Sampling settings in active learning for investigating inconsistency

by

Mengze Li

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Wednesday September 23, 2020 at 02:00 PM.

Student number:	4956338	
Project duration:	February, 2020 – September, 2020	
Thesis committee:	Prof, Dr. Marco Loog,	Mentor
	Dr. Jan van Gemert,	Graduation committee
	Dr. ir. Sicco Verwer,	Graduation committee

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft

Preface

It is a long journey to get to the end of my master study. I have met many challenges during my master study, both in school and in life. I really appreciate those who have offered help and support to me when I feel hopeless and desperate and those who have provided me with skills and opportunities. I hereby express gratitude to these people and organizations. Thanks to EIT Digital Master School for the scholarship. Thanks to my parents for constant emotional and financial support. Thanks to professors/coordinators who have given me professional suggestions and personal guidance, especially to Marco Loog and Anna Vilanova. I also need to thank friends H Jiang and W Huang for the company during time.

I feel really lucky to have the chance to study in TU Delft. The education here provides me with knowledge and skills needed to start my career. I am sure I will not forget the days I spent in Delft no matter how many years from now on. X(sports and culture center) is an amazing place where I has a really good time. If not for the COVID-19, I could have a happier and easier time in the final days of my master study.

I have spent several months doing my master thesis. It is also a long journey to finish my thesis. It is quite good to work with Prof. Marco Loog. He has given me so many instructions and suggestions for my thesis. I really enjoyed the time to conduct my thesis under the supervision of Marco.

Mengze Li
Delft, September 2020

Sampling settings in active learning for investigating inconsistency

Mengze Li

Abstract

Active learning has the potential to reduce labeling costs in terms of time and money. In practical use, active learning works as an efficient data labeling strategy. Another point of view to look at active learning is to consider active learning as a learning problem, where the training data is queried by the active learner. Under this perspective, an important question is inconsistency: can classifiers trained using active learning converge to the same result as using random sampling given an infinite number of data. In this paper, we discuss the possibility and potential consequences of using new sampling settings other than sampling without replacement in active learning to analyze the inconsistency problem. Moreover, a third sampling setting is defined to simulate the infinite data scenario in inconsistency. We compare the traditional setting, sampling without replacement in active learning with sampling with replacement in active learning, and true active learning. Furthermore, the two unusual sampling settings provide insight into the inconsistency problem. (1)Regularization parameter without adjustment can lead to inconsistency. (2)Querying data "really" close to the decision boundary can also bring threats to active learning.

1 Introduction

How to make use of available data with little annotation cost is a fundamental question in classification problems. In fully supervised learning, samples are drawn randomly from the data pool to obtain labels. The approach using random sampling is called passive sampling in contrast to active learning. The samples acquired by random sampling are considered independent and identically distributed, following the true underlying distribution. Active learning strategies allow the learning algorithm to explore in the unlabeled data pool itself. In one active learning iteration, the active learner draws one or more data samples to be labeled according to active learning strategies. The most simple and commonly used active learning strategy is uncertainty sampling [1,15]. In uncertainty active learning, the active learner finds the sample in the unlabeled data pool that the classifier is least certain about its prediction. Once the label of this sample is acquired, this sample is removed from the current data pool and be added to the training data. In active learning, one samples without replacement, meaning one sample can only be selected once. If no additional stopping criteria is set, active learning stops once all available samples in the pool are labeled.

One goal of active learning is to reduce the cost of data labeling by actively selecting samples to query labels. It is expected to use less labeled data to achieve a competitive classification performance compared to the classification performance when all data in the pool are labeled and used for training. There are concerns about the effect of active learning. Firstly, even though each active learning strategy has its intuition and reasoning to select the most informative instance, there is no hard proof that active learning is guaranteed to achieve a better performance than random sampling when the same amount of training samples are selected. Several studies [5, 10] have shown that active learning strategies can perform worse than random sampling. The second question about the performance of active learning is inconsistency: given infinite number of samples, does active learning converge to the same result as random sample? The problem of inconsistency is fundamental in active learning but receives little attention. This paper focuses on the inconsistency problem. Even though inconsistency is considered a result of sampling bias in active learning [3, 9, 14], the exact link between them is not clear.

Sampling without replacement is a default setting in active learning [1, 15]. Nevertheless, when we investigate the inconsistency problem in active learning, this setting has two limitations: (1) the performance of active is affected by the number of available samples; (2)active learning cannot run infinite times. Therefore, we implement two unusual settings for active learning to illustrate and understand the inconsistency problem. Sampling with replacement [9] allows us to conduct active learning until infinity but cannot simulate a scenario with an infinite number of samples. The idea of simulating an infinite number of samples is further extended into a new setting named true active learning in this article. In true active learning, it is assumed the distribution of data in the feature space is known, and it is possible to query instance located anywhere in the feature space.

In this paper, we illustrate the limitations of sampling with and without replacement and show the consequences of sampling under two unusual settings. We find both two new settings meet the problem of inconsistency. The performance of the two new settings is worse than usual active learning in terms of surrogate loss. We believe this is a result of building a classifier with less optimal reg-

ularization parameter. When using active learning with replacement, sampling eventually gets stuck by sampling two single points and do not query other samples.

