Nonlinear Rescaling of Acquisition Metric Values Based on Distribution Fitting

Jongeui Park JONGEUI.PARK@KAIST.AC.KR
School of Electrical Engineering
Korea Advanced Institute of Science and Technology
Yuseong-gu, Dajeon 34141, Republic of Korea

Youngchul Sung YCSUNG@KAIST.AC.KR
School of Electrical Engineering
Korea Advanced Institute of Science and Technology
Yuseong-gu, Dajeon 34141, Republic of Korea

Abstract

In this paper, we consider active learning for unbalanced datasets. When class imbalance exists, active learning algorithms tend to acquire more samples from the majority class. We present a nonlinear rescaling mechanism to compensate for the effect of class imbalance. Experiments on unbalanced datasets with multiple types of class imbalance show that the proposed scheme yields noticeable performance gain when applied to existing active learning algorithms.

Keywords: Active Learning, Class Imbalance, Extreme Value Theory

1. Introduction

Data inefficiency is one of the main challenges to deep neural networks because they usually require substantial amounts of labeled data for training. Researchers may now easily collect an abundant amount of unlabeled data from the web, but labeling these collected data for training is still done by humans and costs excessive time and effort. Active learning tackles this problem and aims to minimize the labeling cost by devising a scheme to determine the unlabeled sample that, when labeled, would improve the final target performance to the greatest extent. Thus, active learning is currently under vigorous research (Ash et al., 2020; Gal et al., 2017; Sener and Savarese, 2018; Sinha et al., 2019; Pinsler et al., 2019).

However, most works on active learning assume that the dataset is balanced, which is far from reality in most cases. For instance, real-world problems like anomaly detection inherently involve an unbalanced dataset: anomalies are rare by definition. It is known that class imbalance severely degrades the performance of neural networks (Johnson and Khoshgoftaar, 2019). Since naive labeling of data from an unbalanced unlabeled dataset will lead to an unbalanced labeled dataset, advanced active learning is necessary in such cases. Typical active learning algorithms show suboptimal performance when the dataset is unbalanced. Extreme value theory (De Haan and Ferreira, 2007) states that the maximum value out of a pool increases as the size of the pool increases. So in a typical metric-based sample selection scheme, more samples from the majority class will be selected even when their acquisition metric values are much lower on average than samples from the minority class, solely due to their abundance (Figure 1).

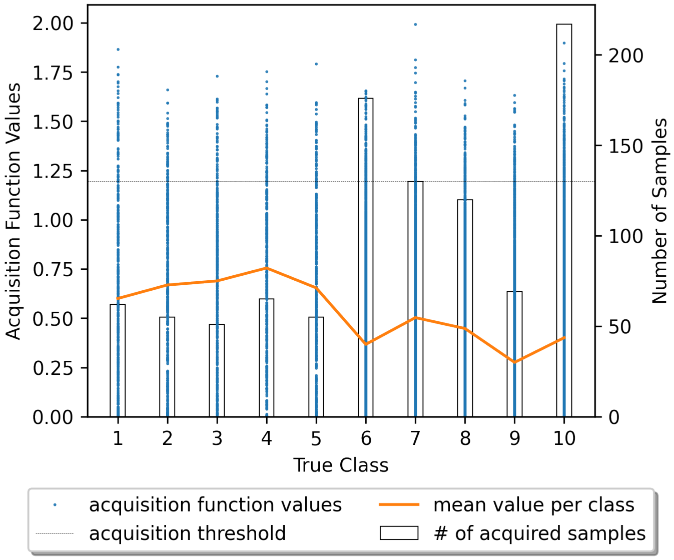


Figure 1: Distribution of acquisition metric values computed using a popular active learning algorithm MAXENT on an unbalanced dataset and the number of samples acquired from each class. The number of samples belonging to a majority class (classes 6–10) is ten times larger than the number of samples belonging to a minority class (classes 1–5). We can see that even though the average metric values of the minority class samples are greater than those of the majority class samples, the agent acquires more samples from the majority class. Refer to Appendix A for a detailed explanation of what each part means.

