AL^2: Learning for Active Learning

Bistra Dilkina **Theodoros Damoulas** **Carla P. Gomes** **Daniel Fink**
Cornell University
Ithaca, 14850, NY, USA
{bistra,damoulas,gomes}@cs.cornell.edu, df36@cornell.edu

Abstract

We introduce AL^2, a pool-based active learning approach that learns how to inform the active set selection. The framework is *classifier-independent*, amenable to different performance targets, applicable to both binary and multinomial classification for batch-mode active learning. Here, we consider a special instantiation, AL$^2_{submodular}$, in which the choice of learning structure leads to a submodular objective function, therefore allowing for an efficient algorithm with optimality guarantee of $1-1/e$. Statistically significant improvements over the state of the art are offered for two supervised learning methods, benchmark (UCI) datasets and the motivating sustainability application of land-cover prediction in the Arctic.

1 Motivation and Related Work

Sustainability research is inherently a predictive science and can be crucially informed by accurate models for e.g. species distributions, land-use and climate change [6, 8]. Consider a predictive model for land-cover in the Arctic that relates ecological covariates to vegetation type. Such a model enables projections of the possible effects of climate scenarios by predicting the future composition of the land cover under drift of the ecological covariates [15]. Predictive accuracy and uncertainty estimates of the model are crucial and depend not only on the model complexity and inherent assumptions but also on the amount and quality of the training data. On one hand, ecological and environmental features such as biomass are readily available from remote sensing data sources. On the other hand though, collecting information on the actual vegetation cover in different parts of the Arctic is an expensive and time-consuming task performed by surveys over areas of large spatial extent. Hence, land-cover survey planning has to be done very carefully, in a targeted way, and with certain constraints in mind. This leads to experimental design and *active learning* (AL); for a comprehensive review, see Settles [18].

In pool-based active learning, one starts with a small training dataset $\mathcal{L}$ of labeled samples and a large pool $\mathcal{U}$ of unlabeled samples. On each iteration the active learner selects one or more samples from $\mathcal{U}$, which are then labeled by an oracle (e.g., a human annotator) and added to the training dataset. The learner then retrains the predictive model and selects more samples for labeling. The goal of active learning is to achieve good performance of the predictive model with as few labeled samples as possible. Most active learning research has focused on *sequential active learning*, in which one greedily selects a single most informative unlabeled sample from $\mathcal{U}$ according to some utility measure. The most commonly used utility measures fall within the family of uncertainty sampling methods such as least confident sampling [3], margin sampling [17], and entropy sampling [21]. Another family of sequential active learning approaches is based on the query-by-committee (QBC) algorithm [20], where active learning selection is based on the disagreement of the committee classifiers about the label of an unlabeled sample. A key limitation of *sequential* active learning is the need for retraining which can be time consuming and in many applications is not even possible due to limited resources and expertise.

In contrast, *batch-mode active learning* selects a set of k ($k > 1$) data points, $\mathcal{A} \subseteq \mathcal{U}$, all at once. Naively extending sequential active learning to the batch-mode setting will not take into account *redundancies* in information and hence can be drastically suboptimal. One ad-hoc way to avoid redundancies is to explicitly incorporate diversity when selecting the batch of samples [2, 19, 22]. A more direct approach to deal with redundancies within the batch is to treat batch-mode active learning as a set optimization problem with an objective defined over possible batch sets. For example, Guo and Schuurmans [11] describe a batch selection method for maximizing a utility function over batch sets that measures the performance of the retrained model as the difference between the log likelihood of the labeled points and the scaled-down sum of the discrete class distribution entropies for the remaining unlabeled points. However, the approach has no optimality guarantees and is strictly tied to binary logistic regression in similar manner to Hoi et al. [12]. Recently, Guo [9] proposed a batch-mode active learning method that maximizes the mutual information between the selected batch and the remaining unlabeled set, but that approach assumes a known Gaussian kernel function and makes no use of the class labels or the predictive model obtained thus far.

