Active Learning in Bayesian Neural Networks: Balanced Entropy Learning Principle

Anonymous Author(s)
Affiliation
Address
email

Abstract

Acquiring labeled data is challenging in many machine learning applications with limited budgets. Active learning gives a procedure to select the most informative data points and improve data efficiency by reducing the cost of labeling. The infomax learning principle maximizing mutual information such as BALD has been successful and widely adapted in various active learning applications. However, this pool-based specific objective inherently introduces a redundant selection. In this paper, we design and propose a new uncertainty measure, Balanced Entropy Acquisition (BalEntAcq), which captures the information balance between the uncertainty of underlying softmax probability and the label variable. To do this, we approximate each marginal distribution by Beta distribution. Beta approximation enables us to formulate BalEntAcq as a ratio between a shifted entropy and the marginalized joint entropy. The closed-form expression of BalEntAcq facilitates parallelization by estimating two parameters in each marginal Beta distribution. BalEntAcq is a purely standalone measure without requiring any relational computations with other data points. Nevertheless, BalEntAcq captures a well-diversified selection near the decision boundary with a margin, unlike other existing uncertainty measures such as BALD, Entropy, or Mean Standard Deviation (MeanSD). Finally, we demonstrate that our balanced entropy learning principle with BalEntAcq consistently outperforms well-known linearly scalable active learning methods, including a recently proposed PowerBALD, a simple but diversified version of BALD, by showing experimental results obtained from MNIST, CIFAR-100, SVHN, and TinyImageNet datasets.

1 Introduction

Acquiring labeled data is challenging in many machine learning applications with limited budgets. As the dataset size gets bigger and bigger for training a complex model, labeling data by humans becomes more expensive. Active learning gives a procedure to select the most informative data points and improve data efficiency by reducing the cost of labeling.

The active learning problem is well-aligned with a subset selection problem that can find the most efficient but minimal subset from the data pool [70, 34, 18, 66, 86, 87, 85]. The difference is that active learning is typically an iterative process where a model is trained and a collection of data points is selected to be labeled from an unlabelled data pool. Therefore, it is still a theoretically very challenging but important problem.

It is now commonly accepted that standard deep learning models do not capture model uncertainty correctly. The simple predictive probabilities are usually erroneously described as model confidence [31]. So there is a risk that a model can be misdirecting its outputs with high confidence. However, the predictive distribution generated from Bayesian deep learning models better captures the uncertainty

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

from the data [26, 51, 67, 17]. Therefore, we focus on developing an active learning framework in the Bayesian deep neural network model by leveraging the Monte-Carlo (MC) dropout method as a proxy of the Gaussian process [26] which may facilitate further analysis.

1.1 Our contributions

Our proposed active learning method is well-aligned with Bayesian experimental design [89, 14, 80, 62, 24] with an assumption that the forward active learning iterative process follows the Bayesian prior-posterior framework. Furthermore, our approach is also aligned with Bayesian uncertainty quantification methods [40, 1, 35, 41, 26, 27, 48, 67, 47] with an assumption that the working neural network model is a Bayesian network [49].

In this paper, we extend and improve recent advances in both aspects of Bayesian experimental design and Bayesian uncertainty quantification. We investigate the generalized notion of the joint entropy between model parameters and the predictive outputs by leveraging a point process entropy [64, 25, 73, 16, 5]. By approximating the marginals using Beta distributions, we then derive an explicit formula of the marginalized joint entropy by estimating Beta parameters from Bayesian deep learning models. As a Bayesian experiment, we revisit the well-known entropy and mutual information measures given expected cross-entropy loss. We show that well-known acquisition measures are functions of marginal distributions through analytical formulas. We propose our new uncertainty measure, Balanced Entropy Acquisition (BalEntAcq), which captures the information balance between the uncertainty of underlying softmax probability and the label variable. Finally, we demonstrate that our balanced entropy learning principle with BalEntAcq consistently outperforms well-known linearly scalable active learning methods, including a recently proposed PowerBALD [47] for mitigating the redundant selection in BALD [27], by showing experimental results obtained from MNIST, CIFAR-100, SVHN, and TinyImageNet datasets.

2 Background

2.1 Problem formulation

We write an unlabeled dataset $\mathcal{D}_{\textbf{pool}}$ and the labeled training set $\mathcal{D}_{\textbf{training}} \subseteq \mathcal{D}_{\textbf{pool}}$ in each active learning iteration. We denote by $\mathcal{D}^{(n)}_{\textbf{training}}$ if it's necessary to indicate the specific n-th iteration step. Given $\mathcal{D}_{\textbf{training}}$, we train a Bayesian deep neural network model Φ with model parameters $\omega \sim \mathbf{p}(\omega)$.

Then for a data point $\mathbf{x}$ given $\mathcal{D}_{\textbf{training}}$, the Bayesian deep neural network Φ produces the prediction probability: $\Phi(\mathbf{x}, \omega) := (P_1(\mathbf{x}, \omega), \cdots, P_C(\mathbf{x}, \omega)) \in \Delta^C$ where $\Delta^C = \{(p_1, \cdots, p_C) : p_1 + \cdots + p_C = 1, p_i \geq 0 \text{ for each } i\}$ and C is the number of classes. For the final class output Y, it is assumed to be a multinoulli distribution (or categorical distribution):

$$Y(\mathbf{x}, \omega) := \begin{cases} 1 & \text{with probability } P_1(\mathbf{x}, \omega) \\ \vdots & \vdots \\ C & \text{with probability } P_C(\mathbf{x}, \omega). \end{cases} \tag{1}$$

For the sake of brevity, we sometimes omit $\mathbf{x}$ or ω by writing $\Phi(\omega)$, $P_i(\omega)$, $Y(\omega)$ or Φ, P_i, Y unless we need further clarifications on each data point $\mathbf{x}$. Under this formulation, the oracle (active learning algorithm) selects a subset of data points to add to the next training set, i.e. at $(n+1)$-th iteration, the training set is determined by $\mathcal{D}^{(n+1)}_{\textbf{training}} = \mathcal{D}^{(n)}_{\textbf{training}} \cup \{\text{Next training batch from Oracle}\}$. Once the next training batch is selected, the selected batch will be labeled. This means that the ground truth label information of the selected data is added in training set $\mathcal{D}^{(n+1)}_{\textbf{training}}$ in the next round. Then the goal in active learning is to minimize the number of selected data points to reach a certain level of prediction accuracy.

2.2 Examples of uncertainty based active learning methods

In this section, we list up well-known uncertainty measures suitable for Bayesian active learning.

1. **Random**: $\text{Rand}[\mathbf{x}] := U(\omega')$ where $U(\cdot)$ is a uniform distribution which is independent to ω. Random acquisition function assigns a random uniform value on $[0, 1]$ to each data point.

2. **BALD** (Bayesian active learning by disagreement) [58, 35, 27]: $\text{BALD}[\mathbf{x}] := \mathfrak{I}\left(\omega, Y\left(\mathbf{x}, \omega\right)\right)$, where $\mathfrak{I}(\cdot, \cdot)$ represents a mutual information between random measures. BALD captures the mutual information between the model parameters and the predictive output of the data point. In practice, we calculate the mutual information between the predictive output and the predictive probabilities.

