THEORY AND PRACTICE OF
ACTIVE LEARNING

RON BEGLEITER
THEORY AND PRACTICE OF ACTIVE LEARNING

RESEARCH THESIS

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

RON BEGLEITER

SUBMITTED TO THE SENATE OF THE TECHNION — ISRAEL INSTITUTE OF TECHNOLOGY
ADAR, 5773 HAIFA MARCH, 2013
ACKNOWLEDGMENTS

First and foremost, I am grateful to my doctoral advisors Prof. Nir Ailon and Prof. Ran El-Yaniv.

They both taught me how to do research, how to ask the right questions and how to answer them, and how to do both mathematics and empirical assessment. Their wisdom, creativity, integrity, humility, and generosity will continue to inspire me. I was indeed fortunate to have them as my advisors.

Nir was always two steps ahead of me, and it was he who paved the way to the successful completion of this work. I am deeply grateful to him for sharing with me his vast knowledge, intuition, and ingenious observations, as well as for his endless support, and encouragement. For all these reasons, Nir is a genuine role model and the ideal advisor — I couldn’t ask for better!

Ran trained and guided me from the very first stages of my research, and to him I owe the very DNA of my work. In fact, without Ran, my work could never have come into being. I thank him deeply for committing precious time and energy to my project and to me, for always keeping my sights trained on the big picture, and for setting the highest possible standards and demonstrating, again and again, that we could attain them. I was privileged to learn from such a true master.

I am in debt to my co-authors, Dr. Esther Ezra, and Dr. Dmitry Pechyony. Esther, who also shared her deep knowledge and expertise with me, provided me with the ideas and the tools that turned out to be critical for the development of the main theme of this thesis. I thank her greatly for her contribution.

I started collaborating with Dmitry in the early stages of my doctoral studies, when he and I were both students. I thank him
for inspiring me with his wisdom, for being a friend, and for teaching me how fun research collaboration can be.

I am thankful to Prof. Masashi Sugiyama for hosting me at the Tokyo Institute of Technology in the summer of 2008. I thank friends and colleagues who assisted along the way: Amit, Meir, Dany Dörr, Marko Jankovic, Gilad Kutiel, Simcha Lechtman, Masashi Sugiyama, Ryota Tomioka, Reut Tsarfaty, Liwei Wang, Yair Wiener, and David Yanay. I thank Shahar Mendelson for encouraging me when I needed it the most. I am thankful to Itai Brickner, and Stas Krichevsky from Kontera Technologies for sharing the enjoyment and fun of doing applied machine learning — I have learned a lot from both of them.

I am thankful to PASCAL – European Network of Excellence, Masashi Sugiyama, organizers of MLSS’06, and organizers of COLT’12 for their financial support.

I deeply thank my grandfather, Ernst; my parents, David and Hanna; my brother, Shy; and my sister, Tal, for their support, love, and care. This thesis is dedicated to them.

List of publications. (I have been a major contributor in all of the following publications.)


Remarks:

THE GENEROUS FINANCIAL HELP OF THE TECHNION IS
GRATEFULLY ACKNOWLEDGED
## Contents

### Abstract

1 Introduction

1.1 Problem Definition ............................................. 7
1.2 Proposed Solution .............................................. 7
1.3 Major Contributions ............................................ 7
1.4 Thesis Outline .................................................. 9

2 Brief Introduction to Active Learning ............................. 11

2.1 Settings and Notations ......................................... 11
2.2 Theory of Active Learning .................................. 13
  2.2.1 Complexity Terms ......................................... 13
  2.2.2 Alternative Settings and Guarantees ..................... 16
2.3 Algorithmic Ideas ............................................. 17
  2.3.1 Sampling by Uncertainty .................................. 17
  2.3.2 Utility Maximization (Error Minimization) ............ 19
  2.3.3 Online Weighting .......................................... 20
  2.3.4 Other Ideas ................................................ 21

3 The Method of Smooth Relative Regret Approximations ....... 23

3.1 Smooth Relative Regret Approximations (SRRA) .............. 23
3.2 Constant Uniform Disagreement Coefficient Implies Efficient SR-RAs ........................................... 27
  3.2.1 The Construction .......................................... 28
3.3 Convex Relaxations .......................................... 32
3.4 Discussion .................................................. 33

4 Active Preference-based Ranking Using SRRAs ............... 35

4.1 Problem Definition ........................................... 36
4.2 The Necessity of Active Learning ............................. 38
4.3 Disagreement Coefficient Arguments Are Not Sufficient for Effective Active Learning ......................... 39
4.4 Better SRRA for LRPP .......................................... 40
List of Figures

2.1 Pool-based active learning protocol ........................................... 12
2.2 Stream-based active learning protocol ................................. 13
3.1 Illustration of Disagreement-SRRA Elements ......................... 29
4.1 Maximal distance by a single pair inversion ....................... 39
4.2 Illustration: Sampling $R_{u,t}$ ................................................. 41
4.3 Example: Arrangement and Duality View of $X$ and $H$ .......... 48
4.4 Synthetic model-based preferences. We permuted the rows and columns on the basis of the $w$-induced order, resulting in a $Y_{\text{perfect}}$ noiseless ranking, which contrary to the upper-right and lower-left triangles depicted here, would have appeared as perfect solid-dark and solid-light triangles, respectively .................. 52
4.5 Results for synthetic datasets averaged over 10 runs, accompanied by standard error of the mean ......................... 53
4.6 Example for an Amazon Turk HIT that provides a label for the COST dataset .......................................................... 55
4.7 Three crowdsourcing preferences. The dark colored $W_{u,v}$ depicts that row alternative, $u$, is preferred over the column alternative $v$. A fully sortable, noiseless preference would have been represented here as solid dark upper triangles, representing anti-symmetric matrices .................................................................... 56
4.8 Comparing SRRA and random samplers. The first row corresponds to COMBI-SVM + CLIMB solver; the second row to CLIMB; and the last to SVM. The first column corresponds to COST, the second to WINE, and the last to DATE. Each result is an average of 10 runs along with the standard error of the mean .................. 57
4.9 Example of a recipe text ........................................................... 58
4.10 Ranking by preference-majority versus local-improvement. The dark color illustrates that the row alternative is preferred over that of the corresponding column. The REF ranking corresponds to row indices, where the top and bottom rows are most and least preferred, respectively .......................................................... 59
5.1 Semi-Supervised Clustering as a Graph Problem ........................................ 65
5.2 An example of an ultrametric tree with $\ell = k = 3$. Let $v_i$ denote the vertex indexed with $i \in [27]$. Observe that $\tau(v_{25}, v_{27}) = 1$, $\tau(v_{22}, v_1) = 3$ is the maximal distance, and $\mathcal{V}(\mathcal{R}_2) = \{C_1, C_2, C_3\}$ defines the tree’s first level. ................................................... 78

6.1 Motivating example. ..................................................................................... 85
6.2 $\texttt{+EXPLORE}$ routine. ............................................................................. 89
6.3 Comparing the three methods: $\texttt{RAND}$, $\texttt{SELF-CONF}$, and $\texttt{+EXPLORE}$. Each point in each axis comprises two mean error results of two methods (in the x-axis and the y-axis) over one of the datasets for a training size $m = 50$. ........................................................................... 93
6.4 The effect of dynamic exploration: Comparing the learning (error) curves of $\texttt{SELF-CONF}$ with $\texttt{SELF-CONF+EXPLORE}$. Queries by exploration (using $Q_k$) are indicated by dark dots. .................. 94
List of Tables

4.1 The number of cycles in the different crowdsource datasets. We present the number and size of strongly connected components (scc’s). ................................................................. 60

6.1 The error (%) of the “best” representatives of the +EXPLORE, Q, and P methods. The lowest error in each row (over all columns) appears in bold font. .......................................................... 92
Abstract

The active-learning model is concerned with ways to expedite the learning phase through interaction with the labeling process. In a pool-based active-learning setting, the learning algorithm receives a set of unlabeled examples, as well as access to an oracle that contains the full labeling information on that particular set. The learner’s goal is to produce a nearly optimal hypothesis, while requiring the minimum possible interactions with the oracle.

In this thesis, we present a novel smoothness condition over empirical risk error estimators and show its usefulness for active pool-based learning. Instead of estimating the risk directly, we target regrets relative to pivot hypotheses. We show that such smooth relative regret estimators yield an active-learning algorithm that converges to a competitive solution at a fast rate.

We show three specific constructions of such smooth estimators. The first is obtained when the only assumptions made are bounds on the disagreement coefficient and the VC dimension. This leads to an active-learning algorithm that almost matches the best-known algorithms that use the same assumptions. On the other hand, we present two problems of vast interest, for which a direct analysis of the relative regret function produces state-of-the-art learning strategies. The two problems we study are concerned with learning relations over a ground set, where one problem deals with order relations and the other with equivalence relations (with a bounded number of equivalence classes).