In summary, most batch-mode active learning approaches either rely on the use of a specific classification method with a closed-form likelihood and take advantage of available information-theoretic quantities, or are dependent on assumptions about knowledge of the data generation process. Hence, such approaches are not classifier-independent and in particular they cannot be employed with black-box ensemble classification approaches [1, 5, 16], which have proved to be very successful when unknown, highly nonlinear interactions exist—the case for many sustainability-related settings.

Ideally, a batch-mode active learning method should a) be *classifier independent*, b) select a batch as a whole, taking into account *redundancies* among the selected samples, and c) optimize a utility measure that captures the *global informativeness* of the batch, with the aim of producing a classification model with low expected generalization error on future test instances. We propose a novel framework for active learning called *Learning for Active Learning* (AL^2) that meets these requirements. An expanded version with additional results is available as a technical report.[1]

2 AL^2: Learning for Active Learning

Given an index set $\mathcal{L}$ for a set of labeled instances $(\mathbf{x}, y)$, an index set $\mathcal{U}$ for a set of unlabeled instances $\mathbf{x}$, and classifier $\mathcal{F}$, we use a loss function Φ that, for the subsets $\mathcal{A}$ of $\mathcal{U}$, evaluates the predictive model obtained by fitting $\mathcal{F}$ with the training set $\mathcal{L} \cup \mathcal{A}$ in terms of the loss computed over the test set $\mathcal{U} \backslash \mathcal{A}$. In particular, we consider any loss function Φ that is additive over the samples in the test set, that is, $\Phi_{\mathcal{L},\mathcal{U},\mathcal{F}}(\mathcal{A}) = \sum_{i \in \mathcal{U} \backslash \mathcal{A}} \phi(\mathcal{F}, \mathcal{L} \cup \mathcal{A}, \mathbf{x}_i)$, where ϕ is any loss measure computed with respect to the prediction by $\mathcal{F}$ of the label for $\mathbf{x}_i$ when $\mathcal{F}$ is trained on $\mathcal{L} \cup \mathcal{A}$. For fixed batch size k, the optimization problem is to select the batch $\mathcal{A}^* \subseteq \mathcal{U}$ of size k that maximizes $\Delta\Phi$:

$$\begin{aligned} \mathcal{A}^* &= \operatorname{argmax}_{\mathcal{A} \subseteq \mathcal{U} \wedge |\mathcal{A}| = k} \Delta\Phi_{\mathcal{L},\mathcal{U},\mathcal{F}}(\mathcal{A}) \qquad (1) \\ \Delta\Phi_{\mathcal{L},\mathcal{U},\mathcal{F}}(\mathcal{A}) &= \Phi_{\mathcal{L},\mathcal{U},\mathcal{F}}(\emptyset) - \Phi_{\mathcal{L},\mathcal{U},\mathcal{F}}(\mathcal{A}) \\ &= \sum_{i \in \mathcal{A}} \phi(\mathcal{F}, \mathcal{L}, \mathbf{x}_i) + \left[\sum_{i \in \mathcal{U} \backslash \mathcal{A}} \phi(\mathcal{F}, \mathcal{L}, \mathbf{x}_i) - \sum_{i \in \mathcal{U} \backslash \mathcal{A}} \phi(\mathcal{F}, \mathcal{L} \cup \mathcal{A}, \mathbf{x}_i) \right] \end{aligned}$$

Maximizing the objective function $\Delta\Phi$ captures both the local and the global informativeness of the set $\mathcal{A}$, as it corresponds to maximizing the current value of ϕ summed over $\mathcal{A}$, which is to be resolved after labeling $\mathcal{A}$, and simultaneously maximizing the reduction in the value of ϕ over the remaining unlabeled data $\mathcal{U} \backslash \mathcal{A}$. Evaluating $\Delta\Phi_{\mathcal{L},\mathcal{U},\mathcal{F}}(\mathcal{A})$ for all possible batch sets $\mathcal{A} \subseteq \mathcal{U}$ of size k from an unlabeled pool of size $n = |\mathcal{U}|$ would entail retraining the classifier $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ times, which is computationally infeasible for large n. Active learning with batch selection is NP-Hard.