Moreover, using the two unusual sampling settings in active learning, we make several attempts to investigate into inconsistency. We find that the regularization parameter under sampling bias is indeed one factor of inconsistency in active learning. If the regularization parameter is not adjusted, it is possible to lead to a model that fits only the training data but not fit the true distribution of the data. Also, if sampling happens really close to the decision boundary, there are more chances of getting more unhelpful data and affect the probability estimation of the classifier. Unhelpful samples refer to the ones whose true class labels is different from the prediction of the classifier, which is trained on a large amount of samples independent and identically distributed.

This paper is organized as follows. Section 2 provides background for the active learning system we implemented, including the active learning strategy and the classifier, and how we evaluate performance for the system. Section 3 explains the three sampling settings used in this article with details. Consequences and explanations of implementing sampling with replacement are provided in section 4. We show our experiment setup and results in section 5, and provide analysis on sampling settings and the problem of inconsistency. Section 6 summarizes the findings of the research and propose future research questions.

2 Background

This section provides additional background for the paper.

2.1 Uncertainty sampling

In uncertainty active learning, there are three main approaches to identify the most uncertain sample [15]. The least confidence method selects the sample with the lowest posterior probability of the class that the sample is assigned to. Margin sampling incorporates the posterior of the second probable class. The sample with the smallest difference between the two posteriors is selected. Entropy is a more general approach since calculating information entropy takes all class labels into consideration. Even though these approaches are based on different intuitions, they produce the same results in binary classification problems. Least confidence is used in this article since it is the most computationally efficient method.

2.2 Logistic regression for binary classification

In this paper, we use logistic regression as a classification method considering it has a linear decision boundary and naturally a probabilistic model. For a binary classification problem with class labels from 1, -1, the following cost function is minimized, considering a L_2 form regularization. C is the regularization parameter. A smaller value of C specifies a stronger regularization. The latter part of the cost function is the sum of logistic loss for a prediction. Furthermore, using logistic regression would have the potential to produce interpretable models. The decision boundary, the estimated posterior probability over dataset distribution, can be directly obtained from the trained classifier.

$$\omega, c = \arg\min_{\omega,c} \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{N} log(exp(-y_i(X_i^T\omega + c)) + 1) \tag{1}$$

2.3 Performance measure in active learning

Error rate or accuracy [2, 6, 16] is a commonly used method for classifier evaluation, in both active learning and supervised learning problems. F1 score, area under ROC curve(AUC) are also used for imbalanced problems and different circumstances [7, 13]. By plotting a learning curve of error rate over running time/number of selected samples, we would be able to evaluate active learning strategies like supervised classification.

Besides error rate, surrogate loss, as a second criterion for performance measure, can be used for plotting learning curves [9]. In logistic regression, logistic loss is directly optimized as a surrogate of accuracy(0-1 loss). Therefore, surrogate loss on a large test set is expected to get smaller as more samples are selected and labeled in supervised classification. It will be a question whether this remains true in active learning. Moreover, there is no guarantee that optimized surrogate loss will lead to good accuracy.

3 Sampling settings in active learning

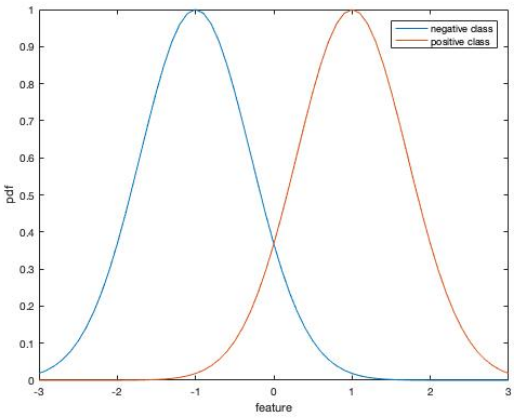

Figure 1: Classification example of two 1D Gaussian distributions

Consider a binary classification task of 2 1-dimensional Gaussian distribution, centered at [1] and [-1] with the

variance of 0.25. The probability density function(fig 1) for each class can be denoted as $P(x|y_-)$ and $P(x|y_+)$. Similar to data collection in practical settings, we can draw a certain number of samples from each class distribution and construct an unlabeled data pool as in pool-based active learning. The available unlabeled pool is denoted as $D_{unlabeled} = \{x_1, x_2, ..., x_n\}$. After annotation, these unlabeled data samples can be utilized as training data for classification. The labeled data set or the training set is denoted as $D_{train} = \{(x_1, y_1), (x_2, y_2), ..., (x_m, y_m)\}$, where y is the class label assigned to the sample. We assume a large amount of independent test set is available for evaluation, and thus the labeled dataset is fully used for training and does not need to be further divided.