This paper mainly focuses on unbalanced data and proposes a new active learning scheme suited for such circumstances, although the proposed scheme can be applied to general situations including balanced datasets. Numerical results show the efficacy of the proposed method on unbalanced datasets with multiple types of class imbalance when applied to existing active learning algorithms.

2. Problem Setting

In the active learning literature, there are mainly two different problem settings: one is the single-pass online setting (Lughofer, 2012; Woodward and Finn, 2017), and the other is the pool-based off-line setting (Ash et al., 2020; Gal et al., 2017; Kirsch et al., 2019; Pinsler et al., 2019; Sener and Savarese, 2018; Sinha et al., 2019). In the online setting, unlabeled data arrives continuously in a stream one by one. Whenever a sample arrives, the agent should decide whether or not to request its label. On the other hand, in the off-line setting, a pool of unlabeled data already exists. The agent searches for informative unlabeled samples at each step and requests their labels. The process continues until the agent is out of budget. In this paper, we assume the off-line setting that we describe in detail below.

Consider a pool of unlabeled samples denoted by $U = \{u_j\}$. Let $L^{(i)} = \{(x_j, y_j)\}$ be the set of labeled samples at step i, where y_j is the true label of sample x_j. With a slight abuse of notation, we will write $U \setminus \{\, x \colon (x, y) \in L^{(i)} \,\}$ as $U \setminus L^{(i)}$, which denotes the set of samples in U that remains unlabeled at step i. Let $p(x; \theta)$ be the classifier[1] we want to optimize. At step i, the agent trains the model parameter θ by minimizing the following empirical loss based on $L^{(i)}$:

$$\theta_i^* = \arg\min_{\theta} \frac{1}{|L^{(i)}|} \sum_{(x,y) \in L^{(i)}} \ell(p(x; \theta), y), \tag{1}$$

where ℓ is the standard cross-entropy loss. After finishing training, the agent searches for an optimal set of samples $R^{(i)} \subseteq U \setminus L^{(i)}$ and requests their true labels. Then, a human expert labels the samples in $R^{(i)}$ and adds the newly labeled samples to $L^{(i)}$, yielding

$$L^{(i+1)} = L^{(i)} \sqcup \{\, (x, y) \colon x \in R^{(i)}, y \text{ is the true label of } x \,\}.$$

This procedure is repeated for each step i.

Most active learning algorithms select the samples for labeling based on some metric that depends on the current model parameter θ_i^*. Following Gal et al. (2017), we call this metric the *acquisition function* and denote it by $a(x; \theta_i^*)$, where $x \in U$. Since training takes a significant amount of time and the effect of adding a single labeled sample is negligible, we usually acquire multiple samples at each step. This leads us to define a *batch acquisition function* $a(B; \theta_i^*)$, where $B \subseteq U \setminus L^{(i)}$. Then, the acquisition step can be expressed as finding the batch $B \subseteq U \setminus L^{(i)}$ of K samples that maximizes $a(B; \theta_i^*)$, where K is the maximum number of samples the agent can request true labels for at each step. That is, at step i the agent selects $B^{(i)}$ as

$$B^{(i)} = \arg\max_{\substack{B \subseteq U \setminus L^{(i)} \\ |B| \leq K}} a(B; \theta_i^*). \tag{2}$$

and the agent requests labels of the samples in $B^{(i)}$. The most straightforward way to define $a(B; \theta_i^*)$ is

$$a(B; \theta_i^*) = \sum_{u \in B} a(x; \theta_i^*), \tag{3}$$

which turns the optimization problem (2) into simply selecting K samples with the highest acquisition function values. Some works discuss the use of batch acquisition functions that incorporate intra-batch sample diversity and thus cannot be written simply as (3), which comes with an increased computational cost (Ash et al., 2020; Kirsch et al., 2019; Pinsler et al., 2019).