3. **Entropy** [82]: $\text{Ent}[\mathbf{x}] := -\sum_i \left(\mathbb{E}P_i\right)\log\left(\mathbb{E}P_i\right)$. Entropy is the Shannon entropy with respect to the expected predictive probability. Entropy can be the uncertainty of the prediction probability. Moreover, under the cross-entropy loss, we may also interpret the entropy measure as an expected loss gain since $-\log\left(\mathbb{E}P_i\right)$ is the cross-entropy loss given the ground truth label is the class i.

4. **Mean standard deviation (MeanSD)** [14, 40, 1]: $\text{MeanSD}[\mathbf{x}] := \frac{1}{C}\sum_i \sqrt{\mathbb{E}P_i^2 - \left(\mathbb{E}P_i\right)^2}$. Mean standard deviation captures the average of the standard deviations for each marginal distribution.

5. **PowerBALD** [21, 47]: $\text{PowerBALD}[\mathbf{x}] := \log \text{BALD}[\mathbf{x}] + Z$, where Z is an independently generated random value from Pareto distribution with the exponent $\alpha > 0$. We use $\alpha = 1$ as a default choice suggested by [47]. The motivation of this randomized acquisition is to mitigate the redundant selection by diversifying selected multi-batch points. In general, we do not know which exponent will be the optimal choice.

In a multiple acquisition scenario, we simply add the above uncertainty values for each data point $\mathbf{x}_i$:

$$\text{AcqFunc}[\mathbf{x}_1, \cdots, \mathbf{x}_n] := \sum_{i=1}^{n} \text{AcqFunc}[\mathbf{x}_i], \tag{2}$$

where $\text{AcqFunc} \in \{\text{Rand}, \text{BALD}, \text{Ent}, \text{MeanSD}, \text{PowerBALD}\}$.

2.3 Summary of other active learning approaches

Cohn et al. [14] provided one of the first statistical analyses in active learning, establishing how to synthesize queries that reduce the model's forward-looking error by minimizing its variance leveraging MacKay's closed-form variance approximation [60]. In this fashion, there exists a line of works in Bayesian experimental design [11, 58, 89, 14, 80, 90, 24, 23, 38] with an assumption that the forward active learning iterative process follows Bayesian prior-posterior framework.

On the other hand, in active learning, accommodating both the information uncertainty and the diversification of the acquired samples is essential to improve the performance under multi-batch acquisition scenarios. In a theoretical perspective, the most natural way to combine the uncertainty and the diversification seems to leverage reasonable sub-modular functions, e.g. Nearest neighbor set function [92], BatchBALD [48], Determinantal Point Process [6] and SIMILAR [50] with sub-modular information measures, and then/or apply a fast linear-time algorithm to find a diversified multi-batch with a provable performance guarantee [69, 70, 20, 95, 79, 36, 37, 57]. Although a fast linear-time solver is available for general sub-modular functions, there still exists a gap with practical implementation, such as high memory requirements, which makes the computation unscalable for identifying multi-batch acquisition points, e.g., BatchBALD [48]. Similar to the sub-modular function optimization, there exist many customized optimization approaches, e.g. CoreSet [81] and more approaches [29, 39, 19, 94, 91].

Another recent approach is to look at parameters of the neural network and to diversify points such as BADGE [4] with gradients and BAIT [3] with Fisher information. There also exist network architectural design focused approaches such as Learning loss by designing loss prediction layers [96], UncertainGCN and CoreGCN [8] with graph neural networks , VAAL [84] and TA-VAAL [42] by applying adversarial learning methods.

3 Bayesian neural network model

We adopt the Bayesian neural network framework introduced in Gal et al. [26]. The core idea in the Bayesian neural network is leveraging the MC dropout feature to generate a distribution of the predictive probability as an output at inference time. Under mild assumptions, it turns out that it is equivalent to an approximation to a Gaussian Process [77, 68, 93, 26, 56].

3.1 Each softmax probability marginal approximately follows Beta distribution

We may consider a Bayesian neural network model Φ as a random measure, i.e., stochastic process parametrized by $\mathcal{D}_{\textbf{training}}$ over the data set $\mathcal{D}_{\textbf{pool}}$. Given a data point $\mathbf{x} \in \mathcal{D}_{\textbf{pool}}$, $\Phi(\mathbf{x}, \omega)$ produces a random probability distribution in a simplex Δ^C. This analogy has a close connection with the construction of random discrete distribution, originally introduced by Kingman [45]. Since then, random measure construction has been extensively developed in Bayesian nonparametrics, and it is well-known that Dirichlet probability having Beta marginals plays the central role in the construction of the random discrete distribution [46, 22, 75, 74, 7, 72, 78]. It is the main motivation of the Beta distribution approximation. Many kinds of literature similarly assume the Dirichlet distribution after the softmax in the Bayesian neural network.

As illustrated by Milios et al. [65], we may follow the construction of Dirichlet distribution. Following the approach by Ferguson [22], a Dirichlet probability can be constructed through a collection of independent Gamma distributions. On the other hand, each marginal in Gaussian Process (approximated by Bayesian neural network) in the softmax output having dependent components follows a log-normal distribution (before the normalization, but after the exponentiation in softmax). Then by applying the shape similarity between a log-normal distribution and Gamma distribution, the construction of random probability from log-normal distributions would produce an approximated Dirichlet distribution. Therefore we may assume that the marginal distribution would approximately follow the Beta distribution.

Alternatively, as an analytical approach, we may see that Beta approximation can be justified through Laplace approximation [61, 32, 33, 17]. There exists a mapping between multivariate Gaussian distribution and Dirichlet distribution under a softmax basis. Then Beta distribution follows as a marginal distribution of Dirichlet distribution. Therefore we may assume that Beta approximation exists through Laplace approximation under the assumption that the Bayesian neural network produces the multivariate Gaussian distribution (as a marginalized Gaussian process over finite rank covariate function) before the softmax layer [68, 93, 26, 56].

In practice, once we estimate the sample mean and sample variance for each marginal of $\Phi(\mathbf{x}, \omega)$, we can estimate two parameters of the Beta distribution as follows. Assume that $P_i \sim \text{Beta}(\alpha_i, \beta_i)$. If $\mathbb{E}P_i = m_i$ and $\text{Var}P_i = \sigma_i^2$, then

$$\alpha = \frac{m^2(1-m)}{\sigma^2} - m, \quad \beta = \left(\frac{1}{m} - 1\right)\alpha. \tag{3}$$

When $P_i \sim \text{Beta}(\alpha_i, \beta_i)$, $\mathbb{E}P_i = \frac{\alpha_i}{\alpha_i+\beta_i} = m$ and $\text{Var}P_i = \frac{\alpha_i\beta_i}{(\alpha_i+\beta_i)^2(\alpha_i+\beta_i+1)} = \sigma_i^2$. Solving the equation with respect to α_i and β_i, then the (3) follows.

3.2 Marginalized joint entropy in Bayesian neural network

In this section, we derive a marginalized joint entropy in the Bayesian neural network, which shall be further discussed in constructing our main results. We may formulate the Bayesian neural network Φ as a well-known encoder-decoder framework. The sender sends a message $(\mathbf{x}, \omega)$ with a random key ω through the Bayesian neural network, then the receiver receives a message $Y(\mathbf{x}, \omega)$.

