Journal of Machine Learning Research 27 (2026) 1-63 Submitted 11/23; Revised 7/26; Published 7/26

Dirichlet Active Learning

Kevin Miller KSMILLER@MATH.BYU.EDU
Department of Mathematics
Brigham Young University, Provo
Provo, UT 84602, USA

Ryan Murray RWMURRAY@NCSU.EDU
Department of Mathematics
North Carolina State University
Raleigh, NC 27607, USA

Editor: Varun Kanade

Abstract

This work introduces Dirichlet Active Learning (DiAL), a Bayesian-inspired approach to the design of active learning algorithms. Our framework models feature-conditional class probabilities as a Dirichlet random field and lends observational strength between similar features in order to calibrate the random field. This random field can then be utilized in learning tasks: in particular, we can use current estimates of mean and variance to conduct classification and active learning in the context where labeled data is scarce. We demonstrate the applicability of this model to low-label rate graph learning by constructing "propagation operators" based upon the graph Laplacian, and offer computational studies demonstrating the method's competitiveness with the state of the art. Finally, we provide rigorous guarantees regarding the ability of this approach to ensure both exploration and exploitation, expressed respectively in terms of cluster exploration and increased attention to decision boundaries.

Keywords: active learning, graph-based learning, semi-supervised classification, learning theory, uncertainty quantification

1. Introduction

The advent of big data applications in machine learning has necessitated the design of efficient methods to label data for downstream learning tasks such as classification. Massive computing capabilities can produce ever-increasing amounts of data, yet obtaining meaningful labels for training accurate machine learning classifiers is often time-intensive and expensive. Hence, while *labeled data* (i.e., data for which the practitioner has access to observed labels) can be difficult to obtain, *unlabeled data* (i.e., data for which the practitioner does *not* currently have access to such labels) is ubiquitous in many practical applications. While supervised machine learning algorithms rely on the ability to acquire an abundance of labeled data, semi-supervised learning methods leverage both unlabeled and labeled data to achieve accurate classification with significantly fewer labeled data. Simultaneously, the choice of training points can significantly affect classifier performance, especially due to the limited size of the training set of labeled data in the case of semi-supervised learning.

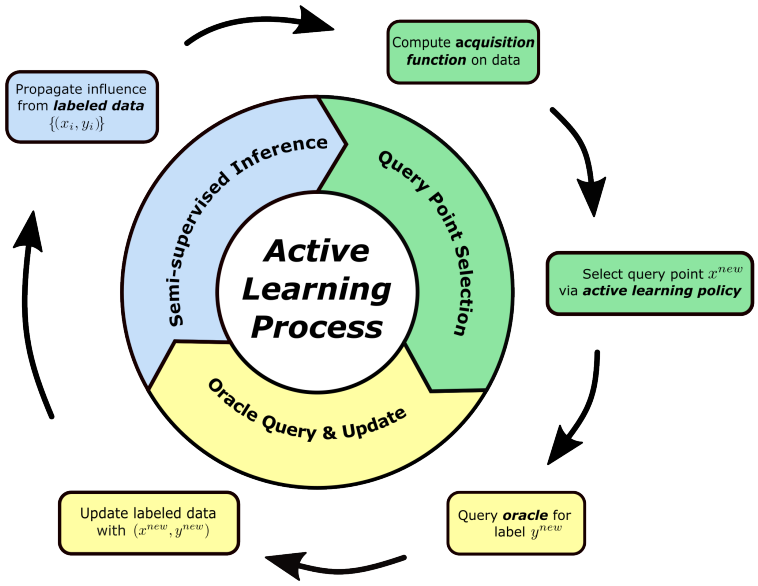


Figure 1: Active learning process flowchart.

Active learning seeks to judiciously select a limited number of currently unlabeled data points that will inform the classification task (Settles, 2012). These points are then labeled by an expert, or human in the loop, with the aim of improving the performance of an underlying classifier. While there are various paradigms for active learning (Settles, 2012), we focus on *pool-based* active learning wherein an unlabeled pool of data is available at each iteration of the active learning process from which query points may be selected. This paradigm is a natural fit for applying active learning in conjunction with semi-supervised learning since the underlying semi-supervised learner also uses the unlabeled pool. These query points are selected by optimizing an *acquisition function* over the discrete set of points available in the unlabeled data pool; an acquisition function is a user-defined quantity that aims to measure the "utility" of expending the effort to label a currently unlabeled input. Figure 1 illustrates the active learning process that iterates between (1) using currently-labeled data to infer a classifier on the unlabeled data, (2) selecting the next query point from the unlabeled data via the acquisition function, and (3) labeling the query point and updating the labeled data. In practical applications, the labeling "oracle" is a domain expert that plays the role of a human in the loop. The number of points that are labeled at each iteration depends upon the application, but in this work, we focus on the setting where points are labeled one at a time (i.e., sequential active learning).

Active learning may have a number of different goals, such as ensuring high classifier accuracy relative to the number of query points or ensuring sampling from all parts of the feature distribution. Part of the challenge in doing so is that we need an appropriate vocabulary, as well as computational experiments, to understand tradeoffs between different goals in active learning. This paper seeks to develop this vocabulary and the associated computational experiments and does so while proposing a novel active learning method, which we call *Dirichlet Active Learning (DiAL)*.

1.1 Setup and Main Contributions

We assume that we have access to a set of inputs $\mathcal{X} \subset \mathbb{R}^d$ which are sampled from an underlying data-generating distribution with density $\rho : \mathbb{R}^d \to [0, \infty)$. While in application, this set $\mathcal{X}$ represents a discrete set of points, we will also consider the continuum case where $\mathcal{X}$ represents the support of ρ. We seek to classify these inputs $x \in \mathcal{X}$ into $K \geq 2$ classes, and we assume access to an initially labeled set $\mathcal{L} = \{x^1, \dots, x^\ell\} \subset \mathcal{X}$ for which we have observed $y^j \in [K] = \{1, 2, \dots, K\}$ for $x^j \in \mathcal{L}$. Let $\mathcal{Y}_\mathcal{L} = \{y^1, \dots, y^\ell\}$ denote the corresponding set of labels (classes) for the labeled set $\mathcal{L}$. The task in active learning will be to iteratively augment the labeled data set $\mathcal{L} \subset \mathcal{X}$ by cycling between (1) learning a semi-supervised learning classifier given current labeled data $\mathcal{L}$ with associated labels y^j for $x_j \in \mathcal{L}$, (2) selecting a query point x^* from the unlabeled data, $\mathcal{U} = \mathcal{X} \setminus \mathcal{L}$, and (3) labeling said query point and adding it to the labeled data accordingly. As such, this process generates a sequence of labeled sets $\mathcal{L}^{(0)} \subset \mathcal{L}^{(1)} \subset \mathcal{L}^{(2)} \subset \dots$ as query points are labeled and added to the labeled set; we refrain from this iteration-explicit notation in favor of readability and denote the (changing) labeled set simply as $\mathcal{L}$.

Active learning serves as a naturally complementary problem to semi-supervised learning. At a high level, semi-supervised learning seeks an answer to the question

> *How do we accurately infer the classification of the unlabeled data $\mathcal{U}$ given the currently labeled data $\mathcal{L}$?*

In a complementary fashion, active learning seeks to answer the question

> *How do we judiciously select currently unlabeled points $x \in \mathcal{U}$ that, once labeled, can maximally improve performance in the underlying semi-supervised problem?*

An implicit constraint is to perform as few queries as possible; in practical applications, labeling of query points may be costly or time-consuming, and so an overall budget of queries may be explicitly imposed. Hence, a goal of active learning is to design methods which are provably efficient at selecting the query points that are the most useful for the underlying semi-supervised learning task.

One important aspect of active learning is the inherent tradeoff between using the limited resource of queries to either (i) *explore* the extent of the given data set or (ii) *exploit* the current classifier's inferred decision boundaries. Exploration in active learning is often characterized by querying points from each of a set of salient regions (clusters) in a data set. In contrast, exploitation in active learning is characterized by querying points where the current classifier predicts there to be a decision boundary in order to refine and improve the accuracy of that boundary. Similar to the exploration-exploitation tradeoff in reinforcement learning (Sutton and Barto, 2018; Agarwal et al., 2021), it is crucial to properly balance between exploration and exploitation to ensure the greatest benefits from active learning. In the current work, we not only provide cluster exploration guarantees for our proposed method but also give a novel asymptotic analysis of its capability to efficiently exploit along decision boundaries during later stages of the active learning process.

The terminology for the mechanism by which one selects query points is not uniform across the literature, and we accordingly suggest the following framework. The active learning query points are selected at each iteration via the combination of what we will term

an *acquisition function* and a *policy*.[1] We define an acquisition function $\mathcal{A}(\cdot, \mathcal{L}, \mathcal{Y}_\mathcal{L}) : \mathcal{U} \to \mathbb{R}$ to be a user-defined criterion to quantify the utility of expending the effort to label a point $x \in \mathcal{U}$. This acquisition function is dependent on the currently labeled data $\mathcal{L}, \mathcal{Y}_\mathcal{L}$, though for ease of notation we drop the explicit dependence and write $\mathcal{A}(\cdot) = \mathcal{A}(\cdot; \mathcal{L}, \mathcal{Y}_\mathcal{L})$. Indeed it may be said that the "art" of active learning primarily depends on the choice of acquisition function to reflect the desired properties of the selected query points. With a chosen acquisition function it remains to decide how to select the next query point $x^* \in \mathcal{U}$ based upon the acquisition function values on the unlabeled data, $\{\mathcal{A}(x)\}_{x \in \mathcal{U}}$. We refer to the method for selecting the query point from said values as the *active learning policy*. For example, a common policy is to select the maximizer, $x^* = \operatorname{argmax}_{x \in \mathcal{U}} \mathcal{A}(x)$, or minimizer $x^* = \operatorname{argmin}_{x \in \mathcal{U}} \mathcal{A}(x)$ depending on the specific properties of the chosen acquisition function. We also consider a proportional sampling policy that fits naturally into our theoretical results (Section 5) and is beneficial in our experiments (Section 4).

A novel idea of the present work is to model the influence of the labeled data on the unlabeled data via what we will term a "Dirichlet random field". Random fields (i.e., collections of random variables $\{F_x\}_{x \in \mathcal{X}}$ indexed by elements in (a subset of) a topological space) are convenient structures for statistical processes, such as active learning. For example, Gaussian random fields are collections of random variables where the joint distribution of any finite subset of the random variables is distributed as a multivariate Gaussian (see Rasmussen and Williams, 2006). Gaussian random fields and Markov random fields are commonly used in the active learning literature to define acquisition functions that reflect measures of uncertainty or variance in the underlying statistical model (Zhu et al., 2003b; Jun and Nowak, 2018; Ji and Han, 2012; Schreiter et al., 2015; Riis et al., 2022; Kapoor et al., 2007; Krause and Guestrin, 2007). Graph-based random fields are an especially relevant case (Zhu et al., 2003b; Jun and Nowak, 2018; Ji and Han, 2012; Ma et al., 2013; Qiao et al., 2019; Miller et al., 2020), wherein the index set $\mathcal{X}$ is finite and the correlation structure is determined by the connectivity structure of an associated graph $G(\mathcal{X}, W)$. Various previous graph-based methods for semi-supervised and active learning have been framed in the context of Gaussian random fields, where the correlation structure between the outputs at nodes (i.e., inputs $x \in \mathcal{X}$) follows a Gaussian distribution influenced by the connectivity structure of a graph Laplacian matrix (Zhu et al., 2003a,b; Bertozzi et al., 2018; Qiao et al., 2019) constructed from the set of inputs. While this approach yields a straightforward Bayesian interpretation of the associated semi-supervised and active learning problems, *the inherent modeling assumption is that of regression, not classification.*

Elaborating further on this point, from a "strictly" Bayesian perspective, the posterior predictive distribution at any given point in a Gaussian random field is inherently supported outside of the probability simplex, regardless of whether all labeled points' observations lie at the corners of the probability simplex. As one uses a measure of variance from Gaussian random fields to prioritize query points in active learning, these variance estimates are inherently influenced by values outside of the sensible domain for classification tasks. Our proposed Dirichlet random field approach, however, constructs a posterior distribution such that the marginal distribution at each $x \in \mathcal{X}$ is supported on the probability simplex in an effort to produce variance estimates that are more faithful to the classification setting.

1. The term "acquisition function" appears in various references such as (Gal et al., 2017).

The details of the proposed semi-supervised model based on a Dirichlet random field are given in Sections 2 and 3, but we briefly summarize here: we model the classification of each $x \in \mathcal{X}$ via a categorical random variable $Y(x) \sim Cat(P(x))$. Inspired by the classical Bayesian approach, we express our current information about the probability vector P as a Dirichlet random variable

$$P(x) = (p_1, \ldots, p_K) \sim Dir(\alpha_1(x), \alpha_2(x), \ldots, \alpha_K(x)).$$

The Dirichlet posterior belief at each point $x \in \mathcal{X}$ is updated according to the amount of information that is propagated from labeled set $\mathcal{L}$ according to the respective classes. We call this information a "pseudolabel", as the α_i parameters of the Dirichlet distribution are interpreted as observed labels in a classical Bayesian context, whereas in the low-label learning context we must resort to treating observed labels at $x' \sim x$ as proxies for label observations at x. Using the vector P we can construct a semi-supervised learning classifier which we call *Dirichlet Learning*. In the graph-based setting, the propagation operator is defined via the graph Laplacian matrix.

As a result of the Bayesian-inspired approach of Dirichlet Learning, the variance of the current estimate of the categorical probabilities at each unlabeled point is readily computable. This variance quantifies classifier uncertainty which we use as an active learning acquisition function. We term this acquisition function *Dirichlet Variance*.

For the sake of clarity, we summarize our proposed method *Dirichlet Active Learning (DiAL)* in Algorithm 1. In addition, we provide Figure 2 to contextualize DiAL as a realization of the active learning loop from Figure 1 with (i) Dirichlet Learning used as the classifier for Semi-supervised Inference and (ii) Dirichlet Variance used as the acquisition function for Query Point Selection.

Having described the context, we now briefly summarize the main contributions of the present work as follows.

1) **Novel semi-supervised learning classifier (Dirichlet Learning) for use in efficient active learning.**

   We introduce a novel semi-supervised learning classifier (Dirichlet Learning) that models the inferred classifications on unlabeled data via a Dirichlet random field. An intuitive measure of "uncertainty" (Dirichlet Variance) is readily available for this classifier, and we find this quantity to be an informative acquisition function for both exploration and exploitation in active learning. The proposed Dirichlet Learning (classifier) and DiAL (active learning) frameworks are flexible and easily adapted to any desired propagation operator from labeled data to the rest of the data set. We specialize the framework to graph-based propagations that are very useful for capturing the clustering structure of the underlying data set. These graph-based propagation operators are naturally defined on the discrete graph structure, but also conveniently have intimate connections to continuum-limit second-order elliptic operators that depend on the underlying data-generating distribution.

2) **Theoretical guarantees for exploration of clustering structure by DiAL.**

   We establish theoretical guarantees that DiAL will initially prioritize the selection of query points in salient regions ("clusters") of the domain (Section 5.3). Under

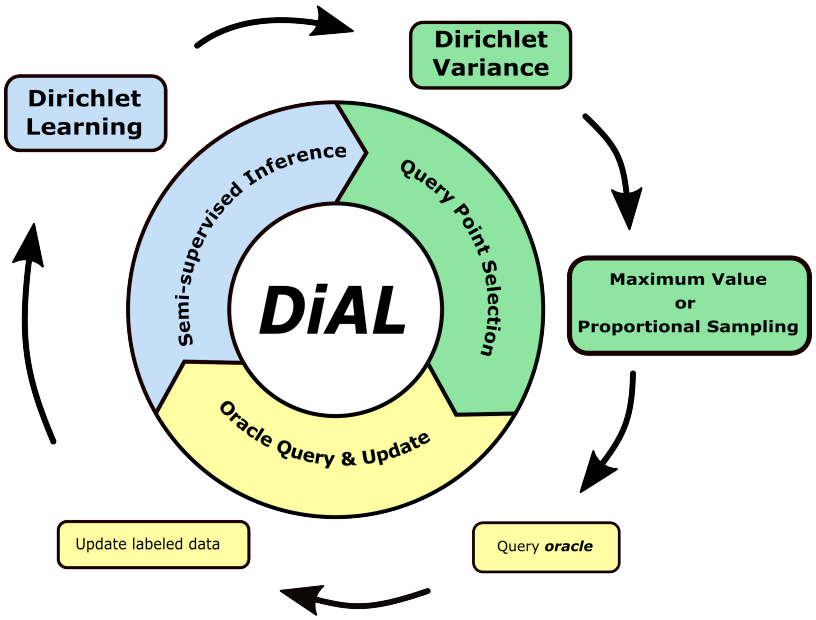


Figure 2: Utilizing the *Dirichlet Learning* semi-supervised classifier introduced in Section 2, *DiAL* selects query points using the *Dirichlet Variance* acquisition function (9) with either the *Maximum Value* or *Proportional Sampling* policies (Section 3).

some simple assumptions about the choice of propagation operator as it relates to the underlying data distribution, we demonstrate that K active learning queries are sufficient to ensure the labeling of points in K clusters of potentially varying sizes.

3) **Novel asymptotic analysis of the later stages of active learning.**

We provide an asymptotic analysis of DiAL in the later stages of the active learning process (Section 5.5). We have not seen previous work, particularly in the graph-based active learning literature, that characterizes the late-stage asymptotics of active learning. This novel analysis elucidates that, if scaled properly, Dirichlet Variance as an acquisition function can lead to querying in regions of the data set where multiple class-conditional probabilities are high—i.e., where there is inherent population-level uncertainty in the underlying data distribution. We suggest this is a natural characterization of beneficial exploitation in active learning.

4) **Computational efficiency of the proposed method.**

We empirically verify the efficacy of DiAL when utilizing a graph-based Dirichlet Learning classifier to explore data set clustering structure in real-world data sets, including a comparison to previous methods on hyperspectral imagery (HSI) pixel classification. We also highlight the very favorable computational complexity of the acquisition function, comparing it to previous graph-based active learning acquisition functions. See our GitHub repository (`https://github.com/millerk22/dirichlet-active-learning.git`) for a Python implementation of DiAL that is (i)

Algorithm 1 Dirichlet Active Learning (DiAL)

Input: data set $\mathcal{X} = \{x_1, \dots, x_n\} \subset \mathbb{R}^d$, # classes K, initially labeled data $\mathcal{L}_1, \dots, \mathcal{L}_K$, propagation operator $\mathcal{K}(\cdot, \cdot)$, active learning budget $B \in \mathbb{N}_+$
Output: Classification predictions $\{\hat{y}(x)\}_{x \in \mathcal{X}}$
$A \leftarrow \alpha \mathbb{1} \in \mathbb{R}^{n \times K}$, $(A_{ik} = \alpha_0$ for all $i \in [n], k \in [K])$ ▷ Dir. hyperparam. matrix
for $k \in [K]$ **do** ▷ Propagate "pseudo-label" from each $\mathcal{L}_k$
  $A_{i,k} = \sum_{x^\ell \in \mathcal{L}_k} \mathcal{K}(x^\ell, x_i)$ for $i \in [n]$
end for
$v_i \leftarrow Var(A_{i,:})$ for each $i \in [n]$ ▷ Initial variances from rows of A
$b \leftarrow 1$
while $b < B$ **do**
  if policy = 'MV' **then**
    $i_* \leftarrow \arg\max_{i \in [n]} \; v_i$ ▷ Max. Value policy
  else
    Sample i_* according to $\mathbb{P}(I_* = i) = \frac{e^{\lambda v_i}}{\sum_{j=1}^n e^{\lambda v_j}}$ ▷ Prop. Sampling policy
  end if
  Query label $y_{i_*} = k_*$ for x_{i_*}
  $\mathcal{L}_{k_*} \leftarrow \mathcal{L}_{k_*} \cup \{x_{i_*}\}$ ▷ Update labeled data
  $A_{i,k_*} \leftarrow A_{i,k_*} + \mathcal{K}(x_{i_*}, x_i)$ ▷ Update Dirichlet hyperparameter matrix
  $b \leftarrow b + 1$
end while

specialized to the graph-based semi-supervised classification setting and (ii) integrated into the GraphLearning (Calder, 2022) Python package.

5) Development of active learning vocabulary and associated computational tests.

We provide various discussions throughout that seek to clarify important ideas that are not necessarily original to this work, but we believe help to elucidate the mechanisms underlying active learning. For example, in Appendix C we discuss two distinct types of "uncertainty" in active learning (population-level and data-conditional uncertainty) and provide an illustrative example that demonstrates the importance of designing acquisition functions that ultimately reflect population-level uncertainty. We suggest that the asymptotic analysis of exploitation provides an important avenue for continued research in active learning methods. Finally, we introduce the terminology of an *active learning policy*. While acquisition functions provide a concrete quantity to reflect the utility of labeling a currently unlabeled point, an active learning policy identifies how these acquisition function values are ultimately used to select the next query point. Traditionally, the maximizer of the acquisition function on the unlabeled data is chosen to be labeled; we find that sampling proportional to the acquisition function values is a useful active learning policy for both our numerical and theoretical results.

Finally, to conclude our introduction, we remark that this work utilizes ideas from several different fields, including active learning, graph-based learning, and PDE theory in machine learning. While we provide the most immediately relevant references throughout the text, in order to make the main results more directly accessible, we delay a more thorough accounting of the related literature to Appendix A.

1.2 Summary of Notation

We here briefly summarize various notational conventions used throughout the paper for the reader's convenience. We will generally use caligraphic capital letters (e.g., $\mathcal{X}, \mathcal{L}$) to denote sets, with the lone exception of $\mathcal{K}(\cdot,\cdot)$ to denote a kernel used to define propagations from labeled data points. We will use subscripts to index entries of a vector and superscripts to index elements of a set; for example, $x^i \in \mathcal{X}$ will represent the i^{th} element of a set $\mathcal{X}$ and x_k will represent the k^{th} entry of a vector $x \in \mathbb{R}^d$. We let $B(x,\delta)$ denote an open ball centered at x of radius $\delta > 0$.

2. Dirichlet Learning: The Semi-supervised Inference Model

We consider a set of features $\mathcal{X} \subset \mathbb{R}^d$, and a set of $K \geq 2$ classes. We assume that there is a relationship between the features and the K classes (categories), which we model with a joint probability distribution ν over the space $\mathbb{R}^d \times [K]$. We consider the marginal of ν on the inputs to be modeled via the mixture model

$$\rho(x) = \sum_{k=1}^{K} w_k \rho_k(x),$$

where each $\rho_k(x) = \mathbb{P}(x|y=k)$ is the class-conditional distribution's density for the k^{th} class and the weights represent the class marginal distribution weights $\sum_{k=1}^{K} w_k = 1, w_k \geq 0$. Let $\rho^K(x) = (\rho_1(x), \rho_2(x), \ldots, \rho_K(x))^\top \in \mathbb{R}^K$ be the vector of class-conditional densities at the input $x \in \mathcal{X}$.

Recall that we are given a set of *labeled data*, $\{(x^\ell, y^\ell)\}_{x^\ell \in \mathcal{L}}$, where $\mathcal{L} \subset \mathcal{X}$ and $y^\ell \in \{1 \ldots K\} =: [K]$ are the observed labels, which are drawn from the joint distribution ν. We will identify the label y^ℓ with its "one-hot encoding representation", $y(x^\ell) = e_{y^\ell} \in \mathbb{R}^K$, where e_k is the k^{th} standard basis vector in $\mathbb{R}^K$. Semi-supervised classification is the task of inferring the classification of the *unlabeled data* $\{x\}_{x \notin \mathcal{L}}$ from the observed labeled data $\{(x^\ell, y^\ell)\}_{x^\ell \in \mathcal{L}}$. As this is an ill-posed problem, one must incorporate a priori assumptions about the ground-truth classification of the points in terms of the underlying geometry of the data set $\mathcal{X}$.

In light of the semi-supervised context, we choose to model the classification $y(x)$ of an input $x \in \mathcal{X}$ probabilistically in terms of a categorical random variable Y such that

$$\mathbb{P}(Y = k|p) = p_k, \qquad p \in \Delta_K,$$

where the probability vector p belongs to the $K-1$ dimensional simplex, $\Delta_K = \{q \in \mathbb{R}^K : q_k \geq 0, \sum_{k=1}^{K} q_k = 1\}$. Thus, we consider semi-supervised learning as estimating input-indexed probability vectors $\{p(x)\}_{x \in \mathcal{X}}$ given observations of labels at the labeled points in $\mathcal{L}$. This leads us to the construction of a Dirichlet random field.

2.1 Dirichlet Random Field

We recall that in the context of classical Bayesian inference, if we have observations of a categorical variable $Y \in [K]$ we model the distribution of the probability of each category using a Dirichlet distribution, which is a probability distribution on the simplex with density function given by

$$\varphi(\alpha, z) := B(\alpha) \prod_{i=1}^{K} z_i^{\alpha_i - 1}, \tag{1}$$

where α is a vector in $\mathbb{R}_+^K$, z is vector in Δ_K, and $B(\alpha)$ is a normalization constant given by $B(\alpha) = \frac{\Gamma(\sum_{i=1}^K \alpha_i)}{\prod_{i=1}^K \Gamma(\alpha_i)}$, where Γ is the standard Gamma function generalizing the factorial. Given a Dirichlet prior distribution with parameters α, if we observe the data $Y = j$ then our posterior probabilities take the form $(\alpha_1, \ldots, \alpha_j + 1, \ldots \alpha_K)$. In this classical setting one may interpret the vector α as the total number of observations we have of each category, usually across repeated experiments.