3. Proposed Method

During the acquisition phase, the agent computes $a(x; (\theta_F)_i^*, \phi_i^*)$ for each $x \in U \setminus L^{(i)}$. Since our dataset is unbalanced, $U \setminus L^{(i)}$ is highly likely to be unbalanced. Under this situation, just selecting the top K samples with the highest acquisition function scores is

[1] In this paper, we only consider the object classification task, but active learning in general can be applied to other tasks such as object detection or image segmentation (Casanova et al., 2020).

problematic because the extreme values (i.e., the maximum and the minimum values) of a population with a shared distribution increase as the population size grows (De Haan and Ferreira, 2007). So the samples from a majority class will have higher acquisition function values than those from a minority class, not because they are more informative but just because there are more of them. This phenomenon would not happen if the dataset were balanced. In order to correct the class imbalance, we should mitigate the effect of the class size on the acquisition function value. Extreme value theory provides the scaling law of the maximum value of a population as the population size increases (De Haan and Ferreira, 2007). However, what we need is not just a scaling law but a concrete method to eliminate the effect of the class size and normalize the acquisition value for a fair comparison across the classes.

We propose a normalization scheme based on the following observation on the relationship between the sample size of a population and its extreme value:

Theorem 1 *Let $X_1, X_2, \ldots, X_N$ be independent and identically distributed (i.i.d.) random variables; let F_X be the cumulative distribution function (cdf) of X_1. Then, for all x,*

$$P\left(\max\{g(X_1), g(X_2), \ldots, g(X_N)\} \leq x\right) = P\left(X_1 \leq x\right),$$

where $g\colon \mathbb{R} \to \mathbb{R}$ is defined by the equation

$$g(y) = F_X^{-1}(F_X(y)^N). \tag{4}$$

Proof Let $Y = \max\{X_1, X_2, \ldots, X_N\}$. It is obvious that

$$\max\{g(X_1), g(X_2), \ldots, g(X_N)\} = g(Y).$$

Since $g(Y) \leq x$ if and only if $Y \leq g^{-1}(x)$, which holds if and only if $X_i \leq g^{-1}(x)$ for all i.

$$P(g(Y) \leq x) = \prod_{i=1}^{N} P(X_i \leq g^{-1}(x)) = [P(X_1 \leq g^{-1}(x))]^N = [F_X(g^{-1}(x))]^N.$$

From (4), we know that

$$F_X(g(y)) = [F_X(y)]^N,$$

so if we let $y = g^{-1}(x)$,

$$P(g(Y) \leq x) = [F_X(y)]^N = F_X(g(g^{-1}(x))) = F_X(x) = P(X_1 \leq x).$$

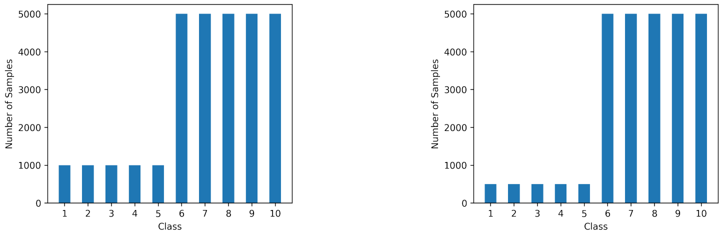
■

This theorem implies that by applying the transform g defined by (4) on the acquisition metric values, we can remove the effect of the population size N from $\max\{X_1, X_2, \ldots, X_N\}$.

Unfortunately, two obstacles block us from applying this rule to sample selection in active learning. First, we do not know what class each unlabeled sample belongs to unless we request their labels. Second, we need access to the cdf in order to compute the transform g. We bypass these hurdles by pseudo-labeling the samples according to the trained classifier $p(x; \theta_i^*)$ and distribution fitting using the maximum likelihood method. The parametric distribution family $\{\, f(\cdot; \phi) : \phi \in \Phi \,\}$ used for modeling the distribution of acquisition metric values should be selected a priori. If the acquisition metric is known to be bounded, the beta distribution will be a reasonable choice. Otherwise, we may use the gamma distribution or the normal distribution. Algorithm 1 presents a pseudo-code for the proposed method.