Under this framework, controlling ω is difficult, but we can control the family of the encoded messages $\Phi(\mathbf{x}, \omega)$ in a tractable manner [27, 43, 88]. We can easily prove that the mutual information between ω and Y is the same as the mutual information between the encoded $\Phi(\mathbf{x}, \omega)$ and the predictive output Y since Y depends only on $\Phi(\mathbf{x}, \omega)$:

$$\begin{aligned} \text{BALD}[\mathbf{x}] :=& \Im(\omega, Y(\mathbf{x}, \omega)) = H(Y(\mathbf{x}, \omega)) - \mathbb{E}_\omega\left[H(Y(\mathbf{x}, \omega)|\omega)\right] & (4) \\ =& H(Y(\mathbf{x}, \omega)) - \mathbb{E}_\Phi\left[H(Y(\mathbf{x}, \omega)|\Phi(\mathbf{x}, \omega))\right] = \Im(\Phi(\mathbf{x}, \omega), Y(\mathbf{x}, \omega)), & (5) \end{aligned}$$

where $H(Y(\mathbf{x}, \omega))$ represents the Shannon entropy by marginalizing out the randomness of ω in $Y(\mathbf{x}, \omega)$ and $\Im(\cdot, \cdot)$ represents a mutual information between two quantities.

The formulations of the mutual information (4) - (5) look natural, but we need to note that ω or $\Phi(\mathbf{x}, \omega)$ is on a continuous domain, and $Y(\mathbf{x}, \omega)$ is on a discrete domain. This combined domain implies that we cannot directly apply Shannon entropy and differential entropy notions [15]. One immediate question is what the joint entropy between $\Phi(\mathbf{x}, \omega)$ and $Y(\mathbf{x}, \omega)$ is. For this, we can leverage point process entropy [64, 25, 73, 16, 5] by generalizing the notion of the entropy in this combined domain. We consider the joint entropy of $\Phi(\mathbf{x},\omega)$ and $Y(\mathbf{x},\omega)$, denoting by $\mathfrak{H}(\Phi(\mathbf{x},\omega), Y(\mathbf{x},\omega))$ through the point process entropy. We write a Janossy density function [16] $j(\mathbf{p}, y=i)$ of $(\Phi(\mathbf{x},\omega), Y(\mathbf{x},\omega))$ on $\Delta^C \times [C]$ as follows:

$$j(\mathbf{p}, y=i) = p_i f(\mathbf{p}), \tag{6}$$

where $\mathbf{p} := (p_1, \cdots, p_C)$ and $f(\cdot)$ is a density function of $\Phi(\mathbf{x},\omega)$. Then the joint entropy of $\Phi(\mathbf{x},\omega)$ and $Y(\mathbf{x},\omega)$ can be defined as

$$\mathfrak{H}(\Phi(\mathbf{x},\omega), Y(\mathbf{x},\omega)) = -\sum_{i=1}^{C} \int_{\Delta^c} j(\mathbf{p}, y=i) \log j(\mathbf{p}, y=i)\, d\mathbf{p}. \tag{7}$$

By plugging (6) into (7), we have the following identity.

$$\mathfrak{H}(\Phi(\mathbf{x},\omega), Y(\mathbf{x},\omega)) = H(Y(\mathbf{x},\omega)) + \mathbb{E}_Y\left[h(\Phi(\mathbf{x},\omega) | Y(\mathbf{x},\omega))\right], \tag{8}$$

where $H(\cdot)$ represents the usual Shannon entropy, and $h(\cdot)$ represents the usual differential entropy. By applying Jensen's inequality, we may derive a marginalized joint entropy as an upper bound of the joint entropy:

$$\mathfrak{H}(\Phi(\mathbf{x},\omega), Y(\mathbf{x},\omega)) \leq -\sum_i \mathbb{E}_{P_i}\left[P_i \log(P_i f(P_i))\right], \tag{9}$$

where we ambiguously write $f(\cdot)$ to be a density function for each P_i. Assume that each $P_i \sim \text{Beta}(\alpha_i, \beta_i)$ by applying Beta approximation. We then define a quantity of the marginalized joint entropy from (9) and we find an equivalent formulation as follows:

$$\text{MJEnt}[\mathbf{x}] := -\sum_i \mathbb{E}_{P_i}\left[P_i \log(P_i f(P_i))\right] = \underbrace{\sum_i (\mathbb{E}P_i)\, h(P_i^+)}_{\text{posterior uncertainty}} + \underbrace{H(Y)}_{\text{entropy}}, \tag{10}$$

where P_i^+ is the conjugate Beta posterior entropy of P_i which follows $P_i^+ \sim \text{Beta}(\alpha_i + 1, \beta_i)$. We remark that $h(P_i^+)$ can be easily calculated by the closed form entropy formula of Beta distribution. i.e.

$$h(P_i^+) = \log B(\alpha_i + 1, \beta_i) - \alpha_i \Psi(\alpha_i + 1) - (\beta_i - 1)\Psi(\beta_i) - (\alpha_i + \beta_i - 1)\Psi(\alpha_i + \beta_i + 1),$$

where $B(\cdot,\cdot)$ is the Beta function, and $\Psi(\cdot)$ is the Digamma function. We call the first term in (10) to be the posterior uncertainty. We may interpret the posterior uncertainty as an expected posterior entropy assuming that we observed a positive sample of the class toward P_i for each i without knowing the true class label. The first term is always non-positive, and is maximized (equals to 0) when each P_i^+ is $\text{Beta}(1,1)$, i.e., Uniform on $[0,1]$. So $-\infty < \text{MJEnt}[\mathbf{x}] \leq H(Y)$. The second entropy term can be decomposed into two uncertainty terms:

$$H(Y) = \underbrace{\mathfrak{I}(\omega, Y)}_{\text{epistemic uncertainty}} + \underbrace{\mathbb{E}_\omega\left[H(Y|\omega)\right]}_{\text{aleatoric uncertainty}}. \tag{11}$$

The epistemic uncertainty captures the model uncertainty (as BALD), and the aleatoric uncertainty captures the data uncertainty [63]. Therefore the marginalized joint entropy, MJEnt[$\mathbf{x}$] is a decomposition of three types of uncertainty values.

3.3 Entropy is for maximizing an expected cross-entropy loss

Given a ground-truth label $\{Y = i\}$, the cross-entropy loss of the neural network can be given as $\text{loss}(\Phi(\mathbf{x},\omega), Y=i) = -\log \mathbb{E}P_i$. Therefore we can calculate the expected cross-entropy loss without knowing the truth label:

$$\text{ExpectedLoss}[\mathbf{x}] := \sum_{i=1}^{C} \mathbb{P}[Y=i]\, \text{loss}(\Phi(\mathbf{x},\omega), Y=i) = -\sum_i (\mathbb{E}P_i)\log(\mathbb{E}P_i) = \text{Ent}[\mathbf{x}].$$

Based on the re-formulation, we may interpret that entropy acquisition is for maximizing an expected cross-entropy loss in a selection of acquisition points, aligning the idea with the learning loss [96]. The natural question is, "Once we acquire a data point that maximizes entropy acquisition, can we remove/or learn this expected cross-entropy amount of loss at the future stage of the active learning?". The answer could be "No." The exhaustive loss acquisition could only happen if the neural network perfectly over-fits the training data. Therefore, there exists a gap between a realistic neural network training scenario and the objective of the entropy acquisition. Our equivalent loss interpretation gives us an insight into why the entropy acquisition might not be successful in practice, even in the single-point acquisition scenario.