The key question is in what manner to select samples and assign labels. In passive learning, samples are selected randomly for annotation. The benefit of passive learning is that no extra system or computation is required to decide which point to label. While in active learning, an extra sample selection process based on a particular sampling strategy(such as uncertainty sampling) is required before labeling one (or one batch of) sample(s). Active learning is usually considered effective for efficient labeling. Usual active learning(AL) and random sampling(RS) are two traditional sampling settings that have been well defined. Each sample can be selected only once in these two settings. As active learning proceeds, the size of the unlabeled pool $D_{unlabeled}$ gets smaller and the labeled pool D_{train} has more training data. Even though uncertainty sampling and active learning could select very different samples, as the available amount of data run out in the unlabeled pool, the choice of possible sample to select becomes smaller. As a result, usual active learning is expected to work well only in the initial stages. In other words, if we decide to select a fixed number of samples, the performance of active learning without replacement is influenced by the size of available samples in the pool. The more data we have in the pool, the more active learning plays its role. For example, in uncertainty sampling, active learners typically choose samples that they are least uncertain about, which are the samples closest to the decision boundary, close to 0 in the above example. The active learners believe the samples closest to the current decision boundary will improve the classifier the most. However, the exact sample to be selected depends on the current available unlabeled data samples. All samples may locate far from the decision boundary. Also, there may be few samples located in the desired area, but after several iterations of active learning, we would have sampled all these samples and have to select samples relatively less close. Another disadvantage is if we label all samples in the data pool, both active learning and random sampling lead to the same result.

A second sampling setting in active learning is sampling with replacement(resampling). The idea of resampling in the unlabeled datapool aims to isolate the effect of active learning from available data to fully utilize the power of active learning. After an active learning iteration, the selected sample is added to the training set with label while the sample remains in the unlabeled data pool. This way of sampling is the so-called sampling with replacement. The motivation to implement sampling with replacement is that it is less affected by the available data samples compared to usual active learning. Sampling with replacement allows querying an infinite number of samples with guarantee that active learning can select the most desirable point without being forced to select the unlabeled but less representative samples. Even though some samples in the desired areas are less represented by the available data, sampling with replacement can reuse the samples.

Following the same intuition, we can expect active learning will reach its best performance if an infinite number of samples are provided and also reused. We define the true active sampling(True AL) scenario for a synthetic dataset as follows. After each training, the classifier reaches its optimal performance based on the current labeled data and then active learner queries the most uncertain sample. In uncertainty sampling, the most uncertain sample will be located closest to the decision boundary of the trained model. This most uncertain sample x_0 is the one located exactly on the current decision boundary of the classifier in the setting of an infinite number of samples. Therefore, a sample(of 1 dimension) located on the decision boundary is generated as the queried sample. We define the label of this sample x_0 as one outcome of a random variable following a Bernoulli distribution, shown in Eq 2, where p is calculated following the Bayesian theorem shown in Eq 3.

$$P_{x_0}(y_+) = p = 1 - P_{x_0}(y_-) \tag{2}$$

$$p = P(y_+|x) = \frac{p(x|y_+)}{p(x|y_+) + p(x|y_-)} \tag{3}$$

4 Consequences of sampling with replacement

Previous work [9] using active learning with replacement to investigate inconsistency found sampling with replacement can lead to increasing surrogate loss. We believe this is due to a combination of 2 reasons: (1)only two samples are queried by active learners after several iterations; (2)As active learning proceeds, the regularization parameter requires adjustment to the fit new training set. Both reasons are related to sampling bias introduced by active learning.

4.1 Sampling between two points

When conducting active learning with these active learners would always end up selecting between two samples. This may seem both expected and strange. As we know, uncertainty sampling likes sampling close to the decision boundary. These final two samples are exactly

the two closest to the decision boundary.How does active learning end up here? The following figure provides a simple example. There is no quantitative approach to measure sampling bias. But when the training set consists mostly of duplicates of two samples, the sampling bias is becoming a problem.

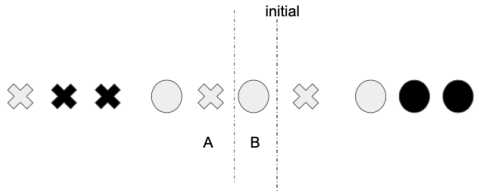

Figure 2: An example illustrating how active learning ends up with selecting between two points. Assume we have four labeled samples(marked with black color) at the beginning of active learning, we could achieve a classifier with the initial classification decision boundary. Considering sample B is the one closest to it, the active learner is very likely to select B for labeling. After adding B to the labeled data set, we get a new classifier with a new decision boundary. After we get this, the active learner will either select sample A or B. If sample A is selected to be labeled data pool, the classification decision boundary shifts a little to the right. And then, the most uncertain sample will be sample B, and thus, the decision boundary moves back. And therefore we can find the decision boundary shifting between these two samples and no longer consider other samples from the data pool.

4.2 Increasing surrogate loss and regularization under sampling bias

A possible reason that the surrogate loss increases are because of regularization term used in logistic regression. Eq 1 is the loss function implemented in "scikit-learn". The loss function consists of two parts, the regularization part and the sum of log loss over all training samples. As the number of selected labeled samples increase, the sum of log loss increases, therefore leaving the effect of the regularization smaller. This means even under the same regularization parameter, the strength of regularization is becoming softer as the number of training samples increase. Normally this would not be a problem in usual classification tasks, which does not have sampling bias, considering that a large number of training samples independent and identically distributed are less likely to overfit the model. However, in active learning with replacement, this is not the case. Training a classifier with a training set consisting of two samples can lead to overfitting and give the active learner blind confidence if the regularization becomes soft.