Furthermore, in order to compute the sum $\sum_{i \in \mathcal{U} \backslash \mathcal{A}} \phi(\mathcal{F}, \mathcal{L} \cup \mathcal{A}, \mathbf{x}_i)$, one needs the labels of the samples in the candidate set $\mathcal{A}$, which are not available at the time of optimization. Some previous approaches have addressed this challenge by evaluating $\mathcal{A}$ with "guessed" labels assigned optimistically with respect to their effect on the retrained model [10, 11]. In a new research direction, our framework uses an additional learning structure that predicts the impact of adding a given sample $(\mathbf{x}, y)$ to $\mathcal{L}$ on the loss ϕ of any other sample.

[1] www.cs.cornell.edu/∼bistra/papers/dilkinaetal-AL-techreport2011.pdf

2.1 Density and Agreement Features

In order to select an active learning set, we infer $\phi(\mathcal{F}, \mathcal{L} \cup \mathcal{A}, \mathbf{x}_i)$ on the basis of *uncertainty features* $\widetilde{\mathbf{x}}$. The uncertainty features depend on the unlabeled sample i and on the labeled points in $\mathcal{L}$. We first introduce the nature of the uncertainty features, and then address the regression $\phi(\mathcal{F}, \mathcal{L}, \mathbf{x}_i) \propto \widetilde{\mathbf{x}}(i, \mathcal{L})$.

The first family are *density* related and follow as $\widetilde{x}_m(i, \mathcal{L}) = \sum_{j \in \mathcal{L}:\ |x_d(j) - x_d(i)| \leq \theta_m} 1$. Uncertainty feature $\widetilde{x}_m(i, \mathcal{L})$ captures the number of samples in $\mathcal{L}$ that are within some specified distance θ_m of sample i w.r.t. a particular input feature d among the the D components of the input feature vector. This can be expressed in the form $\widetilde{x}_m(i, \mathcal{L}) = \sum_{j \in \mathcal{L}} \beta_m(i, j)$ where $\beta_m(i, j) = 1$ for points $j \in \mathcal{L}$ such that $|x_d(j) - x_d(i)| \leq \theta_m$, and $\beta_m(i, j) = 0$ otherwise. Similar uncertainty features can be constructed on multiple dimensions and capture different measures of density with respect to the training set $\mathcal{L}$ around each unlabeled point $i \in \mathcal{U}$.

The second family are *agreement* features that capture the location of the decision boundary. They compute the class agreement, defined as the fraction of the most represented class among "neighbors" as: $\widetilde{x}_m(i, \mathcal{L}) = \max_c \frac{\sum_{j \in \mathcal{L}:\ t_j = c \ \wedge\ |\mathbf{x}_j - \mathbf{x}_i| \leq \theta_m} 1}{\sum_{j \in \mathcal{L}: |\mathbf{x}_j - \mathbf{x}_i| \leq \theta_m} 1}$. Intuitively, a higher value of the uncertainty features (such as more neighbors or higher class agreement between neighbors) implies a lower value of ϕ. Hence, having defined the uncertainty features, we impose nonnegativity constraints on the regression coefficients $\mathbf{w}$ of a linear model and use nonnegative least squares to perform inference.

The regression coefficients for the set of density-based uncertainty features $M' \subseteq \{1, \ldots, M\}$, can be used to estimate $\phi(\mathcal{F}, \mathcal{L}, \mathbf{x}_i) - \phi(\mathcal{F}, \mathcal{L} \cup \{j\}, \mathbf{x}_i)$, that is, the impact in the value of ϕ for sample $i \in \mathcal{U}$ by knowing the label of a single additional point $j \in \mathcal{U}$::[2]

$$R(i, j) = \sum_{m \in M'} w_m \beta_m(i, j) \approx \phi(\mathcal{F}, \mathcal{L}, \mathbf{x}_i) - \phi(\mathcal{F}, \mathcal{L} \cup \{j\}, \mathbf{x}_i)$$

Since $\beta_m(i, j) \in \{0, 1\}$ by construction, and $w_m \geq 0$ for all $m \in M'$, then for all $i, j \in \mathcal{U}$, $R(i, j) \geq 0$.

