ALPS: Active Learning via Perturbations

Dani Kiyasseh [1] **Tingting Zhu** [*1] **David A. Clifton** [*1]

Abstract

Small, labelled datasets in the presence of larger, unlabelled datasets pose challenges to data-hungry deep learning algorithms. Such scenarios are prevalent in healthcare where labelling is expensive, time-consuming, and requires expert medical professionals. To tackle this challenge, we propose a family of active learning methodologies and acquisition functions dependent upon input and parameter perturbations which we call Active Learning via Perturbations (ALPS). We test our methods on six diverse time-series and image datasets and illustrate their benefit in the presence and absence of an oracle. We also show that acquisition functions that incorporate temporal information have the potential to predict the ability of networks to generalize.

1. Introduction

The success of modern day deep learning algorithms has been contingent on the presence of large labelled datasets (Poplin et al., 2018; Tomašev et al., 2019; Attia et al., 2019). The time-consuming nature of and high costs associated with labelling data, however, result in troves of rich, yet unlabelled, datasets. This is particularly the case in the medical domain where expert labels are hard to come by. Data-hungry deep learning models can better leverage such datasets via self-supervised learning (Radford et al., 2015; Lee et al., 2019), semi-supervised learning (Zhu, 2005; Beaulieu-Jones et al., 2016), and active learning (Settles, 2009; Otálora et al., 2017; Smailagic et al., 2018; Shao et al., 2018).

Active learning (AL) is the setting wherein a learner has control over the data it uses for training. Such a setting is characterized by the presence of few labelled instances and an abundance of unlabelled instances that are difficult or expensive to label. Typically, in AL the learner is tasked with querying unlabelled instances during training, obtaining their corresponding labels via an oracle (e.g. medical professional), and adding them to the training dataset such that it is capable of generalizing better in a data-efficient manner. Instances are queried using an acquisition function: a metric that determines the degree of informativeness of a particular unlabelled instance (Settles, 2009).

One way of performing active learning is by querying datapoints such that the version space, the set of hypotheses consistent with the labelled training data, is iteratively reduced. Accurately obtaining this version space, however, can be intractable and is therefore approximated. Monte Carlo Dropout (MCD) (Gal et al., 2017) is a technique that applies stochastic binary masks to hidden vectors within a neural network multiple times and which has been shown to be analagous to a Bayesian network (Gal & Ghahramani, 2016). We believe that MCD can also be interpreted as an approximation to the version space where the number of samples roughly represents the number of consistent hypotheses, as seen in Fig. 1a. This approximated version space can then be reduced by querying datapoints that lie in a region of uncertainty, a region where there is high disagreement between the hypotheses about a particular instance. An acquisition function that attempts to quantify this disagreement is that contained within Bayesian Active Learning by Disagreement (BALD) (Houlsby et al., 2011). We claim that acquisition functions used in conjunction with MCD can be limited in their ability to distinguish between the informativeness of two instances, thus hindering their utility.

In the context of healthcare, active learning can alleviate the overall burden associated with labelling data placed on medical experts. However, in the presence of large, unlabelled datasets, the number of requests made to the oracle for labelling data can still be demanding and time-consuming. Most, if not all, prior AL procedures have assumed the presence of an oracle. We argue that a more realistic and independent (yet challenging) AL approach is one that does not have access to an oracle. Such latter scenarios preclude the availability of physicians, and thus naturally extend to large-scale applications where expert medical professionals are scarce.

Our contributions. In this paper, we challenge the commonly-held assumption of the presence of an oracle,

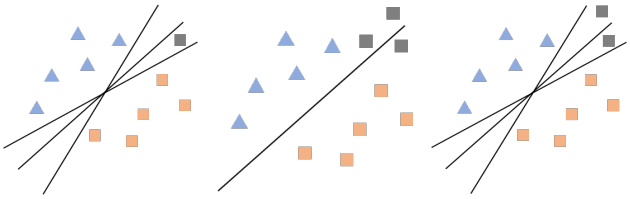
[*]Equal contribution [1]Department of Engineering Science, University of Oxford, Oxford, United Kingdom. Correspondence to: Dani Kiyasseh <dani.kiyasseh@eng.ox.ac.uk>.

and propose several methodologies and acquisition functions:

1. **Area Under the Temporal Acquisition Function** (AUTAF); an acquisition function that incorporates temporal information about the version space.

2. **Monte Carlo Perturbations** (MCP); a novel AL method which stochastically perturbs the input data (Fig. 1b), is a direct substitute for MCD that instead perturbs the hidden vectors of a network, and which can also be used with most existing acquisition functions.

3. **Bayesian Active Learning by Consistency** (BALC); a novel AL method and acquisition function that perturbs both the inputs and the parameters (via MCD) in order to acquire datapoints that the model is least robust towards (Fig. 1c). We will refer to MCP and BALC methods collectively as Active Learning via Perturbations (ALPS).



(a) MCD (b) MCP (c) BALC

Figure 1. Version space of (a) MCD where each MC sample can be viewed as a distinct hypothesis (decision boundary) (b) MCP where there exists one hypothesis but multiple perturbations (gray squares) of the unlabelled instance and (c) BALC where there exist multiple hypotheses in addition to the unlabelled instance and its perturbed counterpart.