We replace the "sum" calculation with "average" in the loss function. The loss function is in Eq 4. Under this condition, the strength of regularization does not change during the process of active learning.

$$\omega, c = \arg\min_{\omega,c} \frac{1}{2}\omega^T\omega + \frac{C}{N}\sum_{i=1}^{N} log(exp(-y_i(X_i^T\omega + c)) + 1) \tag{4}$$

5 Experiments

5.1 Implementation details

In this section, we carry out experiments mainly on artificial datasets for binary classification. The aim of using this dataset is to illustrate the performance of sampling under different settings. Artificial dataset is easier to interpret compared to real-world experiments. A logistic regression classifier is used in the active learning pipeline. We use l2-form logistic regression for the classification task and the regularization parameter is set to $1000(\lambda = 0.001)$. We use Scikit-Learn [11] implementation for the classifier and "liblinear" [4] algorithm to find solution to the minimization problem defined in section 2.2. To evaluate the performance of each sampling settings objectively, an independent test set of 20,000 data samples are generated in all experiments. For both artificial dataset, two instances from each class are randomly selected and labeled before the start of active learning. And in each active learning iteration, one single instance is queried for label according to the most uncertain rule. For each experiment, we report the result for each setting averaged on 1000 repetitions.

Two artificial datasets are constructed, including Gaussian dataset as defined in section 3 and the ABA dataset as described in [8]. The ABA data set consists of three Gaussian distribution where one Gaussian distribution is one class located in the middle while the other class is a mixture of two Gaussian distribution on the two sides of the first one(see figure 3). The three Gaussian distribution have the same variance of 0.04 and located at [-1], [0], and [1] respectively.

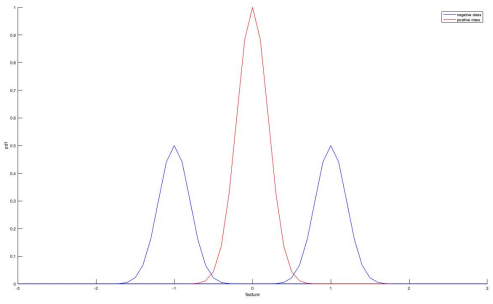

Figure 3: ABA dataset for binary classification

5.2 Performance of different sampling settings

We evaluate the performance of active learning under three different sampling settings.

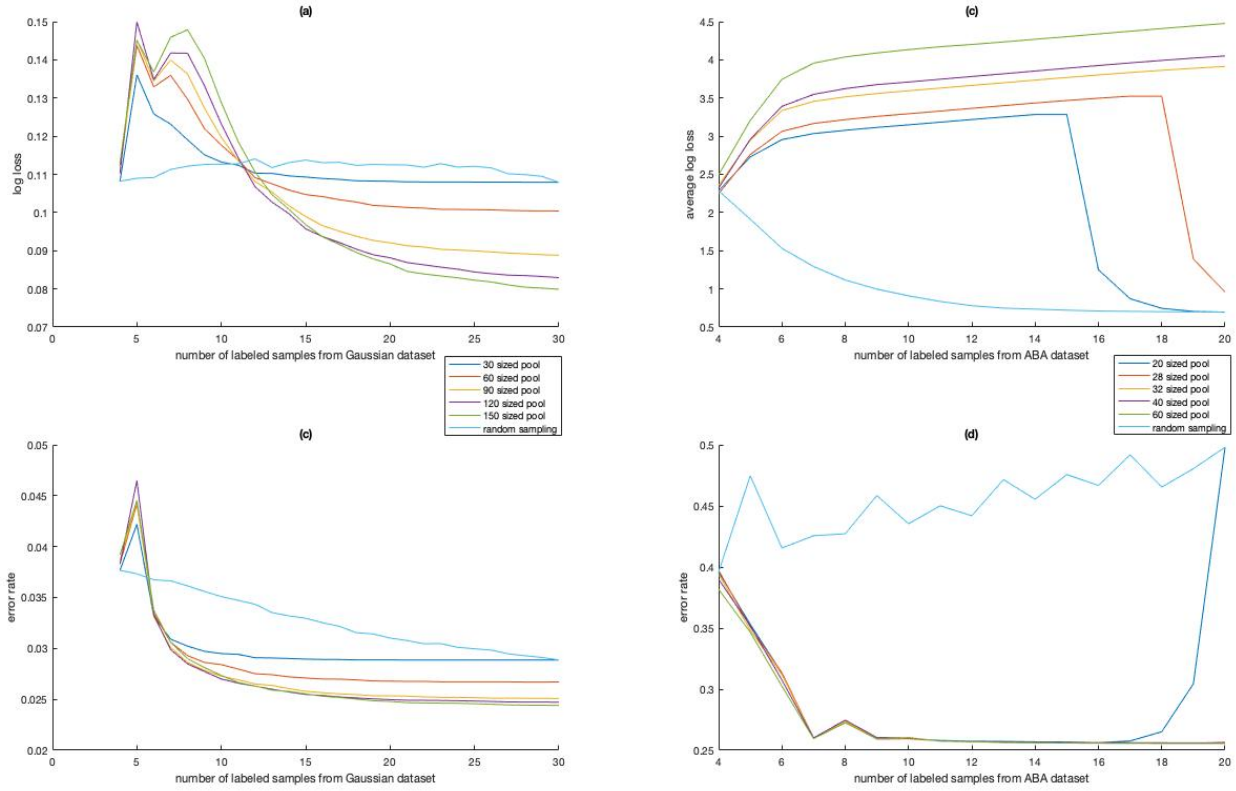

Figure 4: Learning curves of selecting 30(20 for ABA dataset) samples from different pool size. The two figures on the left are the learning curves(log loss and error rate) on Gaussian dataset, while the other two figures on the right side are the learning curves on ABA dataset.