The generalization to the batch-mode case of adding a whole set $\mathcal{A}$ of samples from $\mathcal{U}$ is now just the sum of the single-sample estimates for all $j \in \mathcal{A}$, and to avoid overestimation we bound the impact that the active learning set $\mathcal{A}$ can have on the value of ϕ for the unlabeled point i, $\phi(\mathcal{F}, \mathcal{L}, \mathbf{x}_i) - \phi(\mathcal{F}, \mathcal{L} \cup \mathcal{A}, \mathbf{x}_i)$, by its original value, $\phi(\mathcal{F}, \mathcal{L}, \mathbf{x}_i)$, and obtain the estimate

$$R(i, \mathcal{A}) = \min \left(\phi(\mathcal{F}, \mathcal{L}, \mathbf{x}_i), \sum_{j \in \mathcal{A}} R(i, j) \right)$$

2.2 $\text{AL}^2_{\textit{submodular}}$: Optimizing the Batch-Mode Active Selection

We are now ready to address the issue of optimizing the selection of an active set $\mathcal{A}$. We now have:

$$\Delta\Phi_{\mathcal{L},\mathcal{U},\mathcal{F}}(\mathcal{A}) \quad \approx \quad \sum_{j \in \mathcal{A}} \phi(\mathcal{F}, \mathcal{L}, \mathbf{x}_j) + \sum_{i \in \mathcal{U} \setminus \mathcal{A}} R(i, \mathcal{A})$$

Since $\Delta\Phi_{\mathcal{L},\mathcal{U},\mathcal{F}}(\mathcal{A})$ cannot be optimized directly, we turn to its estimate:

$$\mathcal{A}^* = \text{argmax}_{\mathcal{A} \subseteq \mathcal{U}: |\mathcal{A}| = k} \widetilde{\Delta\Phi}_{\mathcal{L},\mathcal{U},\mathcal{F}}(\mathcal{A}) = \text{argmax}_{\mathcal{A} \subseteq \mathcal{U}: |\mathcal{A}| = k} \sum_{j \in \mathcal{A}} \phi(\mathcal{F}, \mathcal{L}, \mathbf{x}_j) + \sum_{i \in \mathcal{U} \setminus \mathcal{A}} R(i, \mathcal{A}) \qquad (2)$$

We show that the formulation in Eq. 2 is an optimization problem that maximizes a submodular monotonically nondecreasing set function and hence admits a polynomial-time greedy algorithm with optimality guarantees. A set function is submodular if for all A, B with $A \subset B$, and all $x \notin B$, $f(A \cup \{x\}) - f(A) \geq f(B \cup \{x\}) - f(B)$, i.e. it follows the intuitive diminishing returns property. Nemhauser et al. [14] derived the well-known result that for nondecreasing submodular set function maximization, one can use a polynomial-time greedy algorithm to obtain a solution with optimality guarantee of $1 - 1/e$. The overall flow of $\text{AL}^2_{\textit{submodular}}$ is depicted in Fig. 1.

Theorem 1. $\widetilde{\Delta\Phi}_{\mathcal{L},\mathcal{U},\mathcal{F}}(\mathcal{A}) = \sum_{j \in \mathcal{A}} \phi(\mathcal{F}, \mathcal{L}, \boldsymbol{x}_j) + \sum_{i \in \mathcal{U} \setminus \mathcal{A}} R(i, \mathcal{A})$ *is a* ***monotonically nondecreasing submodular set function****.*

[2] Notice that we can compute the values of $\beta_m(i, j)$ for all $i, j \in \mathcal{U}$ in advance and use them in multiple iterations of active learning.