Algorithm 1 Normalizing Acquisition Metric Values

```
$i \leftarrow 0$.
while $|L^{(i)}| < M$ do
    Initialize $\theta$ randomly.
    Train the network with the cross-entropy loss on $L^{(i)}$ to obtain $\theta_i^*$.
    $S_1, S_2, \ldots, S_C \leftarrow \varnothing$
    for $x \in U \setminus L^{(i)}$ do
        $P(x) \leftarrow \arg\max_k p_k(x; \theta_i^*)$                ▷ Pseudo-label
        $S_{P(x)} \leftarrow S_{P(x)} \cup \{x\}$
    end for
    for $j \leftarrow 1, \ldots, C$ do
        $\phi_j^* \leftarrow \arg\max_{\phi_j \in \Phi} \sum_{x \in S_j} \log f(a(x; \theta_i^*); \phi_j)$      ▷ Maximum likelihood estimate
    end for
    $B^{(i)} \leftarrow \arg\operatorname{topk}_{x \in U \setminus L^{(i)}} F^{-1}\left(\left[F\left(a(x; \theta_i^*); \phi^*_{P(x)}\right)\right]^{|S_{P(x)}|}; \phi^*_{P(x)}\right)$      ▷ (4)
    Request labels for $B^{(i)}$ and create $L^{(i+1)}$
    $i \leftarrow i + 1$.
end while
```



(a) Step-5

(b) Step-10

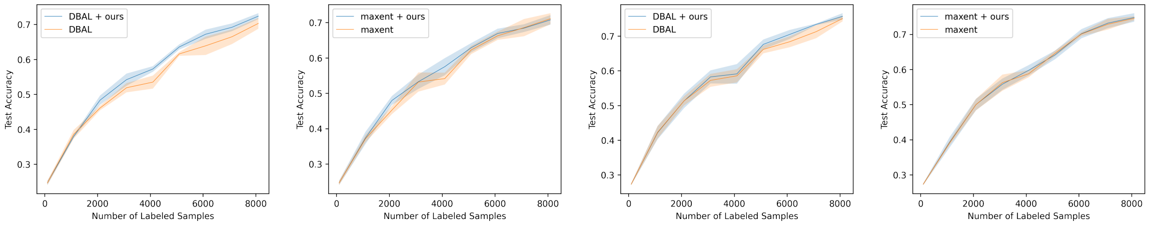
Figure 2: Number of samples that belong to each class for each type of class imbalance.

4. Experiments

This section provides numerical results to evaluate the proposed active learning algorithm. Since it is difficult to find popular datasets with class imbalance, we followed the practice of Buda et al. (2018) and created an artificial imbalance by removing samples from an existing balanced dataset. Each class in our artificial dataset is either a minority class or a majority class. The number of samples is equal across the minority classes, and so it is across the majority classes. That is, $|C_\ell| = \min_k\{|C_k|\}$ if ℓ is a minority class and $|C_\ell| = \max_k\{|C_k|\}$ if ℓ is a majority class, where C_k is the set of samples in U belonging to class k. Two parameters μ and ρ can characterize the imbalance, where they are defined as

$$\mu = \frac{\#\text{ of minority classes}}{d}, \qquad \rho = \frac{\max_k\{|C_k|\}}{\min_k\{|C_k|\}}.$$

We tested our algorithm on two types of imbalance: step imbalance with $\mu = 0.5$ and $\rho = 10$ (step-10) and step imbalance with $\mu = 0.5$ and $\rho = 5$ (step-5). We used CIFAR-



(a) Step-10 (DBAL) (b) Step-10 (maxent) (c) Step-5 (DBAL) (d) Step-5 (maxent)

Figure 3: Performance comparison. The shaded region denotes one standard deviation.

10 (Krizhevsky et al., 2009) as our base dataset. Figure 2 shows the number of samples belonging to each class for each type of imbalance.

We began our experiments with $|L^{(0)}| = 100$ and acquired $K = 1000$ samples at each step until 30 % of the entire dataset was labeled. We used the CIFAR-variant of ResNet-18 (He et al., 2016) as the architecture for our backbone network. We trained for 100 epochs during training phase using the Adam (Kingma and Ba, 2015) optimizer with learning rate 5×10^{-4} and batch size 256. We measured the performance of the classifier at each step, using the balanced test set of the original CIFAR-10 dataset. The algorithm was implemented with PyTorch (Paszke et al., 2019), and the distribution fitting was done using SciPy (Virtanen et al., 2020).