3.4 BALD is a function of marginals and is strongly aligned with maximizing an expected cross-entropy loss difference upto the next iteration

We have the mutual information between ω and Y and it is the same as the mutual information between the encoded message and the channel output since Y depends only on $\Phi(\mathbf{x}, \omega)$ [27]:

$$\mathrm{BALD}[\mathbf{x}] := \mathfrak{I}(\omega, Y) = \mathfrak{I}(\Phi(\mathbf{x}, \omega), Y(\mathbf{x}, \omega)), \tag{12}$$

where $\mathfrak{I}(\cdot, \cdot)$ represents mutual information between two quantities. By assuming that $\Phi(\mathbf{x}, \omega)$ follows a Dirichlet distribution, we can calculate the mutual information analytically [2]. Then by investigating further into the analytical mutual information formula, we see that the marginal distributions P_i's in $\Phi(\mathbf{x}, \omega)$ are sufficient to estimate BALD. Therefore we can state BALD through Beta marginal distributions as follows. See Appendix for more details.

Theorem 3.1. *Under Beta marginal distribution approximation, let $P_i \sim Beta(\alpha_i, \beta_i)$ in $\Phi(\mathbf{x}, \omega)$. Then the mutual information BALD$[\mathbf{x}]$ can be estimated as follows:*

$$\begin{aligned} \textit{BetaMarginalBALD}[\mathbf{x}] := &\sum_{i=1}^{C} (\alpha_i - 1)\Psi(\alpha_i + \beta_i) - \sum_{i=1}^{C} \left(\frac{\alpha_i}{\alpha_i + \beta_i}\right) \log\left(\frac{\alpha_i}{\alpha_i + \beta_i}\right) - \sum_{i=1}^{C} \frac{\alpha_i(\alpha_i - 1)}{\alpha_i + \beta_i}\Psi(\alpha_i) \\ &- \sum_{i=1}^{C} \frac{\beta_i(\alpha_i - 1)}{\alpha_i + \beta_i}\Psi(\alpha_i + \beta_i + 1) + \sum_{i=1}^{C} \left(\frac{\alpha_i^2}{\alpha_i + \beta_i}\right) [\Psi(\alpha_i + 1) - \Psi(\alpha_i + \beta_i + 1)]. \end{aligned}$$

As a Bayesian experimental design process, we may assume that each Beta marginal distribution P_i with the ground-truth label $\{Y = i\}$ of the next trained model would follow the Beta posterior distribution P_i^+. Without this assumption, existing choices of acquisition functions such as BALD or MeanSD might not be well-justified. For example, what is the implication of maximizing mutual information through the active learning process with a Bayesian neural network? How is it different from the maximization of the entropy acquisition? To answer these questions, leveraging our Beta marginalization and considering the similar idea of expected information gain [24], we may consider the expected cross-entropy loss difference between the current stage model and the next stage model.

$$\begin{aligned} \mathrm{ExpectedEffectiveLoss}[\mathbf{x}] &:= \sum_{i=1}^{C} \mathbb{E}P_i \left[-\log \mathbb{E}P_i - \left(-\log \mathbb{E}P_i^+\right)\right] \\ &= \sum_{i=1}^{C} \left(\frac{\alpha_i}{\alpha_i + \beta_i}\right) \left[\log\left(\frac{\alpha_i + 1}{\alpha_i + \beta_i + 1}\right) - \log\left(\frac{\alpha_i}{\alpha_i + \beta_i}\right)\right]. \end{aligned}$$

ExpectedEffectiveLoss captures the effective amount of cross-entropy loss for the model to learn after the acquisition. By definition, we see that ExpectedEffectiveLoss aims to exclude the undesirable over-fitting scenario assumption unlike Entropy acquisition.

Since Digamma function $\Psi(x) \sim \log x - \frac{1}{2x}$ where $f(x) \sim g(x)$ implies $\lim_{x\to\infty} f(x)/g(x) = 1$, we may expect that BetaMarginalBALD$[\mathbf{x}]$ and ExpectedEffectiveLoss$[\mathbf{x}]$ would behave similarly. Figure 1 shows the Spearman's rank correlations among different acquisition measures upto a class dimension $C = 10,000$. We observe that BetaMarginalBALD behaves equally like the original BALD and we confirm that BALD and MeanSD are strongly aligned with maximizing ExpectedEffectiveLoss. Therefore, acquiring points through BALD or MeanSD could be a better strategy than Entropy because BALD or MeanSD takes into account the effective loss acquisition instead of the unrealistic full amount of the loss acquisition.

4 Balanced entropy learning principle

The previous section shows that well-known acquisition measures have an objective toward the cross-entropy loss, and they are closely related to marginal distributions. According to Farquhar et al. [21], to be successful in active learning, they hypothesize that it is crucial to find a good balance between active learning bias and over-fitting bias under over-parametrized neural networks. In parallel to their hypothesis, we define the balanced entropy (BalEnt) to be a ratio between the marginalized joint entropy (9) and the shifted entropy:

$$\mathrm{BalEnt}[\mathbf{x}] := \frac{\mathrm{MJEnt}[\mathbf{x}]}{\mathrm{Ent}[\mathbf{x}] + \log 2} = \frac{\sum_i (\mathbb{E}P_i)\, h(P_i^+) + H(Y)}{H(Y) + \log 2}. \tag{13}$$

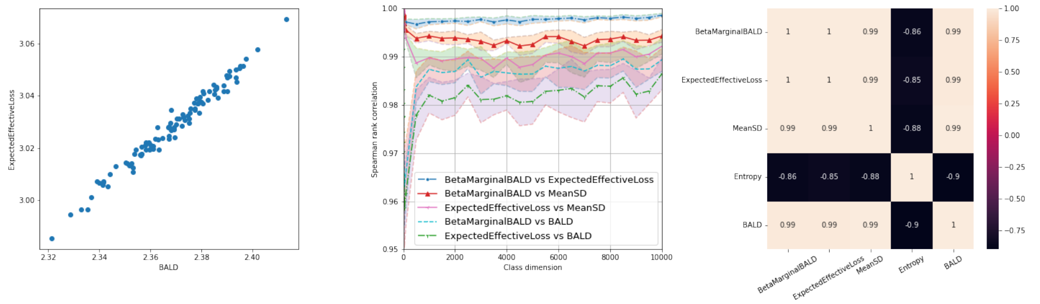


Figure 1: Scatter plot at $C = 10,000$ between BALD and ExpectedEffectiveLoss (left), Spearman's rank correlations over various class dimensions (middle), and Spearman's rank correlation matrix at $C = 10,000$ (right). The relationship between BetaMarginalBALD and ExpectedEffectiveLoss consistently captures a high rank-correlation with $> 99.6\%$ regardless of the class dimensions. BALD and ExpectedEffectiveLoss show $> 97.5\%$ rank-correlation. We randomly generate 100 softmax applied C-dimensional Gaussian samples and repeated the process 10 times. Shaded band shows the standard deviation.

Recall that we call the first term in MJEnt$[\mathbf{x}]$ to be posterior uncertainty, and it is an expected posterior entropy of underlying marginals. BalEnt captures the information balance between the posterior uncertainty from the model Φ and entropy of the label variable Y.

4.1 Implications of balanced entropy

To understand the implication of BalEnt$[\mathbf{x}]$, we can prove the following Theorem 4.1.