Our smoothness condition, we argue, influences sampling methods that should be carefully biased in a way that incorporates exploration of all hypothesis space, along with exploitation of a current candidate solution. Following this idea, we present a heuristic that enhances any active-learning algorithm with systematic explorations. We show that this heuristic significantly improves leading active-learning heuristics within a graph-based transductive setting.
Abbreviations and Notations

\( \varepsilon \) Error approximation parameter
\( \pi, \sigma \) Permutations over a finite set \( V \)
\( \theta \) The universal disagreement coefficient
\( \text{ALG} \) Algorithm \( \text{ALG} \), e.g., \( \text{SVM} \)
\( \mathcal{H} \) Hypothesis class, the set of feasible hypotheses
\( d \) Either VC dimension, or geometric dimension, i.e., \( \mathbb{R}^d \); (context dependent)
\( \mathcal{D} \) Distribution over instance-label pairs \((X, Y) \sim \mathcal{D}\)
\( \mathcal{D}_X \) Marginal distribution over the instance space \((X)\)
\( \text{dist} \) Distance function over hypotheses, \( \text{dist} : \mathcal{H} \times \mathcal{H} \rightarrow \mathbb{R} \)
\( d_{\text{SF}} \) Spearman’s Footrule distance
\( \mathbb{E} [\cdot] \) Expectation
\( \text{er} \) Error rate, \( \text{er}_\mathcal{D}(h) = \mathbb{E}_{X \sim \mathcal{D}} [h(X) \neq Y(X)] \)
\( \nu \) The noise rate, \( \nu = \inf_{h \in \mathcal{H}} \text{er}_\mathcal{D}(h) \)
\( \text{ERM} \) Empirical Risk Minimization
\( h \) Hypothesis in \( \mathcal{H} \)
\( h^* \) The optimal hypothesis, s.t., \( \text{er}_\mathcal{D}(h^*) = \nu \)
\( \text{LRPP} \) Learning to Rank from Pairwise Preferences
\( \mathbb{P} [\cdot] \) Probability
\( \text{PTAS} \) Polynomial Time Approximation Scheme
\( \mathbb{R} \) The set of real numbers
\( \text{reg}_h \) Relative regret function with respect to hypothesis \( h \)
\( \text{SRRA} \) Smooth Relative Regret Approximation
\( V \) A finite set of \( n \) elements
\( v, u \) Elements in \( V \)
\( u \) An Embedding of \( u \in V \) in \( \mathbb{R}^d \) (for some fixed \( d \))
\( \text{VC} \) Vapnik-Chervonenkis (as in VC dimension)
\( \mathcal{X} \) Instance space
\( \mathcal{Y} \) Label space, here it is mostly binary: \( \mathcal{Y} = \{0, 1\} \)
\( Y(X) \) Labeling function (deterministic), \( Y : \mathcal{X} \rightarrow \mathcal{Y} \)
Chapter 1

Introduction

Efficient utilization of unlabeled examples during a learning process can be very advantageous when using limited labeling to construct accurate classifiers. The abundance of information led by “big data” phenomena like cloud computing, personalization, and the social web has motivated the research community to make considerable efforts in this direction. The three most prominent approaches for achieving this goal are semi-supervised, transductive, and active learning. Semi-supervised and transductive learning are close in nature to the standard PAC learning, and mainly deal with ways to benefit from the side information of unlabeled examples. Here we are concerned with active learning, a model that studies ways to accelerate learning by acquiring labels in interactive ways.

Unlike standard supervised learning, in active learning the learner chooses which instances to learn from. In the streaming setting, the learner may reject labels of instances that arrive in a stream; in the pool setting, the learner may collect a pool of instances and then choose a subset from which to request labels. Although it is a relatively young field compared to traditional (passive) learning, there is by now a significant body of literature on the subject (see, e.g., Cohn et al., 1994; Freund et al., 1997; Dasgupta, 2005; Castro et al., 2005; Kääriäinen, 2006; Balcan et al., 2006; Sugiyama, 2006; Hanneke, 2007a,b; Balcan et al., 2007; Dasgupta et al., 2007; Bach, 2007; Castro and Nowak, 2008; Balcan et al., 2008; Dasgupta and Hsu, 2008; Cavallanti et al., 2008; Hanneke, 2009; Beygelzimer et al., 2009, 2010; Koltchinskii, 2010; Cesa-Bianchi et al., 2010; Yang et al., 2010; Hanneke and Yang, 2010; El-Yaniv and Wiener, 2010; Hanneke, 2011; Orabona and Cesa-Bianchi, 2011; Cavallanti et al., 2011; Yang et al., 2011; Wang, 2011; Minsker, 2012). A more comprehensive overview of active learning, along with a proper contextual background will be provided in Chapter 2.

Active learning can be viewed as a search game in which the learner searches for a “good” hypothesis by means of queries that restrict the search space. This approach is especially appealing when the search space contains the underlying labeling function. That is, the search space contains a perfect hypothesis that never errs, a setting known as the realizable. When this is the case, the search
space is essentially the set of hypotheses that are congruent with all revealed labels. This subset of hypotheses is called the version space. Each new exposed label breaks the version space into hypotheses that either agree or disagree with the label. The merit of a query is thus tied to the potential proportion of the corresponding split. The revealed labels may potentially shrink the search space in constant factors (e.g., as in a generalized binary search). This means that the optimal hypothesis may potentially be revealed by a mere logarithmic number of queries, which is an exponential improvement over passive learning rates.

However, it turns out that exploiting the obtained labeled examples is generally not enough. Dasgupta (2005) shows in his breakthrough theoretical work a simple example wherein the probability that a query will split the search space is minuscule, thus the only way to make progress is by trying to locate such “split” examples through exploration.

When the setting is non-realizable, this exploration–exploitation dilemma becomes acute. The search cannot be led by the set of hypotheses that are unanimous on the sampled labels because every candidate solution errs. The labeled examples may be exploited for the sake of shrinking the search space only when there is some “guarantee” that we will not lose potential solutions. In cases where the “confidence” in the labeled examples is low, the learner should enhance it by using new examples to make it more robust to the problem’s inherent error noise.

From the theoretical point of view of active learning, the (few) existing active-learning complexity terms quantify how difficult it is to apply significant search space reductions. We argue that in difficult cases such as these, the search “direction” is static and thus choices are only concerned with better search state exploitations. For example, Dasgupta’s (2005) splitting index captures how common it is for labels to break the search space into significant portions. Thus, the splitting index measures how easy it is to sample a good set of labeled examples that shrink the version space sufficiently. This expresses how well we can exploit the current set of labels. Similarly, Hanneke’s (2007b) disagreement coefficient measures the “volume” of the set of instances on which the $r$-ball of hypotheses around the optimal hypothesis is not unanimous on. Again, this set consists of all query candidates for reducing the search space.

Among the above two complexity terms, the disagreement coefficient of Hanneke (2007b) has become a central data-independent invariant in proving active learning rates. As a result, the analysis of the vast majority of the theoretically justified active learning algorithms is mainly concerned with efficient exploitations. When this is not enough (i.e., when exploration is needed), the analysis is often accompanied by certain structural or Bayesian assumptions about the noise (especially the model of Mammen and Tsybakov, 1999; Tsybakov, 2004). The problem is analogous under “nicely” behaved noise to the realizable case, in which, under sufficient conditions, exploitation can be the main concern. Indeed, this type of assumption provides excellent labeling rate guarantees that usually
outperform passive learning (e.g., Balcan et al., 2007; Castro and Nowak, 2008; Hanneke, 2009; Koltchinskii, 2010; Yang et al., 2010; Wang, 2011; Yang et al., 2011; Minsker, 2012).

On the other hand, a few empirical works consider the role of exploration explicitly (e.g., Lindenbaum et al., 1999; Baram et al., 2004; Osugi et al., 2005). All of them indicate that sensible active learning methods should perform systematic explorations. This serves as real motivation for examining this phenomenon from a theoretical point of view.

1.1 Problem Definition

Our work is concerned with pool-based active learning only. We deal mostly with distribution-free settings, thus we do not consider any structural or Bayesian noise assumptions. Our analysis was carried out under the worst-case analysis framework. We focused on the non-realizable setting, known as the agnostic active learning setting. The hypothesis space in this setting does not necessarily contain the ground truth labeling function; as a result, the analogy of active learning as a search through the version space is no longer valid. The active learner has to trade off between exploitation of the labeled sample, which is a proxy to the labeling process, and exploration of new sample biases.

1.2 Proposed Solution

The essence of this thesis lies in exploration–exploitation tradeoffs in active learning. We define a smoothness condition on actively learned empirical risk minimizers that quantify such tradeoffs (implicitly); the condition is accompanied by a well-justified active learning meta-algorithm.

Recall that empirical risk minimization (ERM) is a learning paradigm, which is based on the idea that it is possible to approximate the expected loss of hypotheses using their empirical mean. Following the ERM philosophy, our condition deals with empirical mean and expected loss differences. We condition the deviation of such differences defined between any hypothesis and some fixed pivotal one. Our condition ensures that such deviations vary smoothly with the hypothesis space distance of their operand and the pivotal hypothesis. Thus, we require that the density of the sample that defines the empirical estimator be correlated with these distances. In other words, the learner should balance between exploitation of hypotheses in close proximity to the pivot and exploration of far hypotheses. Our solution can be intuitively viewed as a holistic approach to the active learning’s explore–exploit Catch-22.
1.3 Major Contributions

We establish a novel approach to pool-based active learning and outline its benefits and drawbacks. The key ingredients of our approach are a novel smoothness condition on ERM estimators of relative regrets and a corresponding active learning algorithm that uses such estimators. Our algorithm is guaranteed to output a hypothesis that errs no more than $(1 + \varepsilon)$ times the optimal error rate and to output such hypothesis at a “fast” rate (exponential in $\varepsilon$). The algorithm depends on specific smooth estimator constructions and solvers for solving the specific ERMs. The specific main contributions are listed below:

A novel approach to active learning. Our smoothness condition, accompanied by a well-justified meta-algorithm, brings a novel approach to active learning. It contrasts with the “shrinking version spaces” algorithmic approach that currently dominates the active learning field. For two important problems in which standard active learning methods fail to provide meaningful guarantees, we use our approach to define active learning algorithms with non-trivial guarantees.

General purpose construction matches state-of-the-art sample complexity. For the generic pool-based setting, we define a construction of smooth estimators that uses knowledge only on disagreement coefficient $\theta$ and VC dimension $d$. We prove that our construction yields (w.h.p.) a query complexity that is in

$$O(\theta d(\log^2(1/\nu))(\log \theta)),$$

where $\nu$ denotes the problem’s inherent agnostic noise. This query-complexity bound is $O(\log(1/\nu))$ times the best known bounds using disagreement coefficient and VC dimension bounds only (e.g., Dasgupta et al., 2007; Beygelzimer et al., 2009).