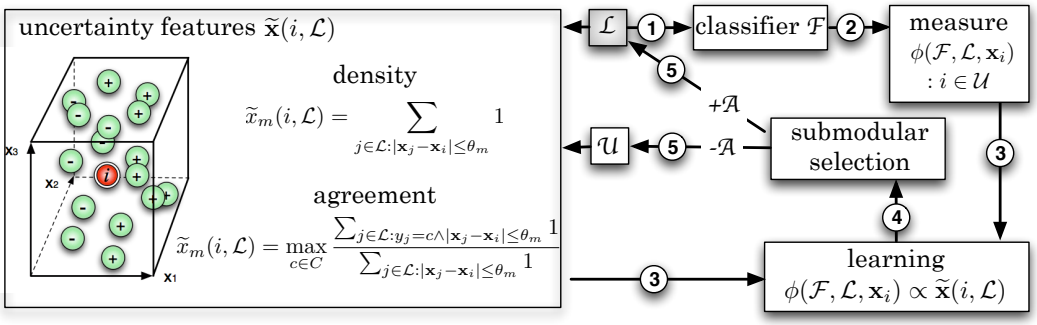

Figure 1: $\text{AL}^2_{submodular}$. 1: training $\mathcal{F}$ on labeled set $\mathcal{L}$. 2: evaluation of ϕ on $\mathcal{U}$. 3: learning ϕ from density and agreement features. 4: selection of set $\mathcal{A}$ by a greedy algorithm applied to a submodular set function. 5: updating of labeled set $\mathcal{L}$ and unlabeled set $\mathcal{U}$ according to $\mathcal{A}$.

3 Experimental Results

We evaluate AL^2 on four of the larger-sized multinomial UCI datasets [7]: *image*, *letter*, *pendigits*, and *landsat*. In addition, we evaluated AL^2 on our motivating application of land-cover prediction in the *Arctic*. Details about the datasets are presented in Table 1. Each of the datasets was randomly split into a query set Q and a holdout set H. Following standard procedure, the initial labeled training set $\mathcal{L}_0$ was a subset of Q that contained two randomly chosen samples from each class; the rest was used as the pool $\mathcal{U}$ of unlabeled samples to choose from. For each dataset, this setup was bootstrapped 10 times. The loss measure ϕ is *margin sampling*, which considers the difference between the probabilities of the two most probable classes for a sample point [17]. $\text{AL}^2_{submodular}$ is compared to random and margin uncertainty sampling under two classifiers: i) multinomial probit kernel regression (KProbit) [4] and ii) a query-by-committee classifier (QBC). The QBC method we used is the query-by-bagging with 20 decision trees as base classifiers. We follow Körner and Wrobel [13] and apply margin sampling to the voting histogram of the base classifiers, interpreted as a posterior probability distribution over the class labels. This has been shown to outperform other disagreement measures for multiclass problems [13] .

Table 1: Dataset statistics: number of classes C, number of features D, size of the query set $|Q|$, size of the holdout set $|H|$. Columns 4–5 apply to each bootstrap set. Columns 6–13 show a comparison of accuracies, based on the 1-sided paired t-test at the 5% level, of three active learning methods (AL^2, Margin Sampling, and Random Sampling) with classifiers KProbit and QBC, using batch size 50 and 8 iterations.

					KProbit				QBC			
					AL^2 vs Marg		AL^2 vs Rand		AL^2 vs Marg		AL^2 vs Rand	
Dataset	C	D	$\lvert Q\rvert$	$\lvert H\rvert$	wins (%)	losses (%)	wins (%)	losses (%)	wins (%)	losses (%)	wins (%)	losses (%)
image	7	18	1500	810	100.0	0.0	75.0	25.0	25.0	0.0	75.0	0.0
letter[a]	26	16	2000	5000	100.0	0.0	50.0	0.0	0.0	0.0	87.5	0.0
pendigits	10	16	2000	8992	100.0	0.0	100.0	0.0	25.0	0.0	87.5	0.0
landsat	6	36	2000	4435	75.0	0.0	87.5	0.0	25.0	0.0	87.5	0.0
Arctic	10	22	4000	6000	100.0	0.0	75.0	0.0	37.5	0.0	62.5	0.0