Influence of pool size on usual active learning The first designed experiment is used to illustrate influence of pool size on active learning. Fig 4 shows the performance of conducting active learning under the different sizes of the data pool. In the Gaussian dataset, the learning curves in fig 4.(a) and 4.(c) are plotted based on active sample selection of 30 samples, under a pool of size 30,60,90,120 and 150 respectively. Besides, the results of 30 randomly selected samples are compared with the active learning solutions. In the ABA dataset, we actively select 20 instances from a pool each of size 20, 24, 32, 40 and 60. The learning curves can be seen in fig 4.(b) and 4.(d). In both applications, the performance of active learning under a small pool approaches the performance of random sampling as active learning continues, especially when the pool has the same amount of samples as the number of intended selected samples. If we look at the Gaussian dataset alone, we find that a larger unlabeled pool can achieve better performance in terms of both error rate and surrogate loss. This example shows that active learning has the power to utilize the current available data we have access to. Regardless of the size of the pool, active learning has the ability to find the most informative sample within and thus making labeling more efficient. However, on the other hand, the potential of active learning is not fully explored when only a relatively small number of samples are available. In this sense, evaluating an active learning strategy under a limited number of samples is guaranteed to be influenced by the pool size. In the second example, ABA dataset, what we find strange is that active learning using a larger sized pool leads to a better result in terms of error rate but also a worse result in terms of surrogate loss. The ABA dataset is designed for linear classifiers not to work. Research [8] has shown that the error rate of such a classifier can be as worse as 50%. In this example, active learning with relatively more available data achieve the error rate of close to 0.25. We find that provided two positive samples in the middle and two negative samples each from a cluster, in the very first active learning iteration, the sampler selects an instance from the positive class in the middle. Afterwards, the queried samples are either close to 0.5 or close to -0.5. Therefore only the samples from the two nearby Gaussian distributions are labeled. Sampling only moves to the third Gaussian distribution when all data in the first two Gaussian distributions are labeled. In this example,

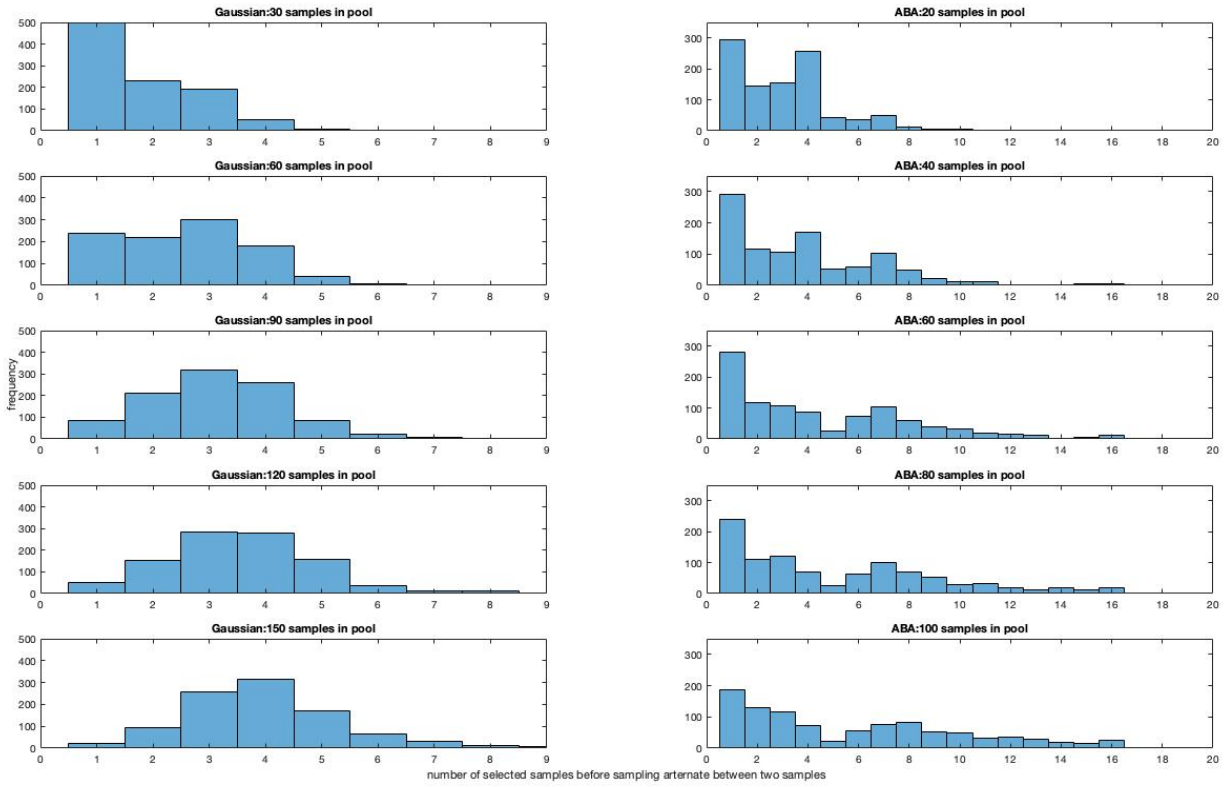

Figure 5: Histograms showing when sampling starts alternate between two samples. Plot on the left is the result of Gaussian dataset while the one on the right is from ABA dataset.

using active learning could lead to a better classifier with a lower error rate sometime. However, this also proves the sampling bias and the inconsistency problem in active learning. We believe if the underlying unlabeled data can be generated infinitely, it is possible that the active learner never go query any instance in the third distribution.

