2K} \sum_{k=1}^{K} \left(\mathrm{p}\left(\hat{y} \mid \hat{\omega}_k\right) + \mathrm{p}\left(\hat{y} \mid \hat{\omega}'_k\right) \right)$
    $\delta(x, \hat{y}) = 0$
    for $k \in [K]$ **do**
      if $\mathbb{I}_{d_{\text{H}}(\hat{\omega}_k, \hat{\omega}'_k) > \tau} = 1$ **then**
        $\delta(x, \hat{y}) \leftarrow \delta(x, \hat{y}) + \frac{1}{K} \left(1 - \lambda_{\hat{\omega}_k, \hat{y}} \lambda_{\hat{\omega}'_k, \hat{y}}\right)$
    $\Delta(x) \leftarrow \Delta(x) + \mathrm{p}(\hat{y}) \cdot \delta(x, \hat{y})$
$x \leftarrow \text{argmax}_{x \in \mathscr{D}_{\text{pool}}} \Delta(x)$
output: $\{x\}$

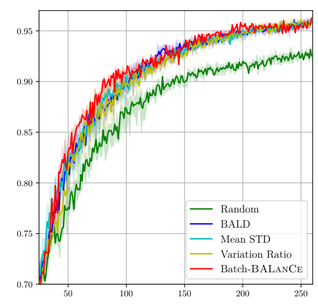
Only consider those pairs that have large distances

7 Experiments

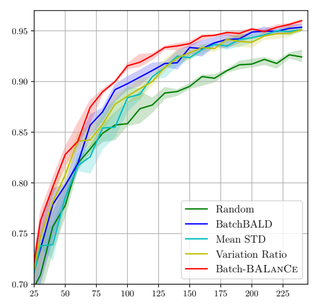
Dataset details

	CINIC-10	EMNIST-Balanced	EMNIST-ByMerge	EMNIST-ByClass	Fashion-MNIST	MNIST
# classes	10	47	47	62	10	10
# samples	270,000	131,600	814,255	814,255	70,000	70,000
Budget	1,400	600	600	450	350	260
τ	$\frac{\text{val err}}{4}$	$\frac{\text{val err}}{4}$	$\frac{\text{val err}}{4}$	$\frac{\text{val err}}{4}$	$\frac{\text{val err}}{4}$	$\frac{\text{val err}}{4}$
$\lvert\mathscr{D}_{\text{pool}}\rvert$	40,000	18,800	18,800	18,800	39,980	10,000

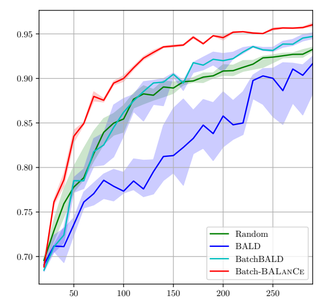
Balanced datasets



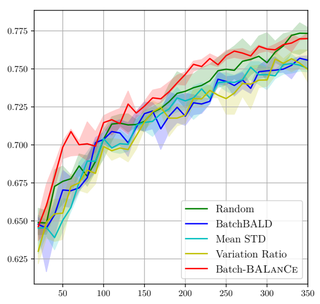
(a) ACC on MNIST B=1, K=100



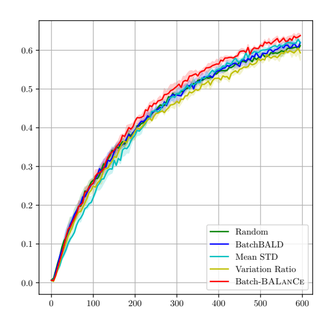
(b) ACC on MNIST B=10, K=100



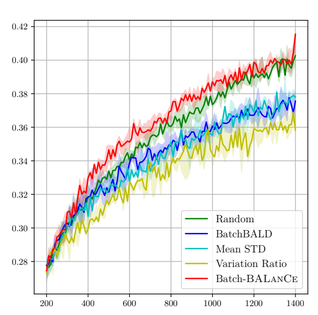
(c) ACC on Repeated-MNIST B=10, K=100



(d) ACC on Fashion-MNIST B=10, K=100

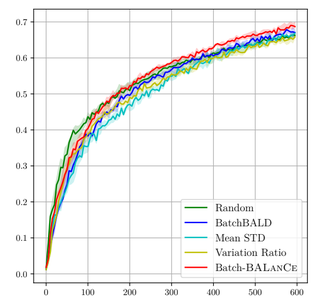


(e) F1 on EMNIST-balanced B=5, K=10

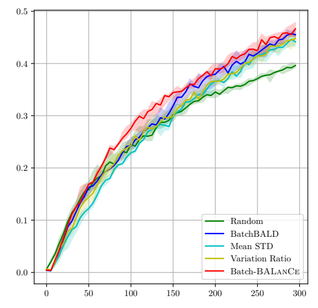


(f) ACC on CINIC-10 B=10, K=50

Imbalanced datasets



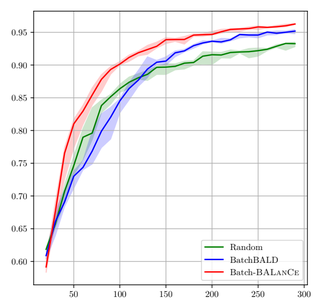
(g) ACC on EMNIST-Bymerge B=5, K=10



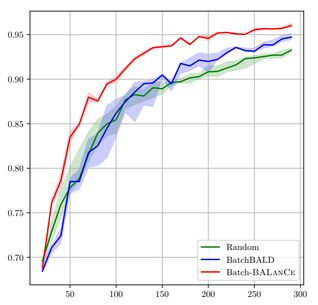
(h) F1 on EMNIST-Bymerge B=5, K=10

Increase repeat number

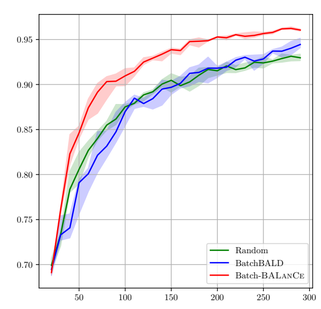
Performance on Repeated-MNIST with different repeat number



(a) Repeat 1 time



(b) Repeat 2 times



(c) Repeat 3 times

QR code of our paper