Extending the state-of-the-art of learning to rank from pairwise preferences. We define a specific construction of smooth estimator for the problem of active learning to rank from pairwise preferences (also known as query-efficient version of minimum feedback in arc-set tournaments). The resulting algorithm improves on what were previously the best results (Ailon, 2012) in two ways: Most importantly, it is more general and can be applied over any set of hypotheses (whereas the former solution is restricted to a specific set); secondly, it slightly improves the sample complexity and is guaranteed (w.h.p.) to query no more than $O(\varepsilon^{-3} n \log^5 n)$ labels, where $n$ denotes the number of alternatives we should rank. Additionally, we demonstrate that our specific construction beats both uniform sampling and any known active learning method that is based-on disagreement coefficient and VC dimension arguments only (e.g., our general-purpose construction).
Outlining the state-of-the-art of active correlation clustering. We define a query-efficient variant of correlation clustering (semi-supervised clustering), in which the number of clusters is fixed and known. We present a corresponding construction that achieves (w.h.p.) a query complexity

$$O \left( n \max \{ \varepsilon^{-2}k^3, \varepsilon^{-3}k^2 \} \log^2 n \right),$$

where $n$ is the number of items to cluster and $k$ is the number of clusters. We demonstrate that this result is not trivial in the sense that it beats both uniform sampling and any known active learning method that is based on disagreement coefficient and VC dimension arguments only.

1.4 Thesis Outline

Chapter 2 provides definitions and a brief contextual background on active learning. In Chapter 3, we lay out the key ingredients of our method, which we term Smooth Relative Regret Approximations (SRRA). Our approach relies on specific constructions of SRRA estimators, and on a way to solve corresponding ERM optimizations. We thus also present a general-purpose construction of smooth estimators, analyze the corresponding query complexity of the resulting algorithm, and finally, discuss optimizations under convex relaxations. Chapters 4 and 5 deal with query-efficient variants of learning to rank from pairwise preferences and correlation $k$-clustering respectively. In each of these chapters, after presenting the problem, we discuss the pitfalls of applying uniform sampling and disagreement-based approaches. We then present a corresponding SRRA construction and show that it defines the state-of-the-art guarantees. In Chapter 4 we additionally provide empirical proof of concept of our approach using a novel benchmark consisting of several synthetic and most importantly, three real-world datasets. In Chapter 5, we also discuss a hierarchical variant of the problem in which we try to actively learn ultrametrics. In our concluding Chapter 6 we present a powerful heuristic strategy for adding systematic exploration abilities to any active learner. Focusing on transductive graph-based active learners, we empirically demonstrate the usefulness of our approach and show that it defines the corresponding state-of-the-art empirical result with respect to standard and common benchmarks.
Chapter 2

Brief Introduction to Active Learning

This chapter provides a contextual background of active learning and introduces the main concepts and notations we will use in the following chapters. We start by presenting two main active learning settings analogous to batch and online supervised learning. Then we sketch the main active learning theoretical concepts and complexity terms. Last, we outline the most important active learning algorithmic ideas.

2.1 Settings and Notations

We begin with the core learning elements that are common both to the supervised learning (passive) and active learning settings. Let $\mathcal{X}$ be an instance space and $\mathcal{Y}$ be a label space. Let a distribution over $\mathcal{X} \times \mathcal{Y}$ be denoted by $\mathcal{D}$, with corresponding marginals $\mathcal{D}_X$ and $\mathcal{D}_Y$. Here we focus on a classification scenario in which the label space is discrete and finite (the case in which $\mathcal{Y}$ is continuous is known as regression). We will also assume that unless explicitly stated, the label space is binary, $\mathcal{Y} = \{0, 1\}$, and the labeling is a deterministic function over instances, that is, $Y(x): \mathcal{X} \rightarrow \{0, 1\}$. These assumptions do not restrict the results and can be generalized.

Let $\mathcal{H}$ be a set of functions mapping $\mathcal{X}$ to $\mathcal{Y}$. We call $\mathcal{H}$ a hypothesis class, and function $h \in \mathcal{H}$ a hypothesis. The labeling function $Y(\cdot)$ is unknown, and the hypothesis class represents the set of “feasible” estimations of $Y$. The discrepancy between estimations $h(x)$ and true labels $Y(x)$ for $x \in \mathcal{X}$ is being quantified by a loss function $\ell: \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}^+$. The performance of a hypothesis $h$ is measured upon its expected (in $\mathcal{D}$) loss, which is known as the risk function

$$\text{risk}_{\mathcal{D}, \ell}(h) = \mathbb{E}_{(X,Y) \sim \mathcal{D}} [\ell(h(X), Y)].$$

Here we will mostly use the zero-one loss function $\ell(a, b) = 1_{a \neq b}$ (where $1_{(\cdot)}$ is the
indicator function). In this case, the risk simply becomes the hypothesis error rate

\[ \text{er}_D(h) = \mathbb{E}_{D_X} [h(X) \neq Y(X)] . \]

We define the noise rate of the class \( \mathcal{H} \) to be \( \nu = \inf_{h \in \mathcal{H}} \text{er}_D(h) \). We will assume here that there exists an optimal hypothesis \( h^* \in \mathcal{H} \) that achieves the noise rate \( \text{er}_D(h^*) = \nu \). The case in which \( h^* \equiv Y \), that is \( \nu = 0 \), is called the realizable setting. The other case where \( \nu > 0 \) is known as the agnostic setting. Here we will consider the agnostic setting only.

We say that hypothesis \( h \) is \( \varepsilon \)-competitive if \( |\text{er}_D(h) - \nu| < \varepsilon \nu \). In other words, its risk does not exceed \((1 + \varepsilon)\) times the hypothesis class noise rate. We measure the performance of the active learning algorithm by its convergence rate: the minimal amount of labels it consumes for producing (w.h.p.) an \( \varepsilon \)-competitive hypothesis, \( h \in \mathcal{H} \). This is known as sample or query complexity.

There are two definitions of the sample complexity. The first is called the self-verifying sample complexity, which counts the minimal number of labels required both to produce a competitive hypothesis and prove (w.h.p.) it is indeed competitive. This is the commonly used definition, and we will use it as well. The alternative definition counts the labeling overhead needed only to produce a competitive hypothesis (without verifying it). Indeed, there are cases in which the difference between the two definitions is significant. We discuss this further in Section 2.2 below.

In this thesis, we focus only on what is known as the pool-based active learning setting; which is one of the two commonly analyzed active learning settings. Both settings are presented below.

**Require:** active learning algorithm \( \text{ALG} \), a pool \( X \subseteq \mathcal{X} \), labeling oracle \( Y_X \), and query budget \( T \)

\[ S_0 \leftarrow \emptyset \], denotes the labeled sample

\[ \text{for } i = 1, \ldots, T \text { do } \]

\( \text{ALG uses } (S_0, S_1, \ldots, S_i) \) and \( X \) to choose an instance \( x \in X \) (or a batch)

\[ S_{i+1} \leftarrow S_i \cup \{(x, Y_X(x))\} \]

end for

**return** hypothesis \( h = \text{ALG}((S_0, \ldots, S_T), X) \)

**Figure 2.1:** Pool-based active learning protocol.

**Pool-Based Active Learning.** Pool-based active learning is analogous to batch supervised learning, wherein the learner obtains access to a labeling oracle that possesses the true labeling information of some subset of \( \mathcal{X} \). This subset is called the pool. A protocol of pool-based active learning is presented in Figure 2.1. The protocol is iterative — in each round the algorithm queries some label, based on its current set of labeled instances.
**Stream-Based Active Learning.** Stream-based active learning is analogous to online learning. Here, the active learner receives a stream of instances $x_1, x_2, \ldots \in \mathcal{X}$ one at a time (each is drawn i.i.d. and paired with its label, $(x_i, y_i)$, from $\mathcal{D}$). At each step it decides whether to request or ignore the instance’s label. The online nature of the algorithm usually imposes restricted resources (e.g., $O(1)$ space complexity). Thus, label requests are mostly based on a current solution $h_i$. In comparison, in the pool-based setting the learner can maintain the sampled examples and sequence of solutions. See Figure 2.2 for the corresponding protocol. (Note that a few stream-based algorithms maintain additional resources.)

**Require:** active learning algorithm $\text{ALG}$, and number of rounds $T$

\begin{verbatim}
    $h_0$ initial hypothesis
    for $i = 1, \ldots, T$ do
    
    an instance-label pair $(x_i, y_i)$ is drawn from $\mathcal{D}$
    \hspace{1em} $\text{ALG}$ receives $x_i$ and decides whether to ask for $y_i$ (based on $h_{i-1}$)
    \hspace{1em} $\text{ALG}$ emits hypothesis $h_i$
    
    end for

    return hypothesis $h_T$
\end{verbatim}

*Figure 2.2:* Stream-based active learning protocol.

### 2.2 Theory of Active Learning

We discuss below the main theoretical active-learning results. We start by listing known active-learning complexity terms and then we present alternative settings and guarantees for the agnostic setting.

#### 2.2.1 Complexity Terms

Complexity terms capture the essence of the problem at hand and accordingly, their usefulness is measured by their generality, simplicity, and expressiveness. In other words, they should rely upon assumptions that are as minimal as possible (generality), easy to describe and compute (simplicity), and useful for creating relations between the performance of algorithms and the nature of a problem’s instance at hand. Active learning is a relatively young research field, and until recently it lacked a theoretical foundation. This was changed by the breakthrough work of Dasgupta (2005) who defined the first active learning specific complexity term, the **splitting index**. Since then only two more terms have been introduced: the **teaching dimension** (Hanneke, 2007a; Goldman and Kearns, 1995),\(^1\) and the

---

\(^1\)Hanneke was the first to demonstrate the usefulness of the teaching dimension measure for analyzing sample complexities in active learning.
disagreement coefficient (Hanneke, 2007b). The disagreement coefficient is currently the popular term to prove sample bounds of active learners. We also use it for analyzing our general-purpose construction (Section 3.2.1).

**Splitting Index.** The idea of splitting indices proposed by Dasgupta (2005) facilitated the first general theory of convergence rates for active learning. In the realizable case, we can view the active learning process as a (binary) search, where the algorithm searches a competitive hypothesis by “flipping” instance labels. Using this analogy, a point \( x \in X \) informativeness is determined by the ratio it splits the “search set.” That is, for a finite \( Q \subseteq \{ \{ h_1, h_2 \} : h_1, h_2 \in \mathcal{H} \} \), define \( Q_x^y = \{ \{ h_1, h_2 \} : h_1(x) = h_2(x) = y \} \). An instance \( x \) is said to \( \rho \)-split \( Q \) if

\[
\max_{y \in \{0,1\}} |Q_x^y| \leq (1 - \rho) |Q| .
\]

**Definition 2.1 (Splitting Index)** We say \( A \subseteq \mathcal{H} \) is \((\rho, \Delta, \tau)\)-splittable if, for all finite \( Q \subseteq \{ \{ h_1, h_2 \} \in \binom{\mathcal{H}}{2} : P[h_1(X) \neq h_2(X)] > \Delta \} \), then \( P(X \rho\text{-splits } Q) \geq \tau \).