We applied our algorithm to two active learning algorithms DBAL (Gal et al., 2017) and maxent. Since DBAL requires the network to contain a dropout layer, we added a dropout layer with dropout probability 0.5 right before the last fully-connected layer. We sampled 25 times for the Monte Carlo approximation of the acquisition function used by DBAL. For fair comparison, we also sampled the maxent acquisition function 25 times and used its mean value. The acquisition metrics used in both algorithms are bounded from below by 0 and bounded from above by $\log C$ (See Appendix B). So we used the beta distribution scaled by $\log C$ to fit the data. Figure 3 shows the change in test accuracy as samples get labeled. We can see that our algorithm outperforms the baselines for both types of datasets.

5. Conclusion

In this paper, we have proposed an active learning algorithm for unbalanced datasets that compensates for the class imbalance. In order to resolve the class imbalance effect, we introduced a special type of nonlinear transform based on distribution fitting. The transform allows us to remove the dependency of the extreme value in each class on the class population size, while preserving their statistical information. Although the proposed algorithm shows its effectiveness for active learning on unbalanced datasets, it has some limitations and rooms for improvement. First, the usage of imperfect pseudo-labels are not justified and may lead to performance degradation. Second, it is not straightforward to apply our method to batch acquisition functions introduced in recent works such as Ash et al. (2020), Kirsch et al. (2019), or Pinsler et al. (2019). Finally, the experiments were performed on artificial datasets constructed from CIFAR-10. Experiments on more realistic unbalanced datasets are necessary. We leave them as future work.

Acknowledgments

This work was conducted by Center for Applied Research in Artificial Intelligence(CARAI) grant funded by Defense Acquisition Program Administration(DAPA) and Agency for Defense Development(ADD) (UD190031RD).

References

Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL `https://openreview.net/forum?id=ryghZJBKPS`.

Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018. doi: 10.1016/j.neunet.2018.07.011. URL `https://doi.org/10.1016/j.neunet.2018.07.011`.

Arantxa Casanova, Pedro O. Pinheiro, Negar Rostamzadeh, and Christopher J. Pal. Reinforced active learning for image segmentation. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL `https://openreview.net/forum?id=SkgC6TNFvr`.

Laurens De Haan and Ana Ferreira. *Extreme value theory: an introduction*. Springer Science & Business Media, 2007.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1183–1192. PMLR, 2017. URL `http://proceedings.mlr.press/v70/gal17a.html`.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.90. URL `https://doi.org/10.1109/CVPR.2016.90`.

Justin M. Johnson and Taghi M. Khoshgoftaar. Survey on deep learning with class imbalance. *J. Big Data*, 6:27, 2019. doi: 10.1186/s40537-019-0192-5. URL `https://doi.org/10.1186/s40537-019-0192-5`.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL `http://arxiv.org/abs/1412.6980`.

Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 7024–7035, 2019. URL `https://proceedings.neurips.cc/paper/2019/hash/95323660ed2124450caaac2c46b5ed90-Abstract.html`.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Edwin Lughofer. Single-pass active learning with conflict and ignorance. *Evol. Syst.*, 3 (4):251–271, 2012. doi: 10.1007/s12530-012-9060-7. URL `https://doi.org/10.1007/s12530-012-9060-7`.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035, 2019. URL `https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html`.

Robert Pinsler, Jonathan Gordon, Eric T. Nalisnick, and José Miguel Hernández-Lobato. Bayesian batch active learning as sparse subset approximation. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 6356–6367, 2019. URL `https://proceedings.neurips.cc/paper/2019/hash/84c2d4860a0fc27bcf854c444fb8b400-Abstract.html`.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL `https://openreview.net/forum?id=H1aIuk-RW`.

Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 5971–5980. IEEE, 2019. doi: 10.1109/ICCV.2019.00607. URL `https://doi.org/10.1109/ICCV.2019.00607`.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Mark Woodward and Chelsea Finn. Active one-shot learning. *CoRR*, abs/1702.06559, 2017. URL http://arxiv.org/abs/1702.06559.