2. Related Work

Active learning and healthcare have been relatively underexplored within machine learning. Early work in AL acquires instances using a mixture of Gaussians to minimize the variance of a learner (Cohn et al., 1995; 1996) and a support vector machine (SVM) to reduce the size of the version space (Tong & Koller, 2001). A more complete review of active learning methodologies can be found in Settles (2009). In the healthcare domain, Gong et al. (2019) propose a Bayesian deep latent Gaussian model to acquire important features from electronic health record (EHR) samples in the MIMIC dataset (Johnson et al., 2016) to improve mortality prediction. Also dealing with EHR data, Chen et al. (2013) use the distance of unlabelled samples from the hyperplane in an SVM to acquire datapoints. Wang et al. (2019) implement an RNN with active learning to acquire ECG samples during training. Unlike our work, the latter uses an entropy-based acquisition function. Zhou et al. (2017) propose using transfer learning in conjunction with a convolutional neural network to acquire biomedical images in an online manner. Smailagic et al. (2018; 2019) actively acquire unannotated medical images by measuring their distance in a latent space to images in the training set. Such similarity metrics, however, are sensitive to the amount of available labelled training data. Gal et al. (2017) adopt BALD (Houlsby et al., 2011) in the context of Monte Carlo Dropout to acquire datapoints that maximize the Jensen-Shannon divergence (JSD) across MC samples. Konyushkova et al. (2017) explicitly deal with the *cold-start problem* which is characterized by the poor performance of AL strategies as a result of insufficient initial training data. Most similar to our work is that of (Ducoffe & Precioso, 2018) where sampled instances are associated with the lowest magnitude adversarial attack required to change the network prediction. Unlike our simple Gaussian perturbations, their iterative adversarial attack, is computationally expensive. They also do not explore the combination of input and parameter perturbations, as we do with BALC.

Consistency training has recently emerged in the context of semi-supervised learning in order improve the generalization performance of neural networks. For instance, Interpolation Consistency Training (Verma et al., 2019) enforces a linearity condition on a network by penalizing it for not generating a linear combination of outputs in response to a linear combination of inputs. Unsupervised data augmentation introduced by Xie et al. (2019) perturbs unlabelled samples using image-based data augmentation methods and penalizes networks for generating subsequent drastically different posterior predictive distributions. The intuition is that resultant networks should become more robust to perturbations. In BALC, we acquire unlabelled samples for which the model generates drastically different distributions in response to perturbations. The acquisition function introduced by McCallumzy & Nigamy (1998) uses ensembles of networks and an acquisition function based on the average Kullback-Leibler divergence, $\mathcal{D}_{KL}$, between the posterior predictive distribution from a particular network and the consensus posterior predictive distribution across all networks. Most similar to our work is that of Gao et al. (2019), who use unlabelled data simultaneously to introduce a consistency-loss in the form of a $\mathcal{D}_{KL}$ and actively acquire instances using the variance of the probability assigned to each class by the network in response to perturbed inputs. In contrast, our work perturbs time-series data and proposes a divergence-based acquisition function.

3. Methods

3.1. Area Under the Temporal Acquisition Function

Active learning is known to benefit from improved approximations of the version space (Cohn et al., 1994). One way of doing so is by increasing the number of MC samples: however this becomes computationally expensive for large datasets. Instead, we propose to *track* an acquisition function over time (e.g. epochs) before employing it to acquire instances. In the absence of temporal information, it is difficult to discern between two instances that are currently ranked equivalently yet have been ranked differently in the past. The intuition is that by incorporating temporal information, we are enumerating a larger number and more diverse set of hypotheses. This, in turn, should allow for a better approximation of the version space and thus help to eliminate hypotheses at a greater rate.

In this approach, our temporal acquisition function calculates the area under the trajectory of the acquisition function. For any tracked acquisition function, $\alpha(t)$, the corresponding Area Under the Temporal Acquisition Function (AUTAF $\in \mathbb{R}^1$) is calculated as follows:

$$\begin{aligned} AUTAF &= \int_0^{\tau} \alpha(t)dt \\ &\approx \sum_{t=0}^{\tau} \left(\frac{\alpha(t+\Delta t) + \alpha(t)}{2} \right) \Delta t \end{aligned} \tag{1}$$

where the integral is approximated using the trapezoidal rule, Δt is the time-step between epochs when the acquisition values are calculated, and τ is the epoch at which an acquisition of unlabelled instances is performed.

3.2. Monte Carlo Perturbations

We exploit the idea that instances closer to the decision boundary are likely to be informative for training by stochastically perturbing unlabelled instances and observing the network's outputs. We call such perturbations Monte Carlo Perturbations (MCP). If, for a single instance, the network predictions differ significantly across the perturbations, then such an instance is likely to be in proximity to a decision boundary, as seen in Fig. 1b.

As described, MCP is analogous to MCD in several ways: (1) both deal with perturbations; parameter perturbations (in the form of dropout masks) for MCD, and input perturbations for MCP and (2) both are generic approaches that can accommodate most acquisition functions in the literature e.g., Variance Ratio, Entropy, BALD, etc. Below, we show how to calculate BALD, the JSD between posterior predictive distributions, $p(y|x)$, across *T* MC samples, when used in conjunction with MCP. A more detailed derivation can be found in Appendix B.

$$\begin{aligned} \text{BALD}_{\text{MCP}} &= \text{JSD}(p_1, p_2, \ldots, p_T) \\ &= \text{H}\left(p(y|x)\right) - \mathbb{E}_{p(z|D_{train})}\left[\text{H}\left(p(y|x,z)\right)\right] \end{aligned} \tag{2}$$

where $\text{H}(p(y|x))$ represents the entropy of the posterior predictive distribution averaged across MC samples, and $p(z|D_{train})$ represents the distribution of the perturbations $z \sim \mathcal{N}(0, \sigma^2)$, and where σ was chosen based on the amplitude of the original time-series.

3.3. Bayesian Active Learning by Consistency

Acquisition functions that depend solely on MCD or MCP may still misidentify instances as lying outside the region of uncertainty, and thus erroneously rank them as uninformative. This could be due to improper specification of the perturbations applied. To alleviate this issue and better capture the disagreement set, we propose Bayesian Active Learning by Consistency (BALC) which is motivated by recent work in consistency training whereby models are encouraged to be robust to input perturbations. In BALC, *both* the input and the hidden space of networks are exposed to perturbations, as seen in Fig. 1c. Whereas those applied to the former take on the form of MCP and are deterministic across MC samples, those applied to the latter take on the form of MCD and are stochastic across MC samples.