Table 1 presents statistical significance results comparing the accuracy of the proposed AL^2 method to the accuracies achieved by margin sampling and random sampling, all with batch size 50, 8 iterations of active learning, and 10 independent bootstrap runs per dataset. To compare each pair of methods, we applied a paired t-test at each active learning iteration point. The "wins" column shows the percentage of iteration points where AL^2 outperformed the competing algorithm on the 1-sided paired t-test at the level of $p < 0.05$, and the "losses" column shows the percentage of iteration points at which the competing algorithm outperformed AL^2 on that test. The statistical results in the table demonstrate that our approach yields significant improvement over the competing methods across datasets.

Bibliography

[1] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[2] Klaus Brinker. Incorporating diversity in active learning with support vector machines. In *ICML: International conference on Machine learning*, pages 59–66, 2003.

[3] Aron Culotta and Andrew McCallum. Reducing labeling effort for structured prediction tasks. In *AAAI: National conference on Artificial intelligence*, pages 746–751, 2005.

[4] T. Damoulas and M. A. Girolami. Combining feature spaces for classification. *Pattern Recognition*, 42:2671–2683, 2009.

[5] T. G. Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, 40(2):139–157, 2000.

[6] Thomas G. Dietterich. Machine learning in ecosystem informatics and sustainability. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence. Pasadena, Calif.: IJCAI*, pages 8–13, 2009.

[7] A. Frank and A. Asuncion. UCI machine learning repository, 2010. URL `http://archive.ics.uci.edu/ml`.

[8] Carla Gomes. Computational sustainability: Computational methods for a sustainable environment, economy, and society. *The Bridge, National Academy of Engineering*, 39(4), Winter 2009.

[9] Y. Guo. Active instance sampling via matrix partition. In *NIPS: Conference on Neural Information Processing Systems*, 2010.

[10] Y. Guo and R. Greiner. Optimistic active learning using mutual information. In *IJCAI: International Joint Conferences on Artificial Intelligence*, 2007.

[11] Y. Guo and D. Schuurmans. Discriminative batch mode active learning. In *NIPS: Conference on Neural Information Processing Systems*, 2007.

[12] S. C. H. Hoi, R. Jin, Jianke J. Zhu, and M. R. Lyu. Batch mode active learning and its application to medical image classification. In *ICML: International conference on Machine learning*, pages 417–424, 2006.

[13] Christine Körner and Stefan Wrobel. Multi-class ensemble-based active learning. In *ECML: European Conference on Machine Learning*, pages 687–694, 2006.

[14] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions-I. *Mathematical Programming*, 14(1):265–294, 1978.

[15] Richard G. Pearson, Steven J. Phillips, Pieter S. A. Beck, Michael M. Loranty, Theo Damoulas, and Scott J. Goetz. Arctic greening under future climate change predicted using machine learning. American Geophysical Union (AGU) Fall Meeting, San Francisco, Calif., 2011.

[16] R. E. Schapire. The boosting approach to machine learning: An overview. In *Nonlinear estimation and classification*, Lecture Notes in Statistics, pages 149–172. Springer, 2003. ISBN 9780387954714.

[17] T. Scheffer, C. Decomain, and S. Wrobel. Active hidden markov models for information extraction. In *International Conference on Advances in Intelligent Data Analysis*, pages 309–318, 2001.

[18] B. Settles. Active learning literature survey. Technical Report Computer Sciences 1648, University of Wisconsin–Madison, 2010.

[19] Burr Settles and Mark Craven. An analysis of active learning strategies for sequence labeling tasks. In *EMNLP: Conference on Empirical Methods in Natural Language Processing*, pages 1070–1079, 2008.

[20] H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *COLT: Annual workshop on Computational learning theory*, pages 287–294, 1992.

[21] C. E. Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27:379–423, 1948.

[22] Zuobing Xu, Ram Akella, and Yi Zhang. Incorporating diversity and density in active learning for relevance feedback. In *ECIR: European conference on IR research*, pages 246–257, 2007.