The splitting index intuitively indicates how common informative instances are. Perhaps the most elegant property of the splitting index is that it links between the number of labels required and the number of unlabeled instances (size of the pool); this property is unique to the splitting index, not shared by any other active learning complexity term. The next theorem gives a general performance result that uses this term:

**Theorem 2.1 (Dasgupta (2005))** For any \( \mathcal{H} \) with VC dimension \( d \), there is an algorithm such that if the set \( \{ h : P[h(X) \neq h^*(X)] \leq 4\Delta \} \) is \((\rho, \Delta, \tau)\)-splittable for all \( \Delta \geq \varepsilon/2 \) , then with a probability of at least \( 1 - \delta \) the algorithm draws \( \tilde{O}((1/\varepsilon) + (d/\rho\tau)) \) instances and uses \( \tilde{O}(d/\rho) \) labels and returns a hypothesis with error at most \( \varepsilon \). (\( \tilde{O}(\cdot) \) is the analog of big-\( O \), which suppresses poly-logarithmic terms.)

**Teaching Dimension.** The teaching dimension is a label complexity term defined by Goldman and Kearns (1995), designating the minimum number of labeled examples required to present to any consistent passive learning algorithm in order to uniquely identify any hypothesis in the hypothesis class. Related extensions (Hegedüs, 1995; Hellerstein et al., 1996) were used to tightly characterize the number of membership queries sufficient for exact learning.\(^2\)

\(^2\)Learning with membership queries is a setting that comes from the COLT community and is related to active learning. The main difference between the two is that in membership queries, \( \mathcal{X} \equiv \mathbb{R}^d \). Note that in many natural problems this cannot be assumed to be true. For example, it is not reasonable that every point in \( \mathbb{R}^d \) represents a bag-of-words of some document. In the exact learning setting the goal is to identify an errorless hypothesis in the class.
Hanneke (2007a) adjusted the (extended) teaching dimension to the PAC setting in which the learner is required to provide an approximately correct hypothesis rather than the exact errorless one. He defined a “binary search style” active-learning algorithm and provided a corresponding sample complexity analysis using his PAC-extension teaching dimension term. Note, that in this sense, there seems to be some relationship between this term and Dasgupta’s splitting index; however, as noted by Hanneke (2007a) the connection is not clear.

The (extended) teaching dimension did not gain popularity, and in general cases it is not clear how to utilize it to analyze and construct active learning algorithms. For this reason, we do not discuss it further.

**Disagreement Coefficient.** In parallel with the presentation of the teaching dimension variant, Hanneke (2007b) introduced a complexity term that became the standard active learning complexity term. Hanneke’s disagreement coefficient intuitively captures how difficult it is to estimate a hypothesis within some threshold in terms of the probabilistic volume of instance labels that are needed.

To define the disagreement coefficient we need the following definitions: Define the distance \( \text{dist}(h_1, h_2) \) between two hypotheses \( h_1, h_2 \in \mathcal{H} \) as \( \mathbb{P}_{X \sim D_X}[h_1(X) \neq h_2(X)] \); observe that \( \text{dist}(\cdot, \cdot) \) is a pseudo-metric over pairs of hypotheses. For a hypothesis \( h \in \mathcal{H} \) and a number \( r \geq 0 \), the ball \( B(h, r) \) around \( h \) of radius \( r \) is defined as \( \{ h' \in \mathcal{H} : \text{dist}(h, h') \leq r \} \). For a set \( V \subseteq \mathcal{H} \) of hypotheses, let \( \text{DIS}(V) \) denote

\[
\text{DIS}(V) = \{ x \in X : \exists h_1, h_2 \in V \text{ such that } h_1(x) \neq h_2(x) \}.
\]

**Definition 2.2 (Uniform Disagreement Coefficient)** The disagreement coefficient of \( h \) with respect to \( \mathcal{H} \) under \( D_X \) is defined as

\[
\theta_h = \sup_{r > 0} \frac{\mathbb{P}_{D_X}[\text{DIS}(B(h, r))]}{r},
\]

where \( \mathbb{P}_{D_X}[W] \) for \( W \subseteq X \) denotes the probability mass of \( W \) with respect to the distribution \( D_X \). Define the uniform disagreement coefficient \( \theta \) as \( \sup_{h \in \mathcal{H}} \theta_h \), namely

\[
\theta = \sup_{h \in \mathcal{H}} \sup_{r > 0} \frac{\mathbb{P}_{D_X}[\text{DIS}(B(h, r))]}{r}.
\]

**Remark 2.1** A slight though useful variation of the definitions of \( \theta_h \) and \( \theta \) can be obtained by replacing \( \sup_{r > 0} \) with \( \sup_{r \geq \nu} \) in (2.1) and (2.2).

The disagreement coefficient is useful in bounding the query complexity of many algorithms (see e.g., Dasgupta et al., 2007; Balcan et al., 2008; Beygelzimer et al., 2009, 2010; Koltchinskii, 2010; Hanneke, 2011; Wang, 2011; El-Yaniv and Wiener, 2012) probably due to its simplicity, and applicability beyond the realizable case (where the splitting index fails). We will therefore compare our ideas and results with those of this complexity term.
2.2.2 Alternative Settings and Guarantees

We focus here on the agnostic setting, notably, the lower bound and the current state-of-the-art upper-bound on the sample complexity.

**Agnostic Setting: Lower Bound.** Kääriäinen (2006) proves an information-theoretic limit on what we can hope to achieve with agnostic active learning. He was able to prove that for virtually any non-trivial marginal $D_X$, noise rate $\nu$, number $n$, and active-learning algorithm, there exists a distribution $D$ with marginal $D_X$ and noise rate $\nu$ such that

$$P(\text{er}_D - \nu) \geq c \sqrt{\frac{\nu^2 \log(1/\delta)}{n}}.$$ 

This result was improved by Beygelzimer et al. (2009) to a lower bound of $c \sqrt{\frac{\nu^2 d}{n}}$. Mind that in passive learning, VC convergence rates are proportional to $\sqrt{\frac{\nu d \log(1/d)}{n}}$ for a hypothesis class with VC dimension $d$. This negative result indicates that, in a worst-case sense, active learning does not provide a significant advantage over passive learning. Nevertheless, this is not much of a restriction and can be removed by making for example, some assumptions on the distribution $D$, the noise structure, or even by using a modified definition of the sample complexity (see discussion on the *non-verifiable sample complexity* below). Here, all the specific problems that we target in Chapters 4-6 are concerned with a (transductive) distribution-free setting, for which this lower bound is irrelevant.

**Agnostic Setting: State-of-the-art Guarantees.** We are interested in the best guarantees for methods that use the disagreement coefficient and VC dimension arguments only. Thus, we are concerned with methods that use minimal assumptions (better bounds require more assumptions). The current state-of-the-art sample complexity guarantee corresponds with the algorithms of Dasgupta et al. (2007) and Beygelzimer et al. (2009). Note that both of these algorithms are described under the stream-based active-learning setting. The guarantee can be summarized as follows. Assume that the (universal) disagreement coefficient $\theta$ and the VC dimension $d$ are bounded. The number of labels the algorithm of Dasgupta et al. (2007) needs in order to achieve an error $O(\nu)$ is (w.h.p.),

$$\hat{O}(\theta d \log(1/\nu)).$$

For further details, see the discussion following Theorem 2 in Dasgupta et al. (2007).

**Non-Verifiable Sample Complexity.** Balcan et al. (2008) provides an alternative definition of the sample complexity based on the subtle observation that
bifurcate into the following two cases. The first is the standard definition in which the emitted hypothesis error bound can be verified upon the given sample (called self-verifying), and the other case is when the error rate cannot be verified from the sample; the latter is called non-verifiable sample complexity. Interestingly, Balcan et al. (2008) show that under the non-verifiable sample complexity setting, active learning gains a significant advantage over passive learning in terms of sample complexity rates. The key aspect of this remarkable advantage is that this setting can explicitly encapsulate a knowledge of the minority class’ probability mass. Recall our discussion in Chapter 1 about exploration–exploitation tradeoffs in active learning. The probability of the minority class is the exact quantity that influences the exploration overhead (i.e., when it is minuscule, the algorithm can only succeed by performing a uniform exploration). The success of the non-verifiable setting can be intuitively attributed to prior knowledge of the exploration overhead.

**Structural Noise Assumptions.** One way to overcome the negative result of Kääriäinen (2006) is by considering some assumptions about the structure of the noise (e.g., Balcan et al., 2007; Castro and Nowak, 2008; Hanneke, 2009; Koltchinskii, 2010; Yang et al., 2010; Wang, 2011; Yang et al., 2011; Minsker, 2012). The most popular assumption are the noise conditions of Mammen and Tsybakov (1999). These noise conditions informally capture the density of the noise around the boundary; thus, they are also referred to as margin conditions. Assuming that the (label) noise is inverse to the distance of the instance from the decision boundary, the conditions describe how “quickly” the label mass $\mathbb{P}[Y(X) = 1]$ is being changed around the decision boundary. It intuitively parameterizes how rapidly the diameter of the hypothesis class shrinks as we eliminate sub-optimal hypotheses. This relates in a natural way to the idea behind the definition of the disagreement coefficient and therefore provides a way to describe “distribution structures” that provide fast sample rates.

2.3 Algorithmic Ideas

This section provides an overview of the main active-learning algorithmic ideas. These ideas not only yield theoretically justified methods but also motivate heuristics that gain good empirical evidence over real-world data. For an even more comprehensive review, see, for example, Settles (2012).