We propose two acquisition functions to be used in conjunction with BALC: BALC_{JSD} and BALC_{KLD}. Given a network parameterized by ω, the first involves subtracting the Kullback-Leibler divergence D_{KL} between the mean posterior distribution in response to the original data $p(y|x,\omega)$ and its perturbed counterpart $p(y|z,\omega)$ from the average D_{KL} between the respective posterior distributions. The second variant calculates the D_{KL} between two C-dimensional Gaussians, where C is the number of classes, that empirically model the posterior predictive distributions given the original and perturbed data. Further details can be found in Appendix C.

$$\begin{aligned} BALC_{JSD} = \mathbb{E}_{p(\omega|D_{train})}\left[\mathcal{D}_{KL}(p(y|x,\omega) \parallel p(y|z,\omega))\right] \\ - \mathcal{D}_{KL}(p(y|x) \parallel p(y|z)) \end{aligned} \tag{3}$$

$$BALC_{KLD} = \mathcal{D}_{KL}(\mathcal{N}(\mu(x), \Sigma(x) \parallel \mathcal{N}(\mu(z), \Sigma(z))) \tag{4}$$

where $\mu = \frac{1}{T}\sum_{t=1}^{T} p(y|\hat{\omega}_t, x)$ is the empirical mean of the posterior distributions across *T* MC samples and $\hat{\omega} \sim q_\theta(\omega)$ represents parameters sampled from the MC distribution (Gal et al., 2017). $\Sigma = (Y-\mu)^T(Y-\mu)$ is the empirical covariance matrix of the posterior distributions where $Y \in \mathbb{R}^{T \text{x} C}$ is the matrix of posterior distributions for all MC samples.

Algorithm 1 illustrates the BALC procedure with the option of incorporating temporal information shown in blue.

Algorithm 1 Bayesian Active Learning by Consistency

Input: acquisition epochs τ, temporal period Δt, labelled data $\mathcal{L}$, unlabelled data $\mathcal{U}$, network parameters ω, MC samples T, acquisition percentage b
while training **do**
  if epoch in Δt **then**
    for $x \sim \mathcal{U}$ **do**
      $z = x + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2)$
      for MC sample in T **do**
        obtain $p(y|x, \omega)$
        obtain $p(y|z, \omega)$
      end for
      calculate α using eq. 3 or eq. 4
      $\alpha(t) = \alpha$
    end for
  end if
  if epoch in τ **then**
    calculate α using eq. 1
    SortDescending(α)
    $\mathcal{U}_b \subseteq \mathcal{U}$
    $\mathcal{U} \in (\mathcal{U} \setminus \mathcal{U}_b)$
    $\mathcal{L} \in (\mathcal{L} \cup \mathcal{U}_b)$
  end if
end while

4. Experimental Design

4.1. Datasets

Experiments were implemented using PyTorch (Paszke et al., 2019) and were conducted on six diverse datasets. These consist of image and time-series data such as the electrocardiogram (ECG). We use $\mathcal{D}_1$ = PhysioNet 2015 PPG, $\mathcal{D}_2$ = PhysioNet 2015 ECG (Clifford et al., 2015) (5-way), $\mathcal{D}_3$ = PhysioNet 2017 ECG (Clifford et al., 2017) (4-way), $\mathcal{D}_4$ = Cardiology ECG (Hannun et al., 2019) (12-way), $\mathcal{D}_5$ = PTB ECG (Bousseljot et al., 1995) (2-way), and $\mathcal{D}_6$ = CIFAR10 (Krizhevsky et al., 2009) (10-way).

To observe the impact of the availability of labelled training data on the active learning procedure, we take a fraction β = (0.1, 0.3, 0.5, 0.7, 0.9) of the training dataset and place it into the labelled set. Its complement is placed into the unlabelled set. Details about the data splits, preprocessing steps, and the network architecture can be found in Appendices D-E.

4.2. Baselines

Since ALPS proposes novel AL methodologies *and* independent improvements to acquisition functions, we compare it to the state-of-the-art method (MCD) used in conjunction with strongly-performing acquisition functions (such as Var Ratio, Entropy, and BALD). Corresponding definitions can be found in Appendix A. We also compare to the training procedure that does not employ active learning (No AL). In the absence of an oracle, such a baseline will help to illustrate the effect of network-generated noisy labels on AL. In the presence of an oracle where noisy labels are less likely, this baseline will help to illustrate data-efficiency. The No AL performance curves are a function of the fraction, β, chosen and this relationship can be seen in Appendix F.

4.3. Active Learning Hyperparameters

For all time-series experiments, we chose the number of MC samples T = 20 to balance computational complexity and accuracy of the approximation of the version space. Acquisitions of unlabelled instances were performed at pre-defined epochs during training which we refer to as acquisition epochs $\tau = 5n$, $n \in \mathbb{N}^+$. Moreover, the amount of instances acquired during each acquisition epoch is $b = 2\%$ of the remaining unlabelled instances. We also run experiments to see the effect of changing the aforementioned hyperparameter values on performance. Lastly, we chose the temporal period Δt = 1 for all experiments involving temporal variants of acquisition functions. For CIFAR10 experiments, the hyperparameters are T = 5, $\tau = 2n$, and $b = 10\%$.

5. Experimental Results

5.1. Absence of Oracle

Physiological Time-Series. In this section, we conduct experiments in the *absence* of an oracle. In other words, once an unlabelled instance is chosen for acquisition, its corresponding label is obtained via the network prediction. Fig. 2 illustrates the validation AUC of various methodologies implemented on $\mathcal{D}_2$ at β = 0.3.