In light of this aim, for each $x \in \mathcal{X}$, let $P(x) \sim Dir(\alpha_1(x) + \alpha_0, \alpha_2(x) + \alpha_0, \ldots, \alpha_K(x) + \alpha_0)$ be a Dirichlet-valued random variable (with uniform prior $Dir(\alpha_0, \alpha_0, \ldots, \alpha_0)$) to model our belief about the multinomial probability vector $p(x)$. In contrast with the classical Bayesian setting, in the semi-supervised setting we do not expect to observe the categorical variable $Y(x)$ associated with every feature x: instead, we will observe values of $Y(\tilde{x})$ where $\tilde{x}$ and x are similar.

We therefore will consider a vector $\alpha(x) := (\alpha_1(x), \alpha_2(x), \ldots, \alpha_K(x))^\top \in \mathbb{R}_+^K$ which reflects the continuous values of implicit "observations"—what we will term "pseudolabel"—from each possible class at the feature x. We emphasize that the semi-supervised setting requires us to lend the strength of an observation at one vertex to nearby vertices which we will accomplish by using a propagation mechanism that we introduce in Section 2.2. Hence we call the $\alpha(x)$ pseudo-labels or implied observations because the labels were only observed at nearby points.

The set of $\{P(x)\}_{x \in \mathcal{X}}$ defines a Dirichlet-valued random field. To interpret this as a semi-supervised classification model, we consider the mean estimator $\hat{p}(x) := \mathbb{E}[P(x)]$, where

$$\hat{p}_k(x) = \frac{\alpha_k(x) + \alpha_0}{\alpha_0 K + \sum_{m=1}^K \alpha_m(x)}, \tag{2}$$

to be the multinomial probability vector used to determine the inferred classification $Y(x)$ of x; that is, $\mathbb{P}(Y(x) = k | \hat{p}(x)) = \hat{p}_k(x)$, so that we infer the classification via

$$\hat{y}(x) := \operatorname*{argmax}_{k=1,2,\ldots,K} \hat{p}_k(x) = \operatorname*{argmax}_{k=1,2,\ldots,K} \alpha_k(x). \tag{3}$$

Here, by design, we have constructed $P(x)$ to be independent from $P(x')$ (for $x \neq x'$). This stands in contrast to the Gaussian process setting, wherein measurements at different x's are permitted to have non-trivial covariance with each other. In a sense, one does not have as many degrees of freedom to maintain non-trivial correlation between different x's because the classification problem takes discrete values while the regression problem takes real values. Roughly speaking, while pseudo-label information is propagated from

labeled points (as we will discuss in Section 2.2), our proposed statistical model treats the classification of each $x \in \mathcal{X}$ individually.

Furthermore, the uncertainty in our inference can be modeled by the covariance structure of the random variable $P(x)$. For example, we can model the uncertainty in $P(x)$ by considering the trace of the covariance matrix $C(x) \in \mathbb{R}^{K \times K}$, where $C_{km}(x) = Cov(P_k(x), P_m(x))$. Define $\tilde{\alpha}_k(x) = \alpha_k(x) + \alpha_0$ and $\beta(x) = \sum_{k=1}^{K} \tilde{\alpha}_k(x)$ so we can write

$$Tr[C(x)] = \sum_{k=1}^{K} Var(P_k(x)) = \sum_{k=1}^{K} \frac{\tilde{\alpha}_k(x)(\beta(x) - \tilde{\alpha}_k(x))}{(\beta(x))^2(\beta(x)+1)} = \frac{(\beta(x))^2 - \sum_{k=1}^{K}(\tilde{\alpha}_k(x))^2}{(\beta(x))^2(\beta(x)+1)}. \quad (4)$$

Later on, we will use this measure of uncertainty as the basis for our Dirichlet Variance acquisition function for DiAL. In summary, the salient features of our proposed model, namely (i) the inferred classification and (ii) the uncertainty in our belief about the inferred classification, are directly determined by the vectors $\{\alpha(x)\}_{x \in \mathcal{X}}$.

Example 1 (Running example: inference on finitely sampled data) *In many contexts, we have a large fixed quantity of unlabeled data $\mathcal{X} = \{x^i\}_{i=1}^n$. In that context, we will denote the point of interest by superscripts, namely $p(x^i) = p^i, \alpha(x^i) = \alpha^i, \hat{y}(x^i) = \hat{y}^i, \beta(x^i) = \beta^i$ and $C(x^i) = C^i$, and we will subsequently (see Example 3) view the x^i as the nodes of a graph. It is worth noting here that viewing the set of nodes as a random field is a common framework for imposing a Bayesian structure with the goal of quantifying uncertainty in semi-supervised learning, especially in graph-based methods (Zhu et al., 2003b; Miller and Bertozzi, 2024) which usually interpret learning problems in terms of a Gaussian random field with covariance structure derived from the graph Laplacian (see Section A.5). This is a powerful approach, which admits a direct Bayesian interpretation for graph-based regression problems, but the statistical interpretation for categorical problems is not as clear. To the best of our knowledge, this is the first work to consider a Dirichlet-valued random field (which is inherently categorical) for graph-based semi-supervised learning and our application to active learning.*

We also remark that in this finite data context, it may be useful to view the y's associated with the x's as deterministic (but unobserved). In this context, the ground truth $\nu(x^i, \cdot)$ would be concentrated on a single y value.

Remark 1 (Bayesian prior interpretation.) *Although the Dirichlet Learning classifier is not directly derived from a strictly "proper" Bayesian setup, we can interpret the given model in a Bayesian framework. In the interest of space, we defer details of this interpretation to Appendix B.*

2.2 Propagation Operators

Given the framework in the previous section, we are now faced with the question of how to determine the vectors $\{\alpha(x)\}_{x \in \mathcal{X}}$ from the currently observed labeled data $\{(x^\ell, y^\ell)\}_{\ell \in \mathcal{L}}$. We introduce propagation of "pseudo-label" from the labeled data via the use of kernels. In particular, given a positive definite kernel $\mathcal{K} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$, we define the *propagation from input $x \in \mathcal{X}$* to be the function $\mathcal{K}_x(\cdot) = \mathcal{K}(x, \cdot) : \mathcal{X} \to \mathbb{R}_+$.

Now let $\mathcal{L}_k = \{x^\ell \in \mathcal{L} : y^\ell = k\}$ be the subset of the labeled data $\mathcal{L}$ that belong to class k. Then we can define the functions

$$\mathcal{K}_{\mathcal{L}_k}(x) = \sum_{x^\ell \in \mathcal{L}_k} \mathcal{K}_{x^\ell}(x),$$

which are simply the accumulation of individual propagations from labeled points $x^\ell \in \mathcal{L}_k$ to the point x. We then set the concentration parameter at x, $\alpha(x) \in \mathbb{R}_+^K$, to be

$$\alpha(x) = (\mathcal{K}_{\mathcal{L}_1}(x), \mathcal{K}_{\mathcal{L}_2}(x), \ldots, \mathcal{K}_{\mathcal{L}_K}(x))^\top = \left(\sum_{\ell \in \mathcal{L}_1} \mathcal{K}(x^\ell, x), \ldots, \sum_{\ell \in \mathcal{L}_K} \mathcal{K}(x^\ell, x) \right)^\top.$$

The core idea behind our use of a propagation operator is a mechanism for lending observation at one point to "nearby" points. That is, an observation of a label at one input $x \in \mathcal{X}$ can be used to *propagate* some amount of an observation of the same label (i.e., pseudo-label) to points that are similar to x. From a modeling standpoint, it could be reasonable to require this propagation operator, $\mathcal{K}$, to satisfy various properties such as symmetry ($\mathcal{K}(x, z) = \mathcal{K}(z, x)$), normalization ($\mathcal{K}(x, x) = 1$), or a maximum principle ($\mathcal{K}(x, x) \geq \mathcal{K}(x, z)$ for all $z \in \mathcal{X}$).

For our analysis, however, we find that a notion of the ability of $\mathcal{K}$ to "separate" the clustering structure of the underlying class-conditional density over the inputs is quite useful for providing rigorous guarantees of exploration of clustering structure through the active learning process. Simply put, we desire that $\mathcal{K}(x, z)$ is large when x, z belong to the same region of large class-conditional density in a mixture model (i.e., cluster) and small when they do not. We discuss the particulars of this class-separator property in Section 5.3, which leads to a general-purpose result for guaranteeing exploration of clustering structure with the use of DiAL when this property holds. Furthermore, key to our numerical results, we advocate the use of *data-dependent* propagation operators that better reflect this class-separation property for clustered data (see Examples 2 and 3).

Example 2 *Consider a distribution of data on the square $\mathcal{X} = [0, 1]^2$, with density described by the classical "two moons" distribution. While generally sources do not provide an exact expression for such a density, we construct this density using a kernel density estimator of a finite sample: the associated density is displayed in Figure 3(a).*

For such continuum data, there are a variety of possible choices for $\mathcal{K}$ to describe the strength of the relationship between two points. One of the simplest such approaches is based upon radial basis functions (RBF), namely

$$\mathcal{K}(x, x') = \exp(-\|x - x'\|_2^2 / 2\sigma^2).$$

This choice of kernel has been previously utilized in active learning tasks (Karzand and Nowak, 2020). We notice that such a choice of kernel is isotropic and therefore independent of the underlying distribution of the data.

An alternative choice, which has previously been considered in the discrete setting under the name "Poisson Learning" (Calder et al., 2020), utilizes partial differential equations to

construct a data-informed propagation operator. In particular, we can define our propagation operator via the expression

$$\mathcal{K}_P(x, x^*) = u_{x^*}(x),$$

where u_x^* is the solution to the partial differential equation

$$\begin{aligned} -\Delta_\rho u_{x^*}(x) + \tau u_{x^*}(x) &= \delta_{x^*}(x) \quad \text{for } x \in [0,1]^2 \setminus \{x^*\} \\ \nabla u_{x^*}(x) \cdot \boldsymbol{\nu} &= 0 \quad \text{on } \partial[0,1]^2 \end{aligned} \tag{5}$$

where here we define $\Delta_\rho v := \frac{1}{\rho}\nabla \cdot (\rho \nabla v)$. This kernel will not be symmetric, but will satisfy our normalization and Maximum Principle assumptions, and will be strongly data-adapted. A finite difference approximation of this solution is displayed in Figure 3, and demonstrates very attractive data-adapted propagation.

While this approach gives elegant propagation functions which respect data density and topology, kernel density estimation and finite difference approximation are computationally challenging in higher dimensions. As a result, this "continuum" Poisson propagation operator is essentially unusable in practice; this motivates the graph-based approximation of the Poisson propagation that we introduce next in Example 3.

Example 3 (Running example (ctd): Graph-based propagation operators) *Let $G(\mathcal{X}, W)$ be a similarity graph with finite node set $\mathcal{X} = \{x^i\}_{i=1}^n \subset \mathbb{R}^d$ with edge weight matrix $W \in \mathbb{R}^{n \times n}$, where the weight $0 \leq W_{ij})$ captures the similarity between $x^i, x^j \in \mathbb{R}^d$; that is, W_{ij} is to be larger (smaller) when x^i and x^j are similar (dissimilar). Let $D = \text{diag}(d_1, d_2, \ldots, d_n)$ be the diagonal degree matrix with $d_i = \sum_{j=1}^n W_{ij}$ denoting the degree of node x^i. We consider pseudo-label propagation from labeled $x^\ell \in \mathcal{L}$ to the rest of the nodes in the graph via the use of graph Laplacian matrices, as mentioned in Section A.5. The combinatorial graph Laplacian, $L = D - W$, is a standard graph Laplacian matrix that is known to be positive, semi-definite with real eigenvalues and eigenvectors, including a non-trivial null space. The geometric structure of the eigenvectors corresponding to the k smallest eigenvalues of graph Laplacians forms the basis for spectral clustering (von Luxburg, 2007).*

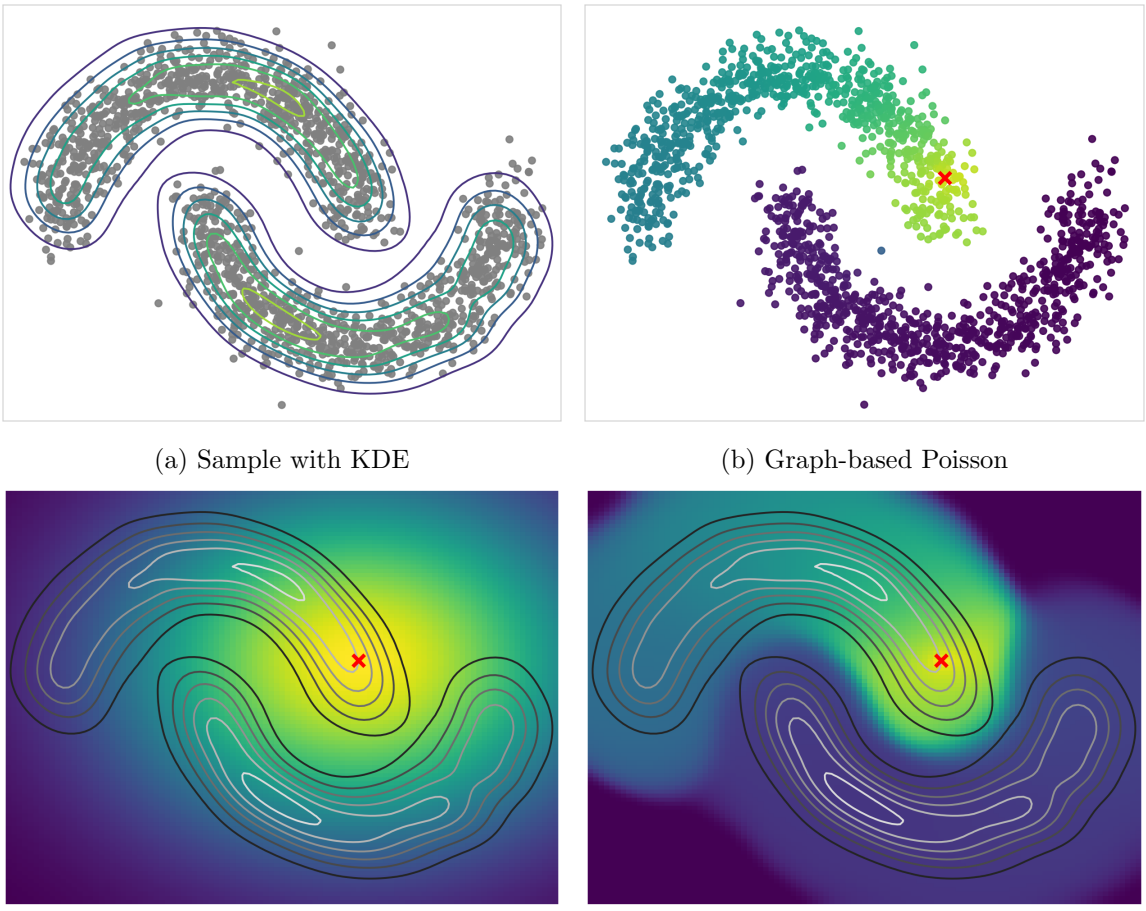
Define a node function $g^\ell : \mathcal{X} \to \mathbb{R}$ (equivalently written as a vector $g^\ell \in \mathbb{R}^n$ assuming the ordering on the nodes $\mathcal{X} = \{x^1, x^2, \ldots, x^n\}$ of the graph) to be the solution to the following "Poisson propagation" (Calder et al., 2020; Miller and Calder, 2023)

$$(L + \tau \mathrm{I}_n)\, g^\ell = e_\ell, \tag{6}$$

for a given $\tau > 0$ where $e_\ell \in \{0,1\}^n$ is the ℓ^{th} standard basis vector in $\mathbb{R}^n$. We can then define a corresponding "Poisson" graph propagation operator $\mathcal{K}_P$ to be

$$\mathcal{K}_P(x^\ell, x) = \frac{g^\ell(x) - (\min_{\tilde{x} \in \mathcal{X}} g^\ell(\tilde{x}))}{g^\ell(x^\ell) - (\min_{\tilde{x} \in \mathcal{X}} g^\ell(\tilde{x}))}. \tag{7}$$

We note that this graph-based propagation operator is inherently transductive; that is, it only applies to the finite node set $\mathcal{X}$ that we used to construct our graph. This is emblematic of graph-based methods for semi-supervised and active learning, and it contrasts



(a) Sample with KDE

(b) Graph-based Poisson

(c) RBF Kernel

(d) Continuum Poisson

Figure 3: Example propagation operators on a synthetic data set (Two Moons) in two dimensions. Panel (a) shows both the level sets of the underlying data-generating distribution $\rho(x)$ and an empirical sample of 1500 points from ρ. Panels (b-d) show different propagation operators on the domain, where (b) and (d) are respectively graph-based and finite-difference approximations to the Poisson propagation of Equation (5). These are inherently data *adaptive* propagations that consider the data density ρ. In contrast, radial basis function (RBF) propagation, as shown in panel (c), is *independent* of the data distribution, whereas the Poisson propagation depends on the data-generating distribution.

with the inductive capability of other propagation operators like the RBF kernel discussed in Example 2 to apply to previously unseen data points. Despite this, the pool-based active learning setting—wherein the active learner has access to a (large) pool of unlabeled data for labeling and classification—is quite common, and so this data-dependent Poisson propagation operator is an effective choice for effective propagation of labeled information that respects clustering structure.

Our analysis in Section 5.3 will then justify this choice of propagation that satisfies the class-separating property for sufficiently clustered data sets; this will, in turn, imply efficient exploration of clustering structure through our Dirichlet Active Learning (DiAL) process.

It is insightful to contrast our setup and choice of propagation operator from Example 3 with previous Bayesian frameworks for graph-based semi-supervised and active learning. For simplicity, consider the binary classification task. In the Gaussian process/random field setting of works such as (Zhu et al., 2003a,b), the graph Laplacian is incorporated into the prior distribution as over node functions $u : \mathcal{X} \to \mathbb{R}$ as $u \sim \mathcal{N}(0, (L+\tau I)^{-1})$ and reflects an *a priori* assumption regarding the smoothness of likely node functions with respect to the graph topology. The covariance matrix of this prior distribution is intimately connected to the graph-based propagation operator that we consider here. Intuitively, this graph-based prior distribution in the Gaussian random field biases the posterior belief given labeled data toward node functions that have similar outputs for nodes that are connected in the graph. In this way, the ample supply of available unlabeled data can straightforwardly be incorporated into the Bayesian framework to admit more sample-efficient learning of the classification task under the assumption that the classification structure aligns with the clustering structure of the unlabeled data.

The motivation for our related graph-based propagation operator does not admit the same direct interpretation in terms of Bayesian inference. The implicit modeling assumption with its use in Dirichlet Learning is that a data-dependent propagation operator should reflect the clustering (geometric) structure of the data set. This intuition is indeed what underlies the choice of the graph Laplacian-based prior distribution in the Gaussian random field setting but is not directly modeled in the prior belief for each Dirichlet random variable in our setting. Instead, the data-dependent propagation operator is incorporated implicitly into the likelihood model, wherein continuously valued amounts of pseudo-label influence from labeled points are given to other points in a manner that reflects the underlying clustering structure of the data set. Intuitively, this choice of data-dependent propagation in the Dirichlet Learning classifier aims to achieve sample efficient exploration of clusters by DiAL which we discuss in the next section (Section 3).

3. DiAL: Query Point Selection

The vector-valued function $\alpha(x) = (\mathcal{K}_{\mathcal{L}_1}(x), \mathcal{K}_{\mathcal{L}_2}(x), \ldots, \mathcal{K}_{\mathcal{L}_K}(x))^\top : \mathcal{X} \to \mathbb{R}_+^K$ (which may be organized as a matrix in the case where $\mathcal{X}$ is finite) expresses our information about the probabilities of the different class labels at every point in $\mathcal{X}$, in terms of the underlying Dirichlet random field. As discussed earlier, given a particular semi-supervised method there are many different possible approaches for identifying new points at which to acquire data. For clarity, we introduce now a distinction between an active learning *acquisition function* and *policy*.

3.1 Acquisition Functions and Active Learning Policies

An acquisition function $\mathcal{A}(x; \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_K)$ evaluated on inputs $x \in \mathcal{X}$ quantifies how useful our model believes it would be for the active learner to query its label. This acquisition function is user-defined and is designed to reflect the desired properties of the query points to be labeled throughout the active learning process. While this function depends on the data points in $\mathcal{L}_1, \ldots, \mathcal{L}_K$ and their associated labels, we will forego explicitly writing this dependence in favor of readability; namely, we will write $\mathcal{A}(x) = \mathcal{A}(x; \mathcal{L}_1, \ldots, \mathcal{L}_K)$ with the understanding of the dependence on the currently labeled data.

Now, with a chosen acquisition function it remains to decide how to select the next query point x^* from the set of acquisition function values on the unlabeled data, $\{\mathcal{A}(x)\}_{x \in \mathcal{U}}$. We refer to the method for selecting the query point from said values as the active learning *policy*. Among many possible choices, we focus our attention on two natural choices: (1) *Maximum Value* and (2) *Proportional Sampling*.

Maximum Value policy chooses to query the label of the point that maximizes $\mathcal{A}$ on the unlabeled data

$$x^*_{MV} = \operatorname*{argmax}_{x \in \mathcal{U}} \ \mathcal{A}(x).$$

The majority of sequential active learning methods previously proposed fall under this category (Settles, 2012; Miller and Bertozzi, 2024; Balcan et al., 2007; Miller and Calder, 2023; Zhu et al., 2003b; Ma et al., 2013; Ji and Han, 2012; Jiang and Gupta, 2019), where cases of acquisition functions that are to be minimized can be equivalently rephrased to maximize the negative of acquisition functions values.

Proportional Sampling selects query points via randomly sampling according to a probability distribution derived from the acquisition function values over the set of unlabeled inputs; for example, we can select $x^*_{PS} = x$ for $x \in \mathcal{U}$ with probability

$$q(x) \propto \exp\left(\lambda \mathcal{A}(x)\right)$$

for scaling factor $\lambda > 0$. This distribution over $x \in \mathcal{U}$ encourages the selection of points with larger acquisition function values at a given iteration. Note that as $\lambda \to \infty$, this distribution concentrates on the maximizer $x^* = \operatorname{argmax} \mathcal{A}(x)$, whereas as $\lambda \to 0^+$, the distribution converges to a uniform distribution over the unlabeled data. In Appendix D.1, we discuss how we choose $\lambda > 0$ for our numerical experiments, and we identify some properties of the choice of λ in an asymptotic regime of DiAL with Prop. Sampling.

We note that this kind of "softmax" scaling for a sampling distribution has recently been used in other active learning works to encourage diverse batches of query points (Kirsch et al., 2023) and to correct for the sampling bias of uncertainty sampling with classifiers found via empirical risk minimization (Zhan et al., 2022). Furthermore, a similar idea of "proportional sampling" has been used in randomized numerical linear algebra methods such as (Deshpande and Vempala, 2006; Musco and Woodruff, 2017; Chen et al., 2025) for column subset selection and low-rank matrix approximation of positive semi-definite matrices. Depite this shared idea of proportional sampling distributions for selecting inputs, the nature of our theoretical results is quite distinct from these previous works.

3.2 Uncertainty Sampling & Other Common Acquisition Functions

One common active learning approach is to query points where the current classifier has the most "uncertainty" about the class selection, a framework often referred to as *uncertainty sampling* (Settles, 2012; Miller and Calder, 2023). This is often expressed by selecting points where the classifier's output class probabilities are most alike. In the context of binary classification, we could quantify this uncertainty using the "smallest margin" acquisition function

$$\mathcal{A}_{unc}(x) = -\left(\max_{k=1,2} \hat{p}_k(x) - \min_{k=1,2} \hat{p}_k(x)\right) = -|\hat{p}_1(x) - \hat{p}_2(x)|, \tag{8}$$

where here the $\hat{p}$ are class probabilities outputted by a semi-supervised algorithm, for example using (2) from our proposed Dirichlet Learning model. We have defined the acquisition function to be the *negative* of the margin value $|\hat{p}_1(x) - \hat{p}_2(x)|$ in order to obey our policy convention of *maximization*. With this choice of acquisition function, then the MV policy would select the query point $x^*_{MV} = \text{argmax}_{x\in\mathcal{X}}\, \mathcal{A}(x) = \text{argmin}_{x\in\mathcal{X}}\, |\hat{p}_1(x) - \hat{p}_2(x)|$: this should focus on labeling points that are "closest" to the current classifier's decision boundary. For the multi-class setting various generalizations are possible for the margin value given above: for example, one could use $\max_{k\in[K]} \hat{p}_k(x) - \min_{k\in[K]} \hat{p}_k(x)$ or the difference in the probabilities of the two most likely classes.

Variance-based acquisition functions are also common for use in active learning, especially in graph-based contexts (Ji and Han, 2012; Ma et al., 2013; Qiao et al., 2019). For example, both Variance Optimization (VOpt) (Ji and Han, 2012) and Σ-Optimality (ΣOpt) (Ma et al., 2013) utilize functions of the variance of Gaussian random fields with covariance structure dependent on a similarity graph as a measure of uncertainty. However, in those contexts, the observed class labels do not affect the variance of the random fields, and hence VOpt and ΣOpt serve primarily to ensure that labeled points are spread evenly across the available data. In a sense, these algorithms are primarily exploratory in nature, and in a way that is label ambivalent. Our use of Dirichlet random fields for semi-supervised inference allows the use of the variance of each Dirichlet random variable as an acquisition function; this will inherently be informed by the observed labels at the labeled data, contrasting with these other variance-based acquisition functions.

3.3 Dirichlet Variance: DiAL's Acquisition Function

While one could directly apply the uncertainty sampling acquisition function (8) to our Dirichlet Learning model outputs at each iteration, we possess significantly more information than just the mean probability estimator. In particular, we could instead define uncertainty in the classifier's outputs for x to be the variance of the Dirichlet distribution of the class labels at the point x, which we recall from (4) to be

$$\mathcal{A}_{var}(x) := Tr[C(x)] = \frac{(\beta(x))^2 - \sum_{k=1}^{K} (\tilde{\alpha}_k(x))^2}{(\beta(x))^2(\beta(x)+1)}. \tag{9}$$

where $\beta(x) = \sum_{k=1}^{K} \tilde{\alpha}_k(x)$. This approach uses a very different conceptual approach to defining uncertainty: instead of uncertainty being an issue of similarly probable class labels under a classifier, it instead becomes a lack of information about the probabilities of

those class labels. We will term the acquisition function of (9) to be *Dirichlet Variance*. Using this acquisition function, we may apply either active learning policy for selecting the next query point. To be clear, we respectively term the maximum value and proportional sampling policies of the Dirichlet Variance acquisition function to be "Dir. Var." and "Dir. Var. (Prop)"; see (Table 1) for a summary.