Theorem 4.1. *Let $\Delta^{-1} := \lfloor 2e^{H(Y)} \rfloor$ and $\Upsilon := \{I_n\}$, a collection of evenly divided intervals in $[0,1]$ where $I_n := \big[(n-1)\Delta, n\Delta\big)$ for $n = 1, \cdots, (\Delta^{-1} - 1)$ and $I_{\Delta^{-1}} := [1-\Delta, 1]$. Let $\bar{P}_i$ be a discretized random variable over Υ of P_i from $\Phi(\mathbf{x}, \omega)$. For any estimator $\hat{P}_i$ of $\bar{P}_i$ given the label $\{Y = i\}$ we have*

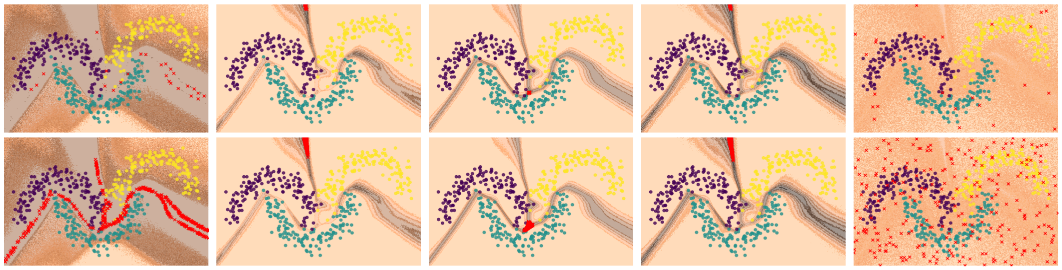
$$\mathbb{E}\left[\mathbb{P}\left[\hat{P}_i \neq \bar{P}_i \middle| Y = i\right]\right] \geq \frac{\sum_i (\mathbb{E}P_i)\, h(P_i^+) + H(Y)}{H(Y) + \log 2}(1+\epsilon_1) - \epsilon_2 = \mathit{BalEnt}[\mathbf{x}](1+\epsilon_1) - \epsilon_2,$$

where $\epsilon_1, \epsilon_2 \geq 0$ are adjustment terms depending on Δ such that $\epsilon_1 \to 0$ and $\epsilon_2 \to 0$ as $\Delta \to 0$.

Theorem 4.1 tries to answer the following inverse problem. For the unlabeled data point, $\mathbf{x}$, if we know the information of the label $\{Y = i\}$, how much can we reliably estimate the underlying probability P_i from the model Φ? As we know that $-\log P_i$ is the cross-entropy loss of the trained model with Y, it equivalently answers the estimation error probability of the loss prediction under a unit precision up to $-\log\Delta$ level. For the precision level, we are assuming to carry $-\log\Delta \approx H(Y) + \log 2$ nats - natural unit of information, re-scaled amount of bits, matching the enumerator with MJEnt$[\mathbf{x}]$ term. It is not clear how to determine a better choice of the precision level $-\log\Delta$. But we may understand the denominator $H(Y) + \log 2$ is for normalizing the term BalEnt$[\mathbf{x}] \leq 1$ as a probability. Then the sign of BalEnt$[\mathbf{x}]$ becomes very important. BalEnt$[\mathbf{x}] \geq 0$ implies that it could be impossible to perfectly predict the loss $-\log P_i$ given currently available information. i.e., there could exist information imbalance between the model and the label approximately starting from BalEnt$[\mathbf{x}] = 0$. Therefore, insight from Theorem 4.1 suggests us a new direction for our main active learning principle. We define our primary acquisition function, namely, balanced entropy learning acquisition (BalEntAcq), as follows:

$$\text{BalEntAcq}[\mathbf{x}] := \begin{cases} \text{BalEnt}[\mathbf{x}]^{-1} & \text{if BalEnt}[\mathbf{x}] \geq 0, \\ \text{BalEnt}[\mathbf{x}] & \text{if BalEnt}[\mathbf{x}] < 0, \end{cases}$$

Since the information imbalance exists at least from BalEnt$[\mathbf{x}] = 0$, we prioritize to fill the information gap from BalEnt$[\mathbf{x}] = 0$ toward positively increasing direction. If we try to fill the information imbalance gap from the highest BalEnt$[\mathbf{x}]$, the information imbalance would still exist around BalEnt$[\mathbf{x}] = 0$ area. Therefore, it might not improve the active learning performance much. See Appendix A12.2 and A12.3 for different prioritization and precision level results. That's the motivation why we take the reciprocal of BalEnt$[\mathbf{x}]$ when BalEnt$[\mathbf{x}] \geq 0$.



(a) BalEntAcq (ours) (b) BALD (c) Entropy (d) MeanSD (e) PowerBALD

Figure 2: Top-K selected points are marked by red color. The first row shows the top $K = 25$ points. The second row shows the top $K = 500$ point selections among around 0.6 million grid points.

4.2 Toy example illustration

To illustrate the behavior of BalEntAcq and its relationship with other uncertainty measures, we train a simple Bayesian neural network with a 3-class moon dataset in $\mathbb{R}^2$. Then we calculate each acquisition measure for all fixed lattice points in the square domain by assuming that the unlabeled pool is highly regularized (or uniform). i.e., by evenly discretizing the domain, we obtain each uncertainty value for each lattice point. The total number of lattice points is around 0.6 million. Then we choose top-K high uncertainty values for each method to observe the prioritized region for each method. We use $K = 25$ and $K = 500$. Figure 2 illustrates the top-K points selected by each method. The most significant phenomenon is that BalEntAcq's selection is highly diversified near the decision boundary showing a bifurcated margin because we are prioritizing the surface area of $\{\text{BalEnt}[\mathbf{x}] \geq 0\}$. This is well-aligned with the strategy avoiding high aleatoric points. (See Appendix A.13) Then we can imagine to conduct a uniform sampling on each contour surface $\{\text{BalEnt}[\mathbf{x}] = \lambda\}$ for each $\lambda \geq 0$, as we move to the surface for each $\lambda < 0$. That's why we observe bifurcated but diversified and balanced selection near the decision boundary with BalEngAcq in Figure 2-(a) when $K = 25$. On the other hand, there is a preferred area for each method from other measures except PowerBALD. PowerBALD shows a good diversification, but it could select non-informative points.

5 Experimental Results

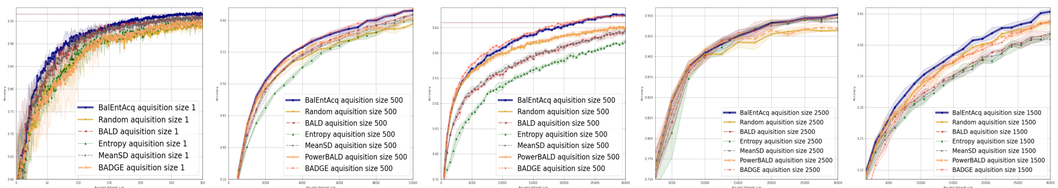
In this section, we demonstrate the performance of BalEntAcq from MNIST [55], CIFAR-100 [52], SVHN [71], and TinyImageNet [54] datasets under various scenarios. We used a single NVIDIA A100 GPU for each experiment, and details about the experiments are explained in Appendix A.12. We test Random, BALD, Entropy, MeanSD, PowerBALD, and BalEntAcq measures. We add BADGE for additional baseline. Note that all acquisition measures except BADGE in our experiments are standalone quantities, so all can be easily parallelized.