2.3.1 Sampling by Uncertainty

The core idea behind the sampling-by-uncertainty query strategy is the assumption that the chance of label-estimation mistake is inversely proportional to the estimator’s confidence. Assuming this is indeed true, querying the least confident
instances has two intuitive motivations. First, there is no point in “spending” queries on labels that the current model “almost surely” knows. Secondly, revealing the least confident labels may have a large impact on the model.

In the next sections we discuss four methods that apply the natural idea of sampling-by-uncertainty. The first two are the classical CAL and Query-by-Committee (QBC) methods, both of which measure the confidence with respect to a set of hypotheses. Each queried label potentially shrinks the search space (usually taken to be the version space). The last two methods maintain an intermediate estimator and query instances near the decision boundary. Each acquired label refines the estimator (and the boundary, as a result). They are effective whenever long distances from the decision boundary are proportional to the estimator’s confidence.

**CAL and Its Variants.** CAL is a classical active-learning algorithm introduced by Cohn et al. (1994). The algorithm that was originally designed for the realizable setting maintains two sets: the uncertainty region which is defined by the set of points on which two version space hypotheses disagree, and the version space. The original algorithm samples the next query from this set according to the marginal distribution $D_X$.

The basic idea of Cohn et al. (1994) began gaining popularity with Balcan et al. [2006, 2009a] who extended the idea to the agnostic setting, where the notion of the version space is no longer valid; thus, it cannot be used to reduce the region of uncertainty. Balcan et al. (2006), on the other hand, define a noisy variant of the version space and assume that they can bound the error of the hypotheses (from above and below). Accordingly, their algorithm uses these bounds to control the region of uncertainty.

Another example of a CAL-inspired algorithm is provided by Balcan et al. (2007) whose algorithm deals with linear separators (half-spaces) that are uniformly distributed in the unit high-dimensional ball. In their elegant geometrical interpretation of the region of uncertainty, the region is controlled by the margin of an intermediate solution.

Note that the algorithm of Balcan et al. (2006) motivated the definition of the disagreement coefficient (Hanneke, 2007b) and as a result was the first agnostic active-learning algorithm to gain a sample complexity analysis.

**Query-by-Committee (QBC).** The QBC algorithm of Seung et al. (1992) also considers the realizable case and tracks the version space. The original algorithm works in the streaming active-learning setting and assumes a Bayesian prior on the hypothesis class. For each streamline decision, the algorithm draws two hypotheses from the version space (according to the Bayesian prior). If the sampled hypotheses disagree on the instance classification, the algorithm queries
the label; otherwise, the algorithm ignores it. In the more general case, the committee consists of an arbitrarily (or parameterized) number of hypotheses (e.g., Dagan and Engelson, 1995; McCallum and Nigam, 1998; Melville and Mooney, 2004). The intuition behind such a notion of confidence is related to version space shrinkage rates. The intuition here is that the empirical split ratio of the committee correlates with the split ratio of the version space and accounts for how much it can be reduced.

The QBC algorithm was analyzed by Freund et al. (1997). The uniform sampling of surfaces in high dimensions, which is in the core of QBC, makes this algorithm intractable. Several follow-up studies addressed this problem (Bachrach et al., 2002; Gilad-Bachrach et al., 2003, 2005).

**Most Uncertain Point.** The motivation behind the most uncertain querying method, suggested in parallel by Campbell et al. (2000), Schohn and Cohn (2000), and Tong and Koller (2001) comes from the geometrical relationship between confidence and distance from the Margin in Support Vector Machines. The algorithmic idea is very simple: query the closest point to the margin. Indeed, the algorithm is also widely known as SIMPLE. The original algorithm is designed for the pool-based setting and works as follows: At each stage, the algorithm maintains a soft-valued function over the pool’s instances; this function is a surrogate for the confidence of the best hypothesis with respect to the current labeled set, e.g., an ERM solution with respect to the training examples; the algorithm then queries the example that has a minimal confidence value.

The motivation of Schohn and Cohn (2000) and Tong and Koller (2001) emerged from geometric considerations of the hyperplane generated by the underlying SVM. Schohn and Cohn (2000) look at the geometric interpretation of classification confidence which is, in SVM, proportional to the example’s distance from the hyperplane. Tong and Koller (2001) argue that the most uncertain point should split the version space into proportional parts, and such query will reduce the search space in constant factor.

While the former motivations are specific for the SVM algorithm, Campbell et al. (2000) provided a more general form of this query strategy. They expressed the most uncertain point querying strategy via an optimization problem. (The minimizer of that problem defines the solution of the SVM algorithm.) The form of the optimization problem is general and can be used with most baseline (passive) algorithms and not only with SVM.

**Coarse-to-Fine.** The idea here is to start with a coarse sample that “covers” the region of interest and then gradually refine the sampling toward uncertainty regions. The uncertainty regions are defined by the Bayesian decision boundary; thus, this process can be interpreted as a search for the decision boundary. Castro et al. (2005) presented a decision-tree algorithm that implements this idea in two
dimensions. Later, Castro and Nowak (2008) presented a different coarse-to-fine algorithm and matched it with a corresponding sample complexity-guarantee.

### 2.3.2 Utility Maximization (Error Minimization)

The utility maximization approach uses a utility function that defines the potential “gain” of acquiring a label for some example. The gain is usually related to error reduction. The unlabeled example with the maximal utility value is then chosen to be queried.

MacKay (1992) analyzed active learning in a Bayesian setting and suggested that the change in entropy (information gain) can be used as a utility function. Lindenbaum et al. (1999, 2004) suggested this method to evaluate a generic utility over a step of the next possible labeling values. The former features an elegant game-tree interpretation, and the utility value function is calculated by looking ahead over this tree structure. The utility function implementations defined in Lindenbaum et al. (2004), on the other hand, are all based on soft classification values of an underlying passive learner.

Other implementations of the utility maximization approach choose to query the point whose addition to the training set maximizes the expected accuracy of a corresponding underlying passive learner. Roy and McCallum (2001) suggested this idea using the Naive-Bayes classifier, and later Baram et al. (2004) suggested an implementation with an SVM classifier. Zhu et al. (2003b) implemented this idea in the transductive setting using their transductive learning algorithm (Zhu et al., 2003a) as the passive learner.

### 2.3.3 Online Weighting

**Importance Weighting.** The importance-weighting active-learning framework was recently described by Beygelzimer et al. (2009) and has been developed further through several follow-up works (e.g., Beygelzimer et al., 2010; Karampatziakis and Langford, 2011). This framework maintains a distribution over instances that expresses the usefulness of each instance. The original algorithms work in the streaming active-learning setting where they receive examples, flipping a biased coin that reflects the example’s importance. In case of success (e.g., “heads”), it requests the example’s label to be revealed. The acquired labeled sample defines an unbiased weighted empirical estimator of the true error where the weights are inversely proportional to the importance-probability. The ERM solution defines an intermediate solution (hypothesis), which is used to define the importance weights in the next learning round.

The specific smooth-estimator construction that we define in this thesis defines importance-weights. We work in the pool-based setting, and the weights are set with respect to our novel smooth-relative-regrets condition.
Perceptron-based Approaches. Many of the streaming active-learning algorithms are variations on the Perceptron algorithm. Essentially, these methods are based on augmenting Perceptron-type algorithms with a margin-based filtering rule.

The algorithm of Cesa-Bianchi et al. (2004) determines whether to query a label by flipping a biased coin. The bias of the coin is proportional to the margin of the example with respect to the current hypothesis. A standard Perceptron update is performed whenever a query of a label finds that the algorithm’s prediction is incorrect.

The algorithm of Dasgupta et al. (2005) retains the standard Perceptron logic for deciding whether to perform a hypothesis update and modifies the standard update rule. The update rule used by Dasgupta et al. (2005) applies a dynamic learning rate that is proportional to the margin of the example with respect to the current hypothesis. Note that this is exactly opposite to Cesa-Bianchi et al. (2004), who modified the standard update indication (using the margin) and kept the standard update rule.

Another interesting implementation of the online weighting approach was proposed by Herbster et al. (2005). The main concern of their paper was applying Perceptron methods over graphs; however, they also defined an active-learning method similar to the approach that queries the most uncertain point. The underlying passive learner is applied with a Perceptron variant that belongs to the “passive-aggressive” family of algorithms (Herbster, 2001; Crammer et al., 2006).

2.3.4 Other Ideas

Clustering-based Methods. A natural approach to active learning is based on a clustering assumption, namely, that similar instances share the same label. When this assumption holds, the labeling overhead can be reduced to the number of clusters. Below we describe the elegant algorithm of Har-Peled et al. (2007), which applies the clustering-based method.

Har-Peled et al. used a margin-based active learning algorithm that applies the SVM optimization solver. To start with, the algorithm’s goal is satisfied with an approximated optimal solution. Hence, instead of constructing an exact-margin maximization solution, it utilizes an approximated one (according to a notion of cluster approximation).

The algorithm replaces instance singletons with their corresponding (geometrical) core-sets, a method that was suggested in the context of passive learning by Bădoiu and Clarkson (2003) and Tsang et al. (2005). In the context of SVM, the core-set is a subset of the training set that induces an \( \epsilon \)-maximal margin solution. Recall that SVM’s solution has the large margin characteristic; thus, in

\[3\] Though the results of Tsang et al. (2005) have been criticized by Loosli and Canu (2007) the validity of the core-set approach remains intact.
this sense, the core-set induces an \( \epsilon \)-maximal solution.

Recall that one of the motivations for applying active learning is to reduce the labeling cost. Thus, in this sense, the core-sets approach is naturally suggested for trading the labeling cost with an almost “good” solution. Har-Peled et al. (2007) noted the usefulness of active learning with core-sets and provided a corresponding algorithm in the spirit of the approach that queries the most uncertain point.

**Ensemble Methods.** Ensemble methods combine the decisions of a set of learners into a single decision. In practice, any single learner is likely to fail on a few datasets. The use of a set of learners can provide a robust decision where members of the ensemble can cover up for the deficiencies of other members.