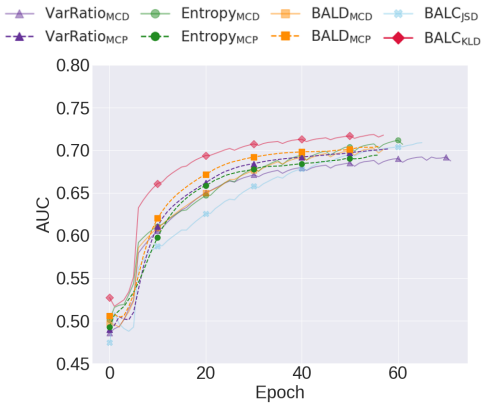


Figure 2. Mean validation AUC for the various methodologies and acquisition functions on $\mathcal{D}_2$ at β = 0.3. BALC and MCP methods are referred to as ALPS in the main text. Results are averaged across 5 seeds and do not include temporal acquisition functions.

We see that the BALC$_{KLD}$ approach leads to faster and improved generalization relative to the others. After 20 epochs,

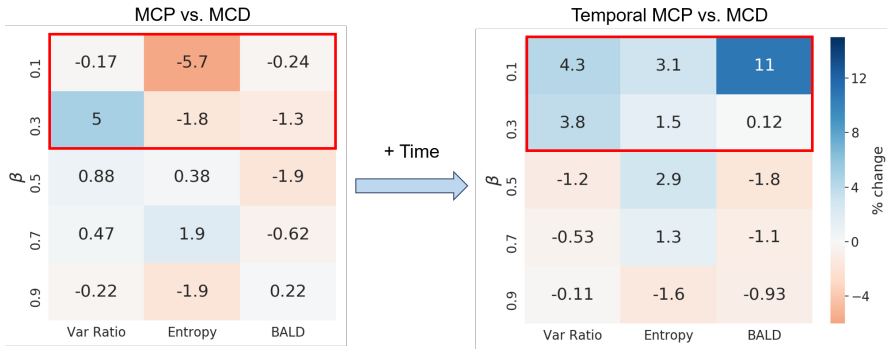


Figure 3. Mean percent change in test AUC when comparing (left) MCP acquisition functions and (right) Temporal MCP acquisition functions to their static MCD counterparts. Results are shown for the acquisition functions variance ratio, entropy, and BALD, and for all fractions, β, of $\mathcal{D}_1$

it achieves the same AUC ($\approx$0.69) as BALD_{MCD} does at epoch 40, representing a two-fold reduction in training time. It also arrives at a higher final AUC ($\approx$0.72) relative to the other methods. We hypothesize that this outcome is due to the implicit separation by BALC_{KLD} of the act of acquisition from that of labelling. Namely, it acquires instances according to the idea of consistency which is not mutually exclusive with network-generated predictions that are correct. More specifically, although acquired instances are *closer* to the decision boundary than their non-acquired counterparts, the former can still lie on the correct side of the boundary. This argument is harder to make with purely uncertainty-based acquisition functions. Nonetheless, to investigate the effect of potentially noisy labels on the AL procedure, we conduct experiments in the presence of an oracle in Section 5.5.

Typically, low values of β in the *presence* of an oracle are expected to lead to the *cold-start problem*; a situation in which networks, initially trained on a labelled dataset that is too small, are too weak to acquire informative unlabelled instances, thus hindering performance. We also happen to observe this phenomenon in our experiments (see Appendix G) and thus refer to it as the *no-oracle cold-start* problem. Surprisingly, the relatively strong performance of BALC_{KLD} in the absence of an oracle and at low β values alludes to its potential utility in overcoming the no-oracle cold-start problem. Furthermore, we show that when acquisition functions are used with MCP (dashed lines), they perform marginally better than their MCD counterparts (solid lines) as seen by their higher AUC values in Fig. 2. For a table of results summarizing the performance of all methods and acquisition functions on all datasets and fractions, please refer to Appendix H.

To quantify better the impact of MCP (our method) relative to MCD (the state-of-the-art), we illustrate in Fig. 3 the percent change in the AUC when using the former method relative to the latter. We also show the added benefit of incorporating *temporal* acquisition functions into the MCP procedure. This benefit is most significant at low β values (red rectangle). For instance, when incorporating temporal information using BALD at $\beta = 0.1$, generalization is improved by 11%. We hypothesize that these improvements are due to a larger and more diverse enumeration of hypotheses in the version space. In addition to the hypotheses enumerated by MCP, those from the past are also considered when incorporating temporal information.

CIFAR10. To convey the robustness of ALPS to various input modalities, we conduct experiments on CIFAR10 and illustrate the validation AUC in Fig. 4. We show that BALC_{KLD} marginally outperforms the remaining methods, as seen by the higher validation AUC throughout training. At epoch 6, it performs on par with $\text{Entropy}_{\text{MCP}}$, achieving an AUC$\approx$0.906. Such a finding corroborates those arrived at with physiological time-series data. Consequently, ALPS has the potential to benefit a diverse set of modalities.