Name	Policy	Formula
Dir. Var.	*Maximum Value*	$x^* = \text{argmax}_{x \in \mathcal{U}} \ \mathcal{A}_{var}(x)$
Dir. Var. (Prop),	*Proportional Sampling*	$x \in \mathcal{U},\ q(x) \sim \exp\left(\lambda \mathcal{A}_{var}(x)\right)$

Table 1: Summary of two methods which we study based upon the proposed "Dirichlet Variance" acquisition functions, $\mathcal{A}_{var}(x) = \frac{(\beta(x))^2 - \sum_{k=1}^{K} (\tilde{\alpha}_k(x))^2}{(\beta(x))^2(\beta(x)+1)}$. The first, Dir. Var. selects a query point via the maximizer of the Dirichlet variance, while Dir. Var. (Prop) selects the next query point with probability $q(x) \propto \exp\left(\lambda \mathcal{A}_{var}(x)\right)$

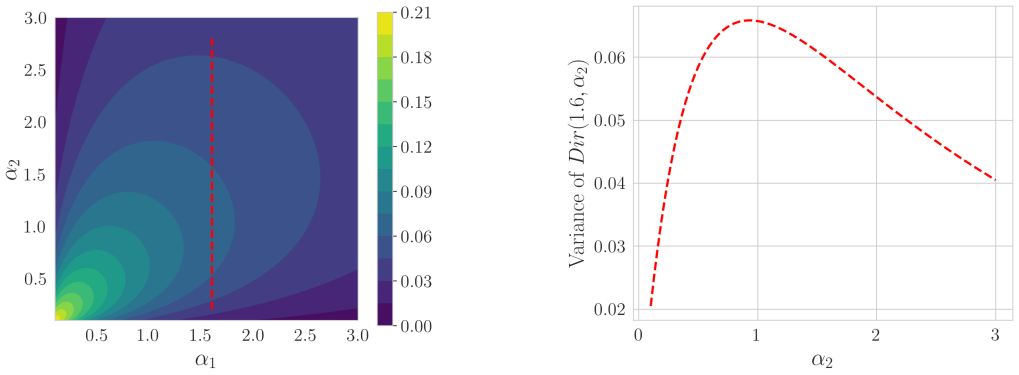
It is useful here to recall a few properties of the variance of a Dirichlet distribution, and give their interpretation in the context of active learning.

Property 1 (Variance decreases in total observations) *Given Dirichlet distributions with parameter $t\alpha$ with $t > 0, \alpha \in \mathbb{R}_+^K$ we have that the variance is monotonically decreasing in t. When t is large, the variance is of order t^{-1}. This indicates that, holding the proportions of class observations constant, the variance decreases as we observe more data. This means that after each new acquisition of data in our active learning algorithm our variance will decrease in expected value.*

Property 2 (Non-monotonicity of variance) *The variance of the Dirichlet distribution is not monotone in its individual components. This is readily seen in Figure 4. This means that following a new query, it is possible that the variance of the Dirichlet distribution at some points may actually increase. Consequently, our proposed algorithm is not guaranteed to be monotone in its uncertainty, a property which is used to provide performance guarantees for some classes of acquisition functions (e.g., VOpt and ΣOpt (Ma et al., 2013)).*

Property 3 (Reduction to uncertainty sampling) *Given a particular constant $\mathcal{O} = \sum_{k=1}^{K} \alpha_k$ the variance of the Dirichlet distribution is maximized when $\alpha_{k_1} = \alpha_{k_2}$ for all $k_1, k_2 \in [K]$. This implies that if the number of total pseudo-labels (i.e., implicit class observations) is comparable across different x's then the uncertainty is measured to be higher at points where class probabilities are similar. Described in another way, if $\beta(x)$ for each x were approximately constant during the active learning process, then the algorithm would reduce to a form of uncertainty sampling that would select query points in regions where the current classifier has large levels of "data-conditional" uncertainty, see Section 6.*

We have chosen to use the variance of the Dirichlet random field as a means of modeling the current level of uncertainty of a semi-supervised classification problem. This is due to



(a) Variance of $Dir(\alpha_1, \alpha_2)$

(b) Variance as function of α_2, holding $\alpha_1 = 1.6$ constant.

Figure 4: Demonstration of the non-monotonicity of variance of Dirichlet random variables *in individual components*, α_i. Panel (a) is a heatmap of $Var[P] = \frac{\alpha_1\alpha_2}{(\alpha_1+\alpha_2)^2(\alpha_1+\alpha_2+1)}$ for $P \sim Dir(\alpha_1, \alpha_2)$. The red dotted line in (a) represents a slice where $\alpha_1 = 1.6$ is held constant, and panel (b) highlights the corresponding non-monotonicity of the variance as α_2 is varied.

its flexibility to capture more information than other previously utilized measures of uncertainty. We provide a more detailed discussion about types of uncertainty in classification in Appendix 6. In that appendix, we also provide a simple 1D example where using classifier margin as a measure of uncertainty leads to an unsuccessful active learning algorithm, while Dirichlet variances lead to success.

4. Numerical Experiments

In this section, we present numerical results to evaluate the utility of our proposed DiAL framework utilizing a graph-based Dirichlet Learning model and the straightforward use of Dirichlet Variance as an acquisition function for active learning. In Subsection 4.1, we begin with an application to pixel classification in hyperspectral imagery (HSI) in order to compare with the Learning by Active Non-linear Diffusion (LAND) algorithm introduced in (Murphy and Maggioni, 2019). The experiments in Subsection 4.2 use a setup involving the common machine learning benchmark data sets MNIST (LeCun and Cortes, 2010) and FASHIONMNIST (Xiao et al., 2017) to assess the explorative capabilities of active learning methods; this setup was introduced recently in (Miller and Calder, 2023). In Section 4.3, we discuss the following two main takeaways from our numerical experiments:[2]

2. We provide some initial steps in understanding the advantages of Proportional Sampling as an active learning policy in Section 5, but further work in this vein is warranted.

Abbr. Name	Name	Policy	Ref.
Dir. Var.	Dirichlet Variance	MV	(present work)
Dir. Var. (Prop)	Proportional Dirichlet Variance	PS	(present work)
LAND	Learning by Active Nonlinear Diffusion	MV	(Murphy et al., 2019)
Unc. (SM)	smallest-margin uncertainty Sampling	MV	(Settles, 2012)
VOpt	Variance Optimization	MV	(Ji and Han, 2012)
ΣOpt	Σ-Optimality	MV	(Ma et al., 2013)
MCVOpt	Model Change with VOpt heuristic	MV	(Miller et al., 2022)
Rand.	Random Selection	N/A	N/A

Table 2: Acquisition functions of interest in the experiments in Subsections 4.1 and 4.2. See Section 3 for a detailed explanation of MV (maximum value) and PS (proportional sampling) policies.

1. Dir. Var. (Prop) (Table 1) performs most favorably for exploration and overall performance across the range of experiments compared to the array of acquisition functions using Maximum Value policies.

2. The Dirichlet Variance acquisition function selects query points that empirically improve classifier accuracy comparable to the prior state of the art (LAND, ΣOpt, VOpt, MCVOpt) with a much lower computational cost and much greater interpretability.

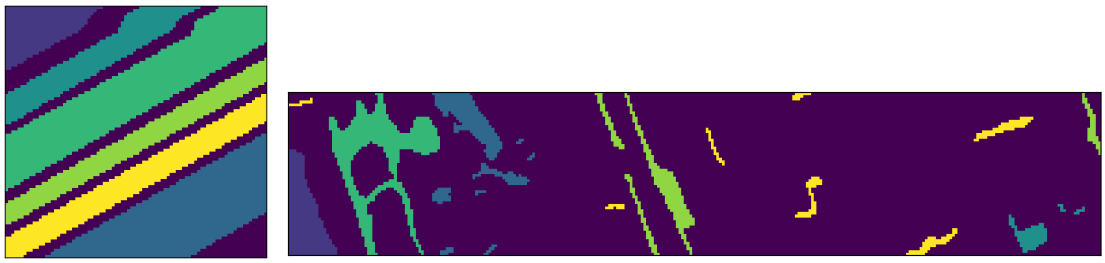
In each experiment, we calculate the average semi-supervised classification accuracy over 10 trials as a function of the size of the labeled set to compare DiAL with previous graph-based active learning methods. For consistent comparisons in each experiment, we compute these accuracies with our proposed Dirichlet Learning classifier using a value of $\tau = 0.1$, which was chosen via trial-and-error. We note that an interesting line of inquiry is to investigate how to properly choose τ; we leave this for future work. In Table 2, we list the various acquisition functions that we use for comparison in our experiments.

In order to assess the explorative capabilities of the compared methods in the experiments of Subsection 4.2, we additionally track the proportion of clusters that each acquisition function has sampled from as a function of the labeled set size. See Subsection 4.2 for additional details of this setup. Please see our GitHub repository (`https://github.com/millerk22/dirichlet-active-learning.git`) for the code to reproduce our results and figures.

Lastly, we refer the reader to Appendix D.1 for our justification of heuristic choices of the following hyperparameters: (i) the strength of the uniform prior over classes for each datapoint (α_0) and (ii) the inverse temperature parameter, $\lambda > 0$, used in the Proportional Sampling policy.

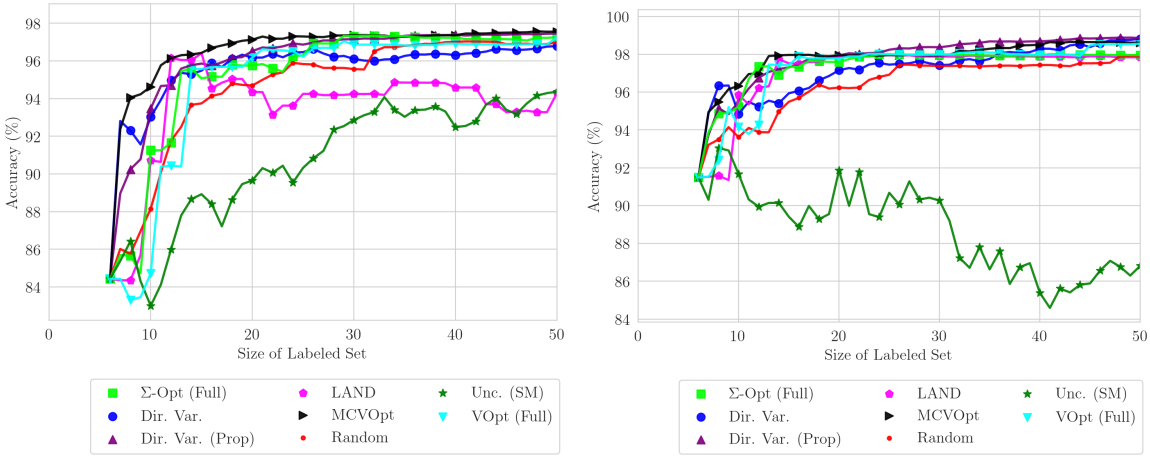
4.1 Hyperspectral Imagery (HSI) Pixel Classification

We first demonstrate the effectiveness of using DiAL for improving pixel classification in two commonly studied hyperspectral imagery (HSI) data sets, Salinas-A and Pavia, mimicking



(a) Salinas A (b) Pavia

Figure 5: Ground truth classifications of Salinas A (a) and Pavia (b) HSI data sets.



(a) Salinas (b) Pavia

Figure 6: Plots of accuracy results on HSI data sets, Salinas-A and Pavia data sets.

the setup presented in (Murphy and Maggioni, 2019; Cloninger and Mhaskar, 2021). The goal is to classify the pixels in the image into material classes based on the samples from the different wavelengths. The Salinas A data set is a common HSI data set that contains 7,138 total pixels in an 83×86 image in $d = 224$ wavelengths. This is an image of Salinas, USA taken with the Aviris sensor and contains 6 classes of plant types arranged in a diagonal pattern (see Figure 5a). The Pavia data set we use here consists of a 270×50 subset of the original Pavia data set and has spatial resolution 1.3m/pixel. The image contains 6 spatial classes, and was taken over Pavia, Italy by the ROSIS sensor (Figure 5b). Both of these hyperspectral data sets are available online at `http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes`.

We construct k-nearest neighbor graphs with $k = 20$ using cosine similarity

$$\mathcal{K}(x^i, x^j) = \langle x^i, x^j \rangle / \|x^i\|_2 \|x^j\|_2,$$

a common similarity metric for HSI applications.[3] The semi-supervised classification task is to infer the classification of unlabeled hyperspectral pixels in the image into one of a predetermined number of classes given a subset of labeled pixels. Beginning with one initially labeled pixel per class, we select pixels sequentially via the acquisition functions described in the previous section.

In both Figure 6 (a) and (b), we plot the accuracy of our Dirichlet Learning graph-based semi-supervised classifier 3 on the unlabeled datapoints as labeled sets are chosen via the different acquisition functions throughout the active learning process.

We briefly comment on the relatively poor performance of smallest-margin uncertainty sampling in the Laplace learning classifier (i.e., Unc. (SM)) in our comparison. smallest-margin uncertainty sampling has been known to produce especially poor results due to the selection of overly-exploitative query points; that is, this commonly used method for uncertainty sampling can lead to query points that do not properly explore the extent of the data set and result in poor empirical performance.

We note that the LAND acquisition function is not originally designed for this underlying semi-supervised learning model. For a consistent comparison of results, we have reported the accuracy in our Dirichlet Learning semi-supervised learning model, though we do note that the corresponding accuracy in the Learning by Active Nonlinear Diffusion (LAND) (Murphy and Maggioni, 2019) was poorer than the results in our model shown in Figure 6.

4.2 Cluster Exploration Experiments

In (Miller and Calder, 2023), the authors introduce an experimental setup designed to evaluate how effectively an active learning method selects query points that both (1) explores the clustering structure of the data set and (2) leads to optimal increases in the accuracy of the underlying semi-supervised classifier. The ground truth classes in the MNIST (LeCun and Cortes, 2010) and FASHIONMNIST (Xiao et al., 2017) benchmark machine learning data sets are used to define "clusters" in an auxiliary problem wherein a modified classification structure is imposed; namely, the true class labelings $y_i \in \{0, 1, \ldots, K\}$ (e.g., digits 0-9 for MNIST) and reassign them to one of $k < K$ classes by taking $y_i^{new} \equiv y_i \bmod k$; see Table 3.

For each trial with an acquisition function, we select one initially labeled point per *"modulo"* class; therefore, only a subset of clusters (i.e., the original true classes) has an initially labeled point. In order to perform active learning successfully in these experiments, query points chosen by the acquisition function over the trial must sample from each cluster. In this way, we have created an experimental setup with high-dimensional data sets with potentially more complicated clustering structures. We perform 10 trials for each acquisition function, where each trial begins with a different initially labeled set, and 100 query points are chosen sequentially by each of the different acquisition functions. To clarify, trials begin with only 3 labeled points in the MNIST and FASHIONMNIST experiments, so only 3 out of the 10 clusters begin with labeled points.

With this experimental setup, we can track not only the accuracy of the underlying semi-supervised classifier on the unlabeled data (accuracy plots), but also the proportion of

3. It should be noted that this similarity metric is *different* than the kernel used for the propagation operator in DiAL. This similarity kernel is only used for comparing input pixels for graph construction.

Resulting Mod Class	0	1	2
MNIST	0,3,6,9	1,4,7	2,5,8
FASHIONMNIST	0,3,6,9	1,4,7	2,5,8

Table 3: *"modulo" class labels.* Each ground truth class is interpreted as a different "cluster" so that the resulting class structure has multiple clusters per class. For MNIST and FASHIONMNIST, there are 10 ground truth classes, and we take these classes modulo $k = 3$.

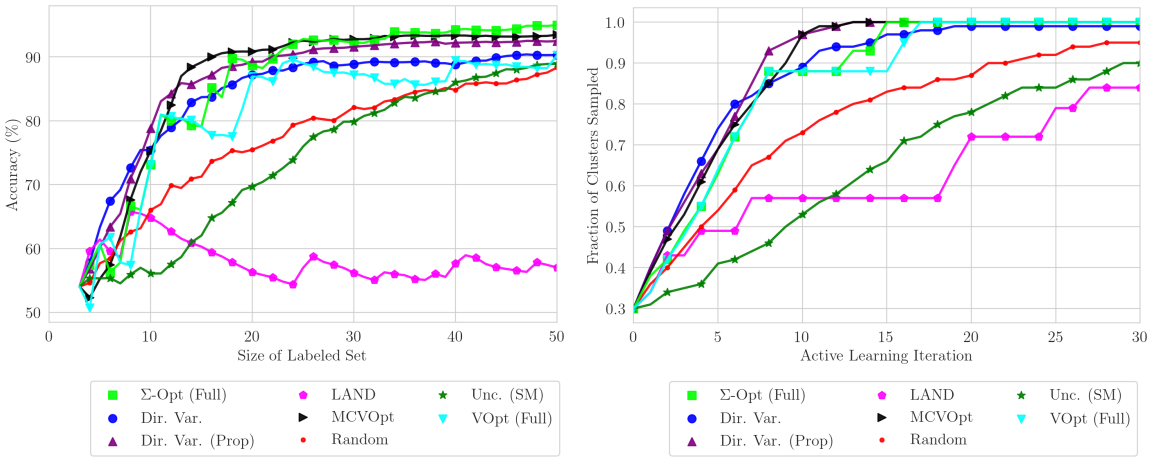
clusters that contain labeled data (cluster exploration plots) throughout the active learning process. While most work in active learning has only analyzed accuracy (and similar metrics) as a function of labeled set size to evaluate the performance of acquisition functions, we suggest that other metrics such as these cluster exploration plots can be informative.

Remark 2 *We note that while we use the experimental setup from (Miller and Calder, 2023) for assessing exploration capabilities in graph-based active learning, our DiAL method is markedly different from their Unc. Norm method. Whereas they use a Poisson propagation to reweight the graph around labeled nodes, we use the Poisson propagation to define how pseudo-label is distributed to similar nodes in the graph for our Dirichlet random field. Thus, both the underlying semi-supervised classifier and the acquisition function in the current work are distinct from those used in (Miller and Calder, 2023).*

4.2.1 Reduced-size Data Sets for Comparison to LAND

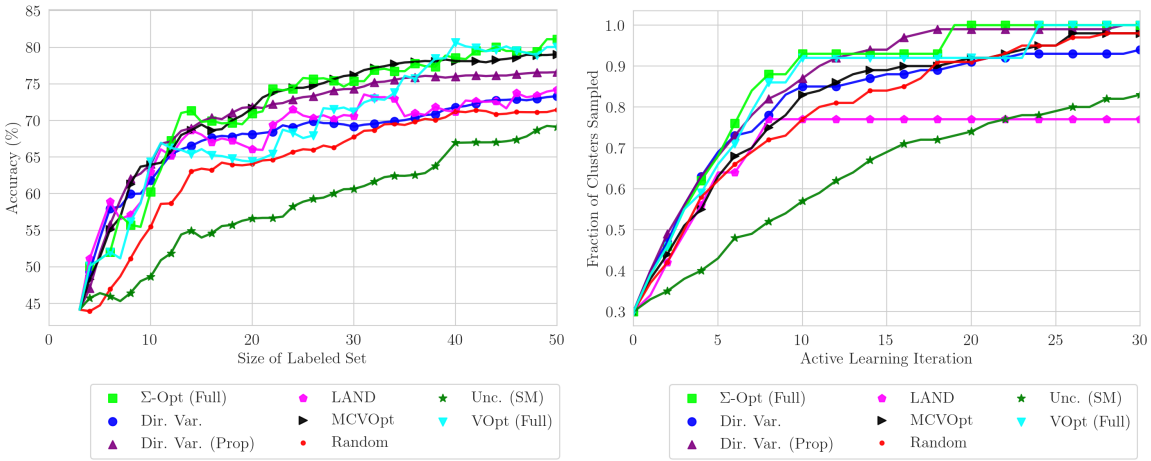
We additionally run active learning experiments on a predetermined, random subset (10% of the total) of each of these "modified" data sets. This reduced setting allows us to compare with the LAND algorithm with the publicly-available MATLAB implementation (Murphy) the authors use since we ran into memory issues when applying to the full MNIST and FASHIONMNIST data sets. Furthermore, the original forms of the VOpt and ΣOpt acquisition functions are prohibitively expensive for the full MNIST and FASHIONMNIST data sets; these reduced data sets allow us to also compare against the full calculation of VOpt and ΣOpt. The random subset is chosen by sampling uniformly at random 10% of the points from each original class, and we refer to these smaller data sets as MNIST-SMALL and FASHIONMNIST-SMALL. Figures 7a and 8a display the accuracy results while Figures 7b and 8b display the cluster exploration results for the MNIST-SMALL and FASHIONMNIST-SMALL experiments, respectively.

The full MNIST and FASHIONMNIST experiments then exclude comparison with the LAND acquisition function as well as the full VOpt, and ΣOpt acquisition functions. We use an approximate VOpt and ΣOpt calculation that utilizes a dimensionality reduction by projecting onto the eigenvectors corresponding to the $r = 50$ smallest eigenvalues of the graph Laplacian, similar to what is done with the MCVOpt (Miller et al., 2022) criterion. Accuracy and cluster exploration results for both MNIST and FASHIONMNIST data sets are reported in Figures 9 and 10.



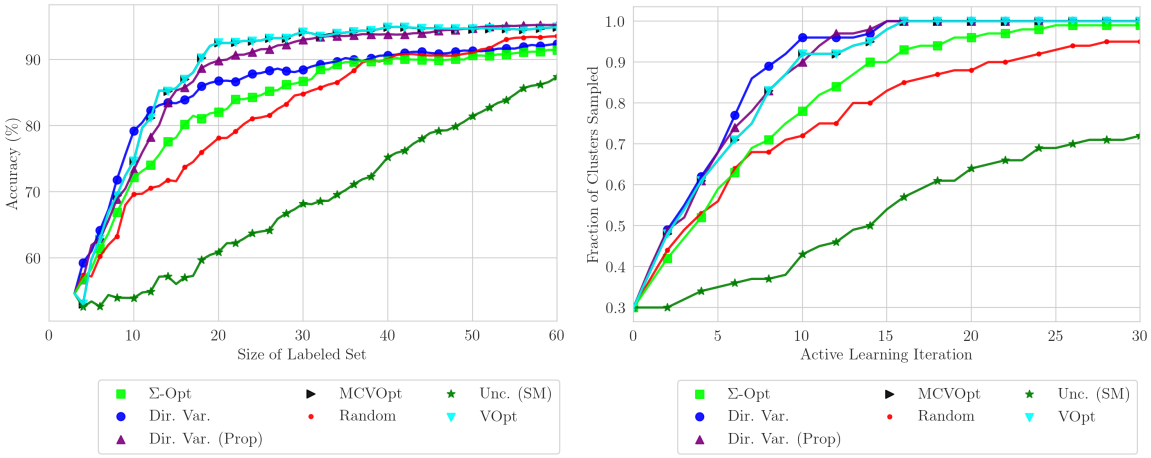
(a) Accuracy (b) Cluster Exploration

Figure 7: Plots of results on MNIST-SMALL data set, showing both accuracies (a) and cluster proportion (b) as a function of the active learning iteration (i.e., the size of the labeled data).



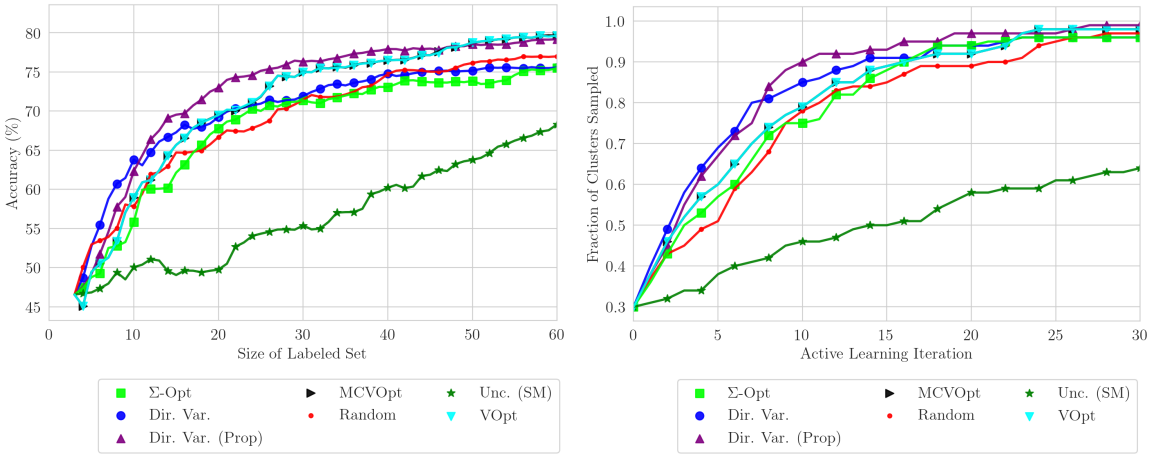
(a) Accuracy (b) Cluster Exploration

Figure 8: Plots of results on FASHIONMNIST-SMALL data set, showing both accuracies (a) and cluster proportion (b) as a function of the active learning iteration (i.e., the size of the labeled data).



(a) Accuracy (b) Cluster Exploration

Figure 9: Plots of Results on MNIST data set, showing both accuracies (a) and cluster proportion (b) as a function of the active learning iteration (i.e., the size of the labeled data).