Baram et al. (2004) provided motivation for this approach by showing that the most-uncertain and utility-maximization approaches tend to fail on XOR-like problems. This deficiency relates to the classical exploration–exploitation problem, where these methods tend to exploit points near their current decision boundary rather than explore new regions. The solution of Baram et al. (2004) is based on the multi-armed bandit algorithmic framework of Auer et al. (1995) for combining an ensemble of active learners, and provides a general framework for combining a set of active-learning schemes.

Another work tackling the same problem is that of Osugi et al. (2005), who used an ensemble of only two learners: *exploring* and *exploiting*. The switch between exploration and exploitation is achieved by flipping a biased coin, where the bias reflects how much the hypothesis changed as a result of the last query. This change is measured by the distance between the new hypothesis and the one that resulted in the previous learning round.

An even simpler approach was set forth by Guo and Greiner (2007), who combined two active-learning methods using a round-robin-like scheme. Interestingly, this very simple idea achieves excellent empirical results.
Chapter 3

The Method of Smooth Relative Regret Approximations

This chapter discusses a novel approach to pool-based active learning that is based on a novel smoothness condition on empirical estimators of relative regrets. We focus entirely on the agnostic active learning within the pool-based setting and use the notations of Section 2.1. Thus, \( \nu = \inf_{h \in \mathcal{H}} \text{er}_D(h) > 0 \) (and there exists \( h^* \in \mathcal{H} \), so that \( \text{er}_D(h^*) = \nu \)).

For the sake of simplicity, we use the following assumptions. The label space is binary \( \mathcal{Y} = \{0,1\} \), and each label \( Y \) is a deterministic function of \( X \), so that if \( X \sim \mathcal{D}_X \), then \( (X,Y(X)) \) is distributed according to \( \mathcal{D} \). Extensions to multi-label spaces and soft (probabilistic) labels are straightforward. Additionally, we assume that the pool takes \( \mathcal{X} \) in its entirety. Again, this is not a real restriction for two reasons: First, it matches a fairly common assumption that the pool is “large enough,” at least in the sense that it does not restrict the learner (i.e., it “properly” represents \( \mathcal{D}_X \)); Secondly, and somewhat more importantly, in the rest of the dissertation, we discuss only the transductive setting, in which \( \mathcal{X} \) is finite and given in advance.

3.1 Smooth Relative Regret Approximations (SRRA)

The underlying idea of the SRRA method is to utilize intermediate solutions (hypotheses) as focal points for designing smooth error estimators. The merit of these estimators is presented below in Corollary 3.1. However, in this section we will not describe how to induce query mechanisms from the smoothness condition, mainly because, as we show later, there is solid indication that specific estimator definitions give the best results for specific problem domains.

Start by fixing a pivotal hypothesis, \( h \in \mathcal{H} \). This pivot serves as our focal error-point via a function \( \text{reg}_h : \mathcal{H} \mapsto \mathbb{R} \) that we call the relative regret function
with respect to \( h \), defined as

\[
\text{reg}_h(h') = \text{er}_D(h') - \text{er}_D(h).
\]

Note that, for \( h = h^* \), this is simply the usual regret, or (in-class) excess risk\(^1\) function.

**Definition 3.1 (Smooth Relative Regret Approximation)** Let \( f : \mathcal{H} \mapsto \mathbb{R} \) be any function, and \( 0 < \varepsilon < 1/5 \), and \( 0 < \mu \leq 1 \). We say that \( f \) is an \((\varepsilon, \mu)\)-smooth relative regret approximation (\((\varepsilon, \mu)\)-SRRA) with respect to \( h \) if for all \( h' \in \mathcal{H} \),

\[
|f(h') - \text{reg}_h(h')| \leq \varepsilon \cdot (\text{dist}(h, h') + \mu).
\]

If \( \mu = 0 \), we simply call \( f \) an \( \varepsilon \)-smooth relative regret approximation with respect to \( h \).

Think of the function \( f \) as an empirical version of \( \text{reg}_h \). In other words, \( f \) is completely defined by the labeled sample at hand, meaning that the (pool-based) active learner has full control over the design of \( f \). The SRRA definition intuitively relates to the explore–exploit principle of active learning (discussed in Chapter 1). The SRRA condition hints that the sample should “cover” the spectrum of disagreement distances, while being “denser” in closer circles of the pivot \( h \). Think of the former as exploring the hypothesis class and the latter as exploiting the intermediate solution \( h \). This notion will become clearer as we present specific SRRA estimators for several problem domains.

**Remark 3.1** We compare the definition of SRRA with other known smoothness conditions.

**Lipschitz Continuity.** The Lipschitz continuity condition is defined as follows: Given two metric spaces \((A, d_A), (B, d_B)\), a function \( f : A \mapsto B \) is called a Lipschitz continuity if there exists a real valued constant \( K \geq 0 \) such that, for all \( a_1, a_2 \in A \)

\[
d_B(f(a_1), f(a_2)) \leq Kd_A(a_1, a_2).
\]

It is immediately noticeable that the left-hand sides of the \((\varepsilon, \mu)\)-SRRA definition and the above smoothness condition differ and that the SRRA has an extra \( \varepsilon\mu \) term.

\(^1\)Assuming that the risk is defined with respect to the zero-one loss function.
Massart’s Condition. Perhaps the most similar to our Definition 3.1 is the smoothness condition due to Massart (2000, Page 288, Assumption A2). Massart was the first to consider analyzing learning processes directly with excess differences $h - h^*$. Massart (2000) defines an abstraction of the contribution of $h$ to this discrepancy, which he calls the contrast function $\gamma(h, X)$ (where $X \sim D_X$ is a random variable). He defines the condition

$$\text{Var} [\gamma(h', X) - \gamma(h, X)] \leq \text{dist}^2(h', h),$$

and $\text{dist}^2(h, h^*) \leq O(\text{E} [\gamma(h, X) - \gamma(h^*, X)])$ for all $h, h' \in \mathcal{H}$. The condition is matched with a (structural) regularization term and corresponding empirical estimator. The estimator is accompanied by a matching error upper bound.

We compare Massart’s condition with our SRRA condition: Assuming $f(h') = f_{h'} - f_h$ and taking $\gamma(h, X) = f_h - 1_{\{h(X) \neq Y(X)\}}$, we get

$$\text{Var} [f_h - 1_{\{h(X) \neq Y(X)\}} - f_h + 1_{\{h(X) \neq Y(X)\}}] \leq \text{dist}^2(h, h').$$

The left-hand side can be further developed into

$$\text{E} \left[ (f(h') - 1_{\{h'(X) \neq Y(X)\}} + 1_{\{h(X) \neq Y(X)\}})^2 \right] - \left( \text{E} [f(h') - 1_{\{h'(X) \neq Y(X)\}} + 1_{\{h(X) \neq Y(X)\}}] \right)^2,$$

which is exactly $\text{Var} [1_{\{h(X) \neq Y(X)\}} - 1_{\{h'(X) \neq Y(X)\}}]$ (note that expectations are taken with respect to $X$ and $f(h')$ is a constant). In other words, the condition bounds the variance of hypotheses distances.\(^2\) Additionally, note that the second part of Massart’s condition yields a very restrictive “structural” assumption on $\mathcal{H}$: $\text{dist}^2(h, h^*) \leq O(f(h) - \text{reg}_{h^*}(h))$. Our SRRA condition is concerned directly with empirical processes that estimate relative regrets; also, it does not imply any structural assumptions. We conclude that our SRRA condition is considerably different than Massart’s condition.

When used sequentially the SRRA condition turns out to be useful. The following theorem and corollary show that a sequence of $(\varepsilon, \mu)$-SRRA estimators define an $\varepsilon$-competitive hypothesis. This results in a simple iterative meta-algorithm. These results constitute the main ingredients of our SRRA method.

**Theorem 3.1** Let $h \in \mathcal{H}$ and $f$ be an $(\varepsilon, \mu)$-SRRA with respect to $h$. Let $h_1$ be a minimizer of $f(\cdot)$ in $\mathcal{H}$ ($h_1 = \text{argmin}_{h' \in \mathcal{H}} f(h')$). Then,

$$\text{er}_D(h_1) = (1 + O(\varepsilon)) \nu + O(\varepsilon \cdot \text{er}_D(h)) + O(\varepsilon \mu).$$

\(^2\)Note that the term $\left( \text{E} [f(h') - 1_{\{h'(X) \neq Y(X)\}} + 1_{\{h(X) \neq Y(X)\}}] \right)^2$ equals $(f(h') - \text{reg}_{h'}(h'))^2$, however here, we can only devise a lower bound on this term.
Proof Applying the definition of \((\varepsilon, \mu)-\text{SRRA}\) we have:

\[
\begin{align*}
\text{er}_D(h_1) &\leq \text{er}_D(h) + f(h_1) + \varepsilon \cdot \text{dist}(h, h_1) + \varepsilon \mu \\
&\leq \text{er}_D(h) + f(h^*) + \varepsilon \cdot \text{dist}(h, h_1) + \varepsilon \mu \\
&\leq \text{er}_D(h) + \nu - \text{er}_D(h) + \varepsilon \cdot \text{dist}(h, h^*) + \varepsilon \cdot \text{dist}(h, h_1) + 2\varepsilon \mu \\
&\leq \nu + \varepsilon \left(2\text{dist}(h, h^*) + \text{dist}(h, h_1)\right) + 2\varepsilon \mu.
\end{align*}
\]

(3.1)

The first inequality is from the definition of \((\varepsilon, \mu)-\text{SRRA}\); the second is from the fact that \(h_1\) minimizes \(f(\cdot)\) by construction; the third is again from the definition of \((\varepsilon, \mu)-\text{SRRA}\), and the definitions of \(h^*\) and \(\text{reg}_h\); and the fourth is from the triangle inequality. The proof is completed by plugging \(\text{dist}(h, h^*) \leq \text{er}_D(h) + \nu\), and \(\text{dist}(h_1, h^*) \leq \text{er}_D(h_1) + \nu\) into Equation 3.1, subtracting \(\varepsilon \cdot \text{er}_D(h_1)\) from both sides, and dividing by \((1 - \varepsilon)\).

A simple inductive use of Theorem 3.1 proves that the following corollary bounds the excess risk of an ERM-based active-learning algorithm (see Algorithm 1 for corresponding pseudocode). The algorithm’s query complexity depends on the specific constructions of \((\varepsilon, \mu)-\text{SRRA}\) estimators.