Appendix A. Experiments on the Class Imbalance Effect

To investigate the effect of class imbalance, we conducted a simple experiment on the step-10 dataset. The result is depicted in Figure 1. The blue scatter plot shows how the MAXENT acquisition function values are distributed for each class. That is, each sample in $U \setminus L^{(0)}$ is depicted as a blue dot in the plot, where its x-axis is the sample's true label, and the y-axis is the acquisition function value. The orange line plot indicates the mean MAXENT acquisition function value of samples that belong to each class. Based on the computed MAXENT acquisition function values of the samples in $U \setminus L^{(0)}$, we acquired $K = 1000$ samples with the highest entropy values. The bar graph shows the class distribution of the $K = 1000$ selected samples. It is evident that samples belonging to the majority classes (6–10) are acquired more than those belonging to the minority classes $(1 - 5)$, even though the mean acquisition function values of the majority classes are lower than those of the minority classes.

Appendix B. Bound Analysis of Acquisition Functions

It is well known that the entropy of a discrete probability distribution is nonnegative and is maximized by the uniform distribution. So the maxent acquisition function is bounded below by 0 and bounded above by

$$-\sum_{k=1}^{C} p_k \log p_k = -\sum_{k=1}^{C} \frac{1}{C} \log \frac{1}{C} = \sum_{k=1}^{C} \frac{1}{C} \log C = \log C,$$

where p_k denotes $p_k(x; \theta_i^*)$.

DBAL uses the following acquisition function:

$$-\sum_{k=1}^{C} \left(\frac{1}{T} \sum_{t=1}^{T} p_k^{(t)} \right) \log \left(\frac{1}{T} \sum_{t=1}^{T} p_k^{(t)} \right) + \frac{1}{T} \sum_{k=1}^{C} \sum_{t=1}^{T} p_k^{(t)} \log p_k^{(t)}, \tag{5}$$

where $p_k^{(t)}$ is the t-th Monte Carlo sample $p_k(x; \theta_i^{(t)})$. Note that $\hat{p}_k = \frac{1}{T} \sum_{t=1}^{T} p_k^{(t)}$ forms a probability mass function of a discrete random variable that can take C values because

$$\sum_{k=1}^{C} \hat{p}_k = \sum_{k=1}^{C} \frac{1}{T} \sum_{t=1}^{T} p_k^{(t)} = \frac{1}{T} \sum_{t=1}^{T} \sum_{k=1}^{C} p_k^{(t)} = \frac{1}{T} \sum_{t=1}^{T} 1 = 1.$$

Since the first term of (5) is the entropy value of this distribution, it is bounded from above by $\log C$, and the maximum is attained when

$$\hat{p}_k = \frac{1}{T}\sum_{t=1}^{T} p_k^{(t)} = \frac{1}{C} \tag{6}$$

for each $k = 1, 2, \ldots, C$. For each $t = 1, 2, \ldots, T$, $\sum_{k=1}^{C} p_k^{(t)} \log p_k^{(t)}$ is the negative entropy value of the distribution $(p_1^{(t)}, p_2^{(t)}, \ldots, p_C^{(t)})$, so it is bounded above by 0, and the maximum is attained when

$$p_k^{(t)} = \delta_{jk} \text{ for some } j \in \{1, 2, \ldots, C\}. \tag{7}$$

Conditions (6) and (7) can both be satisfied when

$$p_k^{(t)} = \begin{cases} 1 & \text{if } t \equiv k \pmod{C} \\ 0 & \text{otherwise.} \end{cases}$$

Therefore (5) is bounded from above by $\log C$.

To compute the lower bound, observe that by the convexity of the function $g(x) = x \log x$,

$$g\left(\frac{1}{T}\sum_{t=1}^{T} p_k^{(t)}\right) \leq \frac{1}{T}\sum_{t=1}^{T} g(p_k^{(t)})$$

for each $k = 1, 2, \ldots, C$. Therefore,

$$(5) = -\sum_{k=1}^{C} g\left(\frac{1}{T}\sum_{t=1}^{T} p_k^{(t)}\right) + \sum_{k=1}^{C}\sum_{t=1}^{T} g(p_k^{(t)}) \geq 0,$$

that is, (5) is bounded below by 0. The minimum is attained when

$$p_k^{(t)} = \begin{cases} 1 & \text{if } k = 1, \\ 0 & \text{otherwise.} \end{cases}$$