(a) Accuracy (b) Cluster Exploration

Figure 10: Plots of results on FASHIONMNIST data set, showing both accuracies (a) and cluster proportion (b) as a function of the active learning iteration (i.e., the size of the labeled data).

4.3 Discussion of Numerical Results

We now discuss the results and subsequent implications of the various experiments of Sections 4.1 and 4.2, as stated briefly in the summary box at the start of Section 4.

4.3.1 Assessing Exploration Capabilities

Recall that plotting cluster proportion as a function of active learning iteration (labeled set size) allows us to monitor the *explorative* nature of the selected query points by each acquisition function; namely, we can characterize a "sufficiently" explorative method as one that samples from *every* cluster in fewer iterations than other methods. Then, by computing the accuracy at each active learning iteration, we can measure how useful the corresponding query points were for the classification task. Combining both measurements yields an arguably more complete picture of the utility of a proposed acquisition function.

In the MNIST-SMALL (Figure 7a) and FASHIONMNIST-SMALL (Figure 8a) experiments, we see that the MCVOpt and ΣOpt (Full) acquisition functions often led to the greatest marginal gains in accuracy compared to all the other methods shown. Dir. Var. (Prop) performed comparably in terms of accuracy in both experiments, but with a more consistent exploration of clusters across the two experiments (see Figures 7b and 8b). By comparison, it seems that the query points selected by LAND led to poor performance in the MNIST-SMALL task while merely sub-optimal performance in the FASHIONMNIST-SMALL task. An investigation of the cluster exploration plots of both experiments suggests that the LAND acquisition function struggled to identify query points from each of the 10 clusters in as few queries as the other presented methods. We suggest that this points to the importance of cluster exploration for the subsequent performance of the classifier in the active learning process.

In the larger experiments, MNIST and FASHIONMNIST, we observe similar results with the exception of the ΣOpt acquisition function (possibly due to the spectral approximation performed to reduce the computational costs for these larger experiments). We highlight that our proposed Dir. Var. (Prop) method still performs comparably to the best-performing acquisition functions on the MNIST task, and achieves the best performance of all methods on the FASHIONMNIST task. These results are encouraging, as across both the large and small tasks our proposed method has consistently performed well in terms of both accuracy and cluster exploration.

As a final observation from these experiments, we highlight the consistently poor performance of smallest-margin uncertainty sampling (Unc. (SM)) computed in the Laplace Learning (Zhu et al., 2003a) model. Such catastrophic behavior of this type of uncertainty sampling has been observed previously (Ji and Han, 2012; Miller and Calder, 2023), and further supports our suggestion that exploration of cluster behavior is crucial for success in the active learning task. We posit that early on, when only a few of the clusters contain labeled samples, the resulting Laplace Learning classifier's decision boundaries are most likely too poor to utilize as the sole mechanism for selecting query points.

4.3.2 Comparing Computational Expense

In addition to the favorable explorative behavior of our proposed DiAL method, we emphasize the scalability of both the underlying semi-supervised learning model (Dirichlet

	Aux. Overhead	Query Point Selection		Semi-Sup. Inference
Abbr. Name	*Initial Compute*	*Cost Per Unl.*	*Aux. Update Cost*	*Class. Update Cost*
Unc. Sampl.	-	$\mathcal{O}(K)$	-	$K \cdot SpSolve(L, n)$
Dir. Var.	-	$\mathcal{O}(K)$	-	$SpSolve(L, n)$
VOpt (Full)	$\mathcal{O}(n^3)^*$	$\mathcal{O}(n)$	$\mathcal{O}(n^2)^*$	-
ΣOpt (Full)	$\mathcal{O}(n^3)^*$	$\mathcal{O}(n)$	$\mathcal{O}(n^2)^*$	-
MCVOpt	$\mathcal{O}(n^2 r)$	$\mathcal{O}(r + K)$	$\mathcal{O}(nr)$	$K \cdot SpSolve(L, n)$

Table 4: Computational comparison between acquisition functions, where $n = |\mathcal{X}|$, K is the number of classes, and $r \ll n$ is the number of eigenvalues computed in the auxiliary matrix used in MCVOpt. $SpSolve(L, n)$ represents the cost of a sparse linear solve with the graph Laplacian matrix $L \in \mathbb{R}^{n \times n}$, which we assume to be sparse pursuant our use of k-nearest neighbor graphs.

Learning) and the Dirichlet Variance acquisition function. That is, updating the Dirichlet Learning classifier at each active learning iteration and the subsequent calculation of Dirichlet Variance are relatively cheap to compute. Expanding on this second point, the computation of Dirichlet Variance scales similarly to the "optimal" scaling of Uncertainty Sampling (Table 4 and Figure 11) for this pool-based active learning setting. This scaling is very favorable compared to other acquisition functions like VOpt and ΣOpt that have been proposed to encourage exploration of clusters (Ma et al., 2013; Ji and Han, 2012; Miller and Bertozzi, 2024; Miller and Calder, 2023). In their originally proposed form, both VOpt and ΣOpt require the computation and storage of the inverse of a perturbed graph Laplacian matrix of size n^2, where n is the size of the data set $\mathcal{X}$. We have referred to this as VOpt/ΣOpt (Full) in the plots of results. MCVOpt (Miller et al., 2022) was proposed as a more computationally-efficient heuristic for combining VOpt and a type of uncertainty sampling, without requiring the inversion of said graph Laplacian. However, MCVOpt still requires the computation of a subset of $r \ll n$ eigenvalues and eigenvectors of said graph Laplacian matrix to provide a low-rank approximation to its inverse. We refer to these additional variables (i.e., the inverse of the graph Laplacian matrix or its low-rank approximation) as *auxiliary matrices* since they are not directly used by the underlying semi-supervised classifiers.

In contrast, the Dirichlet Variance acquisition function requires no such auxiliary matrix prior to the start of the active learning process and therefore has no additional computational overhead. Furthermore, VOpt, ΣOpt, and MCVOpt all require updating these auxiliary matrices in addition to updating the classifier outputs at each iteration. By requiring no such auxiliary matrix, Dirichlet Variance requires fewer operations per unlabeled point than these other acquisition functions.

In Table 4, we display a comparison of the computational requirements of the Dir. Var. , VOpt, ΣOpt, Unc. Sampling, and MCVOpt acquisition functions. *Initial Compute* refers to the computational cost to initially calculate the associated auxiliary matrix if used by the acquisition function. *Cost Per Unl.* refers to the cost of the acquisition function applied to a single unlabeled point and *Aux. Update Cost* refers to the cost to update said auxiliary matrix. Finally, *Class. Update Cost* refers to the cost of updating the corresponding classifier

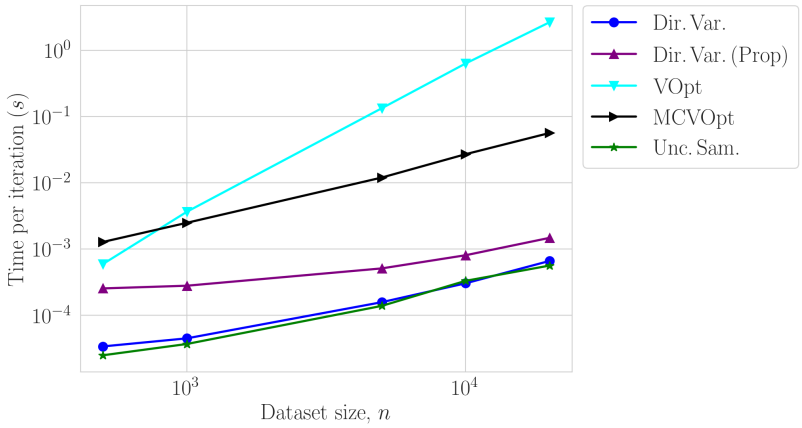


Figure 11: Timing comparison of acquisition function computation per iteration. For each acquisition function and data set size (n), we plot the average time it took to compute the corresponding acquisition function over the unlabeled data. Note that both Dir. Var. acquisition functions scale like $\mathcal{O}(n)$ similar to Unc. Sampling, while VOpt scales quadratically as $\mathcal{O}(n^2)$. MCVOpt scales roughly like $\mathcal{O}(n)$, with the additional overhead due to computations of order $\mathcal{O}(r)$, where r is the number of eigenvalues of the graph Laplacian used. Note that the additional cost incurred by sampling proportional to acquisition function values accounts for the timing difference between Dir. Var. and Dir. Var. (Prop).

at each iteration (if the acquisition function uses the classifier outputs). Costs with an asterisk ($*$) are shown in their originally proposed form—such as full matrix inversion of the graph Laplacian matrix and storage of manipulations of this $n \times n$ dense matrix throughout the active learning process for VOpt and ΣOpt.

Figure 11 shows a timing comparison between these acquisition functions. For each acquisition function, we computed the average time over 10 trials to perform one iteration of the active learning process (i.e., the time required to compute the acquisition function over the unlabeled data and subsequent selection of query point). Due to the near-identical form of ΣOpt and VOpt, we only include VOpt in the plot. Figure 11 empirically verifies the *Cost Per Unl.* column of Table 4; namely, one iteration of the active learning process incurs computational cost roughly like

$$(\text{cost per iteration}) = \underbrace{(\text{size of unlabeled data})}_{\approx \mathcal{O}(n)} \times (\text{cost per unlabeled point}).$$

Thus, VOpt (Full) scales quadratically with data set size, n, while the other acquisition functions scale roughly linearly with n, with additional overhead for MCVOpt due to the extra computations associated with the r eigenvalues and eigenvectors of the graph Laplacian matrix. As a result, to facilitate more comparisons on the largest numerical experiments we utilize a low-rank approximation of VOpt and ΣOpt; see Appendix D.2 for more details.

5. Theory

In this section, we present a theoretical analysis regarding the use of Dirichlet Active Learning (DiAL) to both (i) explore in low-data regimes and (ii) asymptotically exploit in high-data regimes. For analytical convenience, we will work in the continuum regime and make various assumptions upon the propagation operator and acquisition functions. However, these theoretical results help to match and explain the type of performance we observed in our numerical results (Section 4), and we will point out where some of the analysis could be extended to the discrete data setting or to other kernels and acquisition functions.

5.1 Assumptions and the Continuum Setting

For the sake of concreteness, we will consider a domain $\mathcal{X} \subset \mathbb{R}^d$ that is open and bounded throughout this section, though the analysis could be extended without significant change to a compact manifold with proper boundary conditions. Furthermore, we consider a joint distribution over inputs $(X, Y) \in \mathcal{X} \times [K]$ that is identified by a density $\nu(x, y)$, as well as the marginal distribution over the inputs identified by a density $\rho : \mathcal{X} \to [0, \infty)$. Lastly, we will assume that the class-conditional distributions have densities $\{\rho_k(x)\}_{k=1}^K$ and that the marginal density over inputs can be written as the mixture model $\rho(x) = \sum_{k=1}^K w_k \rho_k(x)$ where $\sum_{k=1}^K w_k = 1, w_k \geq 0$. It is worth noting that the properties of $\rho(x)$ capture the continuum analog of the clustering structure of a finite data set of points $\{x^i\}_{i=1}^n \subset \mathcal{X}$ when they are sampled $x \sim^{iid} \rho$; that is, regions of $\mathcal{X}$ wherein a class-conditional $w_k \rho_k(x)$ is significantly larger than the other components can be used to model clustering structure in data.

Finally, in contrast to how we have previously defined the domain of the acquisition function to be the unlabeled data, $\mathcal{U} = \mathcal{X} \setminus \mathcal{L}$, we will consider the Dirichlet Variance acquisition function evaluated on the entire domain $\mathcal{X}$ at each iteration in this continuum setting. This can be interpreted as allowing for "repeated trials" or observations at points in the domain $\mathcal{X}$. Furthermore, in the continuum setting, the labeled data $\mathcal{L}$ constitutes a discrete set of measure zero with respect to the underlying marginal density ρ, and so considering the acquisition function on the entire domain $\mathcal{X}$ seems natural.

5.2 Summary and Discussion of Theoretical Results

The numerical experiments in Section 4 indicate that DiAL flexibly transitions from low-label to high-label regimes, or in other words from an exploration phase to an exploitation phase. As such, our theory seeks to address both regimes. We begin in Section 5.3 by defining a type of *class-separating propagation operators* which are, with high probability, able to discriminate between different components of the mixture model (see Definition 3). This allows us to describe (Proposition 4) the explorative tendencies of DiAL to effectively cover the underlying distribution by providing reasonable guarantees that the algorithm will eventually capture all of the distributional structure and not miss components or clusters.

This flexible framework of a class-separating propagation operator generalizes several different notions of clustering previously used in the literature. In particular, we show in Section 5.4 that these assumptions are satisfied under a flexible definition of clustering described in (García Trillos et al., 2021), and we build upon their analysis in the context

of spectral clustering to show that Laplacian-based kernels will be discriminating for such clustering structures.

Of course, other notions of clustering have previously been utilized to provide exploration guarantees for active learning algorithms. These include, for example, explicit conditions on the inter- and intra-cluster distances (Murphy and Maggioni, 2019), ℓ^p balls that are well-separated (Karzand and Nowak, 2020), and high-density regions of the data-generating distribution that are well-separated (Miller and Calder, 2023). As in all of those works, at this stage we focus on exploration guarantees for our active learning algorithm. We follow the definitions in (García Trillos et al., 2021) due to (i) the direct connection to Laplacian-based methods, upon which the proposed methods in this work focus, and (ii) the fact that the assumptions given below are much less restrictive than many utilized in other similar works.

In Section 5.5 we then analytically study the manner in which DiAL will asymptotically seek more information near classification boundaries. In this case, our analysis is more formal particularly because one needs to be careful in designing a proper mathematical model that can capture what one means by asymptotic exploitation. To this end, we derive a large-sampling limit for DiAL which takes the form of an integro-differential equation (24). In a small kernel bandwidth limit, this equation demonstrates a clear sampling bias of DiAL towards regions with greater population-level classification uncertainty: a concrete statement of this phenomenon can be found in Equation (28). We consider this behavior to be asymptotically exploitative, in the sense that DiAL with Prop. Sampling (1) will, in high data regimes, spend most of its time sampling near decision boundaries if the scaling of λ is chosen properly. This is comparable to the behavior observed for uncertainty sampling, which focuses all of its attention on such regions, and contrasts strongly with the behavior of VOpt, which never transitions its attention towards decision boundaries. Our formal analysis also suggests that the scaling of the inverse-temperature parameter λ plays an important role in ensuring convergence to this stationary distribution that focuses along decision boundaries; we provide a numerical demonstration of this behavior in 1D in Section 5.5.1.

Other works have previously considered this broad question of exploitative behavior (i.e., the focusing of query point selection along ground truth decision boundaries between classes) (Dasgupta, 2011; Dasarathy et al., 2015; Balcan et al., 2007; Rittler and Chaudhuri, 2023). To our knowledge, however, there are relatively few algorithms with rigorous guarantees that can successfully *transition from exploring to exploiting*, and the analysis in this section demonstrates that this is the case with DiAL.

5.3 Exploration Guarantees: General Cluster Discovery

Our first goal will be to establish high-probability cluster discovery guarantees. At a high level, we will show that given K classes which are derived from K distinct clusters, then with high probability Dirichlet Learning will sample from each of the classes in K steps. The subsequent generalization to $C > K$ clusters follows in similar fashion and is addressed in Remark 6. Of course, such behavior is intimately connected with the particular choice of kernel, propagation function, and underlying probability densities. In this section, we will always assume that $\mathcal{X}$ is a subset of $\mathbb{R}^d$.

Our first aim will be to provide an abstract notion describing the ability of a particular kernel to separate classes. To this end, we give the following definition:

Definition 3 *We call a particular propagation operator $\mathcal{K}$ a $(\delta, \zeta, \varepsilon)$ class separator of a mixture model $\{\rho_i\}$ if there exist disjoint sets $\mathcal{X}_i$ such that $\rho_i(\mathcal{X}_i) \geq 1 - \delta$ and so that for any $x \in \mathcal{X}_i$ we have that $\mathcal{K}(x, x') > \zeta$ if $x' \in \mathcal{X}_i$ and $\mathcal{K}(x, x') < \varepsilon$ if $x' \in \mathcal{X}_j$ with $j \neq i$.*

This notion, of course, will be highly dependent upon the kernel and underlying densities, and we will prove that such a property holds for specific situations in Section 5.4. Under the assumption that our operator is separating, we then provide the following concrete result about cluster exploration, which follows from a direct, probabilistic argument.

Proposition 4 *Suppose that $\mathcal{K}$ is a $(\delta, \zeta, \varepsilon)$ class-separating propagation operator used in the Dirichlet Learning classifier. If query points are selected using Dirichlet Variance with Prop. Sampling (see Table 1), then with probability*

$$\geq \; 1 - \frac{K}{(1-\delta)w_{min}} \left[\delta e^{2\lambda/\alpha_0^2} + (1 - (1-\delta)w_{min}) e^{\lambda(C-1)/(K(\alpha_0+\varepsilon)+1)} \right] - K^2 w_{max} \delta$$

Dirichlet Active Learning (DiAL) will sample from each of the K classes in K steps, where

$$C := C(\alpha_0, \varepsilon, \zeta, K) := \left(1 + \frac{4(\alpha_0 + 1)}{K(\alpha_0 + \varepsilon)} \right) \left(\frac{(K-1)(\alpha_0 + \varepsilon)^4}{(K+1)\alpha_0^2 \left(\alpha_0 + \frac{\zeta}{K} \right)^2} \right) \tag{10}$$

and w_{min}, w_{max} are respectively the smallest and largest class weightings.

Remark 5 *Note that the bound in Proposition 4 is meaningful when $C(\alpha_0, \varepsilon, \zeta, K) < 1$ so that the exponential term can be reasonably bounded as $\lambda > 0$ is increased. The inner term will always worsen as λ increases, but this term is kept small by the constant $\delta \ll 1$. Consider the elucidating case when $\varepsilon = 0, \zeta = 1$, and $\delta = 0$ (i.e., $\mathcal{K}$ "perfectly" separates the classes in the mixture ρ) and $\alpha_0 = \frac{1}{K^2}$, then straightforwardly we have that*

$$C\left(\frac{1}{K^2}, 0, 1, K \right) = \left(1 + \frac{4(K^2+1)}{K} \right) \frac{(K-1)}{(K+1)^3} \leq \frac{1}{2} < 1,$$

and so our probability estimate becomes

$$1 - \frac{K}{w_{min}} \left((1 - w_{min}) e^{-\lambda K/2} \right),$$

which evidently goes to 1 as $\lambda \to \infty$.

Example 4 *The most straightforward application of the previous results would be in the setting when we use radial basis functions with support on $B(0, R)$ and the supports of the different mixture components are separated by a distance greater than R and each cluster has diameter less than R. In that context, it is immediate to check that the kernel would be a class separator with $\delta, \varepsilon = 0$ and with ζ determined by the kernel and the maximum diameter of the clusters. By making appropriate choices of α_0, λ, we can then make the probability of sampling from the unexplored clusters arbitrarily close to one. This type of result is analogous to the cluster exploration results given in (Karzand and Nowak, 2020).*

Of course, the simplified setting of Example 4 (used in (Karzand and Nowak, 2020)) requires rather restrictive assumptions regarding the data components in order to obtain reasonable exploration guarantees. In the next subsection, we provide details for a more flexible, data-adaptive framework in which we can demonstrate the class-separation property.

Proof [Proof of Proposition 4] We will provide a proof by induction. Suppose that the first $J < K$ samples belong to the sets $\mathcal{X}_{i_j}$ with $i_j \neq i_k$ when $j \neq k$, and that $y_j = i_j$, where the $\mathcal{X}_i$ are as in Definition 3. With $C(\alpha_0, \varepsilon, \zeta, K)$ as defined in (10) and letting $\omega_{min} = (1-\delta)w_{min}$, we claim that with probability at least

$$(1 - Kw_{max}\delta)\left(1 - \frac{\delta}{\omega_{min}} \exp\left(\frac{2\lambda}{\alpha_0^2}\right) - \frac{(1-\omega_{min})}{\omega_{min}} \exp\left(\frac{\lambda\left(C(\alpha_0, \varepsilon, \zeta, K) - 1\right)}{K(\alpha_0 + \varepsilon) + 1}\right)\right)$$

that x_{J+1} will belong to $\mathcal{X}_{i_{J+1}}$, with $i_{J+1} \neq i_j$, for all $j \leq J$, and that $y_{J+1} = i_{J+1}$. In words: with the given probability, the subsequent sample will be from a $\mathcal{X}_i$ for which we have not yet observed a label, and will have the correct label associated with $\mathcal{X}_i$.

We notice that for any $x \in \mathcal{X}_{i_j}$, for some $j = 1, \ldots, J$, we have that $\alpha_{i_j}(x) > \zeta + \alpha_0$, whereas all other $\alpha_i(x)$ are less than $\varepsilon + \alpha_0$. This then implies for $x \in \mathcal{X}_{i_j}$ that

$$\mathcal{A}_{var}(x) = \frac{2\sum_{i \neq k} \alpha_i \alpha_k}{\beta^2(\beta+1)} \leq \frac{4(K-1)(\alpha_0+\varepsilon)(\alpha_0+1) + K(K-1)(\alpha_0+\varepsilon)^2}{(K\alpha_0+\zeta)^2(K\alpha_0+\zeta+1)} =: \mathcal{A}_{var}^{lab}.$$

On the other hand, for any $x \in \mathcal{X}_i$ such that $i_j \neq i$ for all $j = 1, \ldots, J$ we have that $\alpha_k(x) < \varepsilon$ for all k. We then obtain that

$$\mathcal{A}_{var}(x) = \frac{2\sum_{i \neq k} \alpha_i \alpha_k}{\beta^2(\beta+1)} \geq \frac{K(K+1)\alpha_0^2}{K^2(\alpha_0+\varepsilon)^2(K(\alpha_0+\varepsilon)+1)} =: \mathcal{A}_{var}^{unl}.$$

Let $\hat{\mathcal{X}}$ denote the complement of all of the $\mathcal{X}_i$. We notice that at all points we have that the variance is smaller than

$$\mathcal{A}_{var}(x) \leq \frac{K(K+1)(\alpha_0+1)^2}{(K\alpha_0)^2(K\alpha_0+1)} = \frac{(K+1)(\alpha_0+1)^2}{K\alpha_0^2(K\alpha_0+1)} =: \mathcal{A}_{var}^{back}.$$

Now, we notice that the normalizing constant for our sampling distribution will be at least

$$\int_{\mathcal{X}} e^{\lambda V(\alpha(x))}\rho(x)\,dx \geq \sum_{i_k : k > J} w_{i_k} \int_{\mathcal{X}_{i_k}} e^{\lambda V(\alpha(x))}\rho_{i_k}(x)\,dx \geq (1-\delta)w_{min} \exp\left(\lambda \mathcal{A}_{var}^{unl}\right).$$

Now, we can bound

$$\begin{aligned}
\mathcal{A}_{var}^{back} - \mathcal{A}_{var}^{unl} &\leq \frac{(K+1)(2\alpha_0+1)}{K\alpha_0^2(K\alpha_0+1)} \\
\frac{\mathcal{A}_{var}^{lab}}{\mathcal{A}_{var}^{unl}} &= \left(\frac{4(K-1)(\alpha_0+\varepsilon)(\alpha_0+1) + K(K-1)(\alpha_0+\varepsilon)^2}{K(K+1)\alpha_0^2}\right) \\
&\quad \times \left(\frac{K^2(\alpha_0+\varepsilon)^2(K(\alpha_0+\varepsilon)+1)}{(K\alpha_0+\zeta)^2(K\alpha_0+\zeta+1)}\right)
\end{aligned}$$

$$\leq \frac{(K-1)(\alpha_0+\varepsilon)^4}{(K+1)\alpha_0^2\left(\alpha_0+\frac{\zeta}{K}\right)^2}\left(1+\frac{4(\alpha_0+1)}{K(\alpha_0+\varepsilon)}\right)$$
$$= C(\alpha_0,\varepsilon,\zeta,K),$$

so that

$$\mathcal{A}_{var}^{lab} - \mathcal{A}_{var}^{unl} \leq (C(\alpha_0,\varepsilon,\zeta,K)-1)\,\mathcal{A}_{var}^{unl}.$$

Hence we have that the probability of sampling in either $\hat{\mathcal{X}}$ or from one of the already labeled clusters $\mathcal{X}_{i_j}$ will be at most

$$\begin{aligned}
\mathbb{P}\left(X_{J+1} \in \cup_{i_k:k\leq J}\mathcal{X}_{i_k} \cup \hat{\mathcal{X}}\right) &= \mathbb{P}\left(X_{J+1}\in\hat{\mathcal{X}}\right) + \mathbb{P}\left(X_{J+1}\in\cup_{i_k:k\leq J}\mathcal{X}_{i_k}\right)\\
&\leq \frac{\int_{\hat{\mathcal{X}}} e^{\lambda V(\alpha(x))}\rho(x)\,dx + \int_{\cup_{i_k:k\leq J}\mathcal{X}_{i_k}} e^{\lambda V(\alpha(x))}\rho(x)\,dx}{\int_{\mathcal{X}} e^{\lambda V(\alpha(z))}\rho(z)\,dz}\\
&\leq \frac{\delta}{(1-\delta)w_{min}}\exp\left(\lambda\left(\mathcal{A}_{var}^{back}-\mathcal{A}_{var}^{unl}\right)\right)\\
&\quad + \frac{(1-w_{min}(1-\delta))}{(1-\delta)w_{min}}\exp\left(\lambda\left(\mathcal{A}_{var}^{lab}-\mathcal{A}_{var}^{unl}\right)\right)\\
&< \frac{\delta}{(1-\delta)w_{min}}\exp\left(\frac{\lambda(K+1)(2\alpha_0+1)}{K\alpha_0^2(K\alpha_0+1)}\right)\\
&\quad + \frac{(1-w_{min}(1-\delta))}{(1-\delta)w_{min}}\exp\left(\lambda\left(C(\alpha_0,\varepsilon,\zeta,K)-1\right)\mathcal{A}_{var}^{unl}\right),
\end{aligned}$$

where the constant $C(\alpha_0,\varepsilon,\zeta,K)$ satisfies

$$\frac{\mathcal{A}_{var}^{lab}}{\mathcal{A}_{var}^{unl}} \leq C(\alpha_0,\varepsilon,\zeta,K) := \left(1+\frac{4(\alpha_0+1)}{K(\alpha_0+\varepsilon)}\right)\left(\frac{(K-1)(\alpha_0+\varepsilon)^4}{(K+1)\alpha_0^2\left(\alpha_0+\frac{\zeta}{K}\right)^2}\right) < 1.$$

We further simplify this expression for the probability of interest as:

$$\begin{aligned}
\mathbb{P}\left(X_{J+1}\in\cup_{i_k:k\leq J}\mathcal{X}_{i_k}\cup\hat{\mathcal{X}}\right) &\leq \frac{\delta}{(1-\delta)w_{min}}\exp\left(\frac{2\lambda}{\alpha_0^2}\right)\\
&\quad + \frac{(1-(1-\delta)w_{min})}{(1-\delta)w_{min}}\exp\left(\frac{\lambda(C(\alpha_0,\varepsilon,\zeta,K)-1)}{K(\alpha_0+\varepsilon)+1}\right)\\
&= \frac{\delta}{\omega_{min}}\exp\left(\frac{2\lambda}{\alpha_0^2}\right) + \frac{(1-\omega_{min})}{\omega_{min}}\exp\left(\frac{\lambda(C(\alpha_0,\varepsilon,\zeta,K)-1)}{K(\alpha_0+\varepsilon)+1}\right).
\end{aligned}$$

It only remains to estimate the probability that a sample which is in $\mathcal{X}_{i_{J+1}}$ will have label $y_{J+1} = i_{J+1}$. Using the definition of our class separators, this conditional probability is larger than $1 - Kw_{max}\delta$.