Single acquisition active learning with MNIST. MNIST [55] is the most popular and elementary dataset to validate the performance of image-based deep learning models initially. We use a simple convolutional neural network (CNN) model applying dropouts to all layers with a single acquisition size. The primary purpose of this single acquisition experiment is to validate our proposed balanced entropy approach by removing the contribution of diversification unlike multi-batch acquisition scenario.

Fixed features with CIFAR-100 and 3×CIFAR-100. In recent years, significant efforts have been made on building an efficient framework of unsupervised or self-supervised feature learning such as SimCLR [12, 13], MoCo [30], BYOL [28], SwAV [9], DINO [10], etc. As an application in active learning, we may leverage the feature space from the unsupervised feature learning without explicitly knowing true labels but construct a good representation space. In our experiments, we adopt SimCLR [12] for simplicity with ResNet-50 to build a feature space for CIFAR-100.

With 3×CIFAR-100 dataset, we observe the effect of the redundant information treatment for each method by adding three identical points. We use the same fixed feature obtained from SimCLR with CIFAR-100. We may observe how each method effectively diversifies the selection under a redundant data pool scenario by fixing the feature space.

Pre-trained backbone with SVHN and strong data augmentation with TinyImageNet. In this experiment, we follow a typical image classification scenario in practice. We use the ResNet-18 backbone for SVHN and the ResNet-50 backbone for TinyImageNet with ImageNet pre-trained model for model architecture, and the last linear classification layer is replaced with a simple Bayesian neural network with dropouts. We apply strong data augmentations for TinyImageNet, including random crop, random flip, random color jitter, and random grayscale. Under this scenario, the feature space from the backbone is continuously evolving and keeps confused as the training and active learning process proceeds. Because of the strong data augmentation and batch normalization in ResNet-18 or ResNet-50, the decision boundary keeps confused, implying that the Bayesian experimental design assumption might not hold. However, we still want to observe the general behavior of each measure and how to improve the accuracy under a more dynamic feature space.



(a) MNIST (b) CIFAR-100 (c) 3×CIFAR-100 (d) SVHN (e) TinyImageNet

Figure 3: Active learning accuracy curves obtained from various scenarios. Our proposed BalEntAcq outperforms well-known acquisition measures, and we repeated the experiment 3 times.

Scenario	Full dropouts + CNN			Fixed feature		Redundant images + Fixed feature		Backbone		Backbone + Augmentation	
Dataset/Acq. Size/Test size	MNIST/1/10,000			CIFAR-100/500/10,000		3×CIFAR-100/500/10,000		SVHN/2,500/26,032		TinyImageNet/1,500/10,000	
Train Size/Pool Size	50/60,000	100/60,000	300/60,000	5,000/50,000	10,000/50,000	15,000/150,000	30,000/150,000	15,000/73,257	30,000/73,257	15,000/100,000	30,000/100,000
Random	$78.6 \pm 4.9\%$	$86.4 \pm 2.7\%$	$93.6 \pm 0.7\%$	$55.5 \pm 0.4\%$	$59.4 \pm 0.5\%$	$61.9 \pm 0.2\%$	$64.9 \pm 0.3\%$	$91.8 \pm 0.6\%$	$93.2 \pm 0.2\%$	$37.1 \pm 0.3\%$	$43.8 \pm 0.1\%$
BALD	$82.6 \pm 1.3\%$	$90.5 \pm 0.8\%$	$95.3 \pm 0.4\%$	$56.2 \pm 0.5\%$	$60.8 \pm 0.3\%$	$58.8 \pm 0.2\%$	$64.6 \pm 0.6\%$	$92.5 \pm 0.8\%$	$94.8 \pm 0.2\%$	$35.2 \pm 0.7\%$	$41.8 \pm 0.4\%$
Entropy	$77.4 \pm 2.6\%$	$87.7 \pm 2.0\%$	$94.8 \pm 0.3\%$	$54.9 \pm 0.4\%$	$60.0 \pm 0.3\%$	$56.7 \pm 0.8\%$	$62.3 \pm 0.4\%$	$\mathbf{92.6} \pm 0.4\%$	$94.8 \pm 0.2\%$	$35.1 \pm 0.4\%$	$41.8 \pm 0.4\%$
MeanSD	$83.4 \pm 2.2\%$	$90.6 \pm 1.1\%$	$96.0 \pm 0.2\%$	$56.0 \pm 0.1\%$	$60.9 \pm 0.4\%$	$59.4 \pm 0.5\%$	$64.3 \pm 0.3\%$	$92.5 \pm 0.6\%$	$94.3 \pm 0.2\%$	$34.7 \pm 0.4\%$	$40.9 \pm 0.6\%$
PowerBALD	-	-	-	$56.5 \pm 0.1\%$	$60.3 \pm 0.2\%$	$62.2 \pm 0.2\%$	$65.0 \pm 0.7\%$	$92.2 \pm 0.6\%$	$93.5 \pm 0.2\%$	$37.4 \pm 0.7\%$	$43.4 \pm 0.3\%$
BADGE (not-scalable)	$77.0 \pm 6.1\%$	$86.5 \pm 4.2\%$	$94.8 \pm 0.4\%$	$\mathbf{57.4} \pm 0.1\%$	$\mathbf{61.8} \pm 0.1\%$	$\mathbf{64.0} \pm 0.2\%$	$\mathbf{67.4} \pm 0.1\%$	$\mathbf{92.9} \pm 0.4\%$	$95.0 \pm 0.3\%$	$37.2 \pm 0.6\%$	$43.9 \pm 0.3\%$
BalEntAcq (ours)	$\mathbf{85.4} \pm 1.0\%$	$\mathbf{91.4} \pm 1.3\%$	$\mathbf{96.5} \pm 0.1\%$	$\mathbf{57.2} \pm 0.2\%$	$\mathbf{61.5} \pm 0.2\%$	$\mathbf{63.5} \pm 0.5\%$	$\mathbf{67.4} \pm 0.1\%$	$92.5 \pm 0.8\%$	$\mathbf{95.2} \pm 0.1\%$	$\mathbf{38.5} \pm 0.2\%$	$\mathbf{45.3} \pm 0.4\%$

Table 1: Selected accuracy table. Mean and standard deviation are from 3 repeated experiments.

Discussion. BalEntAcq consistently outperforms other linearly scalable baselines in all datasets, as shown in Table 1. BADGE performs similarly with Entropy under a single acquisition scenario in MNIST because BADGE focuses on maximizing the loss gradient similar to Entropy, as we explained in Section 3.3. BADGE shows better performances at first when we fix the feature space, but our BalEntAcq eventually merges with the performance of BADGE. We also note that BADGE is not a linearly scalable method. Under dynamic feature scenarios in SVHN or TinyImageNet, we observe that our BalEntAcq performs better. Considering the acquisition calculation time (see Appendix A.14), our BalEntAcq should be a better choice. Figure 3 shows the full active learning curves. For CIFAR-100 and 3×CIFAR-100 cases, by fixing features, we control/remove all other effects possibly affecting the model's performance, such as data augmentation or the role of backbone in the classification. As demonstrated in Figure 2, BalEntAcq is very efficient in selecting diversified points along the decision boundary. Instead, PowerBALD suffers from improving accuracy because it focuses more on diversification/randomization by missing the information near the decision boundary. For SVHN or TinyImageNet, BalEntAcq shows better performance again. We suppose that diversification near the decision boundary in BalEntAcq also plays the data exploration because the representation space keeps evolving with the backbone training.