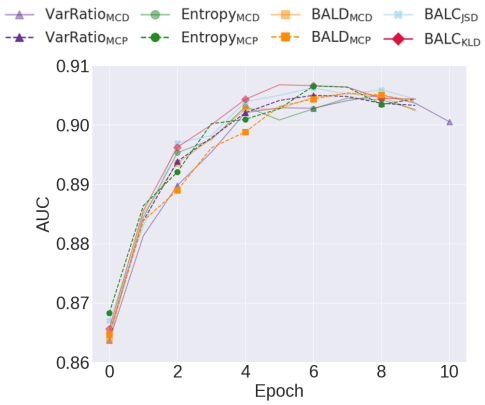
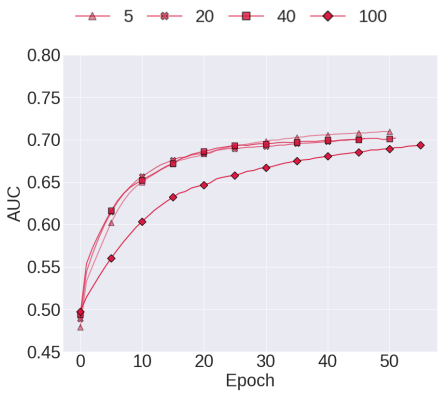
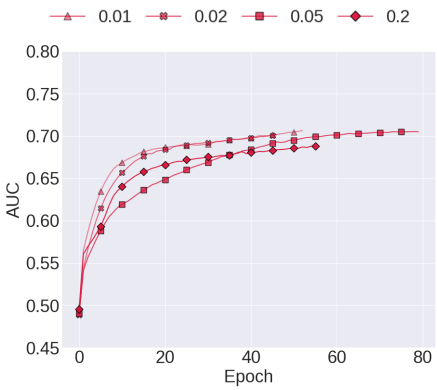


Figure 4. Mean validation AUC for the various methodologies and acquisition functions on $\mathcal{D}_6$ at $\beta = 0.9$. BALC and MCP methods are referred to as ALPS in the main text. Results are averaged across 5 seeds and do not include temporal acquisition functions.



(a) Temporal BALC_{KLD}

Figure 5. Mean validation AUC for range of MC samples, T, on $\mathcal{D}_2$ at $\beta = 0.5$. A higher value of T corresponds to a better approximation of the version space. Results are averaged across 5 seeds.



(a) Temporal BALC_{KLD}

Figure 6. Mean validation AUC for range of acquisition percentages, b, on $\mathcal{D}_2$ at $\beta = 0.5$. A lower value of b implies that fewer instances are acquired during acquisition epochs. Results are averaged across 5 seeds.

5.2. Effect of Number of Monte Carlo Samples, *T*

As the number of MC samples may improve the approximation of the version space, we set out to investigate this effect on the performance of the models in the MCP and BALC scenarios. We choose T = 5-100 and illustrate, in Fig. 5, the mean validation AUC on $\mathcal{D}_2$ at β = 0.5 when using Temporal BALC_{KLD}. The results for the remaining acquisition functions can be found in Appendix I.

We show that, contrary to expectations, increasing the number of MC samples in the range chosen does *not* guarantee improved performance. Such a finding suggests that ALPS can perform sufficiently well with as few as T = 5 MC samples, thus decreasing its computational overhead. Interestingly, the worst performance is exhibited at T=100. This outcome may be explained by the idea that a large number of MC samples may lead to a region of uncertainty that consists of many instances, particularly at early stages of the AL procedure, therefore making it difficult for α to distinguish between the informativeness of instances.

5.3. Effect of Acquisition Percentage, *b*

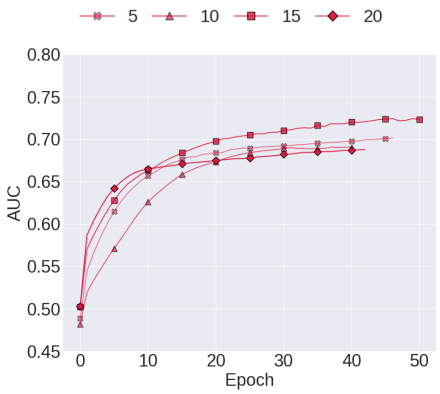
In this section, we perform experiments to observe the effect of the percentage, b, of the unlabelled instances that are acquired on model performance while keeping the number of MC samples fixed at T = 20. We choose b = 1-20% and illustrate the results in Fig. 6. The results for the remaining acquisition functions can be found in Appendix J.

Contrary to expectations that more data is always better, we show that a lower acquisition percentage of b = 1% leads to performance that is greater than that achieved by higher acquisition percentage values, e.g. b = 20% (AUC 0.71 vs. 0.68). There exists a potential twofold explanation. First, acquiring a larger number of instances may introduce distribution shift with respect to that of the current training set that is too extreme for the network to learn from. Secondly, such high acquisition strategies could compound the amount of label noise associated with the unlabelled instances in the absence of an oracle, thereby confounding the learning process.

5.4. Effect of Acquisition Epochs, τ

The degree of label noise in the absence of an oracle can depend on the starting acquisition epoch, τ_{start}, and the subsequent intervals, τ. To investigate this, we conduct experiments and set $\tau_{start} = \tau$, choose τ = 5 (early start and frequent) - 20 (late start and infrequent), and illustrate its effect on the validation AUC in Fig. 7. The results for the remaining acquisition functions can be found in Appendix K.

In Fig. 7, we show that a late start and infrequent acquisitions hinder performance, as seen by the lower final AUC (0.69) achieved by $\tau = 20$ than that achieved with lower acquisition epoch values. Moreover, $\tau = 15$ is the ideal value for this scenario as seen by its highest final AUC (0.73). We hypothesize that these findings are due to the following. Starting acquisition too early means that the network has more control over its training data but may not have been given enough time to correctly classify classes. Similarly, although frequent acquisitions could introduce diversity into the training data and better expose the true data distribution, such a strategy could prevent the network from learning previously acquired datapoints sufficiently well and thus lead to instabilities during training. We term these opposing forces as the control vs. shock trade-off.



(a) Temporal BALC_{KLD}

Figure 7. Mean validation AUC for range of acquisition epochs, τ, on $\mathcal{D}_2$ at $\beta = 0.5$. A lower value of τ implies that acquisitions start earlier and are performed more frequently during training. Results are averaged across 5 seeds.

5.5. Presence of Oracle

The absence of an oracle represents a challenging and scalable scenario (does not require experts) yet may lead to network-generated noisy labels, especially early during training. To investigate the effect of these labels on the AL procedure, we conduct a subset of the original experiments in the *presence* of an oracle and illustrate the results in Table 1. In the last column, we show the ratio of the total number of instances in the best AL scenario to that in the non-AL scenario required to achieve the bolded AUC. Therefore, values less than 100% indicate data-efficiency.