This proves our induction step, and then by iterating this bound (which was constructed to be independent of the particular group of indices i_j) we obtain our desired result by using the inequality $(1-z)^K \geq 1 - Kz$. ■

Remark 6 *We briefly note that while Definition 3 and Proposition 4 are stated in terms of K clusters (one for each of the K different classes), this setup straightforwardly generalizes to the situation of multiple, disjoint clusters in each class. If there are a total of M different clusters—each one belonging to precisely one of the K distinct classes—then the same reasoning used to lower bound the Dirichlet Variance values on the unexplored classes extends to the unexplored clusters belonging to classes that have labeled points in other clusters.*

5.4 Exploration Guarantees: Laplacian-based Class Separation

The results of the previous section provide exploration guarantees for DiAL, under specific assumptions on the propagation operator and upon the underlying class-conditional probabilities. However, most of the computational examples which we gave in Section 4 utilized propagation operators which were based upon graph Laplacians, and the corresponding properties of the propagation operator are not immediately obvious. Accordingly, in this Section we seek to establish class separation guarantees for such propagation operators. As we do so, we make a number of choices in order to simplify the analysis, and we attempt to justify them as they are introduced.

5.4.1 Heat Kernel Propagation for Theoretical Analysis

While we have focused on the graph-based Poisson propagation operator introduced in (6) and (7) for our numerical results, we turn our attention in the continuum regime to a heat-kernel propagation operator as opposed to the continuum-limit analog of the Poisson propagation. For a domain $\mathcal{X} \subset \mathbb{R}^d$, the *density-dependent* heat kernel $\mathcal{K}_t(z, x)$ with source $z \in \mathcal{X}$ that solves

$$\begin{cases} \partial_t \mathcal{K}_t(z,x) - \Delta_\rho \mathcal{K}_t(z,x) = 0 & x \in \mathcal{X}, t > 0 \\ \mathcal{K}_0(z,x) = \delta_z(x) \end{cases} \tag{11}$$

where the diffusion operator

$$\Delta_\rho = \frac{1}{\rho} \operatorname{div}(\rho \nabla \cdot) \tag{12}$$

is *self-adjoint* with respect to the ρ-weighted inner product. We note that the differential operators in (11) are with respect to the variable x, while the variable z is the source. Furthermore, depending on the domain $\mathcal{X}$, one must assume boundary conditions to make (11) well-defined. In general, we will state the necessary assumptions on the heat kernel, $\mathcal{K}_t(z, x)$, and provide reasonable example situations in which these assumptions hold.

We choose to use the heat kernel for a number of reasons. One main reason that we choose not to analytically study Poisson propagation is that it induces analytical complications in continuum settings. For example, it is still an open problem to establish a formal continuum limit for the graph-based Poisson propagation operator (Calder et al., 2020) due to the highly singular nature of the right-hand side of the corresponding governing equation

$$-\Delta_\rho u + \tau u = (\tau I - \Delta_\rho) u = \delta_z(x).$$

Furthermore, by considering the fundamental solution of the Laplacian on $\mathbb{R}^d$, we expect solutions of this equation to go to infinity at z, a point which is handled in a delicate way

by solution rescaling in (Calder et al., 2020). The heat equation, in contrast, is much more amenable to analysis in the continuum setting.

We note, however, that the Poisson propagation can be viewed as an approximation to a corresponding heat-kernel propagation. Informally, one could consider using a single backwards Euler step of the heat equation to write

$$\mathcal{K}_t(z,x) = e^{t\Delta_\rho}\delta_z(x) \approx (I - t\Delta_\rho)^{-1}\delta_z(x) = tu_{1/t}(x),$$

where $(\tau I - \Delta_\rho)u_\tau(x) = \delta_z(x)$ is the continuum-limit analog of the Poisson propagation (6). This shows the relationship between using the heat kernel propagation $\mathcal{K}_t(z,x)$ and the Poisson propagation, and so we continue with our theory in the heat kernel setting in the continuum.

5.4.2 A Well-separated Mixture Model

Many of the examples throughout the paper focus on Laplacian-based propagation operators. These kernels are data-adapted, and it is natural to guess that they will therefore be well-adapted to cluster separation. An analog of the type of separation that we consider here has recently been studied in the context of spectral clustering in (García Trillos et al., 2021). In the interest of clarity, we only consider population-level distributions and mixture models, but their work also considers guarantees for finite samples of such a mixture, and the results we give in this section should extend to that setting as well.

One starting point from (García Trillos et al., 2021) is the concrete description that they give for a mixture model to be "well-separated" in an appropriate sense.

Definition 7 (García Trillos et al. (2021)) *Consider a probability density*

$$\rho(x) = \sum_{k=1}^{K} w_k \rho_k(x)$$

on $\mathbb{R}^d$ with $w_k \geq 0$ and ρ_k being probability densities. We call such a density a mixture model*, and each of the ρ_k the mixture components. We then define the following parameters:*

1. ***Overlapping.*** *The overlapping, $\mathcal{S}$, of a mixture model is defined to be*

$$\mathcal{S} := \max_{i \neq j} \int \frac{\rho_i \rho_j}{\rho}\, dx,$$

2. ***Coupling.*** *The coupling, $\mathcal{C}$, of a mixture model is defined to be*

$$\mathcal{C} := \max_{k} \int \left| \frac{\nabla \rho_k}{\rho_k} - \frac{\nabla \rho}{\rho} \right|^2 \rho_k\, dx,$$

3. ***Indivisibility.*** *The indivisibility, Θ, of the mixture model is defined to be*

$$\Theta := \min_{k} \min_{u \perp 1, \int u^2 \rho_k = 1} \int |\nabla u|^2 \rho_k.$$

In García Trillos et al. (2021) the authors' definition also applies to probability distributions defined on manifolds. We choose not to state our results in terms of manifolds as it doesn't align with other portions of this work, but all of the results in this subsection would apply in the manifold setting as well. One of the main assumptions (Assumption 7 in García Trillos et al. (2021)) in their work is that

$$\begin{cases} \rho, \rho_k \in C^1 \\ \Delta_\rho, \Delta_{\rho_k} \text{ have discrete spectrum} \\ L^2(\rho), L^2(\rho_k) \text{ each have an orthogonal eigenbasis} \end{cases} \tag{S}$$

where Δ_ρ and Δ_{ρ_k} are the Laplace-Beltrami operators depending on the densities, ρ and ρ_k, respectively (see (12)). We will take this as a standing assumption throughout this section.

In the previous definition, the overlapping $\mathcal{S}$ describes the amount to which pairs of the mixtures have coincident densities, relative to the underlying density. In a case where the mixtures have disjoint supports, this parameter will be zero. The indivisibility Θ is a measure of how difficult it is, in terms of Dirichlet energy, to divide any one of the clusters: this parameter would be large in the case of strongly log-concave mixture components. The coupling parameter $\mathcal{C}$ is somewhat more subtle, and measures a type of relative entropy between ρ and ρ_k: again this parameter would be zero for disjoint mixture components.

García Trillos et al. (2021) then define a *well-clustered mixture model* via the relationships

$$\mathcal{S} \ll 1 \quad \text{and} \quad \frac{\mathcal{C}}{\Theta} \ll 1. \tag{13}$$

These conditions, of course, require some knowledge of the underlying distributions, and may be difficult to verify for a particular data set. However, most natural models of well-clustered data would satisfy these assumptions. For example, in the case where we have K clusters with disjoint support we will have $\mathcal{S}$ and $\mathcal{C}$ both equal to zero. They would also hold for Gaussian mixtures which are sufficiently separated (and $\mathcal{S}$ and $\mathcal{C}$ could be quantified in terms of the means and variances). In this sense, we view these conditions as applicable to many different models of separated data components.

5.4.3 Laplacian-based Exploration Main Result

With these definitions in hand, we now state our main exploration result, which guarantees that clusters are explored efficiently: this proposition largely turns out to be a direct consequence of the estimates given by García Trillos et al. (2021). In summary, we have that Proposition 10 gives explicit values for which the heat equation kernel $\mathcal{K}_t(x^*, x) = e^{t\Delta_\rho}\delta_{x^*}$ is a $(\delta, \zeta, \varepsilon)$ class separator for the mixture model with parameters $\mathcal{S}, \mathcal{C}, \Theta$ by using values determined by Proposition 12 presented thereafter. For convenience, we begin by stating Theorem 8 as a simplified result to provide intuition regarding the main results.

Theorem 8 *Consider a mixture model density ρ with parameters $(\mathcal{S}, \mathcal{C}, \Theta)$ satisfying Assumption* (S)*, and let $\mathcal{K}_t(x^*, x) = e^{t\Delta_\rho}\delta_{x^*}$ be the heat kernel with Neumann boundary conditions and parameter t. Suppose further there exist constants $\kappa, M > 0$ such that*

$$\mathcal{C} \leq \kappa, \quad \mathcal{S} \leq \kappa, \quad \Theta \geq \sqrt{\kappa},$$

and $\|e^{\Delta_\rho t}\|_{L^1(\rho)\to L^\infty(\rho)} \le \sqrt{M}$. *Then,* $\mathcal{K}_t$ *is a* $(\delta,\zeta,\varepsilon)$ *class separator with*

$$\delta \le C\kappa^{1/4}, \quad \zeta \ge w_{max}^{-1} - C\left(|\log(\kappa)\kappa^{1/2} + M\kappa)\right), \quad \varepsilon \le C\left(\kappa^{1/4} + M\kappa\right),$$

where C *is a constant that depends only upon* K, w, *and* $\|\rho\|_\infty$.

Theorem 8 simply states specific values of δ,ζ,ε for which the continuum heat kernel defined on a well-separated mixture model ρ is indeed a class separator. In light of our generalized exploration result (Proposition 4), this implies that using a Laplacian-based heat kernel for the propagation operator in DiAL will efficiently explore the clustering structure (i.e., selecting query points in each of the different clusters of the mixture model).

By combining Theorem 8 and Proposition 4 we then immediately obtain the following cluster exploration result:

Corollary 9 *Consider a mixture model as in Theorem 8, assume that* t *is chosen so that* $M \le \kappa^{-1/2}$, $\alpha_0 = w_{max}^{-1}K^{-2}$, $\lambda \sim -\frac{1}{16w_{max}^2 K}\log\kappa$ *and assume that* $\kappa \ll 1$. *Then, for some exponent* $\alpha > 0$ *and constant* $C > 0$, *with probability at least* $1 - C\kappa^\alpha$, *DiAL with the general heat kernel propagation will sample from all* K *classes in exactly* K *steps.*

Corollary 9 intuitively shows that the (data-dependent) heat kernel propagation $\mathcal{K}_t$ allows DiAL to efficiently explore the clustering structure (i.e., select query points in each of K clusters in K queries) of a well-clustered mixture model with high probability.

We continue with our main result (Proposition 10) that provides explicit expressions for which the heat-kernel is a $(\delta,\zeta,\varepsilon)$ class separator for a well-separated mixture model with general values of $\mathcal{C},\mathcal{S}$, and Θ. We note that Theorem Theorem 8 is, after tracking constants, a simplified version of this Proposition.

Proposition 10 *Consider a mixture model with parameters* $(\mathcal{S},\mathcal{C},\Theta)$ *which satisfies Assumption* (S). *Let* $\mathcal{K}_t(x^*,x)$ *be the heat kernel with parameter* t, *namely the solution to the heat equation satisfying* $\mathcal{K}_t(x^*,x) = e^{t\Delta_\rho}\delta_{x^*}$ *with Neumann boundary conditions. Consider parameters* a, b, σ *as in Proposition 12. Then, under the assumptions on parameters stated in Proposition 12, the kernel* $\mathcal{K}_t$ *will be a* $(\delta,\zeta,\varepsilon)$ *class separator with*

$$\delta = \tilde{\delta} + a^2, \quad \zeta = \frac{(1-\sin(b))^2}{w_{max}}\cos(2(\sigma+b)) - \xi, \quad \varepsilon = \frac{(1+\sin(b))^2}{w_{min}}\cos(\pi/2 - 2(\sigma+b)) + \xi,$$

where

$$\begin{aligned}\xi := |e^{\lambda_K t} - 1|\ &\frac{(1+\sin(b))^2}{w_{min}}\|\rho\|_\infty \\ &+ \min_{t_1,s>0,t_1+s<t} e^{\lambda_{K+1}(t-t_1-s)}\|e^{t_1\Delta_\rho}\|_{L^1(\rho)\to L^2(\rho)}\|e^{s\Delta_\rho}\|_{L^2(\rho)\to L^\infty(\rho)}.\end{aligned}$$

Proof We consider a sequence of eigenvalue, eigenfunction pairs (λ_k, e_k) of the operator Δ_ρ, normalized in the ρ-weighted L^2 norm. Define $g(y) = e^{t_1\Delta}\delta_{x^*} - \sum_{k=1}^K e^{\lambda_k t_1}e_k(x^*)\rho(x^*)e_k(y)$, with t_1 being a parameter that we choose later. We can interpret g as the projection of the solution to the heat equation onto the "high" Fourier modes. We find that

$$\mathcal{K}_t(x^*,x) = \sum_{k=1}^K e_k(x^*)\rho(x^*)e_k(x) \ + \ \sum_{k=1}^K (e^{\lambda_k t} - 1)e_k(x^*)\rho(x^*)e_k(x)$$

$$+ \; e^{s\Delta} \sum_{k=K+1}^{\infty} e^{\lambda_k(t-t_1-s)} e_k(x)\langle e_k, g\rangle_\rho. \tag{14}$$

We can then bound

$$\left|\sum_{k=1}^{K}(e^{\lambda_k t}-1)e_k(x)e_k(x^*)\rho(x^*)\right| \leq |e^{\lambda_K t}-1| \max_{k=1\ldots K} \|e_k\|^2_{L^\infty(E_k)}\|\rho\|_\infty,$$

where we will only need to estimate this term in regions (i.e., a family of disjoint sets E_k that are defined in Proposition 12, for the purposes of proving $(\delta,\zeta,\varepsilon)$ class separation) where the eigenfunctions e_k are not too large. We can also bound

$$\begin{aligned}\left|e^{s\Delta}\sum_{k=K+1}^{\infty} e^{\lambda_k(t-t_1)}e_k(x)\langle e_k,g\rangle_\rho\right| &\leq \|e^{s\Delta}\|_{L^2(\rho)\to L^\infty(\rho)}\left\|\sum_{k=K+1}^{\infty} e^{\lambda_k(t-t_1)}e_k(x)\langle e_k,g\rangle\right\|_{L^2(\rho)} \\ &\leq e^{\lambda_{K+1}(t-t_1)}\|g\|_{L^2(\rho)}\|e^{s\Delta}\|_{L^2(\rho)\to L^\infty(\rho)} \\ &\leq e^{\lambda_{K+1}(t-t_1-s)}\|e^{t_1\Delta}\|_{L^1(\rho)\to L^2(\rho)}\|e^{s\Delta}\|_{L^2(\rho)\to L^\infty(\rho)}.\end{aligned}$$

Hence we define

$$\begin{aligned}\xi := \min_{t_1,s>0,t_1+s<t} &|e^{\lambda_K t}-1| \max_{k=1\ldots K}\|e_k\|^2_{L^\infty(E_k)}\|\rho\|_\infty \\ &+ \quad e^{\lambda_{K+1}(t-t_1-s)}\|e^{t_1\Delta_\rho}\|_{L^1(\rho)\to L^2(\rho)}\|e^{s\Delta_\rho}\|_{L^2(\rho)\to L^\infty(\rho)}.\end{aligned}$$

In turn, we only need to estimate the first term in Equation (14). This is considered in Proposition 12. That Proposition also provides an upper bound upon $\|e_k\|^2_{L^\infty(E_k)}$, which completes the proof. ■

The quantity ξ tracks the error induced by using the heat kernel in two ways. The first term, which is proportional to $e^{\lambda_K t}-1$, is small when the lowest eigenvalues are too close to zero. The second term is, in contrast, reliant on higher-frequency information, and decays like $e^{\lambda_{K+1}t}$. The operator norm bounds are relatively standard in the literature on parabolic equations, and are primarily for analytical convenience in estimating the different terms. As the magnitude of ξ is highly dependent upon the difference in magnitude of the eigenvalues, we now recall the following result from García Trillos et al. (2021).

Proposition 11 (Propositions 27 and 38, García Trillos et al. (2021)) *Consider a mixture model with parameters $(\mathcal{S},\mathcal{C},\Theta)$ which satisfies Assumption* (S)*, and suppose that $K\mathcal{S}<1$. Then the following bounds hold:*

$$\left(\sqrt{\Theta(1-K\mathcal{S})}-\frac{\sqrt{\mathcal{C}K\mathcal{S}}}{1-\mathcal{S}}\right)^2 \leq \lambda_{K+1}$$
$$\lambda_K \leq \frac{K\mathcal{C}}{1-K\mathcal{S}^{1/2}}.$$

In turn, the main consideration in bounding ξ relates to the choice of t and the operator norm bounds on the heat kernel. The types of bounds that are implicitly assumed in the definition of ξ are called *ultracontractivity* estimates for the heat kernel, and are known to be finite under various classes of assumptions: a standard reference on the subject is (Davies, 1989). In particular, these bounds will hold for densities which are compactly supported and do not degenerate, or for mixtures whose densities are smooth and (asymptotically) log-concave. While for empirical data it will be challenging to verify bounds on ξ, there are many important models, including those used previously in the active learning literature (Karzand and Nowak, 2020; Cloninger and Mhaskar, 2021), where it is possible to bound ξ (e.g., Gaussian mixtures, uniform distributions on sets with very small overlap). Furthermore, if we restrict our "hypothesis class" to be densities for which the heat kernel obeys desired bounds (for example by assuming uniform log-concavity on the mixture components), then we can completely control all of the remaining terms in Proposition 10 using the parameters $\mathcal{S}, \mathcal{C}, \Theta$. As a simple illustration, we provide an example detailing the scaling of the associated constants in the context of a simple 2-component Gaussian mixture model in Appendix E.

Now, we turn to a key estimate for the heat kernel, regarding the effect of the low Fourier modes after neglecting decay. For convenience, and following the notation in (García Trillos et al., 2021), we define the spectral embedding F to be the mapping from $\mathbb{R}^d$ to $\mathbb{R}^K$ given by

$$F(x) := (e_i(x))_{i=1}^K.$$

Proposition 12 *Consider a mixture model with parameters $(\mathcal{S}, \mathcal{C}, \Theta)$. We define the parameters*

$$\Xi := K\frac{(\Lambda - \sqrt{\mathcal{S}})^2}{4} + 4K^{3/2}\left(\frac{1}{\sqrt{1-K\Lambda}} - 1\right),$$

with

$$\Lambda := 4\left(\sqrt{\frac{\Theta(1-K\mathcal{S})}{\mathcal{C}} - \frac{\sqrt{K\mathcal{S}}}{(1-\mathcal{S})}}\right)^{-1} + \sqrt{\mathcal{S}}.$$

We assume that

$$a, b > 0 \tag{15}$$

$$\frac{a\sin(b)}{\sqrt{w_{max}}} \geq \sqrt{\Xi} \tag{16}$$

$$\mathcal{S} < \frac{w_{min}(1-\cos^2(\sigma))}{w_{max}\cos^2(\sigma)K^2} \tag{17}$$

$$\Lambda - \sqrt{\mathcal{S}} > 0 \tag{18}$$

$$\Lambda K < 1 \tag{19}$$

$$b + \sigma < \frac{\pi}{4} \tag{20}$$

and we let

$$\tilde{\delta} := \frac{w_{max}K^2\mathcal{S}}{w_{min}^2(1-\cos^2(\sigma))} + \frac{1-\cos^2(\sigma)}{w_{min}}$$

Then there exists disjoint sets E_k so that $\rho_k(E_k) > 1 - (\tilde{\delta} + a^2)$, which satisfy the inequalities, for $x \in E_j, x^* \in E_k$

$$\left|\sum_{k=1}^{K} e_k(x)e_k(x^*)\right| = |\langle F(x), F(x^*)\rangle| \leq \frac{(1+\sin(b))^2}{w_{min}} \cos\left(\frac{\pi}{2} - 2(\sigma + b)\right) \qquad \text{if } j \neq k \quad (21)$$

$$\left|\sum_{k=1}^{K} e_k(x)e_k(x^*)\right| = |\langle F(x), F(x^*)\rangle| \geq \frac{(1-\sin(b))^2}{w_{max}} \cos(2(\sigma + b)) \qquad \text{if } j = k \quad (22)$$

$$\left|\sum_{k=1}^{K} e_k(x)e_k(x^*)\right| = |\langle F(x), F(x^*)\rangle| \leq \frac{(1+\sin(b))^2}{w_{min}} \qquad \text{if } j = k \quad (23)$$

Proof The ingredients for this proof are all present in the proof of Theorem 10 in García Trillos et al. (2021). However, their statements are all made in terms of the overall probability of points which are in the support of a set of approximately orthogonal cones: in other words, they estimate probabilities in terms of ρ. This was natural in their unsupervised setting, whereas in ours we are closer to the supervised setting and care about the labels associated with each mixture component; that is, we want to estimate the size of sets in terms of the ρ_k. We simply sketch how the estimates we use can be obtained by small modifications of their proofs.

In one of the first main steps in their proof, they identify the functions $\tilde{q}_k := \sqrt{\frac{\rho_k}{\rho}}$ as a system of approximate eigenfunctions, which are almost orthogonal in a way that can be quantified by $\mathcal{S}$. Specifically, in the proof of Proposition 29, given an angle $0 < \sigma < \frac{\pi}{4}$ they define the disjoint sets

$$A_k := \left\{ x : \rho_k(x) > \cos^2(\sigma) \sum_{j=1}^{K} \rho_j(x) \right\}$$

They then prove that

$$\rho\left(\bigcup_{k=1}^{K} A_k\right) \geq 1 - \frac{w_{max}K^2\mathcal{S}}{w_{min}(1-\cos^2(\sigma))}.$$

We now extend their estimate to apply to the conditional probabilities ρ_k, as needed in estimating the class separator parameters in Definition 3. We first notice that the previous inequality immediately implies that

$$\rho_k\left(\left(\bigcup_{j=1}^{K} A_j\right)^c\right) \leq \frac{w_{max}K^2\mathcal{S}}{w_{min}^2(1-\cos^2(\sigma))}.$$

On the other hand, by the definition of the A_k we have that for $j \neq k$

$$w_j\rho_k(A_j) \leq (1-\cos^2(\sigma))w_j\rho_j(A_j) \leq (1-\cos^2(\sigma))\rho(A_j),$$

which after summing over $j \neq k$ gives

$$w_{min}\rho_k\left(\bigcup_{j\neq k} A_j\right) \leq (1-\cos^2(\sigma)).$$

This then implies that

$$\rho_k(A_k^c) \leq \frac{w_{max}K^2\mathcal{S}}{w_{min}^2(1-\cos^2(\sigma))} + \frac{1-\cos^2(\sigma)}{w_{min}} =: \tilde{\delta}.$$

In terms of embeddings, they define the mapping $F^Q := (\tilde{q}_k)_{k=1}^K$. The proof of Theorem 10 in their paper shows that there exists an orthonormal matrix O so that

$$W_2^2(OF_\sharp\rho, F_\sharp^Q\rho) \leq K\frac{(\Lambda-\sqrt{\mathcal{S}})^2}{4} + 4K^{3/2}\left(\frac{1}{\sqrt{1-K\Lambda}}-1\right) =: \Xi,$$

with

$$\Lambda := 4\left(\sqrt{\frac{\Theta(1-K\mathcal{S})}{\mathcal{C}} - \frac{\sqrt{K\mathcal{S}}}{(1-\mathcal{S})}}\right)^{-1} + \sqrt{\mathcal{S}}$$

Here we have used the notation $F_\sharp\rho$ to denote the standard push-forward measure, which is defined to be the measure so that $F_\sharp\rho(B) := \rho(F^{-1}(B))$ for Borel sets B.