Sampling with replacement As discussed in section 4, there are two main problems sampling with replacement: the increasing surrogate loss and sample selection eventually stuck between two points. In the two artificial datasets, we experimentally confirmed that active learners always queries two opposing samples eventually. Figure 5 are histograms showing when the phenomenon occurs each time under different pool sizes. When the pool size is small, the active learner stops querying other samples at the beginning of active learning. While it happens later, when the pool size gets larger. When the available number of data in the pool goes as large as to 10,000, we do not observe the problem happening in the first 100 active learning iterations. But according to figure 5, we believe if more active learning iterations are continued, it will still lead to the problem happen. Similar to usual active learning, active learning with replacement is also influenced by the pool size. With more collected data, active learning converges to a solution with a lower error rate but higher log loss. Another problem is, when the pool size is small(smaller than 90 in the Gaussian dataset), the performance of active learning with resampling can converge to a result worse than random sampling in terms of error rate. If we look at the surrogate loss, we believe the increasing surrogate loss is a results of applying the same regularization parameter on classification problems with different levels of sampling bias. While in the ABA dataset, learning curves plotted on error rate does not change much when different pool size are provided. While the log loss shows similar performance as Gaussian dataset.

Performance comparison between different sampling settings We compare the performance of true active learning with usual active learning and active learning with replacement using. A pool size of 1000 is used in the two latter sampling settings in order to decrease the effect of available samples on active learning. The result of such comparison on two dataset is shown in fig 7 and fig 8. In the Gaussian dataset, true active learning suffers from increasing surrogate loss, the same problem as active learning with replacement. In this case, this reason of the increasing surrogate loss is also considered less

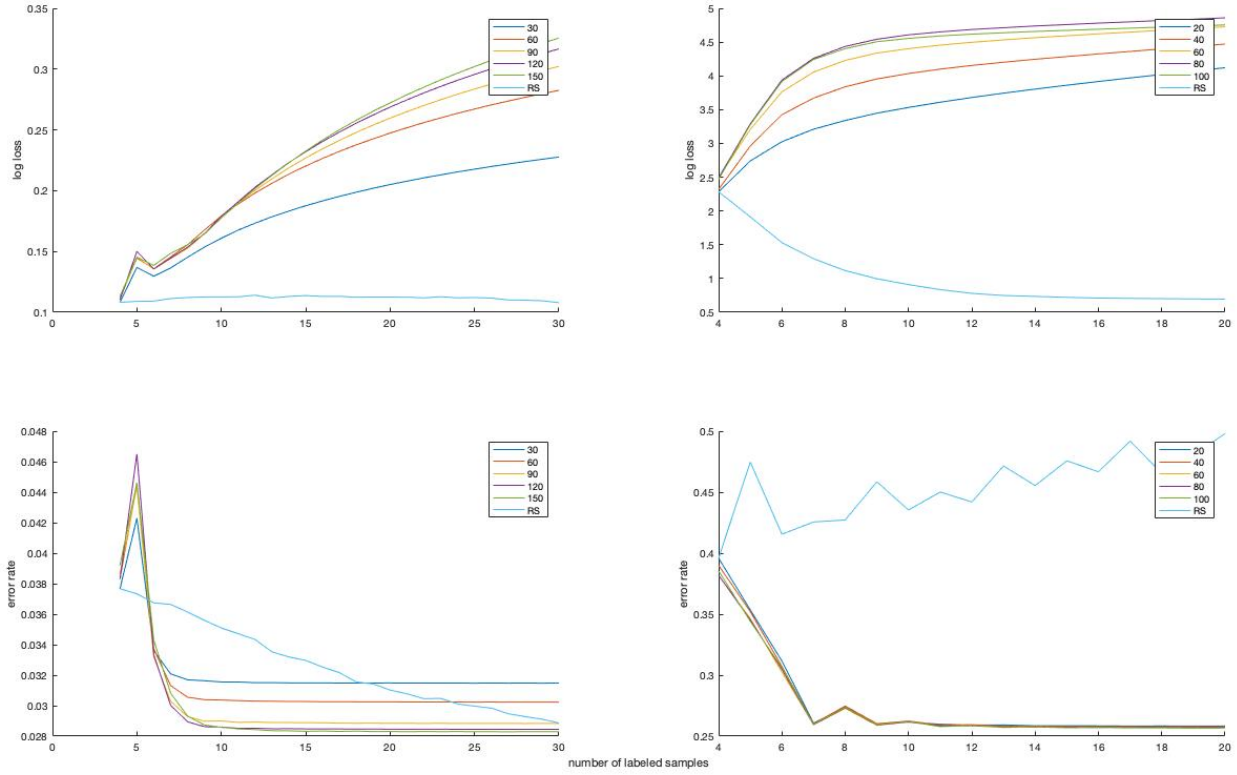

Figure 6: learning curves of selecting 30(20 for ABA dataset) samples from different pool sizes under sampling with replacement. The two figures on the left are the learning curves(log loss and error rate) on Gaussian dataset, while the other two figures on the right side are the learning curves on ABA dataset.

optimal regularization strength and sampling bias. In the Gaussian dataset, true active learning converge to a worst result in terms of error rate, even worse than random sampling. Moreover, both active learning with resampling and true active learning experienced an increasing surrogate loss and thus do not converge the same result as random sampling provides.