Table 1. Test AUC on $\mathcal{D}_1$ - $\mathcal{D}_5$ at $\beta = 0.1$ in the presence of an oracle. Accuracy is reported for $\mathcal{D}_6$ at $\beta = 0.5$. Complexity indicates the ratio of samples needed to achieve the bolded AUC in the corresponding AL vs. non-AL scenario.

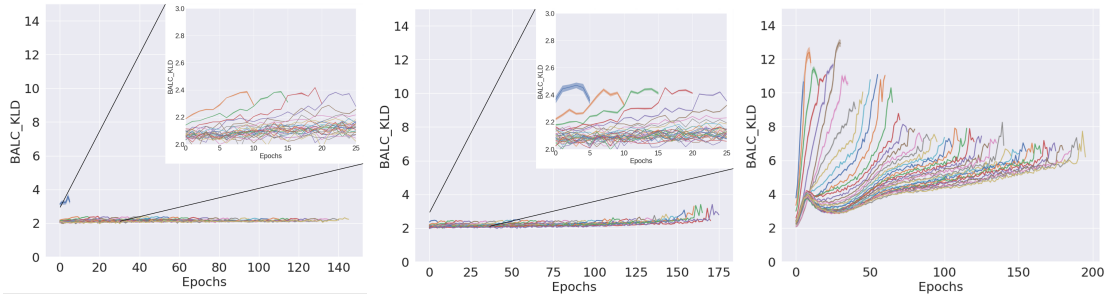
Dataset	Method	Acquisition Metric				Complexity (↓ is better)
		Non-temporal		Temporal		
		BALD	-	BALD	-	
$\mathcal{D}_1$	MCD	0.653 ± 0.013	-	0.643 ± 0.011	-	53%
	MCP	0.676 ± 0.020	-	**0.696 ± 0.029**	-	
	BALC_{JSD}	-	0.693 ± 0.020	-	0.664 ± 0.030	
	BALC_{KLD}	-	0.634 ± 0.030	-	0.659 ± 0.033	
$\mathcal{D}_2$	MCD	0.713 ± 0.053	-	0.706 ± 0.028	-	100%
	MCP	0.735 ± 0.028	-	0.672 ± 0.093	-	
	BALC_{JSD}	-	0.698 ± 0.036	-	0.591 ± 0.101	
	BALC_{KLD}	-	0.722 ± 0.018	-	**0.735 ± 0.011**	
$\mathcal{D}_3$	MCD	0.802 ± 0.008	-	**0.812 ± 0.003**	-	86%
	MCP	0.798 ± 0.007	-	0.787 ± 0.012	-	
	BALC_{JSD}	-	0.789 ± 0.003	-	0.792 ± 0.030	
	BALC_{KLD}	-	0.787 ± 0.008	-	0.794 ± 0.002	
$\mathcal{D}_4$	MCD	0.585 ± 0.011	-	0.573 ± 0.020	-	67%
	MCP	**0.605 ± 0.024**	-	0.555 ± 0.044	-	
	BALC_{JSD}	-	0.574 ± 0.014	-	0.501 ± 0.006	
	BALC_{KLD}	-	0.588 ± 0.033	-	0.532 ± 0.027	
$\mathcal{D}_5$	MCD	**0.937 ± 0.004**	-	0.769 ± 0.103	-	70%
	MCP	0.705 ± 0.013	-	0.901 ± 0.019	-	
	BALC_{JSD}	-	0.917 ± 0.010	-	0.907 ± 0.025	
	BALC_{KLD}	-	0.900 ± 0.036	-	0.708 ± 0.002	
$\mathcal{D}_6$	MCD	**0.596 ± 0.010**	-	0.594 ± 0.009	-	100%
	MCP	0.591 ± 0.009	-	0.593 ± 0.003	-	
	BALC_{JSD}	-	0.590 ± 0.012	-	0.579 ± 0.010	
	BALC_{KLD}	-	0.583 ± 0.013	-	0.584 ± 0.013	

There are several conclusions to be drawn from the results in Table 1. Firstly, AL in the presence of an oracle is either equally or more data-efficient than its non-AL counterpart, as seen by the complexity values that are less than or equal to 100%. Secondly, for $\mathcal{D}_1$ - $\mathcal{D}_3$, we see that the best performing methods are those that incorporate temporal information. However, such information can sometimes be detrimental as seen by the worse performance of Temporal BALC_{JSD} relative to its static counterpart. We hypothesize this to be the case, in the presence of an oracle, because historical information (which is potentially erroneous) should be considered less than that based on the current state of the network. These results suggest that tracking acquisition functions, although does not guarantee improvements, can have a positive impact on generalization. Lastly, by comparing results in the absence and presence of an oracle, we see that the improvement in performance in the latter scenario is greater with MCD than it is with ALPS. Such a finding suggests that, on average, ALPS is more immune to noisy labels than MCD, thus laying the foundations for a family of methodologies that is independent of an oracle.