Using the definition of the Wasserstein distance, we can also infer that

$$W_2^2(OF_\sharp\rho_k, F_\sharp^Q\rho_k) \leq \frac{\Xi}{w_{min}}.$$

We also notice that

$$F^Q(A_k) \subset \left\{z : \frac{z_k}{|z|} > \cos(\sigma), \frac{1}{\sqrt{w_{max}}} \leq |z| \leq \frac{1}{\sqrt{w_{min}}}\right\}.$$

In words, the embedding F^Q separates the A_k into orthogonal cones with angle σ and places the mass in A_k at least distance $\frac{1}{\sqrt{w_{max}}}$ from the origin.

Finally, in Proposition 22 of their paper they show stability bounds of cones under the Wasserstein distance. In particular, given a vector $v \neq 0$ and an angle parameter $0 < \sigma < \frac{\pi}{4}$, if we let the set C be a cone defined by

$$C := \left\{x : \frac{\langle v, x\rangle}{|v||x|} \geq \cos(\sigma)\right\}$$

then we can define the set $C_r := C \cap (B(0,r))^c$. By following the proof of Proposition 22 by García Trillos et al. (2021), as applied to one cone as opposed to an orthogonal system of cones, one has that, for a probability measure μ_1 with $\mu_1(C_r) > 1-\delta$, then for any $a, b > 0$ satisfying

$$ra\sin(b) \geq W_2(\mu_1, \mu_2)$$

and assuming that $0 < \sigma + b < \pi/4$, we will also have that $\mu_2(\tilde{C} \cap (B(0,\tilde{r}))^c) > 1-\hat{\delta}$, with $\tilde{C}$ being the cone centered on v with angle smaller than $\sigma + b$, $\tilde{r} = r(1-\sin(b))$ and $\hat{\delta} = \delta + a^2$. We furthermore note that their proof could be extended so that if we have $\mu_1(C \cap (B(0,R) \setminus B(0,r)))$ with $R > r$ then we will have $\mu_2(\tilde{C} \cap (B(0,\tilde{R}) \setminus B(0,\tilde{r})))$ with $\tilde{R} = R(1+\sin(b))$.

Putting these facts together, we then have that if our parameters satisfy equations (16)-(20) and if we define a cone centered on the k-th coordinate vector with angle $\sigma + b$ to be C_k, then for $\tilde{r} = \frac{1-\sin(b)}{\sqrt{w_{max}}}$ and $\tilde{R} = \frac{1+\sin(b)}{\sqrt{w_{min}}}$ and letting $\tilde{C}_k = C_k \cap (B(0,\tilde{R}) \setminus B(0,\tilde{r}))$ we have

$$OF_\sharp \rho_k(\tilde{C}_k) \geq 1 - (\tilde{\delta} + a^2).$$

We notice that if $z \in \tilde{C}_k$ and $z' \in \tilde{C}_j$ with $j \neq k$ then we have that

$$\langle z, z' \rangle \leq \tilde{R}^2 \cos(\pi/2 - 2(\sigma + b)).$$

On the other hand, if $j = k$ we have that

$$\tilde{r}^2 \cos(2(\sigma + b)) \leq \langle z, z' \rangle \leq \tilde{R}^2.$$

This then concludes the proof.

■

Finally, we give the proof of Theorem 8 which presents a simplified result to give intuition regarding the implications of our main theoretical result (Proposition 10).

Proof (Theorem 8). This simply amounts to choosing the parameters a, b, σ appropriately, and then bounding all of the terms. Throughout this proof, we let C be a constant that varies line by line, and may depend upon $K, w_{min}, w_{max}, \|\rho\|_\infty$, but does not depend upon the other parameters. First, we notice that

$$\left(\sqrt{\frac{\Theta(1 - K\mathcal{S})}{\mathcal{C}} - \frac{\sqrt{K\mathcal{S}}}{(1 - \mathcal{S})}} \right)^{-1} \leq C\kappa^{1/4}$$

In turn

$$\Xi \leq C\kappa^{1/2}$$

Now let $a = \kappa^{1/4}$ and $b = \kappa^{1/4}$, and $\sigma = \kappa^{1/4}$. Then $\tilde{\delta} \leq C\kappa^{1/2}$.

We also note that using our assumptions we have that $|\lambda_K| \leq \kappa$ and $|\lambda_{K+1}| \geq C\kappa^{1/2}$. Therefore by setting $t = -\log(\kappa)\kappa^{-1/2}$ we have that $|e^{\lambda_K t} - 1| \leq C|\log(\kappa)|\kappa^{1/2}$ and $|e^{\lambda_{K+1}(t-2)}| \leq C\kappa$. In turn this implies that $\xi \leq C(|\log(\kappa)|\kappa^{1/2} + M\kappa)$ Putting this all together, then gives that our kernel is a $(\delta, \zeta, \varepsilon)$ class separator with $\delta \leq C\kappa^{1/2}$, $\zeta \geq w_{max}^{-1} - C(|\log(\kappa)|\kappa^{1/2} + M\kappa)$ and $\varepsilon \leq C(\kappa^{1/4} + M\kappa)$. ■

5.5 Asymptotic Behavior of DiAL Queries

In this section, we identify a formal continuum limit associated with Dirichlet Active Learning (DiAL), and identify its steady states. For simplicity, we focus our attention on the case where $\mathcal{X}$ is a bounded open set in $\mathbb{R}^d$, and that the underlying features are associated with a smooth and bounded density $\rho(x) = \sum_{k=1}^K w_k \rho_k(x)$ supported on $\mathcal{X}$ with smooth and bounded class-conditional densities $\rho_k(x)$ for $k \in [K]$. Recalling (9), we furthermore define $V(\alpha(x)) := \mathcal{A}_{var}(x)$ and assume that our acquisition function samples proportional

to $e^{\lambda V(\alpha(x))}$ (i.e., the *Proportional Sampling* policy introduced in 3). Furthermore, we will assume that the value of λ is allowed to vary during the sampling process.

In this context, we can view the expected change in the pseudo-labels from the ℓ-th to $(\ell+1)$-th observation by the relation

$$\mathbb{E}_{X_{\ell+1},Y_{\ell+1}}[\alpha_k^{\ell+1}(x)-\alpha_k^{\ell}(x)|(X_1,Y_1),\ldots,(X_\ell,Y_\ell)] = \int_{\mathcal{X}} \frac{w_k\rho_k(z)}{\rho(z)}\mathcal{K}(z,x)\frac{e^{\lambda(\ell)V(\alpha^\ell(z))}}{\int_{\mathcal{X}} e^{\lambda(\ell)V(\alpha^\ell(\tilde{z}))}\,d\tilde{z}}\,dz,$$

where $\alpha_k^\ell(x) \geq 0$ is the k-th entry of the concentration parameter vector $\alpha^\ell(x) \in \mathbb{R}_+^K$ at the ℓ-th observation.

By assuming that we take a large number of discrete steps in one unit of "continuous time", the law of large numbers then formally leads to the evolution equation

$$\begin{aligned}\partial_t\alpha(x,t) &= \int_{\mathcal{X}} \frac{w_k\rho_k(z)}{\rho(z)}\mathcal{K}(z,x)\frac{e^{\lambda(t)V(\alpha(z,t))}}{\int_{\mathcal{X}} e^{\lambda(t)V(\alpha(\tilde{z},t))}\,d\tilde{z}}\,dz \\ &=: \int_{\mathcal{X}} \eta^K(z)\mathcal{K}(z,x)q(z,t)\,dz, \end{aligned} \tag{24}$$

where we have introduced the notation $\eta^K(x) \in \mathbb{R}_+^K$ as the vector of weighted class-conditional densities concatenated together and $q(x,t)$ as the sampling distribution according to Dirichlet variance. This evolution equation provides a convenient means of understanding the effect of newly observed labels on our future acquisitions in the limit of a large number of observations. Furthermore, it is natural to consider kernels that are increasingly localized when analyzing the large-sample limits of kernel methods (Devroye et al., 1996). With that in mind, it is instructive to consider a limiting case where $\mathcal{K}(z,x)$ is given by the Dirac mass $\delta_z(x)$. In that case, we can write (24) as simply

$$\partial_t\alpha(x,t) = q(x,t)\eta^K(x), \tag{25}$$

from which we see that the trajectories of the $\alpha(x,t) \in \mathbb{R}_+^K$ lie along the corresponding rays $R_x = \{a\eta^K(x) : a \geq 0\}$, whose speed is determined by the sampling $q(x,t)$.

We can then write

$$\alpha(x,t) = \left(\int_0^\top q(x,s)\,ds\right)\eta^K(x), \tag{26}$$

from which we can define the amount of sampling that has occurred at x up to time $t \geq 0$ as

$$\beta_x(t) := \sum_{k=1}^K \alpha_k(x,t) = \int_0^\top q(x,s)\,ds,$$

since the entries of $\eta^K(x)$ sum to 1. By appealing to (26), we can now write the variance at x as

$$\begin{aligned} V(\alpha(x,t)) &= \frac{\sum_{i\neq j}\alpha_i(x,t)\alpha_j(x,t)}{\beta_x^2(t)(\beta_x(t)+1)} = \frac{\sum_{i\neq j}\eta_i(x)\eta_j(x)}{\beta_x(t)+1} = \frac{\sum_{k=1}^K \eta_k(x)(1-\eta_k(x))}{\beta_x(t)+1} \\ &=: \frac{G(x;\eta)}{\beta_x(t)+1}, \end{aligned}$$

where this function

$$G(x;\eta) = \sum_{k=1}^{K} \eta_k(x)(1-\eta_k(x)) \in \left[0, 1 - \frac{1}{K}\right] \tag{27}$$

is a measure of the population level uncertainty as in Section 6.

We now assume that the distribution of points sampled by the acquisition function, namely $q(x,t) = \frac{e^{\lambda(t)V(\alpha(x,t))}}{\int_{\mathcal{X}} e^{\lambda(t)V(\alpha(z,t))}\,dz}$ is asymptotically convergent as $t \to \infty$. The question of whether this actually always occurs for solutions of the evolution equation (25) is not simple, but as q will be weakly compact in t it is natural to guess that it will approach some steady state $\bar{q}(x)$. If one were to use this time-independent distribution throughout the entire stochastic process, we would obtain the simple sampling relationship

$$\bar{\beta}_x(t) = \int_0^{\top} \bar{q}(x) = t\bar{q}(x),$$

with corresponding pseudo-label densities $\bar{\alpha}(x,t) = t\bar{q}(x)\eta^K(x)$.

The assumption that $q(x,t) \to \bar{q}(x)$ as $t \to \infty$ suggests then that $\partial_t \alpha(x,t) \approx \partial_t \bar{\alpha}(x,t)$ for large enough t. From this, we can approximate $\beta_x(t) \approx \bar{\beta}_x(t) = t\bar{q}(x)$ for t large. Thus, we expect that

$$q(x,t) \propto \exp\left(\frac{\lambda(t)G(x;\eta)}{t\bar{q}(x)+1}\right) \approx \exp\left(\frac{\lambda(t)G(x;\eta)}{t\bar{q}(x)}\right) \tag{28}$$

as $t \to \infty$. This gives a non-linear algebraic relation that allows us to narrow our search for possible limiting sampling distributions. At this point, we consider two cases for the scaling of $\lambda(t)$: when (i) $\lambda(t) = \lambda_0 t^p = o(t)$ for $p < 1$ and (ii) $\lambda(t) = \lambda_0 t$ for $\lambda_0 > 0$.

Case 1: When $\lambda(t) = \lambda_0 t^p$ for $p < 1$, then the the exponent in the last expression of (28) decreases like $t^{-(1-p)}$ for every $x \in \mathcal{X}$ and the result is that the limiting distribution is uniform over the domain, $q(x,t) \to \bar{q}(x) = \left(\int_{\mathcal{X}} dz\right)^{-1}$. Roughly, we have that the equivalent expression yields

$$q(x,t) = \left(\int_{\mathcal{X}} \exp\left\{\frac{\lambda_0}{t^{1-p}}\left(\frac{G(z;\eta)}{\bar{q}(z)} - \frac{G(x;\eta)}{\bar{q}(x)}\right)\right\} dz\right)^{-1} \to \left(\int_{\mathcal{X}} dz\right)^{-1}$$

as $t \to \infty$ implying that the limiting distribution $\bar{q}(x)$ is uniform over $\mathcal{X}$ when $\lambda(t) = o(t)$. We can interpret this as asymptotic *exploration* that samples throughout the domain irrespective of the corresponding marginal density $\rho(x)$ in contrast to passive sampling via said marginal distribution.

Case 2: Now consider the case when $\lambda(t) = \lambda_0 t$. The expression (28) the becomes autonomous in the large t limit, and suggests that the limiting distribution in this case satisfies

$$\bar{q}(x) \propto \exp\left(\frac{\lambda_0 G(x;\eta)}{\bar{q}(x)}\right). \tag{29}$$

In the next section, we provide numerical simulations that suggest that for moderate values of λ_0 this $\bar{q}(x)$ emphasizes regions of greater population-level uncertainty; namely, where multiple conditional probabilities ($\eta_i(x) = \mathbb{P}(Y = i | X = x)$) are simultaneously large. In

other words, this limiting distribution asymptotically *exploits* in regions around the true decision boundaries. For $\lambda_0 \ll 1$, however, we expect (29) to imply that $\bar{q}(x)$ behaves like the uniform distribution on $\mathcal{X}$.

We notice that the two cases here both give asymptotic sampling densities which are independent of ρ. In large sample regimes, we posit that independence on ρ ought to be viewed positively, meaning that we will explore (Case 1), or exploit (Case 2), in a fashion that is unbiased by the density ρ.

Of course, several steps in this derivation are formal. The passage to "continuous time" is likely justifiable using stochastic approximation techniques under appropriate scalings, but the details would be significant. The use of non-Dirac kernels certainly will increase the number of possible steady-state sampling densities, but given a density with compact support it also seems plausible that one could show that sampling density steady states need to be "close" to the one given by the Dirac mass kernel. Finally, the existence of steady-state densities is not completely obvious, especially in light of the α large approximations that we made. On the other hand, we do anticipate that it should be possible to demonstrate that sampling densities associated with the evolution equation (25) do converge to the type of steady states we have formally described here.

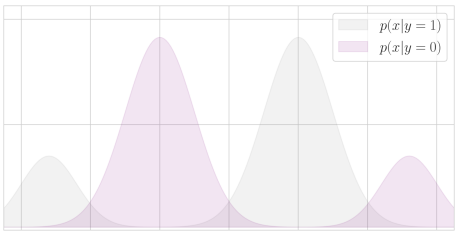
Although the work here is informal, we think that it gives valuable insight into the behavior observed in Figure 14, where DiAL effectively transitions from exploration to exploitation (see Section A.3 for more discussion about this tradeoff in the literature). We leave more rigorous proofs of the types of formulas given in this section for future work. We now present some numerics to illustrate the behavior of (28) in the two cases of $\lambda(t)$ that we've considered here.

5.5.1 Numerical Demonstration

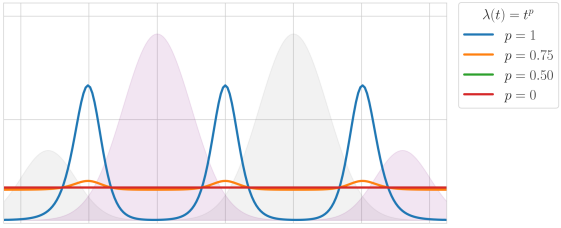
We return to the mixture of Gaussians setup of Section 6 and numerically solve the evolution equation (25) for $\alpha(x,t)$ up to $t = 10^{10}$ in order to simulate convergence of $q(x,t) \to \bar{q}(x)$ for various scalings of $\lambda(t)$. In panel (b) of Figure 12, we plot the corresponding distributions $\bar{q}(x)$ for algebraic scalings of $\lambda(t) = t^p$ for some values of $p \in [0,1]$; note that the case of $p = 0$ corresponds to $\lambda(t) = \lambda_0 = 1$. In panel (c), we plot $\bar{q}(x)$ for $\lambda(t) = \lambda_0 t$ for a range of values of λ_0.

As suggested by our analysis above, we see in panel (b) that for $\lambda(t)$ constant (and for sufficiently small powers $p < 1$), the limiting distribution $\bar{q}(x)$ corresponds to uniform sampling over the whole domain. We can interpret this as asymptotic *exploration* that samples throughout the domain irrespective of the corresponding marginal density $\rho(x)$ in contrast to passive sampling via said marginal distribution. As $p \to 1$, however, we observe that $\bar{q}(x)$ focuses on the regions of the domain that lie near the true decision boundary between the classes; we can interpret this as asymptotic *exploitation*. Similarly, in panel (c) of Figure 12, we observe the influence of λ_0 on $\bar{q}(x)$, including the expected behavior as $\lambda_0 \ll 1$ corresponding to the uniform distribution.

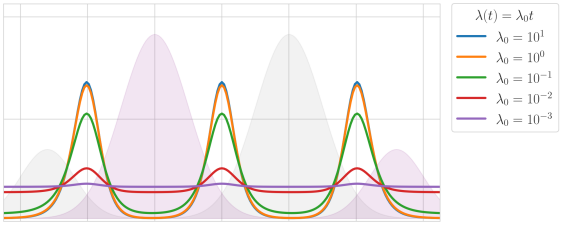
These numerical experiments simply demonstrate the intimately coupled nature of the scaling of $\lambda(t)$ and the limiting distribution $q(x,t) \to \bar{q}(x)$ of this stochastic process that represents Dirichlet Active Learning (DiAL). In addition to further work on analyzing the



(a) Setup of 2 classes



(b) $\bar{q}(x)$ when $\lambda(t) = t^p$



(c) $\bar{q}(x)$ when $\lambda(t) = \lambda_0 t$

Figure 12: Numerical demonstration of the effect of $\lambda(t)$ on the steady state distribution $\bar{q}(x)$ in a 1D setting. With class-conditional distributions $p(x|y=0), p(x|y=1)$ show respectively in gray and purple panel (a), we numerically solve (25) up to $t = 10^{10}$ for different scalings of $\lambda(t)$ to simulate convergence of $q(x,t) \to \bar{q}(x)$. Panels (b) and (c) suggest that the limiting behavior when $\lambda(t) = \lambda_0 t$ for $\lambda_0 \gg 0$ corresponds to *asymptotic exploitation*, whereas when $\lambda(t) = \lambda_0 t^p$ for $p < 1$ or $\lambda_0 \ll 1$ corresponds to *asymptotic exploration*.

steady states of the stochastic process associated with DiAL, we suggest that an in-depth numerical exploration into this process is warranted pursuant our findings here.

Acknowledgments

KM acknowledges partial support from the Peter J. O'Donnell Jr. Postdoctoral Fellowship and NSF IFML grant 2019844. RM acknowledges partial support from NSF-DMS 2307971 and the Simons Foundation MP-TSM.

Appendix A. Related Literature

A.1 Active Learning and Acquisition Functions

Acquisition functions for active learning have been introduced for various machine learning models, including support vector machines (Tong and Koller, 2001; Balcan et al., 2007; Jiang and Gupta, 2019; Hanneke and Yang, 2015), deep neural networks (Ren et al., 2021; Gal et al., 2017; Kushnir and Venturi, 2020; Cai et al., 2017; Sener and Savarese, 2018; Ash et al., 2020), and graph-based classifiers (Miller et al., 2020; Miller and Bertozzi, 2024; Miller et al., 2022; Qiao et al., 2019; Ma et al., 2013; Ji and Han, 2012; Zhu et al., 2003b). Our

experiments primarily focus on graph-based classifiers as the underlying semi-supervised classifier due to their straightforward ability to capture clustering structure in data and their superior performance in the *low-label rate regime*—wherein the labeled data constitutes a very small fraction of the total amount of data (Calder et al., 2020; Miller and Calder, 2023). While there has been progress in adapting deep neural networks to better handle small amounts of labeled data for semi-supervised classification tasks (Berthelot et al., 2019; Sohn et al., 2020; Yang et al., 2023), most active learning methods for deep learning assume a moderate-to-large amount of initially labeled data when evaluating their methods in the active learning process.

A.2 Conformal Prediction and Active Learning

Related to the design of uncertainty criteria for query point selection in active learning, conformal prediction has recently emerged as a prominent model-agnostic technique that can convert heuristic uncertainty scores into statistically rigorous prediction sets under exchangeability assumptions (Vovk et al., 2022; Shafer and Vovk, 2008; Angelopoulos and Bates, 2023). Recent work in various application areas has explored the use of conformal prediction in the active learning process to identify conformally calibrated acquisition function values (Gwon et al., 2025; Nguyen et al., 2026). Whereas conformal prediction traditionally uses a held-out "calibration" set of labeled datapoints for constructing prediction sets, the straightforward application to active learning is delicate: the repeated use of a single calibration set to identify query points makes the subsequent model updates adaptively dependent on the calibration data, which can undermine the standard exchangeability assumptions that underlie the statistical guarantees in conformal prediction. While outside the scope of the current work, a promising line of future work would be to explore a mathematically justified application of conformal prediction techniques to calibrate the uncertainty scores used within the DiAL framework.

A.3 Exploration versus Exploitation

As mentioned in the main body of the paper, an important aspect of active learning is the inherent tradeoff between using the (in-practice) limited resource of queries to either (i) explore the given data set or (ii) exploit the current classifier's inferred decision boundaries. This tradeoff is reminiscent of the similarly named "exploration versus exploitation" tradeoff in reinforcement learning (Sutton and Barto, 2018; Agarwal et al., 2021). Similar to reinforcement learning, there is a clear motivation for ensuring that exploration is performed before exploitation in the active learning process. Broadly speaking, however, most active learning acquisition functions are designed to exhibit one of these two behaviors, though some methods do seem to empirically balance both characteristics (Krause and Guestrin, 2007; Huang et al., 2010; Karzand and Nowak, 2020; Miller and Bertozzi, 2024). An important contribution of the current work is to present a mathematical analysis that gives qualitative guarantees about the explorative and exploitative characteristics of query points that are chosen by the proposed acquisition function, Dirichlet Variance.

Prior work to establish theoretical guarantees for active learning has primarily focused on proving sample-efficiency results for linearly-separable data sets—frequently restricted to the unit sphere (Balcan et al., 2009; Dasgupta, 2006; Hanneke, 2007)—for low-complexity

function classes using disagreement or margin-based acquisition functions (Hanneke, 2014; Hanneke and Yang, 2015; Balcan et al., 2009, 2007). These provide convenient bounds on the number of query points necessary for the associated classifier to achieve (near) perfect classification on these data sets with simple geometry. These results demonstrate that a proposed acquisition function is sufficient to select query points that will (near) optimally refine the associated classifier's decision boundaries to best match the assumed ground-truth decision boundaries; that is, these classical statistical guarantees for active learning focus on the exploitative behavior of the associated methods.

In contrast, theoretical guarantees for graph-based active learning methods primarily demonstrate that a proposed acquisition function sufficiently explores an assumed clustering structure for the data set (Murphy and Maggioni, 2019; Dasarathy et al., 2015; Miller and Calder, 2023; Dasgupta and Hsu, 2008; Dasgupta, 2011; Cloninger and Mhaskar, 2021). Occasionally, this clustering structure is assumed to be hierarchical (Dasgupta and Hsu, 2008; Dasgupta, 2011; Cloninger and Mhaskar, 2021). Sufficient exploration of clustering structure is characterized by a guarantee that—given assumptions about the clustering structure of the observed data set $\mathcal{X}$—the active learning method in question will query points from *all* clusters. The low-label rate regime of active learning—a significant focus of this current work—is the natural setting for establishing such explorative guarantees.

In summary, while classical statistical analysis of disagreement-based and margin-based active learning methods has focused on *exploitative guarantees*, the analysis of graph-based active learning has focused on *explorative guarantees*. In the current work, we not only provide explorative guarantees for our proposed acquisition function but also give a novel asymptotic analysis of the exploitative capabilities of the acquisition function later on in the active learning process.

A.4 Computational Complexity of Active Learning

Computational complexity is an additional consideration that is vital to the practical application of active learning methods. In the assumed setting of sequential active learning, the computational burden at each iteration is encountered in two main ways: (i) the cost to evaluate the acquisition function on a single, currently unlabeled data point and (ii) the size of the set of unlabeled data chosen to evaluate said acquisition function. Additionally, some active learning methods require the computation of "auxiliary" variables, such as covariance matrices (Ji and Han, 2012; Ma et al., 2013; Miller et al., 2020; Miller and Bertozzi, 2024), eigenvector matrices (Murphy and Maggioni, 2019; Miller and Bertozzi, 2024), or graph paths and distances (Murphy and Maggioni, 2019; Cloninger and Mhaskar, 2021; Dasarathy et al., 2015) to be stored and oftentimes updated throughout the active learning process. Finally, some graph-based active learning methods (Murphy and Maggioni, 2019; Cloninger and Mhaskar, 2021) do not exactly follow the assumed interactive labeling scheme displayed in Figure 1; instead, these methods more closely resemble coreset methods (Bachem et al., 2017) since the labels of the selected data points do not influence the choice of any subsequently selected points. Such methods incur other computational costs that do not fit into the paradigm we now discuss.

Reducing the number of unlabeled points on which to evaluate the acquisition function is a way to reduce computational cost, and various heuristics have been suggested previ-

ously such as uniform random subsampling (Gal et al., 2017; Miller and Bertozzi, 2024) or graph-based local restrictions (Chapman et al., 2023). While this is an interesting direction for research, we assume that the acquisition function is evaluated on the entire unlabeled data pool, as is standard in pool-based active learning (Settles, 2012). As such, the primary source of computational complexity follows from the cost of evaluating the acquisition function on a single unlabeled data point.