6 Conclusion

In this paper, we designed and proposed a new uncertainty measure, Balanced Entropy Acquisition (BalEntAcq), which captures the information balance between the underlying probability and the label variable through Beta approximation with a Bayesian neural network. BalEntAcq offers a diversified selection and is unique compared to other uncertainty measures. Moreover, we expect that our proposed balanced entropy measure does not have to be confined to active learning problems in general. BalEntAcq can be applied to improve the diversified selection process or accuracy estimation in a different type of Bayesian neural network frameworks. Therefore, we look forward to having further follow-up studies with broad applications beyond the active learning problems.

Limitations

As we specified in the introduction, our focus is MC-dropout-based Bayesian neural networks; our experiments have been limited to dropout-based Bayesian neural networks. However, our theoretical development does not require special architectural assumptions if we can apply Beta approximation. So one can apply our proposed method to any Bayesian classification network with Beta approximation. Moreover, considering the similarity with recent theoretically guaranteed active learning algorithm with abstention [59, 83, 76, 97] (see Appendix A.13), we expect to replicate the similar out-performance in other types of the Bayesian networks, e.g., Gaussian process [77], ensemble network [53], variational-dropout network [44], Laplace Redux [17], and so on.

References

[1] Vijay Badrinarayanan Alex Kendall and Roberto Cipolla. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 57.1–57.12. BMVA Press, September 2017.

[2] Anonymous. Analytic mutual information in bayesian neural networks. *To appear in 2022 IEEE International Symposium on Information Theory (ISIT)*, 2022.

[3] Jordan Ash, Surbhi Goel, Akshay Krishnamurthy, and Sham Kakade. Gone fishing: Neural active learning with fisher embeddings. *Advances in Neural Information Processing Systems*, 34, 2021.

[4] Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *International Conference on Learning Representations*, 2020.

[5] François Baccelli and Jae Oh Woo. On the entropy and mutual information of point processes. In *2016 IEEE International Symposium on Information Theory (ISIT)*, pages 695–699. IEEE, 2016.

[6] Erdem Bıyık, Kenneth Wang, Nima Anari, and Dorsa Sadigh. Batch active learning using determinantal point processes. *arXiv preprint arXiv:1906.07975*, 2019.

[7] Tamara Broderick, Michael I Jordan, Jim Pitman, et al. Beta processes, stick-breaking and power laws. *Bayesian analysis*, 7(2):439–476, 2012.

[8] Razvan Caramalau, Binod Bhattarai, and Tae-Kyun Kim. Sequential graph convolutional network for active learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9583–9592, 2021.

[9] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.

[10] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660, 2021.

[11] Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, pages 273–304, 1995.

[12] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[13] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020.

[14] David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active learning with statistical models. *Journal of artificial intelligence research*, 4:129–145, 1996.

[15] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

[16] Daryl J Daley and David Vere-Jones. *An introduction to the theory of point processes: volume II: general theory and structure*, volume 2. Springer Science & Business Media, 2007.

[17] Erik Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, Matthias Bauer, and Philipp Hennig. Laplace redux-effortless bayesian deep learning. *Advances in Neural Information Processing Systems*, 34, 2021.

[18] AP Dvoredsky. Some results on convex bodies and banach spaces. 1961.

[19] Ehsan Elhamifar, Guillermo Sapiro, Allen Yang, and S Shankar Sasrty. A convex optimization framework for active learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 209–216, 2013.

[20] Alina Ene and Huy L Nguyen. A nearly-linear time algorithm for submodular maximization with a knapsack constraint. *arXiv preprint arXiv:1709.09767*, 2017.

[21] Sebastian Farquhar, Yarin Gal, and Tom Rainforth. On statistical bias in active learning: How and when to fix it. *International Conference on Learning Representations*, 2021.

[22] Thomas S Ferguson. A bayesian analysis of some nonparametric problems. *The annals of statistics*, pages 209–230, 1973.

[23] Adam Foster, Desi R Ivanova, Ilyas Malik, and Tom Rainforth. Deep adaptive design: Amortizing sequential bayesian experimental design. In *International Conference on Machine Learning*, pages 3384–3395. PMLR, 2021.

[24] Adam Foster, Martin Jankowiak, Elias Bingham, Paul Horsfall, Yee Whye Teh, Thomas Rainforth, and Noah Goodman. Variational bayesian optimal experimental design. *Advances in Neural Information Processing Systems*, 32, 2019.

[25] J. Fritz. An approach to the entropy of point processes. *Periodica Mathematica Hungarica*, 3(1-2):73–83, 1973.

[26] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

[27] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *International Conference on Machine Learning*, pages 1183–1192. PMLR, 2017.

[28] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33:21271–21284, 2020.

[29] Yuhong Guo. Active instance sampling via matrix partition. *Advances in Neural Information Processing Systems*, 23, 2010.

[30] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[31] Matthias Hein, Maksym Andriushchenko, and Julian Bitterwolf. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 41–50, 2019.

[32] Philipp Hennig, David Stern, Ralf Herbrich, and Thore Graepel. Kernel topic models. In *Artificial intelligence and statistics*, pages 511–519. PMLR, 2012.

[33] Marius Hobbhahn, Agustinus Kristiadi, and Philipp Hennig. Fast predictive uncertainty for classification with bayesian deep networks. *arXiv preprint arXiv:2003.01227*, 2020.

[34] Dorit S Hochbaum. Approximating covering and packing problems: set cover, vertex cover, independent set, and related problems. In *Approximation algorithms for NP-hard problems*, pages 94–143. 1996.

[35] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

[36] Rishabh Iyer, Ninad Khargonkar, Jeff Bilmes, and Himanshu Asnani. Generalized submodular information measures: Theoretical properties, examples, optimization algorithms, and applications. *IEEE Transactions on Information Theory*, pages 1–1, 2021.

[37] Rishabh Iyer, Ninad A Khargonkar, Jeffrey A. Bilmes, and Himanshu Asnani. Submodular combinatorial information measures with applications in machine learning. In *The 32nd International Conference on Algorithmic Learning Theory*, Virtual Conference, March 2021.

[38] Aditi Jha, Zoe C Ashwood, and Jonathan W Pillow. Bayesian active learning for discrete latent variable models. *arXiv preprint arXiv:2202.13426*, 2022.

[39] Ajay J Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-class batch-mode active learning for image classification. In *2010 IEEE international conference on robotics and automation*, pages 1873–1878. IEEE, 2010.

[40] Michael Kampffmeyer, Arnt-Borre Salberg, and Robert Jenssen. Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1–9, 2016.

[41] Kirthevasan Kandasamy, Jeff Schneider, and Barnabás Póczos. Bayesian active learning for posterior estimation. In *Proceedings of the 24th International Conference on Artificial Intelligence*, pages 3605–3611, 2015.

[42] Kwanyoung Kim, Dongwon Park, Kwang In Kim, and Se Young Chun. Task-aware variational adversarial active learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8166–8175, 2021.

[43] Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[44] Durk P Kingma, Tim Salimans, and Max Welling. Variational dropout and the local reparameterization trick. *Advances in neural information processing systems*, 28, 2015.

[45] John FC Kingman. Random discrete distributions. *Journal of the Royal Statistical Society: Series B (Methodological)*, 37(1):1–15, 1975.

[46] John FC Kingman. The population structure associated with the ewens sampling formula. *Theoretical Population Biology*, 11(2):274–283, 1977.

[47] Andreas Kirsch, Sebastian Farquhar, and Yarin Gal. A simple baseline for batch active learning with stochastic acquisition functions. *arXiv preprint arXiv:2106.12059*, 2021.