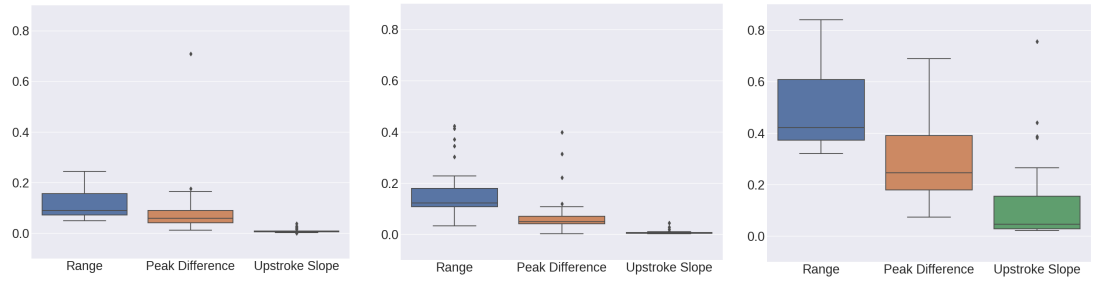
5.6. Behaviour of Temporal Acquisition Function, $\boldsymbol{\alpha}(t)$

In the presence of an oracle, we sought an explanation for the strong but inconsistent performance of certain combinations of temporal acquisition functions and methodologies such as *Temporal* BALC_{KLD}. We did this by tracking the behaviour of the acquisition functions $\alpha(t)$ during training alongside the generalization performance of the models as shown in Fig. 8. The different coloured curves in Fig. 8a represent $\alpha(t)$ averaged across the unlabelled instances that are acquired together at acquisition epochs $\tau = 5n, n \in \mathbb{N}^+$. The apparent dips in the acquisition function values just before instances are acquired can be explained by noise in the function.

The behaviour of *Temporal* BALC_{KLD} suggests that it can be useful in predicting the extent to which a network will generalize in the AL framework. This can be seen in Figs. 8b-8d where the generalization performance of a network exhibits a graded response to notable characteristics of the temporal acquisition function $\alpha(t)$. We identify three such characteristics and calculate them as follows: 1) mean range of values of each upstroke (Range), 2) mean difference in peak values of consecutive upstrokes (Peak Difference), and 3) mean slope of each upstroke (Upstroke Slope). We hypothesize that this behaviour could be indicative of an AL procedure that alters the version space such that it *follows* the remaining unlabelled instances to which it is least consistent. This suggests that the ability of *Temporal* BALC_{KLD} to determine the informativeness of instances does not dissipate during training.



(a) Temporal BALC_{KLD}



(b) Test AUC ≈ 0.66 (c) Test AUC ≈ 0.74 (d) Test AUC ≈ 0.79

Figure 8. (a) *Temporal* BALC_{KLD} values during active learning procedure where each coloured curve represents the average value of $\alpha(t)$ for a distinct set of unlabelled instances that are acquired together at acquisition epochs $\tau = 5n$, $n \in \mathbb{N}^+$. Insets illustrate plots rescaled to show characteristics more clearly. (b)-(d) Statistics derived from the corresponding *Temporal* BALC_{KLD} acquisition function and the associated generalization performance expressed as test AUC. See Sec. 5.6 for definitions used in (b)-(d).

6. Discussion and Future Work

In this paper, we introduced a family of active learning methodologies and acquisition functions, ALPS, that leverages stochastic perturbations to acquire unlabelled instances in the presence and absence of an oracle. We showed that existing acquisition functions employed in conjunction with Monte Carlo Perturbations outperform their Monte Carlo Dropout counterparts. Furthermore, we formulated the Bayesian Active Learning by Consistency framework and illustrated its superiority over prior methods and acquisition functions across diverse datasets. We also showed the potential of this framework when paired with temporal acquisition functions to predict the generalization performance of networks. We now elucidate future avenues worth exploring.

Family of Perturbations. In this work, we limited ourselves to zero-mean Gaussian-distributed perturbations to input samples. Although such simple perturbations were validated on multiple datasets and modalities, they may not have been ideal. An exciting line of research would be to design perturbation generators that can both maximize the informativeness of instances and generalize to different modalities e.g. medical images and videos. These perturbations could be adversarial as in FGSM (Goodfellow et al., 2014) or based on generative adversarial networks (Goodfellow et al., 2014).

Dynamic Hyperparameters. We have showed the effect of altering the hyperparameter values on the generalization performance of networks. These values were then fixed throughout the active learning procedure. An interesting avenue of research would be to design a dynamic hyperparameter strategy that adapts to the online state of the network, somewhat similar to population-based training (Jaderberg et al., 2017).

Explicitly Dealing with Absence of Oracle. We presented active learning in the absence of an oracle and introduced acquisition functions that *implicitly* tease apart the act of acquisition from that of labelling. Due to the potential detrimental effect of noisy labels on the training procedure, a strategy that explicitly alleviates this effect could lead to improved performance and an approach that is truly independent of an oracle.

References

Attia, Z. I., Kapa, S., Lopez-Jimenez, F., McKie, P. M., Ladewig, D. J., Satam, G., Pellikka, P. A., Enriquez-Sarano, M., Noseworthy, P. A., Munger, T. M., et al. Screening for cardiac contractile dysfunction using an artificial intelligence–enabled electrocardiogram. *Nature Medicine*, 25(1):70–74, 2019.

Beaulieu-Jones, B. K., Greene, C. S., et al. Semi-supervised

learning of the electronic health record for phenotype stratification. *Journal of Biomedical Informatics*, 64:168–178, 2016.

Bousseljot, R., Kreiseler, D., and Schnabel, A. Nutzung der ekg-signaldatenbank cardiodat der ptb über das internet. *Biomedizinische Technik/Biomedical Engineering*, 40(s1): 317–318, 1995.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

Chen, Y., Carroll, R. J., Hinz, E. R. M., Shah, A., Eyler, A. E., Denny, J. C., and Xu, H. Applying active learning to high-throughput phenotyping algorithms for electronic health records data. *Journal of the American Medical Informatics Association*, 20(e2):e253–e259, 2013.

Clifford, G. D., Silva, I., Moody, B., Li, Q., Kella, D., Shahin, A., Kooistra, T., Perry, D., and Mark, R. G. The physionet/computing in cardiology challenge 2015: reducing false arrhythmia alarms in the icu. In *2015 Computing in Cardiology Conference*, pp. 273–276, 2015.