Uncertainty sampling (Settles, 2012; Miller and Calder, 2023) is a category of acquisition functions that use the current classifier's outputs to approximate the "uncertainty" of the inferred classification of unlabeled data;[4] the most "uncertain" points are then selected to be queried. Different measures of uncertainty in the classifier outputs (e.g., smallest margin, entropy, ℓ^2-norm) determine the different acquisition functions in uncertainty sampling. While uncertainty sampling has not always empirically demonstrated optimal exploration versus exploitation behavior, it is generally among the most computationally efficient kind of acquisition functions—the current classifier's outputs are the only required quantity at each active learning iteration, which is readily available in our assumed setting (Figure 1). The cost per unlabeled data point simply scales as the number of classes in the classification problem at hand. This is in contrast to other acquisition functions, such as Variance Minimization (Ji and Han, 2012) and Σ-Optimality (Ma et al., 2013), that require computations that scale as the size of the entire data set in order to evaluate the acquisition function at a single unlabeled data point.

Our proposed acquisition function, Dirichlet Variance (see (9)), can indeed be classified as a novel type of uncertainty sampling that naturally fits within the proposed semi-supervised learning model (Dirichlet Learning) which we introduce in Section 2. See Section 4.3.2 for further discussion about and comparison of computational complexity among the compared methods.

A.5 Graph-based Learning

Graph-based methods have been shown to be useful models for semi-supervised and active learning, especially in the low-label rate regime (i.e., when the amount of labeled data is *significantly* smaller than the amount of unlabeled data). Generally speaking, these methods construct a similarity graph $G(\mathcal{X}, W)$ from a finite set of inputs $\mathcal{X} = \{x^1, \dots, x^n\}$, where the edge weight matrix $W \in \mathbb{R}^{n \times n}$ records the pairwise similarities $W_{ij} \geq 0$ between inputs $x^i, x^j \in \mathcal{X}$. For example, edge weights computed by the Gaussian (RBF) kernel are $W_{ij} = \exp(-\|x^i - x^j\|_2^2)$. The task of semi-supervised classification then amounts to identifying how to use both the labeled data (i.e., label y^j at labeled node $x^j \in \mathcal{L}$) and the similarity graph $G(\mathcal{X}, W)$ to infer labels on the set of unlabeled nodes $\mathcal{U} = \mathcal{X} \setminus \mathcal{L}$. Especially relevant to our work is the subset of graph-based methods that are inspired by numerical methods for partial differential equations (PDE), wherein the semi-supervised learning task reduces to identifying a graph function $u : \mathcal{X} \to \mathbb{R}^K$ where the output $u(x) \in \mathbb{R}^K$ reflects the inferred classification of node $x \in \mathcal{X}$ (Bertozzi and Flenner, 2016; Calder et al., 2020; Calder and Slepčev, 2020; Zhu et al., 2003a; Zhou et al., 2004). A central component in nearly all graph-based methods is the graph Laplacian matrix, $L \in \mathbb{R}^{n \times n}$; this matrix is a graph-based analog of the Laplace operator (Chung, 1997; von Luxburg, 2007). Common examples of

4. See Example 6 for a discussion about uncertainty in semi-supervised learning and active learning.

the graph Laplacian matrix are the *combinatorial* $L = D - W$, the *random walk* $L_{rw} = I - D^{-1}W$, and the *symmetric normalized* $L_n = I - D^{-1/2}WD^{-1/2}$ graph Laplacians, where $D = \text{diag}(d_1, d_2, \ldots, d_n)$ is the diagonal degree matrix with $d_i = \sum_{j=1}^{n} W_{ij}$. A common rationale that motivates the use of graph-based methods for semi-supervised learning is that the graph Laplacian matrix is effective for identifying clustering structure in the underlying data set. This is a primary observation and motivation for the use of spectral clustering in unsupervised clustering (see von Luxburg, 2007, and the references therein).

We also mention that these graph-based methods for semi-supervised learning (classification and regression) have intimate connections to second-order elliptic PDEs. Namely, significant work has been done to connect the graph Laplacian matrix, its eigenvalues and eigenvectors, and the corresponding semi-supervised classifiers to "continuum limit" counterparts which can be thought of the limit of the discrete graphs as the number of nodes $n \to \infty$, under proper assumptions regarding the graph scaling (see, for example, Calder and Slepčev, 2020; Calder et al., 2023). This analysis is enlightening since the continuum-limit PDE formulation acts as a proxy for the large data limit scenarios often faced in applications, wherein the amount of unlabeled data is especially large. This allows one to analyze the properties of the second-order elliptic equations as opposed to the discrete graph structure thereby informing the behavior of the methods in the discrete setting. For example, continuum limit analysis has been used to propose a "properly-weighted" graph-based semi-supervised classifier that resolves the degenerate behavior of the Laplace learning classifier (Zhu et al., 2003a) in the presence of extremely low label rates (Calder and Slepčev, 2020). Portions of our theoretical analysis in Section 5 for Dirichlet Learning in the graph-based setting rely on a continuum limit formulation and we introduce further notation and setup at that point in the paper.

Similar to graph-based methods for semi-supervised learning, graph-based active learning can then be framed in terms of selecting unlabeled nodes $x \in \mathcal{U}$ based on properties such as the graph-based classifier u given the currently labeled data, $\mathcal{L}$. As mentioned previously, however, various acquisition functions have been proposed from a statistical perspective of the graph-based semi-supervised learning problem (Zhu et al., 2003b; Jun and Nowak, 2018; Ji and Han, 2012; Ma et al., 2013; Qiao et al., 2019; Miller et al., 2020). Namely, some graph-based semi-supervised classifiers can be viewed as the *maximum a posteriori* (MAP) estimator of a Gaussian random field whose correlation structure is related to the graph Laplacian matrix of the associated graph $G(\mathcal{X}, W)$.

For example, the Laplace learning semi-supervised classifier $u : \mathcal{X} \to \mathbb{R}$ for binary classification introduced by Zhu, Ghahramani, and Lafferty (2003a) can be viewed as the MAP estimator of the Gaussian random field with density

$$p(u) \propto \exp\left(-\frac{1}{\gamma}\mathcal{E}(u)\right) = \exp\left(-\frac{1}{\gamma}\sum_{i,j=1}^{N} w_{ij}(u(x^i) - u(x^j))^2\right),$$

where $\mathcal{E}(u) = \sum_{i,j=1}^{N} w_{ij}(u(x^i) - u(x^j))^2$ is called the graph Dirichlet energy and $\gamma > 0$ is interpreted as a "temperature" parameter (see Zhu et al., 2003b). Given observations (labels) $y^j \in \{\pm 1\}$ at the labeled nodes $x^j \in \mathcal{L}$, the corresponding posterior distribution reflects fixing the outputs $u(x^j) = y^j$ while interpolating the values at the unlabeled nodes according to the graph topology. Other works have considered variants with Gaussian ob-

servation models (Bertozzi et al., 2021; Zhou et al., 2004), binary Markov random fields (Jun and Nowak, 2018), and other non-Gaussian observation models (Qiao et al., 2019; Bertozzi et al., 2021). Each admits computation (or numerical approximation) of the uncertainty in the classifier under the respective Bayesian models. The statistics of the underlying posterior distribution, given the observed labeled data, can be computed for use as acquisition functions to identify which currently unlabeled points would reduce measures of the underlying posterior covariance matrix (Ji and Han, 2012; Ma et al., 2013) or the overall expected error (Zhu et al., 2003b; Jun and Nowak, 2018).

While this convenient Bayesian interpretation underlies each of these aforementioned graph-based semi-supervised learning models, there is an implicit assumption that the prior distribution over node functions u follows a *Gaussian* law—thus, the most "natural" Bayesian modeling choice is that of continuous-valued outputs like that of regression problems, *not* classification problems. We directly address this shortcoming with our novel Dirichlet Learning classifier (see Section 2) for semi-supervised learning that explicitly models the classification task in a Bayesian-inspired manner. Our proposed Dirichlet Variance acquisition function is then a natural choice for both its computational efficiency and theoretical interpretation for measuring the uncertainty in the underlying Dirichlet Learning classifier.[5]

Appendix B. Bayesian Prior Interpretation

Although the Dirichlet Learning model is not directly derived from a properly Bayesian setup, we can interpret the given model in a Bayesian framework. For simplicity, assume that the set $\mathcal{X}$ is finite, and consider the random matrix $\mathrm{P} \in \mathbb{R}_+^{n\times K}$ whose i^{th} row corresponds to the Dirichlet random variable $\mathrm{P}(x^i)$ defined at x^i; that is, P is the concatenation of all the Dirichlet random variables over our set $\mathcal{X}$. Let the k^{th} entry of the probability vector $\mathrm{P}(x^i) \in \Delta_K$ (i.e., the $(i,k)^{th}$ entry of the matrix $\mathrm{P} \in \mathbb{R}^{n\times K}$) be denoted as $\mathrm{P}_k(x^i) \in [0,1]$. Let $(X,Y) \in \mathcal{X} \times [K]$ represent an observation of an input-output pair with instance $(X,Y) = (x,y)$ for some $x \in \mathcal{X}$.

We seek a formula for the prior belief (with density $\pi(\mathrm{P}|\alpha_0)$) on P given our modeling assumption captured in the pseudo-label propagation from labeled points in Dirichlet Learning defined via $\mathcal{K} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$. Recalling the definition of the probability density function $\varphi(\alpha, z)$ for a Dirichlet random variable in (1), then by appealing to Bayes' law we can then write

$$\begin{aligned}
\pi(\mathrm{P}|\alpha_0) &= \sum_{x\in\mathcal{X}} \sum_{y=1}^{K} \mathbb{P}(\mathrm{P}|x,y,\alpha_0)\pi(x,y) \\
&= \left(\prod_{z\in\mathcal{X}} \varphi(\alpha_0 \mathbb{1}, \mathrm{P}(z))\right) \sum_{x\in\mathcal{X}} \sum_{y=1}^{K} \prod_{z\in\mathcal{X}} \frac{\Gamma(\alpha_0)^{K-1}\Gamma(\alpha_0 + \mathcal{K}(x,z))\mathrm{P}_y(z)^{\mathcal{K}(x,z)}}{\Gamma(K\alpha_0 + \mathcal{K}(x,z))} \mathbb{P}(Y = y|x)\pi(x) \\
&\propto \left(\prod_{z\in\mathcal{X}} \varphi(\alpha_0 \mathbb{1}, \mathrm{P}(z))\right) \sum_{x\in\mathcal{X}} \left(\prod_{z\in\mathcal{X}} \frac{\Gamma(\alpha_0 + \mathcal{K}(x,z))}{\Gamma(K\alpha_0 + \mathcal{K}(x,z))}\right) \pi(x) \left\{\sum_{y=1}^{K} \prod_{z\in\mathcal{X}} \mathrm{P}_y(z)^{\mathcal{K}(x,z)} \mathrm{P}_y(x)\right\}.
\end{aligned}$$

5. Classifier "uncertainty" is admittedly an ambiguous term, and we refer the reader to Section 6 for a discussion of different types of uncertainty in active learning.

Now, if we set $\pi(x) = \frac{1}{|\mathcal{X}|}$ to be an *uninformative* prior on the observation data's input, then we can further simplify as

$$\begin{aligned}\pi(\mathrm{P}|\alpha_0) &\propto \left(\prod_{z\in\mathcal{X}} \varphi(\alpha_0 \mathbb{1}, \mathrm{P}(z))\right) \sum_{x\in\mathcal{X}} \overbrace{\left(\prod_{z\in\mathcal{X}} \frac{\Gamma(\alpha_0 + \mathcal{K}(x,z))}{\Gamma(K\alpha_0 + \mathcal{K}(x,z))}\right)}^{w(x):=} \overbrace{\left\{\sum_{y=1}^{K} \mathrm{P}_y(x) \prod_{z\in\mathcal{X}} \mathrm{P}_y(z)^{\mathcal{K}(x,z)}\right\}}^{s(\mathrm{P},\mathcal{K}_x):=} \\ &= \left(\prod_{z\in\mathcal{X}} \varphi(\alpha_0 \mathbb{1}, \mathrm{P}(z))\right) \sum_{x\in\mathcal{X}} w(x) s(\mathrm{P}, \mathcal{K}_x),\end{aligned}$$

where have recalled the definition of $\mathcal{K}_x : \mathcal{X} \to \mathbb{R}^+$ as the propagation function from $x \in \mathcal{X}$. The quantity $w(x)$ then corresponds to a measure of "centrality" that weights each $x \in \mathcal{X}$, while $s(\mathrm{P}, \mathcal{K}_x)$ captures a measure of "alignment" between the probability distribution represented by P and the chosen kernel's propagation at $x \in \mathcal{X}$, $\mathcal{K}_x(\cdot)$. The interpretation of $w(x)$ as a measure of centrality simply follows from the observation that the ratio of the Γ functions increases as $\mathcal{K}(x,z)$ increases; thus, a more "central" point x that is similar to a greater proportion of the data set will have a larger weight $w(x)$. See Example 5 for a simplified example to demonstrate how the quantity $s(\mathrm{P}, \mathcal{K}_x)$ captures the "alignment" between p and the kernel propagation, $\mathcal{K}_x$.

Example 5 (Kernel and probability alignment) *Consider a binary classification setting where the data set is clustered simply into two disjoint sets $\mathcal{X} = \mathcal{X}_1 \cup \mathcal{X}_2$ and that the kernel $\mathcal{K}$ perfectly discriminates between these clusters:*

$$\mathcal{K}(x,z) = \begin{cases} 1 & \text{if there exists } k \in \{1,2\} \text{ s.t. } x, z \in \mathcal{X}_k \\ 0 & \text{otherwise} \end{cases}.$$

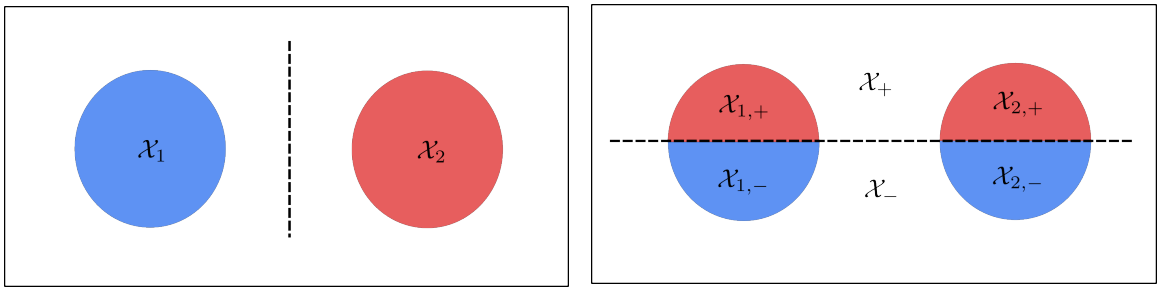
If the probability matrix P *reflects this clusteredness (e.g.,* $\mathrm{P}_1(x) = 1$ *if* $x \in \mathcal{X}_1$ *and* 0 *otherwise), then*

$$\begin{aligned}\prod_{z\in\mathcal{X}} \mathrm{P}_y(z)^{\mathcal{K}(x,z)} &= \prod_{z\in\mathcal{X}_y} \mathrm{P}_y(z)^{\mathcal{K}(x,z)} \prod_{z\notin\mathcal{X}_y} \mathrm{P}_y(z)^{\mathcal{K}(x,z)} = \prod_{z\in\mathcal{X}_y} 1^{\mathcal{K}(x,z)} \prod_{z\notin\mathcal{X}_y} 0^{\mathcal{K}(x,z)} \\ &= \begin{cases} \prod_{z\in\mathcal{X}_y} 1^1 \prod_{z\notin\mathcal{X}_y} 0^0 & \text{if } x \in \mathcal{X}_y \\ \prod_{z\in\mathcal{X}_y} 1^0 \prod_{z\notin\mathcal{X}_y} 0^1 & \text{if } x \notin \mathcal{X}_y \end{cases} = \begin{cases} 1 & \text{if } x \in \mathcal{X}_y \\ 0 & \text{if } x \notin \mathcal{X}_y \end{cases} \\ \implies s(\mathrm{P}, \mathcal{K}_x) &= \sum_{y=0,1} \mathrm{P}_y(x) \cdot \chi\{x \in \mathcal{X}_y\} = 1,\end{aligned}$$

where we have defined $0^0 = 1$.

In contrast, consider a $\tilde{\mathrm{P}}$ *that is very misaligned with the clustering structure, such as one that splits the clusters in half as shown in Figure 13(b):*

$$\tilde{\mathrm{P}}_1(z) = \begin{cases} 1 & \text{if } z \in \mathcal{X}_+ \\ 0 & \text{if } z \in \mathcal{X}_- \end{cases}, \qquad \tilde{p}_2(z) = \begin{cases} 0 & \text{if } z \in \mathcal{X}_+ \\ 1 & \text{if } z \in \mathcal{X}_- \end{cases}.$$



(a) Perfectly aligned P (i.e., $s = 1$)

(b) Misaligned $\tilde{\mathrm{P}}$ (i.e., $s = 0$)

Figure 13: Example of clusters and probability matrix P setup for Example 5. The data set can be written as the union of 2 disjoint clusters, $\mathcal{X} = \mathcal{X}_1 \cup \mathcal{X}_2$, where the circles respectively represent $\mathcal{X}_1$ and $\mathcal{X}_2$. The coloring of regions represents the classification of the points in each cluster and the dotted line denotes the decision boundary between classes according to the different probability matrices P. In the case of panel (a), the classification proposed by P perfectly aligns with the clustering structure and leads to a large value of $s = 1$. In contrast, the horizontal decision boundary characterizing $\tilde{\mathrm{P}}$ in panel (b) is exactly *misaligned* with the clustering structure and results in a value of $s = 0$.

Then, defining $\mathcal{X}_{k,\pm} = \mathcal{X}_k \cap \mathcal{X}_\pm$, *we have*

$$\begin{aligned}\prod_{z\in\mathcal{X}} \tilde{\mathrm{P}}_1(z)^{\mathcal{K}(x,z)} &= \prod_{z\in\mathcal{X}_{1,+}} 1^{\mathcal{K}(x,z)} \prod_{z\in\mathcal{X}_{1,-}} 0^{\mathcal{K}(x,z)} \prod_{z\in\mathcal{X}_{2,+}} 1^{\mathcal{K}(x,z)} \prod_{z\in\mathcal{X}_{2,-}} 0^{\mathcal{K}(x,z)} \\ &= \begin{cases} 0 & \text{if } x \in \mathcal{X}_1 \\ 0 & \text{if } x \in \mathcal{X}_2 \end{cases}.\end{aligned}$$

With a similar computation for $\tilde{\mathrm{P}}_2(z)$, *we can see then that* $s(\tilde{\mathrm{P}}, \mathcal{K}_x) = 0$. *This simple example highlights how this quantity* $s(\mathrm{P}, \mathcal{K}_x)$ *measures the alignment between the class probabilities of an instance* P *and the geometry of the data as reflected by the kernel.*

To summarize, the prior probability for a very misaligned $\tilde{\mathrm{P}}$ *is* $\pi(\tilde{\mathrm{P}}|\alpha_0) = 0$, *while the prior probability for a very well-aligned* P *is* $\pi(\mathrm{P}|\alpha_0) = \left(\prod_{z\in\mathcal{X}} \varphi(\alpha_0 \mathbb{1}, \mathrm{P}(z))\right) \sum_{x\in\mathcal{X}} w(x) > 0$. *This demonstrates the corresponding prior's preference for probability outputs* p *that are well-aligned with the inherent clustering structure of the data set.*

Appendix C. Discussion of Two Types of Uncertainty

We briefly discuss here the various concepts of "uncertainty" in active learning. Namely, we suggest there are (at least) two types of uncertainty to consider and model in the active learning process: (1) data-conditional uncertainty and (2) underlying population-level classification uncertainty. Data-conditional uncertainty reflects the idea that given the current labeled data and assumed hypothesis class, how uncertain is the current classifier about the inferred classifications on the unlabeled data? Both the traditional notion of "uncertainty

sampling" and our proposed Dirichlet random variable's measure of variance reflect two ways of modeling this data-conditional uncertainty. We notice that for Dirichlet Variance, we expect the data-conditional uncertainty to go to zero in the limit of infinitely many labeled data points (see Property 1 at the end of this section). However, the variance goes to zero at different rates in regions with different population-level uncertainty, a phenomenon we explore in the computations below and in Section 5.5.

On the other hand, the underlying population-level uncertainty reflects the inherent uncertainty of the data-generating distribution (e.g., regions where class-conditional distributions are large for multiple classes). We also note that in the large, labeled data limit it is natural to guess that the data-conditional uncertainty associated with "uncertainty sampling" will approach what we call the population-level uncertainty.

Various types of goals for active learning algorithms can be explained in terms of these uncertainties. For example, in settings with very few labeled data points the data-conditional uncertainty is expected to be quite high, and the goal of an active learning algorithm is often to appreciably decrease this uncertainty across a wide range of points. This type of behavior is sometimes called exploratory behavior. On the other hand, as the number of labeled data points increases, a possible goal for active learning is to focus attention on regions with high population-level uncertainty, with the goal being to effectively learn high-quality decision boundaries. Good active learning algorithms likely need to balance these two goals and transition reasonably from one to the other as more labels are obtained: we discuss this more in Section 5.

Example 6 (1D visualization of two types of uncertainty) *Consider the binary classification case (with labels $y \in \{0, 1\}$ as opposed to $y \in \{1, 2\}$) with ground-truth, class-conditional distributions $\rho_0(x) = p(x|y = 0)$ and $\rho_1(x) = p(x|y = 1)$ shown as the green and gray shaded regions in Figure 14(a). The* ***black*** *dashed line represents the population-level uncertainty, wherein these class-conditional probabilities are both large. In particular, this black dashed line is computed as $\min\{p(x, y = 0), p(x, y = 1)\}$, peaking at the locations when the conditionals are both relatively large. The implicit assumption when applying uncertainty sampling acquisition functions for active learning is that they should focus on sampling in these regions of "large" population-level uncertainty. However, as we illustrate in this example, regions of high data-conditional uncertainty according to a given model class of functions do not necessarily reflect population-level uncertainty.*

The example begins with two initially labeled points, chosen from the largest clusters of the respective class-conditional distributions; in panel (b) these are labeled as blue x's while in panel (c) these are labeled as red squares. Panels (b) and (d) show the evolution of the Dirichlet Variance acquisition function as thirty query points are sequentially selected to maximize Dirichlet Variance at each iteration. The selected query point at each iteration is randomly assigned a label of $y = 1$ with probability $p(y = 1|x)$. Similarly, panels (c) and (e) show the evolution of the smallest margin acquisition function (Unc. (SM)) using an SVM classifier using the RBF kernel (specifically for $\mathcal{K}(z, x) = \exp\left(-\gamma|z - x|^2\right)$ where we set the kernel bandwidth $\gamma = 2$). Note that while the blue line of panel (d) has larger values in the three regions between the clusters of opposing labels, the red line of panel (e) has maximum values at only the two rightmost regions between clusters.

The blue and red dotted lines in Figure 15 respectively show kernel density estimators from the selected query points of the two experiments; note that while query points selected by Dirichlet Variance concentrate around the regions where population-level uncertainty is large, the query points from Unc. (SM) sampling have not sampled from leftmost "uncertainty region". This illustrates that the regions where a classifier induces data-conditional uncertainty (e.g., smallest margin) do not necessarily coincide with the true population-level uncertainty regions.

We note that the smallest margin acquisition function has not identified the leftmost decision boundary between the large green cluster and the small gray cluster. This is simply due to the label of the initially chosen point and its influence in this setting with simple geometry. Namely, the initial point labeled by the Unc. (SM) acquisition function in panel (c) lies halfway in between the initially labeled points. The label of this first query point is assigned randomly according to the relative values of the class-conditional densities, which are approximately equal. With probability roughly 1/2, the resulting label will be class $y = 1$ and then the subsequent margin values will focus only on the left half of the domain of the experiment. Our point is not to suggest corrections for this type of smallest-margin uncertainty sampling, but rather illustrate that it is important to design acquisition functions that will properly explore the extent of the clustering structure of the data set so that the resulting regions of data-conditional uncertainty are properly aligned with the regions of population-level uncertainty.

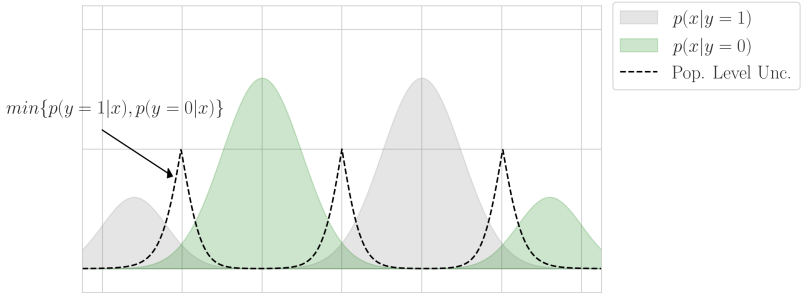
While this example demonstrates the explorative capabilities of using Dirichlet Variance as an acquisition function and the subsequent reflection of population-level uncertainty, we also note the potential gains of considering the proportional sampling active learning policy we have discussed (see Table 1) as opposed to always selecting the maximizer of the chosen acquisition function. For example, while the Unc. (SM) acquisition function is maximized in the right-hand side of the domain throughout the active learning process, proportional sampling would allow queries on the left-hand side and lead to a distribution of query points in all regions of large population-level uncertainty.

Appendix D. Extra Numerical Experiment Details

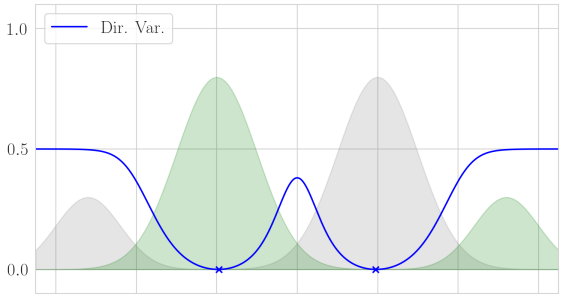
D.1 Choices of Hyperparameters α_0 and λ

We briefly comment on the choice of the uniform prior constant $\alpha_0 \geq 0$ and the inverse temperature parameter $\lambda > 0$ (for the Proportional Sampling active learning policy, see Table 1) in our experimental results. While the derivation of "optimal" values for either of these hyperparameters would be an interesting line of inquiry with practical importance, it lies outside the scope of this current work.