5.3 Analyzing inconsistency problems

Each sampling setting suffers from a certain problems and has its disadvantages. Usual active learning is affected by the number of available sample in the pool and cannot really sample until infinity. Active learning with replacement eventually selects between two instances and all other data are overlooked. Also, the performance of sampling with replacement is influenced by the available pool size, even if, theoretically, sampling can be done infinite times. True active learning is not affected by the available data, but it leads to a worse model fit. Despite the fact that sampling with replacement in active learning and true active learning are hard to apply to real-world applications and is less practical, we believe they can be used to provide insight into the inconsistency problem of active learning.

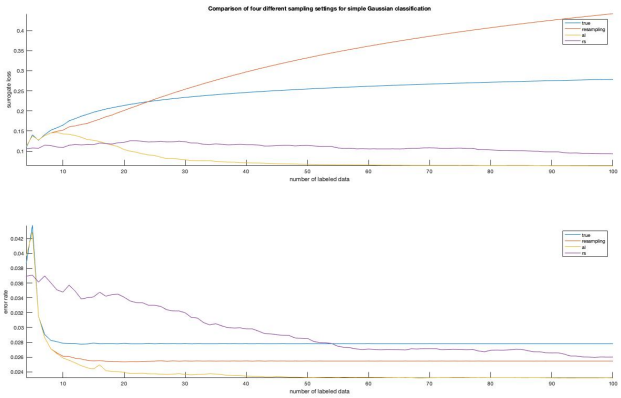

Figure 7: Comparison of three sampling settings on Gaussian dataset

Threats of sampling close to the decision boundary True active learning is designed to be the perfect active learning setting but underperforms compared to other sampling settings as shown in figure 7 and fig-

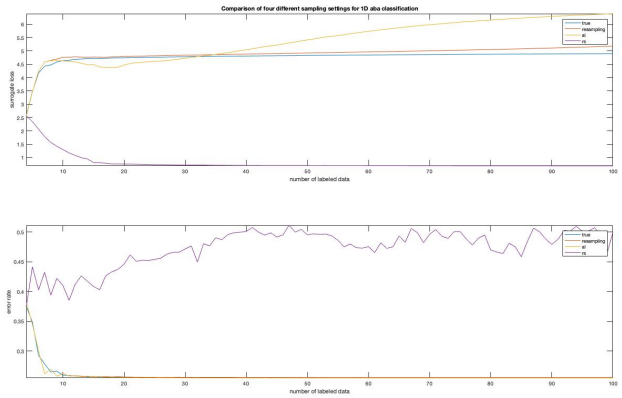

Figure 8: Comparison of three sampling settings on ABA dataset

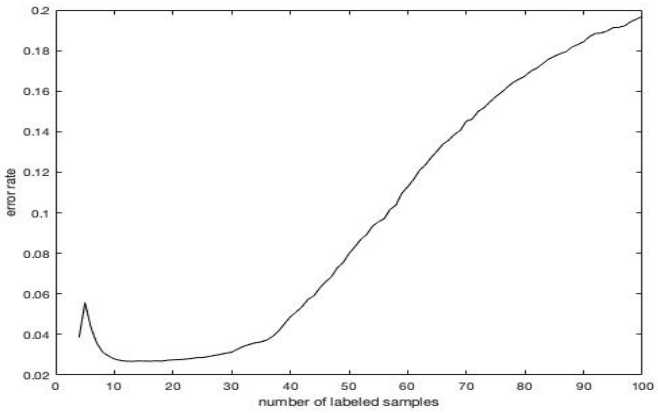

Figure 9: Results of training a classifier using samples near the decision boundary

ure 8 in terms of error rate on two data sets. One extra experiment is carried out to investigate the reason why true active learning converge to a solution worse than random sampling. In this experiment, for the classification task on the Gaussian data set, samples are not queried by an active learner. Instead, each time, we add a sample from the region nearby the decision boundary. More specifically, we randomly sample a data from a Gaussian distribution with mean [0] and variance 0.01 and add this sample to the training set. Similar to true active learning, the label of each selected sample is generated from a correspondent random variable following a Bernoulli distribution. A learning curve based on error rate is shown in figure 9 when 100 samples are selected following this manner. As we can see, adding more training data is giving a worse performance in terms of error rate. A possible reason is that samples close to the decision boundary have a relatively similar probability from each class, i.e, $p(x|y_+) \approx p(x|y_-)$. One thing we observe in the experiment is that many unhelpful samples are selected when acquiring data. The unhelpful samples have a different class label from what a classifier which is trained with sufficient i.i.d. data predicts. We believe sampling unhelpful data does harm to probability estimation and can even give a completely wrong prediction where the error rate can drop to as low as 0.975.

Regularization and sampling bias When the regularization parameter is averaged upon the existing number of labels as in Eq4, we find that the surrogate loss does not deteriorate as shown in fig 10. All three sampling settings have smaller log loss when the log loss is averaged in the loss function. Under random sampling, the learning curve has little difference no matter which loss function is used. While in the other two settings, the problem of inconsistency is not observed when calculating the average of each sample in the loss function. Therefore, we argue that the increasing surrogate loss under sampling with replacement in active learning and true active learning is a result of unmatched regularization strength.