Clifford, G. D., Liu, C., Moody, B., Li-wei, H. L., Silva, I., Li, Q., Johnson, A., and Mark, R. G. Af classification from a short single lead ECG recording: the physionet/computing in cardiology challenge 2017. In *2017 Computing in Cardiology*, pp. 1–4, 2017.

Cohn, D., Atlas, L., and Ladner, R. Improving generalization with active learning. *Machine Learning*, 15(2): 201–221, 1994.

Cohn, D. A., Ghahramani, Z., and Jordan, M. I. Active learning with statistical models. In *Advances in neural information processing systems*, pp. 705–712, 1995.

Cohn, D. A., Ghahramani, Z., and Jordan, M. I. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145, 1996.

Ducoffe, M. and Precioso, F. Adversarial active learning for deep networks: a margin based approach. *arXiv preprint arXiv:1802.09841*, 2018.

Gal, Y. and Ghahramani, Z. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *international Conference on Machine Learning*, pp. 1050–1059, 2016.

Gal, Y., Islam, R., and Ghahramani, Z. Deep Bayesian active learning with image data. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1183–1192. JMLR. org, 2017.

Gao, M., Zhang, Z., Yu, G., Arik, S. O., Davis, L. S., and Pfister, T. Consistency-based semi-supervised active learning: towards minimizing labeling cost. *arXiv preprint arXiv:1910.07153*, 2019.

Gong, W., Tschiatschek, S., Turner, R., Nowozin, S., and Hernández-Lobato, J. M. Icebreaker: element-wise active information acquisition with bayesian deep latent gaussian model. *arXiv preprint arXiv:1908.04537*, 2019.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Hannun, A. Y., Rajpurkar, P., Haghpanahi, M., Tison, G. H., Bourn, C., Turakhia, M. P., and Ng, A. Y. Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. *Nature Medicine*, 25(1):65, 2019.

Houlsby, N., Huszár, F., Ghahramani, Z., and Lengyel, M. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

Jaderberg, M., Dalibard, V., Osindero, S., Czarnecki, W. M., Donahue, J., Razavi, A., Vinyals, O., Green, T., Dunning, I., Simonyan, K., et al. Population based training of neural networks. *arXiv preprint arXiv:1711.09846*, 2017.

Johnson, A. E., Pollard, T. J., Shen, L., Li-wei, H. L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L. A., and Mark, R. G. Mimic-III, a freely accessible critical care database. *Scientific Data*, 3:160035, 2016.

Konyushkova, K., Sznitman, R., and Fua, P. Learning active learning from data. In *Advances in Neural Information Processing Systems*, pp. 4225–4235, 2017.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Lee, J., Oh, K., Kim, B., and Yoo, S. K. Synthesis of electrocardiogram V lead signals from limb lead measurement using R peak aligned generative adversarial network. *IEEE Journal of Biomedical and Health Informatics*, 2019.

McCallumzy, A. K. and Nigamy, K. Employing em and pool-based active learning for text classification. In *Proc. International Conference on Machine Learning*, pp. 359–367, 1998.

Otálora, S., Perdomo, O., González, F., and Müller, H. Training deep convolutional neural networks with active learning for exudate classification in eye fundus images. In *Intravascular Imaging and Computer Assisted Stenting, and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis*, pp. 146–154. Springer, 2017.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pp. 8024–8035, 2019.

Poplin, R., Varadarajan, A. V., Blumer, K., Liu, Y., McConnell, M. V., Corrado, G. S., Peng, L., and Webster, D. R. Prediction of cardiovascular risk factors from retinal fundus photographs via deep learning. *Nature Biomedical Engineering*, 2(3):158, 2018.

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

Settles, B. Active learning literature survey. Technical report, University of Wisconsin-Madison, Department of Computer Sciences, 2009.

Shao, W., Sun, L., and Zhang, D. Deep active learning for nucleus classification in pathology images. In *IEEE International Symposium on Biomedical Imaging*, pp. 199–202, 2018.

Smailagic, A., Costa, P., Noh, H. Y., Walawalkar, D., Khandelwal, K., Galdran, A., Mirshekari, M., Fagert, J., Xu, S., Zhang, P., et al. Medal: Accurate and robust deep active learning for medical image analysis. In *IEEE International Conference on Machine Learning and Applications*, pp. 481–488, 2018.

Smailagic, A., Costa, P., Gaudio, A., Khandelwal, K., Mirshekari, M., Fagert, J., Walawalkar, D., Xu, S., Galdran, A., Zhang, P., et al. O-medal: Online active deep learning for medical image analysis. *arXiv preprint arXiv:1908.10508*, 2019.

Tomašev, N., Glorot, X., Rae, J. W., Zielinski, M., Askham, H., Saraiva, A., Mottram, A., Meyer, C., Ravuri, S., Protsyuk, I., et al. A clinically applicable approach to continuous prediction of future acute kidney injury. *Nature*, 572(7767):116–119, 2019.

Tong, S. and Koller, D. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2(Nov):45–66, 2001.

Verma, V., Lamb, A., Kannala, J., Bengio, Y., and Lopez-Paz, D. Interpolation consistency training for semi-supervised learning. *arXiv preprint arXiv:1903.03825*, 2019.

Wang, G., Zhang, C., Liu, Y., Yang, H., Fu, D., Wang, H., and Zhang, P. A global and updatable ecg beat classification system based on recurrent neural networks and active learning. *Information Sciences*, 501:523–542, 2019.

Xie, Q., Dai, Z., Hovy, E., Luong, M.-T., and Le, Q. V. Unsupervised data augmentation for consistency training. 2019.

Zhou, Z., Shin, J., Zhang, L., Gurudu, S., Gotway, M., and Liang, J. Fine-tuning convolutional neural networks for biomedical image analysis: actively and incrementally. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7340–7351, 2017.

Zhu, X. J. Semi-supervised learning literature survey. Technical report, University of Wisconsin-Madison, Department of Computer Sciences, 2005.