**Corollary 3.1** Let \(h_0, h_1, h_2, \ldots\) be a sequence of hypotheses in \(\mathcal{H}\) such that for all \(i \geq 1\), \(h_i = \arg\min_{h' \in \mathcal{H}} f_{i-1}(h')\), where \(f_{i-1}\) is an \((\varepsilon, \mu)-\text{SRRA}\) with respect to \(h_{i-1}\). Then, for all \(i \geq 0\),

\[
\text{er}_D(h_i) = (1 + O(\varepsilon)) \nu + O(\varepsilon^i)\text{er}_D(h_0) + O(\varepsilon \mu).
\]

Proof Applying Theorem 3.1 with \(h_i\) and \(h_{i-1}\), we have

\[
\text{er}_D(h_i) = (1 + O(\varepsilon)) \nu + O(\varepsilon \cdot \text{er}_D(h_{i-1})) + O(\varepsilon \mu).
\]

Solving this recursion, one gets

\[
\text{er}_D(h_i) = \sum_{j=1}^{i} \varepsilon^{j-1} (1 + O(\varepsilon)) \nu + O(\varepsilon^j) \cdot \text{er}_D(h_0) + O \left( \sum_{j=1}^{i} \varepsilon^j \right) \mu.
\]

The result follows easily by bounding geometric sums. Recall that due to the definition of \((\varepsilon, \mu)-\text{SRRA}\), we have \(\varepsilon \in (0, 1/5)\). Treating term (i),

\[
\sum_{j=1}^{i} \varepsilon^{j-1} (1 + O(\varepsilon)) \nu = \frac{1 - \varepsilon^{i-1}}{1 - \varepsilon} (1 + O(\varepsilon)) \nu = \frac{(1 + O(\varepsilon))}{1 - \varepsilon} \nu = (1 + O(\varepsilon)) \nu.
\]
We get the last equality from \( \frac{1+\varepsilon}{1-\varepsilon} = 1 + \frac{2\varepsilon}{1-\varepsilon} = 1 + O(\varepsilon) \). Similarly, for (ii) we have,

\[
O\left(\sum_{j=1}^{i} \varepsilon^{j-1}\right) \varepsilon \mu = O\left(\frac{1}{1-\varepsilon}\right) \varepsilon \mu = O(\varepsilon \mu).
\]

Corollary 3.1 is constructive and leads to the following simple iterative meta-algorithm. In each iteration the algorithm revises its current solution to the minimizer of a \((\varepsilon, \mu)\)-smooth relative regret approximation with respect to this current solution. If \((h_0, h_1, \ldots, h_T)\) is the series of intermediate solutions, then there is a corresponding series of \((\varepsilon, \mu)\)-SRRAs \((f^{(0)}, f^{(1)}, \ldots, f^{(T-1)})\). In order to apply this algorithm, we should give the algorithm an access to such a resource, as well as provide a suitable solver for minimizing \(f^{(i)}(h')\) over \(\mathcal{H}\).

**Algorithm 1 An Active-Learning Algorithm from SRRAs**

Require: an initial solution \(h_0 \in \mathcal{H}\), estimation parameters \(\varepsilon \in (0, 1/5)\), \(\mu \geq 0\), and number of iterations \(T\)

for \(i = 0, 1, \ldots, T\) do

\(h_{i+1} \leftarrow \text{argmin}_{h' \in \mathcal{H}} f^{(i)}(h')\), where \(f^{(i)}\) is an \((\varepsilon, \mu)\)-smooth relative regret approximation with respect to \(h_i\)

end for

return \(h_T\)

Below we will show problems of interest in which \((\varepsilon, \mu)\)-SRRAs with respect to a given hypothesis \(h\) can be obtained using labeling-queries at a few randomly (and adaptively) selected points \(X\) from the pool \(\mathcal{X}\), if the uniform disagreement coefficient \(\theta\) is small. This will constitute another proof of the usefulness of the disagreement coefficient in design and analysis of active-learning algorithms. In the next two chapters, we present two problems for which a direct construction of an SRRA yields a significantly better query complexity than that which is guaranteed by using the disagreement coefficient alone.

### 3.2 Constant Uniform Disagreement Coefficient Implies Efficient SRRAs

We present our first design of smooth relative regret approximation functions. The design relies on the ability to split \(\mathcal{X}\) into disjoint (disagreement) sets according to fine neighborhoods around the pivot hypothesis. The idea comes from the definition of disagreement coefficient (Definition 2.2). We show that
a bounded uniform disagreement coefficient implies existence of query-efficient 
\((\varepsilon, \mu)\)-SRRAs. Plugging these SRRAs into Algorithm 1 yields an active-learning 
algorithm with guarantees that match the state of the art (when using similar 
assumptions). This constitutes yet another proof of the usefulness of the dis-
agreement coefficient in design of active-learning algorithms.

### 3.2.1 The Construction

Assume that the uniform disagreement coefficient \(\theta\) corresponding to \(\mathcal{H}\) is finite 
and \(\nu > 0\). Consider the set of all dichotomies \(\mathcal{H}^*\) induced by \(\mathcal{H}\) over \(\mathcal{X}\)

\[
\mathcal{H}^* = \left( \bigcup_{h' \in \mathcal{H}} \{ \{ X \in \mathcal{X} : h'(X) = 0 \} \} \right) \cup \left( \bigcup_{h' \in \mathcal{H}} \{ \{ X \in \mathcal{X} : h'(X) = 1 \} \} \right).
\]

In other words, \(\mathcal{H}^*\) is the collection of all subsets \(S \subseteq \mathcal{X}\), whose elements \(X \in S\) 
are mapped to the same value (0 or 1) by \(h'\), for some \(h' \in \mathcal{H}\).

In computational geometry, the hyper-graph \((\mathcal{X}, \mathcal{H}^*)\) is called a range space, 
and \(r \in \mathcal{H}^*\) is called a range. The VC dimension of \((\mathcal{X}, \mathcal{H}^*)\) is the maximum 
cardinality of a subset \(A \subseteq \mathcal{X}\) for which \(\{ A \cap r : r \in \mathcal{H}^* \}\) contains all subsets of \(A\) (e.g., Har-Peled, 2011, Chapter 5.1, page 61).

Assume \((\mathcal{X}, \mathcal{H}^*)\) has VC dimension \(d\) and fix \(h \in \mathcal{H}\). We define a split of \(\mathcal{X}\) 
into \(\mu\) disjoint sets according to their impact on disagreement distances over \(\mathcal{H}\). 
Let \(L = \lceil \log \mu^{-1} \rceil\). Define \(\mathcal{X}_0 = \text{DIS}(\mathcal{B}(h, \mu))\) and for \(i = 1, \ldots, L\) define \(\mathcal{X}_i\) to be 

\[
\mathcal{X}_i = \text{DIS}(\mathcal{B}(h, \mu^{2^i})) \setminus \text{DIS}(\mathcal{B}(h, \mu^{2^{i-1}})) \).
\]

Observe the illustration of these sets in Figure 3.1, where the top of the 
figure corresponds to the hypothesis class and the bottom to the instance 
space. At the top, the hypotheses in \(\mathcal{H}\) are arranged in ball neighborhoods 
\(\{ \mathcal{B}(h, \mu^{2^i}) : i = 1, \ldots, L \}\) around the pivot \(h\). The bottom of the figure de-
picts the disjoint sets \(\{\mathcal{X}_i\}\). Each \(\mathcal{X}_i\) corresponds to the hypotheses “disc” 
\(\mathcal{B}(h, \mu^{2^i}) \setminus \mathcal{B}(h, \mu^{2^{i-1}})\).

Let \(\eta_i = P_{\mathcal{D}_X} [\mathcal{X}_i]\) be the measure of \(\mathcal{X}_i\) and \(\delta\) a failure probability hyper-parameter. 
For each \(i \geq 0\), draw a sample \(X_{i,1}, \ldots, X_{i,m}\) of 
\(m = O(\varepsilon^{-2}\theta (d \log \theta + \log (\delta^{-1} \log(1/\mu))))\) examples in \(\mathcal{X}_i\), each of which is 
drawn independently from the distribution \(\mathcal{D}_X|\mathcal{X}_i\) (with repetitions). (By \(\mathcal{D}_X|\mathcal{X}_i\), 
we mean the distribution \(\mathcal{D}_X\) conditioned on \(\mathcal{X}_i\).) We will now define an estimator 
function \(f : \mathcal{H} \mapsto \mathbb{R}\) of \(\text{reg}_h\) as follows: For any \(h' \in \mathcal{H}\) and \(i = 0, 1, \ldots, L\) let

\[
fi(h') = \eta_i^{-1} \sum_{j=1}^m (1_{Y(X_{i,j}) \neq h'(X_{i,j})} - 1_{Y(X_{i,j}) \neq h(X_{i,j})}) .
\]

Our estimator is now defined as \(f(h') = \sum_{i=0}^L f_i(h')\).
Figure 3.1: Illustration of the disagreement SRRA construction. The top of the figure corresponds to \( \mathcal{H} \) and the bottom to \( \mathcal{X} \). The hypotheses in \( \mathcal{H} \) are arranged in ball neighborhoods around the pivot \( h \). The instances \( \mathcal{X} \) are being split into disjoint sets that match discs \( \mathcal{B}(h, \mu 2^{i}) \setminus \mathcal{B}(h, \mu 2^{i-1}) \). We sampled a fixed number of \( m \) instances from each subset \( \mathcal{X}_i \) and defined with them an importance weighting estimator \( f \) for \( \text{reg}_h(\cdot) \). The weight of each sampled instance is (inversely) proportional to its sample probability. We showed that the resulting unbiased estimator of \( \text{reg}_h \) is a query-efficient \((\varepsilon, \mu)\)-SRRA and defines an active-learning algorithm via Algorithm 1.

**Theorem 3.2** Let \( f, h, h' \), and \( m \) be as above. With probability at least \( 1 - \delta \), \( f \) is an \((\varepsilon, \mu)\)-SRRA with respect to \( h \).