To confirm a higher sampling bias exists in sampling with replacement and true active learning, we try to measure the sampling bias for Gaussian dataset. Prabhu1 [12] proposed to build a separate support vector machine classifier using the same training data. The number of samples used as support vectors represent the level of sampling bias. While in this experiment, we measure the average distance from each sample to the optimal decision boundary("o" in Gaussian dataset). If samples are independent and identically distributed, it is expected that the average distance to point 0 is 1.0. Under sampling bias, more samples are from the region near the decision boundary. Thus the average distance is smaller than 1 in uncertainty sampling. A smaller value represents a higher bias. The average distance under three sampling settings can be seen in table 1. Random sampling produce results without any sampling bias. While all settings of active learning have a relatively smaller average distance and thus a higher sampling bias. From the table, selected samples by active learning with replacement and true active learning do have a high sampling bias. However, it is not clear why usual active learning does not suffer from the increasing surrogate loss.

6 Conclusion

We compare the performance in terms of error rate and surrogate loss of three sampling settings. Through experiments we find that the usual active learning outperforms the other two sampling settings. We show that both sampling with and without replacement cannot simulate infinite number of available samples. while true active learning is closer to an ideal active learning problem with infinite number of samples. The reason why sampling with replacement underperforms is a combination of regularization parameter and repeated sampling. And the fact that true active learning underperforms than usual active learning provides warning about possible dangers

	Average distance	std
Random sampling	1.01	0.47
Usual active learning	0.23	0.20
Active learning with replacement	0.13	0.19
True active learning	0.11	0.20

Table 1: Average distance to point "0" under different sampling settings

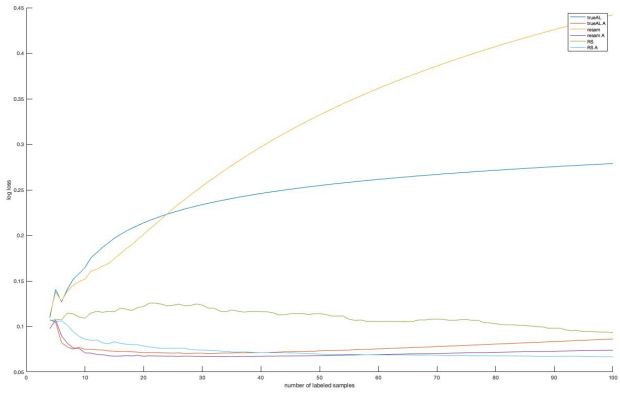

Figure 10: Results of calculating the average of log loss in loss function under three sampling settings. RS A means using average in the loss function under random sampling.

of sampling close to the decision boundary. There are several research questions and directions to be researched upon this paper. To list a few: (1)Can we adjust the regularization parameters according to sampling bias in the training dataset; (2) a deeper investigation into inconsistency problem using sampling with replacement and true active learning; (3) an extension to other sampling strategies; (4) an extension to other classifiers.

References

[1] Charu C Aggarwal, Xiangnan Kong, Quanquan Gu, Jiawei Han, and S Yu Philip. Active learning: A survey. In *Data Classification: Algorithms and Applications*, pages 571–605. CRC Press, 2014.

[2] William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9368–9377, 2018.

[3] Sanjoy Dasgupta. The two faces of active learning.

[4] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *Journal of machine learning research*, 9(Aug):1871–1874, 2008.

[5] Rosa L Figueroa, Qing Zeng-Treitler, Long H Ngo, Sergey Goryachev, and Eduardo P Wiechmann. Active learning for clinical text classification: is it better than random sampling? *Journal of the American Medical Informatics Association*, 19(5):809–816, 2012.

[6] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. *arXiv preprint arXiv:1703.02910*, 2017.

[7] Sheng-Jun Huang, Rong Jin, and Zhi-Hua Zhou. Active learning by querying informative and representative examples. In *Advances in neural information processing systems*, pages 892–900, 2010.

[8] Marco Loog and Robert PW Duin. The dipping phenomenon. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 310–317. Springer, 2012.

[9] Marco Loog and Yazhou Yang. An empirical investigation into the inconsistency of sequential active learning. In *2016 23rd international conference on pattern recognition (ICPR)*, pages 210–215. IEEE, 2016.

[10] Dominic Mazzoni, Kiri L Wagstaff, and Michael C Burl. Active learning with irrelevant examples. In *European Conference on Machine Learning*, pages 695–702. Springer, 2006.

[11] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[12] Ameya Prabhu, Charles Dognin, and Maneesh Singh. Sampling bias in deep active classification: An empirical study. *arXiv preprint arXiv:1909.09389*, 2019.

[13] Wenjun Qiu and David Lie. Deep active learning with crowdsourcing data for privacy policy classification. *arXiv preprint arXiv:2008.02954*, 2020.

[14] Hinrich Schütze, Emre Velipasaoglu, and Jan O Pedersen. Performance thresholding in practical text

classification. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 662–671, 2006.

[15] Burr Settles. Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2009.

[16] Yazhou Yang and Marco Loog. A benchmark and comparison of active learning for logistic regression. *Pattern Recognition*, 83:401–415, 2018.