[48] Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. 2019.

[49] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[50] Suraj Kothawade, Nathan Beck, Krishnateja Killamsetty, and Rishabh Iyer. Similar: Submodular information measures based active learning in realistic scenarios. *Advances in Neural Information Processing Systems*, 34, 2021.

[51] Agustinus Kristiadi, Matthias Hein, and Philipp Hennig. Being bayesian, even just a bit, fixes overconfidence in relu networks. In *International Conference on Machine Learning*, pages 5436–5446. PMLR, 2020.

[52] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2012.

[53] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

[54] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

[55] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[56] Jaehoon Lee, Yasaman Bahri, Roman Novak, Samuel S Schoenholz, Jeffrey Pennington, and Jascha Sohl-Dickstein. Deep neural networks as gaussian processes. *International Conference on Learning Representations*, 2017.

[57] Wenxin Li, Moran Feldman, Ehsan Kazemi, and Amin Karbasi. Submodular maximization in clean linear time, 2020.

[58] Dennis V Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, pages 986–1005, 1956.

[59] Andrea Locatelli, Alexandra Carpentier, and Samory Kpotufe. An adaptive strategy for active learning with smooth decision boundary. In *Algorithmic Learning Theory*, pages 547–571. PMLR, 2018.

[60] David JC MacKay. Information-based objective functions for active data selection. *Neural computation*, 4(4):590–604, 1992.

[61] David JC MacKay. Choice of basis for laplace approximation. *Machine learning*, 33(1):77–86, 1998.

[62] Andrey Malinin and Mark Gales. Predictive uncertainty estimation via prior networks. *Advances in neural information processing systems*, 31, 2018.

[63] Hermann G Matthies. Quantifying uncertainty: modern computational representation of probability and applications. In *Extreme man-made and natural hazards in dynamics of structures*, pages 105–135. Springer, 2007.

[64] J. McFadden. The entropy of a point process. *Journal of the Society for Industrial & Applied Mathematics*, 13(4):988–994, 1965.

[65] Dimitrios Milios, Raffaello Camoriano, Pietro Michiardi, Lorenzo Rosasco, and Maurizio Filippone. Dirichlet-based gaussian processes for large-scale calibrated classification. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[66] Vitali D Milman. A new proof of a. dvoretzky's theorem on cross-sections of convex bodies. *Funkcional. Anal. i Prilozen*, 5:28–37, 1971.

[67] Jishnu Mukhoti, Andreas Kirsch, Joost van Amersfoort, Philip HS Torr, and Yarin Gal. Deterministic neural networks with inductive biases capture epistemic and aleatoric uncertainty. *arXiv preprint arXiv:2102.11582*, 2021.

[68] Radford M Neal. Priors for infinite networks. In *Bayesian Learning for Neural Networks*, pages 29–53. Springer, 1996.

[69] George L Nemhauser and Laurence A Wolsey. Best algorithms for approximating the maximum of a submodular set function. *Mathematics of operations research*, 3(3):177–188, 1978.

[70] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming*, 14(1):265–294, 1978.

[71] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[72] Alon Orlitsky, Narayana P Santhanam, and Junan Zhang. Universal compression of memoryless sources over unknown alphabets. *IEEE Transactions on Information Theory*, 50(7):1469–1481, 2004.

[73] F. Papangelou. On the entropy rate of stationary point processes and its discrete approximation. *Probability Theory and Related Fields*, 44(3):191–211, 1978.

[74] Jim Pitman et al. Combinatorial stochastic processes. Technical report, Technical Report 621, Dept. Statistics, UC Berkeley, 2002., 2002.

[75] Jim Pitman and Marc Yor. The two-parameter poisson-dirichlet distribution derived from a stable subordinator. *The Annals of Probability*, pages 855–900, 1997.

[76] Nikita Puchkin and Nikita Zhivotovskiy. Exponential savings in agnostic active learning through abstention. In *Conference on Learning Theory*, pages 3806–3832. PMLR, 2021.

[77] C Rasmussen and C Williams. Gaussian processes for machine learning. adaptive computation and machine learning, 2006.

[78] Narayana P Santhanam, Anand D Sarwate, and Jae Oh Woo. Redundancy of exchangeable estimators. *Entropy*, 16(10):5339–5357, 2014.

[79] Jacob M Schreiber, Jeffrey A Bilmes, and William Stafford Noble. apricot: Submodular selection for data summarization in python. *J. Mach. Learn. Res.*, 21:161–1, 2020.

[80] Paola Sebastiani and Henry P Wynn. Maximum entropy sampling and optimal bayesian experimental design. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(1):145–157, 2000.

[81] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018.

[82] Claude E Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

[83] Shubhanshu Shekhar, Mohammad Ghavamzadeh, and Tara Javidi. Active learning for classification with abstention. *IEEE Journal on Selected Areas in Information Theory*, 2(2):705–719, 2021.

[84] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5972–5981, 2019.

[85] Daniel A Spielman and Nikhil Srivastava. Graph sparsification by effective resistances. *SIAM Journal on Computing*, 40(6):1913–1926, 2011.

[86] Daniel A Spielman and Shang-Hua Teng. Nearly linear time algorithms for preconditioning and solving symmetric, diagonally dominant linear systems. *SIAM Journal on Matrix Analysis and Applications*, 35(3):835–885, 2014.

[87] Daniel A Spielman and Jae Oh Woo. A note on preconditioning by low-stretch spanning trees. *arXiv preprint arXiv:0903.2816*, 2009.

[88] Dimitris G Tzikas, Aristidis C Likas, and Nikolaos P Galatsanos. The variational approximation for bayesian inference. *IEEE Signal Processing Magazine*, 25(6):131–146, 2008.

[89] Isabella Verdinelli and Joseph B Kadane. Bayesian designs for maximizing information and outcome. *Journal of the American Statistical Association*, 87(418):510–515, 1992.

[90] Benjamin T Vincent and Tom Rainforth. The darc toolbox: automated, flexible, and efficient delayed and risky choice experiments using bayesian adaptive design. *PsyArXiv. October*, 20, 2017.

[91] Zheng Wang and Jieping Ye. Querying discriminative and representative samples for batch mode active learning. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 9(3):1–23, 2015.

[92] Kai Wei, Rishabh Iyer, and Jeff Bilmes. Submodularity in data subset selection and active learning. In *International Conference on Machine Learning*, pages 1954–1963. PMLR, 2015.

[93] Christopher KI Williams. Computing with infinite networks. *Advances in neural information processing systems*, pages 295–301, 1997.

[94] Yi Yang, Zhigang Ma, Feiping Nie, Xiaojun Chang, and Alexander G Hauptmann. Multi-class active learning by uncertainty sampling with diversity maximization. *International Journal of Computer Vision*, 113(2):113–127, 2015.

[95] Grigory Yaroslavtsev, Samson Zhou, and Dmitrii Avdiukhin. “bring your own greedy”+ max: near-optimal 1/2-approximations for submodular knapsack. In *International Conference on Artificial Intelligence and Statistics*, pages 3263–3274. PMLR, 2020.

[96] Donggeun Yoo and In So Kweon. Learning loss for active learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 93–102, 2019.

[97] Yinglun Zhu and Robert Nowak. Efficient active learning with abstention. *arXiv preprint arXiv:2204.00043*, 2022.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section **??**.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]