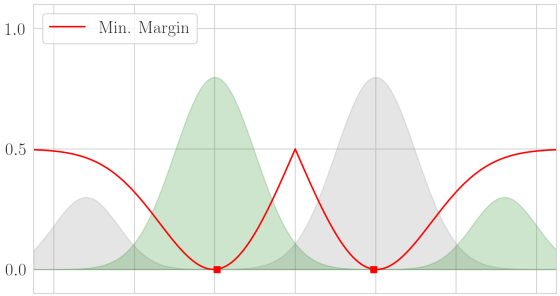
We use a simple heuristic for the selection of both parameters that focuses on locality in the clustering structure. Namely, the choice of α_0 ensures that it is on the order of the expected value of the Poisson graph-based propagation $\mathcal{K}_P(z, x)$ for nodes $z, x \in \mathcal{X}$ that are relatively "local" to one another in the graph. This expectation is roughly approximated via a small random sample of source nodes and the notion of locality comes from the $100(\hat{K}-1)/\hat{K}$th percentile of propagation values, where $\hat{K}$ is an overestimate of the number of clusters K contained in the data set.



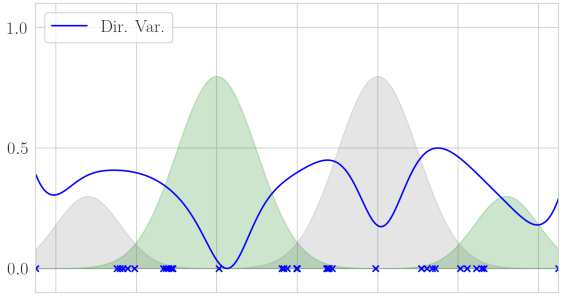
(a) Setup of 2 classes



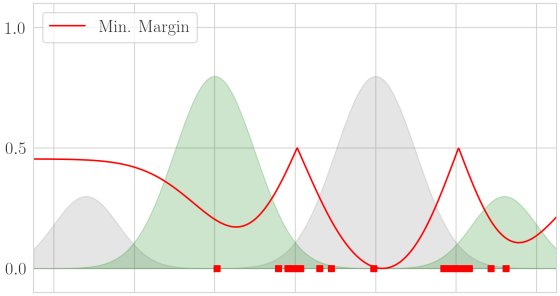
(b) Dir. Var. at Iter 1



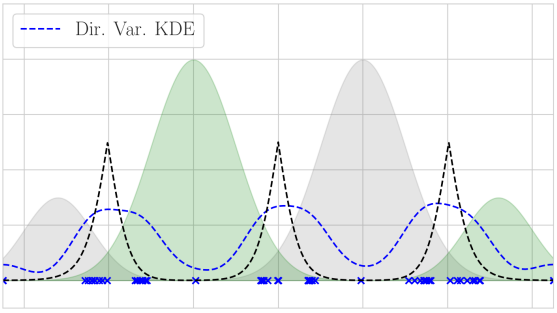
(c) Unc. (SM) at Iter 1



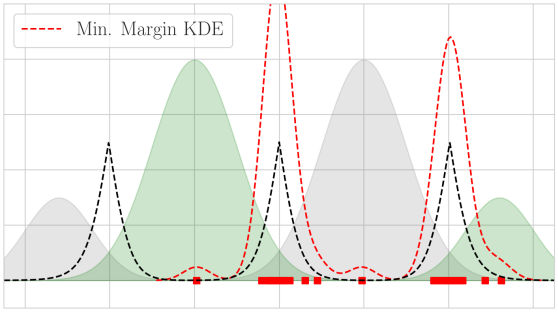
(d) Dir. Var. at Iter 30



(e) Unc. (SM) at Iter 30

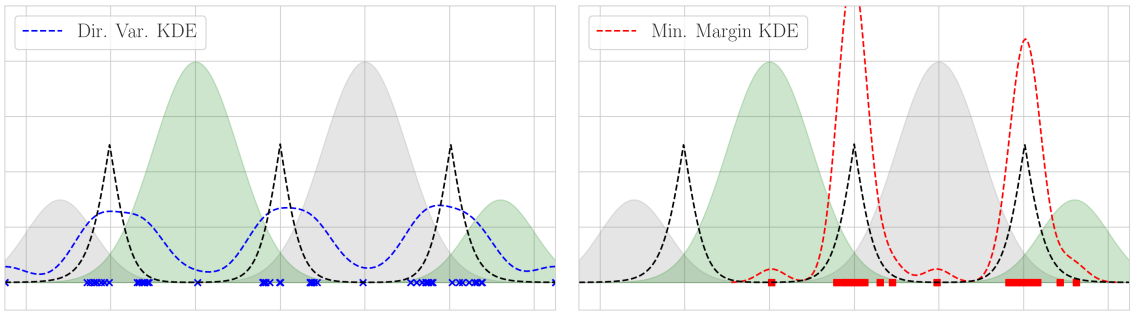


(f) KDE of choices from Dir. Var.



(g) KDE of choices from Unc. (SM)

Figure 14: Demonstration of **population-level** and **data-conditional** uncertainties in a toy example of a mixture of Gaussians in one dimension. The dashed black line represents a measure of the population-level uncertainty for the given setup; this is maximized in regions where the class-conditional densities are equal, $p(x|y = 1) = p(x|y = 0)$. In panels (b)-(e), we plot data-conditional uncertainties for the Dirichlet Variance (blue) and Unc. (SM) (red) acquisition functions at various iterations. See Section 6 for further experiment details.



(a) KDE of choices from Dir. Var.

(b) KDE of choices from Unc. (SM)

Figure 15: Visualization of a kernel density estimator (KDE) of the choices made from the first 50 query points by Dir. Var. and Unc. (SM) from the toy example shown in Figure 14. Note that the empirical distribution (reflected by the KDE) of the query points selected by Unc. (SM) implies that it has not sampled from the left-most region of large population-level uncertainty due to insufficient exploration earlier on in the active learning process.

Similarly, the choice of λ emphasizes the $100(\hat{K}-1)/\hat{K}$th percentile of acquisition function values at each iteration. This has the effect of biasing the sampling toward the $1/\hat{K}$ fraction of largest values in order to "focus the sampling" on regions where the acquisition function value is larger. While one doesn't necessarily have access to the true value of K in practice, we suggest that *overestimating* the number of clusters in the data set is a reasonable idea since the limiting case of $\lambda \to \infty$ corresponds to the Maximum Value policy—which already performs quite well in our experiments here. In our experiments, we take $\hat{K} = 2K$ as an overestimate of the number of clusters.

D.2 Low-rank Approximation of VOpt and ΣOpt in Larger Experiments

For the larger experiments, the matrix inversion involved in the "full" VOpt and ΣOpt is impractical, and so we use a computational workaround similar to the MCVOpt acquisition function. As was done in (Miller and Bertozzi, 2024; Miller et al., 2022), we approximate the VOpt and ΣOpt acquisition functions by projecting onto the first $r = 50$ eigenvectors of the corresponding graph Laplacian in order to reduce the computational burden of these methods. This constitutes a low-rank approximation of the inverse of the graph Laplacian matrix for the corresponding calculations of these acquisition functions. The error in this low-rank approximation may explain the comparative degradation in performance of the ΣOpt acquisition from the smaller to the larger data sets, though we remark that this approximation did not seem to significantly affect the results of VOpt and MCVOpt. An interesting direction for empirical and theoretical analysis would be to quantify how such low-rank approximations affect these acquisition functions that are intimately tied to covariance operators associated with the underlying graph-based Gaussian random field.

Appendix E. Example of Applying Exploration Results to Gaussian Mixture Model

We expand upon Example 1 in García Trillos et al. (2021). Let us consider the case where ρ_0, ρ_1 are Gaussians with unit variance in $\mathbb{R}$, with means separated by distance D, and where $w_1 = w_2$. Using classical theory for Hermite polynomials, we have that $\Theta = 1$. We can estimate

$$\begin{aligned}\mathcal{S} &= \int \frac{e^{-x^2/2}e^{-(x-D)^2/2}}{\sqrt{2\pi}(e^{-x^2/2} + e^{-(x-D)^2/2})}dx \\ &\leq \frac{1}{\sqrt{2\pi}}\int_{D/2}^{\infty} e^{-x^2/2}dx + \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{D/2} e^{-(x-D)^2/2}dx \leq \frac{2}{\sqrt{2\pi}}e^{-D^2/8} \leq e^{-D^2/8},\end{aligned}$$

where we have used the upper bound $\int_s^{\infty} e^{-z^2/2}dz \leq \frac{1}{\sqrt{2\pi}}e^{-s^2/2}$. We can also estimate

$$\begin{aligned}\mathcal{C} &= \frac{1}{\sqrt{2\pi}}\int \left| x - \frac{(xe^{-x^2/2} + (x-D)e^{-(x-D)^2/2}}{e^{-x^2/2} + e^{-(x-D)^2/2}}\right|^2 e^{-x^2/2}dx \\ &= \frac{1}{\sqrt{2\pi}}\int D^2 \frac{e^{-(x-D)^2}e^{-x^2/2}}{(e^{-x^2/2} + e^{-(x-D)^2/2})^2}dx \\ &\leq \frac{D^2}{\sqrt{2\pi}}\int \frac{e^{-(x-D)^2/2}e^{-x^2/2}}{(e^{-x^2/2} + e^{-(x-D)^2/2})}dx \\ &= \frac{2D^2\mathcal{S}}{\sqrt{2\pi}} \leq D^2 e^{-D^2/8}.\end{aligned}$$

Here we note that the bound from Proposition 11 gives a lower bound of

$$\sqrt{1 - D^2e^{-D^2/8}} - \frac{\sqrt{2D^2e^{-D^2/4}}}{1 - e^{-D^2/8}} \leq (\sqrt{\Theta(1 - K\mathcal{S})} - \frac{\sqrt{\mathcal{C}K\mathcal{S}}}{1 - \mathcal{S}}) \leq \lambda_{K+1},$$

and the upper bound

$$\lambda_K \leq \frac{2D^2e^{-D^2/8}}{1 - 2e^{-D^2/16}}$$

Next, we seek to obtain an upper bound on the operator norm of e^{Δ_ρ} (for simplicity, we'll assume that $s = 1$). In order to do so, we use the function

$$\Psi(t, x, x') = \frac{e^t}{\sqrt{8\pi t}}\exp\left(-\frac{|e^{-t}x - x'|^2}{4t}\right) + 2De^4\sqrt{t}e^{-|xe^{-t} - x'|/4D},$$

which one can verify satisfies $(\partial_t - \Delta_\rho)\Psi \geq 0$ and converges to a Dirac mass as $t \to 0$. If we let $G(t, x, x')$ be the fundamental solution of the operator $(\partial_t - \Delta_\rho)$, then by the maximum principle $G(t, x, x') \leq \Psi(t, x, x')$.

Now to estimate the operator norm at time $t = 1$, we can use the Young convolution inequality and a change of variables to obtain

$$\|u(1, \cdot)\|_2 \leq e\|u(0, \cdot)\|_1\|\bar{\Psi}\|_2, \qquad \|u(1, \cdot)\|_\infty \leq e\|u(0, \cdot)\|_2\|\bar{\Psi}\|_2,$$

where $\bar{\Psi}$ is given by

$$\bar{\Psi}(z) = \frac{e^{\gamma}}{\sqrt{8\pi}} e^{-z^2/4} + 2De^4 e^{\frac{-|z|}{4D}},$$

Squaring and integrating, we have

$$\|\bar{\Psi}\|_2 \leq 2\left(\frac{e^{2\gamma}\sqrt{2\pi}}{8\pi} + 16D^3 e^8\right).$$

In summary, for D large, we have that λ_K decays like e^{-D^2}, λ_{K+1} approaches -1, and the operator norm of the generalized heat equation at time 1 is bounded by D^3. We then insert these (with $t_1 = s = 1$) to bound ξ:

$$\begin{aligned}\xi &= |e^{\lambda_K t} - 1|\frac{(1+\sin(b))^2}{w_{min}}\|\rho\|_\infty \\ &\quad + \min_{t_1, s>0, t_1+s<t} e^{\lambda_{K+1}(t-t_1-s)}\|e^{t_1\Delta}\|_{L^1(\rho)\to L^2(\rho)}\|e^{s\Delta}\|_{L^2(\rho)\to L^\infty(\rho)} \\ &\lesssim te^{-D^2}(1+\sin(b))^2 + e^{-(t-1)}D^6\end{aligned}$$

Picking t of order D^2 gives $\lesssim D^6 e^{-D^2}(1-\sin(b))$, which decays rapidly in D.

In order to satisfy the assumption of Proposition 10, we pick $\sigma = e^{-D^2/32}$, and $a, b = D^{1/4}e^{-D^2/64}$. Substituting these choices into our constants, we obtain $\tilde{\delta} \sim e^{-D^2/16}$. After then applying our theorem, we obtain that, when using the heat kernel with $t = D^2$, our model is a $(\delta, \zeta, \varepsilon)$ class separator with $\delta \lesssim D^{1/2}e^{-D^2/32}, \zeta \sim 1$ and $\varepsilon \lesssim D^{1/4}e^{-D^2/64}$.

References

A. Agarwal, N. Jiang, S. M. Kakade, and W. Sun. Reinforcement Learning: Theory and Algorithms, 2021. Accessed via `https://rltheorybook.github.io/rltheorybook_AJKS.pdf`.

A. N. Angelopoulos and S. Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, March 2023. ISSN 1935-8237. doi: 10.1561/2200000101. URL `https://doi.org/10.1561/2200000101`.

J. T. Ash, C. Zhang, A. Krishnamurthy, J. Langford, and A. Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *International Conference on Learning Representations (ICLR)*, 2020.

O. Bachem, M. Lucic, and A. Krause. Practical coreset constructions for machine learning, 2017. arXiv preprint arXiv:1703.06476.

M.-F. Balcan, A. Broder, and T. Zhang. Margin based active learning. In *Conference on Learning Theory (COLT)*, volume 4539, pages 35–50. Springer Berlin Heidelberg, 2007. ISBN 978-3-540-72925-9. doi: 10.1007/978-3-540-72927-3_5.

M.-F. Balcan, A. Beygelzimer, and J. Langford. Agnostic active learning. *Journal of Computer and System Sciences*, 75(1):78–89, 2009. doi: 10.1016/j.jcss.2008.07.003.

D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel. MixMatch: A holistic approach to semi-supervised learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

A. L. Bertozzi and A. Flenner. Diffuse interface models on graphs for classification of high dimensional data. *SIAM Review*, 2016. doi: 10.1137/16M1070426.

A. L. Bertozzi, X. Luo, A. M. Stuart, and K. C. Zygalakis. Uncertainty quantification in graph-based classification of high dimensional data. *SIAM/ASA Journal on Uncertainty Quantification*, 6(2):568–595, 2018.

A. L. Bertozzi, B. Hosseini, H. Li, K. Miller, and A. M. Stuart. Posterior consistency of semi-supervised regression on graphs. *Inverse Problems*, 37(10):105011, September 2021. doi: 10.1088/1361-6420/ac1e80.

W. Cai, M. Zhang, and Y. Zhang. Batch mode active learning for regression with expected model change. *IEEE Transactions on Neural Networks and Learning Systems*, 28(7): 1668–1681, July 2017. doi: 10.1109/tnnls.2016.2542184.

J. Calder. GraphLearning Python package, January 2022.

J. Calder and D. Slepčev. Properly-weighted graph Laplacian for semi-supervised learning. *Applied Mathematics & Optimization*, 82(3):1111–1159, December 2020. ISSN 1432-0606. doi: 10.1007/s00245-019-09637-3.

J. Calder, B. Cook, M. Thorpe, and D. Slepčev. Poisson learning: Graph-based semi-supervised learning at very low label rates. In *International Conference on Machine Learning (ICML)*, pages 1306–1316, November 2020.

J. Calder, D. Slepčev, and M. Thorpe. Rates of convergence for Laplacian semi-supervised learning with low labeling rates. *Research in the Mathematical Sciences*, 10(1):10, February 2023. ISSN 2197-9847. doi: 10.1007/s40687-022-00371-x.

J. Chapman, B. Chen, Z. Tan, J. Calder, K. Miller, and A. Bertozzi. Novel batch active learning approach and its application on the synthetic aperture radar datasets. In *SPIE Defense + Commercial Sensing*, 2023.

Y. Chen, E. N. Epperly, J. A. Tropp, and R. J. Webber. Randomly pivoted Cholesky: Practical approximation of a kernel matrix with few entry evaluations. *Communications on Pure and Applied Mathematics*, 78(5):995–1041, 2025. doi: 10.1002/cpa.22234.

F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.

A. Cloninger and H. N. Mhaskar. Cautious active clustering. *Applied and Computational Harmonic Analysis*, 54:44–74, September 2021. ISSN 1063-5203. doi: 10.1016/j.acha.2021.02.002.

G. Dasarathy, R. Nowak, and X. Zhu. S2: An efficient graph based active learning algorithm with application to nonparametric classification. In P. Grünwald, E. Hazan, and S. Kale, editors, *Conference on Learning Theory (COLT)*, volume 40 of *Proceedings of Machine Learning Research*, pages 503–522, Paris, France, July 2015.

S. Dasgupta. Coarse sample complexity bounds for active learning. In *Advances in Neural Information Processing Systems*, volume 18, pages 235–242. MIT Press, 2006.

S. Dasgupta. Two faces of active learning. *Theoretical Computer Science*, 412(19):1767–1781, April 2011. doi: 10.1016/j.tcs.2010.12.054.

S. Dasgupta and D. Hsu. Hierarchical sampling for active learning. In *International Conference on Machine Learning (ICML)*, pages 208–215, Helsinki, Finland, July 2008. Association for Computing Machinery. ISBN 978-1-60558-205-4. doi: 10.1145/1390156.1390183.

E. B. Davies. *Heat Kernels and Spectral Theory*. Number 92. Cambridge University Press, 1989.

A. Deshpande and S. Vempala. Adaptive sampling and fast low-rank matrix approximation. In J. Díaz, K. Jansen, J. D. P. Rolim, and U. Zwick, editors, *Approximation, Randomization, and Combinatorial Optimization: Algorithms and Techniques (APPROX/RANDOM)*, pages 292–303, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. ISBN 978-3-540-38045-0.

L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*, volume 31 of *Stochastic Modelling and Applied Probability*. Springer, 1996. ISBN 978-1-4612-0711-5.

Y. Gal, R. Islam, and Z. Ghahramani. Deep Bayesian active learning with image data. In *International Conference on Machine Learning (ICML)*, pages 1183–1192, Sydney, NSW, Australia, August 2017.

N. García Trillos, F. Hoffmann, and B. Hosseini. Geometric structure of graph Laplacian embeddings. *Journal of Machine Learning Research*, 22(63):1–55, 2021.

Y. Gwon, S. Hwang, H. Kim, J. Ok, and S. Kwak. Enhancing cost efficiency in active learning with candidate set query. *Transactions on Machine Learning Research*, 2025.

S. Hanneke. A bound on the label complexity of agnostic active learning. In *International Conference on Machine Learning (ICML)*, pages 353–360, New York, NY, USA, June 2007. Association for Computing Machinery. ISBN 978-1-59593-793-3. doi: 10.1145/1273496.1273541.

S. Hanneke. Theory of disagreement-based active learning. *Foundations and Trends® in Machine Learning*, 7(2-3):131–309, June 2014. doi: 10.1561/2200000037.

S. Hanneke and L. Yang. Minimax analysis of active learning. *Journal of Machine Learning Research*, 16(109):3487–3602, 2015. ISSN 1533-7928.

S.-J. Huang, R. Jin, and Z.-H. Zhou. Active learning by querying informative and representative examples. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, pages 892–900. Curran Associates, Inc., 2010.

M. Ji and J. Han. A variance minimization criterion to active learning on graphs. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 556–564, March 2012.

H. Jiang and M. Gupta. Minimum-margin active learning, May 2019. arXiv preprint arXiv:1906.00025.

K.-S. Jun and R. Nowak. Chapter 10 - Bayesian active learning on graphs. In P. M. Djurić and C. Richard, editors, *Cooperative and Graph Signal Processing*, pages 283–297. Academic Press, January 2018. ISBN 978-0-12-813677-5. doi: 10.1016/B978-0-12-813677-5.00010-9.

A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell. Active learning with Gaussian processes for object categorization. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1–8. IEEE, 2007.

M. Karzand and R. D. Nowak. Maximin active learning in overparameterized model classes. *IEEE Journal on Selected Areas in Information Theory*, 1(1):167–177, May 2020. ISSN 2641-8770. doi: 10.1109/JSAIT.2020.2991518.

A. Kirsch, S. Farquhar, P. Atighehchian, A. Jesson, F. Branchaud-Charron, and Y. Gal. Stochastic batch acquisition: A simple baseline for deep active learning. *Transactions on Machine Learning Research*, 2023.

A. Krause and C. Guestrin. Nonmyopic active learning of Gaussian processes: An exploration-exploitation approach. In *International Conference on Machine Learning (ICML)*, pages 449–456, 2007.

D. Kushnir and L. Venturi. Diffusion-based deep active learning, March 2020. arXiv preprint arXiv:2003.10339.

Y. LeCun and C. Cortes. MNIST handwritten digit database. 2010.

Y. Ma, R. Garnett, and J. Schneider. Σ-optimality for active learning on Gaussian random fields. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, pages 2751–2759. Curran Associates, Inc., 2013.

K. Miller and J. Calder. Poisson reweighted Laplacian uncertainty sampling for graph-based active learning. *SIAM Journal on Mathematics of Data Science*, 5(4):1160–1190, 2023. doi: 10.1137/22M1531981.

K. Miller, H. Li, and A. L. Bertozzi. Efficient graph-based active learning with probit likelihood via Gaussian approximations. In *ICML Workshop on Experimental Design and Active Learning*, July 2020.

K. Miller, J. Mauro, J. Setiadi, X. Baca, Z. Shi, J. Calder, and A. Bertozzi. Graph-based active learning for semi-supervised classification of SAR data. In *SPIE Defense + Commercial Sensing*, 2022.

K. S. Miller and A. L. Bertozzi. Model change active learning in graph-based semi-supervised learning. *Communications on Applied Mathematics and Computation*, 6(2):1270–1298, 2024. doi: 10.1007/s42967-023-00328-z.

J. Murphy. Learning by active nonlinear diffusion (LAND) codebase. https://jmurphy.math.tufts.edu/Code/.

J. M. Murphy and M. Maggioni. Unsupervised clustering and active learning of hyperspectral images with nonlinear diffusion. *IEEE Transactions on Geoscience and Remote Sensing*, 57(3):1829–1845, March 2019. ISSN 1558-0644. doi: 10.1109/TGRS.2018.2869723.

C. Musco and D. P. Woodruff. Sublinear time low-rank approximation of positive semidefinite matrices. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 672–683, 2017. doi: 10.1109/FOCS.2017.68.

H. H. Nguyen, C. Jung, S. Salehi, T. Glück, A. Schmeink, and A. Kugi. Conformal cross-modal active learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5147–5157, 2026.

Y.-L. Qiao, C. X. Shi, C. Wang, H. Li, M. Haberland, X. Luo, A. M. Stuart, and A. L. Bertozzi. Uncertainty quantification for semi-supervised multi-class classification in image processing and ego-motion analysis of body-worn videos. *Image Processing: Algorithms and Systems*, 2019. doi: 10.2352/issn.2470-1173.2019.11.ipas-264.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, Mass, 2006. ISBN 978-0-262-18253-9.

P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B. B. Gupta, X. Chen, and X. Wang. A survey of deep active learning. *ACM Computing Surveys*, 54(9), October 2021. ISSN 0360-0300. doi: 10.1145/3472291.

C. Riis, F. Antunes, F. B. Hüttel, C. L. Azevedo, and F. C. Pereira. Bayesian active learning with fully Bayesian Gaussian processes. In A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

N. Rittler and K. Chaudhuri. A two-stage active learning algorithm for k-nearest neighbors. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, editors, *International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pages 29103–29129, July 2023.

J. Schreiter, D. Nguyen-Tuong, M. Eberts, B. Bischoff, H. Markert, and M. Toussaint. Safe exploration for active learning with Gaussian processes. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*, pages 133–149. Springer, 2015.

O. Sener and S. Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations (ICLR)*, 2018.

B. Settles. *Active Learning*, volume 6. Morgan & Claypool Publishers LLC, June 2012. doi: 10.2200/s00429ed1v01y201207aim018.

G. Shafer and V. Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9:371–421, 2008.

K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C.-L. Li. FixMatch: Simplifying semi-supervised learning with consistency and confidence. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 596–608. Curran Associates, Inc., 2020.

R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.

S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2:45–66, November 2001. ISSN 1533-7928.

U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, December 2007. ISSN 1573-1375. doi: 10.1007/s11222-007-9033-z.

V. Vovk, A. Gammerman, and G. Shafer. *Algorithmic Learning in a Random World*. Springer International Publishing, Cham, 2022. ISBN 978-3-031-06648-1. doi: 10.1007/978-3-031-06649-8.

H. Xiao, K. Rasul, and R. Vollgraf. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms, 2017.

S. Yang, Y. Dong, R. Ward, I. S. Dhillon, S. Sanghavi, and Q. Lei. Sample efficiency of data augmentation consistency regularization. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 3825–3853, April 2023.

X. Zhan, Y. Wang, and A. B. Chan. Asymptotic optimality for active learning processes. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2022.

D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In *Advances in Neural Information Processing Systems*, pages 321–328. MIT Press, 2004.

X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using Gaussian fields and harmonic functions. In *International Conference on Machine Learning (ICML)*, pages 912–919, Washington, DC, USA, August 2003a. AAAI Press. ISBN 978-1-57735-189-4.

X. Zhu, J. Lafferty, and Z. Ghahramani. Combining active learning and semi-supervised learning using Gaussian fields and harmonic functions. In *ICML Workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*, pages 58–65, 2003b.