**Proof** A main tool to be exploited in the proof is called relative \( \varepsilon \)-approximations set forth by Haussler (1992) and Li et al. (2000). It is defined as follows: Let \( h \in \mathcal{X} \mapsto \mathbb{R}^+ \) be some function, and let \( \mu_h = \mathbb{E}_{X \sim \mathcal{D}_X}[h(X)] \). Let \( X_1, \ldots, X_m \) denote i.i.d. draws from \( \mathcal{D}_X \), and let \( \hat{\mu}_h = \frac{1}{m} \cdot \sum_{i=1}^{m} h(X_i) \) denote the empirical average. Let \( \kappa > 0 \) be an adjustable parameter. We will use the following measure of distance between \( \mu_h \) and its estimator \( \hat{\mu}_h \) to determine how far the latter diverges from the true expectations:

\[
d_{\kappa}(\mu_h, \hat{\mu}_h) = \frac{|\mu_h - \hat{\mu}_h|}{\mu_h + \hat{\mu}_h + \kappa}.
\]
This measure corresponds to a relative error when approximating \( \mu_h \) by \( \hat{\mu}_h \). Indeed, let \( \varepsilon > 0 \) be our approximation ratio and put \( d_{\kappa}(\mu_h, \hat{\mu}_h) < \varepsilon \). This easily yields

\[
|\mu_h - \hat{\mu}_h| < \frac{2\varepsilon}{1 - \varepsilon} \cdot \mu_h + \frac{\varepsilon}{1 - \varepsilon} \cdot \kappa.
\] (3.2)

In other words, it is implied that \( |\mu_h - \hat{\mu}_h| < O(\varepsilon)(\mu_h + \kappa) \).

Let us fix a parameter \( 0 < \delta < 1 \). Assume that \( \mathcal{H} \) is a set of \( \{0,1\} \) valued functions on \( \mathcal{X} \) of VC dimension \( d \). Li et al. (2000) show that if one samples

\[ m = c(\varepsilon^{-2}\kappa^{-1}(d \log \kappa^{-1} + \log \delta^{-1})) \]

examples as above, then (3.2) holds uniformly for all \( h \in \mathcal{H} \) with probability of at least \( 1 - \delta \).

We now apply this definition of relative \( \varepsilon \)-approximations, and the corresponding results within our context. For any \( h' \), we define the following four sets of instances:

\[
\begin{align*}
R_{h'}^{++} &= \{ X \in \mathcal{X} : h'(X) = Y(X) = 1, \text{ and } h(X) = 0 \} \\
R_{h'}^{+-} &= \{ X \in \mathcal{X} : h'(X) = 1, \text{ and } h(X) = Y(X) = 0 \} \\
R_{h'}^{-+} &= \{ X \in \mathcal{X} : h'(X) = 0, \text{ and } h(X) = Y(X) = 1 \} \\
R_{h'}^{--} &= \{ X \in \mathcal{X} : h'(X) = Y(X) = 0, \text{ and } h(X) = 1 \}.
\end{align*}
\]

Observe that the set \( \{ X \in \mathcal{X} : h(X) \neq h'(X) \} \) is equal to the union of \( R_{h'}^{++} \), \( R_{h'}^{+-} \), \( R_{h'}^{-+} \), and \( R_{h'}^{--} \). For each \( i = 0, \ldots, L \) and \( b \in \{++,+-,-+,--\} \), let \( R_{h,i}^{b} = R_{h}^{b} \cap \mathcal{X}_i \). Let \( \mathcal{R}_{i}^{b} = \{ R_{h,i}^{b} : h' \in \mathcal{H} \} \). It is easy to verify that the VC dimension of the range spaces \( (\mathcal{X}_i, \mathcal{R}_i^{b}) \) is at most \( d \). Each set in \( \mathcal{R}_i^{b} \) is an intersection of a set in \( \mathcal{H}^* \) with some fixed set.

For any \( R \subseteq \mathcal{X}_i \), let \( \rho_i(R) = P_{X \sim D_X|X_i} [X \in R] \) and \( \hat{\rho}_i(R) = m^{-1} \sum_{j=1}^{m} \mathbf{1}_{X_{i,j} \in R} \). Note that \( \hat{\rho}_i(R) \) is an unbiased estimator of \( \rho_i(R) \).

From Equation (3.2), the assumptions on \( \theta \) and \( \nu \), and the choice of \( m \), we have: with a probability of at least \( 1 - \delta/L \) for all \( R \subseteq \mathcal{R}_i^{++} \cup \mathcal{R}_i^{+-} \cup \mathcal{R}_i^{-+} \cup \mathcal{R}_i^{--} \),

\[
|\rho_i(R) - \hat{\rho}_i(R)| = O(\varepsilon) \cdot (\rho_i(R) + \theta^{-1});
\] (3.3)

and by the probability union bound we obtain that this uniformly holds for all \( i = 0, \ldots, L \) with a probability of at least \( 1 - \delta \).

Now fix \( h' \in \mathcal{H} \) and let \( r = \text{dist}(h, h') \) and let \( i_r = \lceil \log(r/\mu) \rceil \). By the definition of \( \mathcal{X}_i \), \( h(X) = h'(X) \) for all \( X \in \mathcal{X}_i \) whenever \( i > i_r \). We can therefore decompose \( \text{reg}_h(h') \) as:
\[
\text{reg}_h(h') = \text{er}_D(h') - \text{er}_D(h) = \sum_{i=0}^{L} \eta_i \cdot (P_{X \sim \mathcal{D}_X|X_i}[Y(X) \neq h'(X)] - P_{X \sim \mathcal{D}_X|X_i}[Y(X) \neq h(X)])
\]

On the other hand, we similarly have that
\[
f(h') = \sum_{i=0}^{i_r} \eta_i \cdot \left( -\hat{\rho}_i(R_{h'}^{++}) + \hat{\rho}_i(R_{h'}^{+-}) + \hat{\rho}_i(R_{h'}^{-+}) - \hat{\rho}_i(R_{h'}^{--}) \right).
\]

Combining these, we conclude by using (3.3) that
\[
|\text{reg}_h(h') - f(h')| \leq O\left( \varepsilon \sum_{i=0}^{i_r} \eta_i \cdot \left( \rho_i(R_{h'}^{++}) + \rho_i(R_{h'}^{+-}) + \rho_i(R_{h'}^{-+}) + \rho_i(R_{h'}^{--}) + 4\theta^{-1} \right) \right) \tag{3.4}
\]

But now notice that \(\sum_{i=0}^{i_r} \eta_i \cdot \left( \rho_i(R_{h'}^{++}) + \rho_i(R_{h'}^{+-}) + \rho_i(R_{h'}^{-+}) + \rho_i(R_{h'}^{--}) \right)\) equals \(r\), since it corresponds to those elements \(X \in \mathcal{X}\) on which \(h, h'\) disagree. Also note that \(\sum_{i=0}^{i_r} \eta_i\) is at most \(2 \max \{P_{\mathcal{D}_X}[\text{DIS}(\mathcal{B}(h, r))], P_{\mathcal{D}_X}[\text{DIS}(\mathcal{B}(h, \mu))]\}\). By the definition of \(\theta\), this implies that the RHS of (3.4) is bounded by \(\varepsilon (r + \mu)\), as required by the definition of \((\varepsilon, \mu)\)-SRRA. \(^3\)

**Corollary 3.2** An \((\varepsilon, \mu)\)-SRRA with respect to \(h\) can be constructed with a probability of at least \(1 - \delta\), using at most
\[
m \left(1 + \lceil \log(1/\mu)\rceil\right) = O\left( \theta \varepsilon^{-2} \left(\log(1/\mu)\right) \left(d \log \theta + \log(\delta^{-1} \log(1/\mu))\right) \right) \tag{3.5}
\]
\(\) label queries.

Combining Corollaries 3.1 and 3.2 (Algorithm 1), we obtain an active-learning algorithm in the ERM-setting with query complexity depending on the uniform disagreement coefficient and the VC dimension. Assume \(\delta\) is a constant; if we are interested in excess risk on the order of at least that of the optimal error \(\nu\),

\(^3\)The \(O\)-notation disappeared because we assume that the constants are properly chosen in the definition of the sample size \(m\).
then we may take $\varepsilon$ to be, say $1/5$, and achieve the sought bound by constructing $(1/5, \nu)$-SRRAs using $O(\theta d(\log(1/\nu))(\log \theta))$ once for each of $O(\log(1/\nu))$ iterations of Algorithm 1. If we seek a solution with error $(1 + \varepsilon)\nu$, we would need to construct $(\varepsilon, \nu)$-SRRAs using $O(\theta d^{-2}(\log(1/\nu))(\log \theta))$ query labels, one for each of $O(\log(1/\nu))$ iterations of the algorithm. The total label query complexity is $O(\theta d(\log^{2}(1/\nu))(\log \theta))$, which is $O(\log(1/\nu))$ times the best known bounds using disagreement coefficient and VC dimension bounds only (e.g., Dasgupta et al., 2007; Beygelzimer et al., 2009).

A few more notes of comparison are in place. First, note that in known arguments that bound query complexity by using the disagreement coefficient, the disagreement coefficient $\theta_{h^*}$ with respect to the optimal hypothesis $h^*$ is used in the analysis, and not the uniform coefficient $\theta$. Also note that both in previously known results that bound the query complexity by using the disagreement coefficient and VC dimension bounds and in our result as well, the slight improvement described in Remark 2.1 applies. In other words, all arguments remain valid if we replace the supremums in (2.1) and (2.2) with $\sup_{r \geq \nu}$.

### 3.3 Convex Relaxations

So far we have focused on theoretical ERM aspects only. By so doing, we have made no assumptions about the computability of the step $h_i = \arg \min_{h' \in H} f_{h_{i-1}}(h')$ in Corollary 3.1 (Step 2 in Algorithm 1). Although ERM results are interesting in their own right, we take an additional step and consider convex relaxations.

Instead of optimizing $e_{\mathcal{D}}(h)$ over the set $\mathcal{H}$, assume that we are interested in optimizing $e_{\mathcal{D}}(\tilde{h})$ over $\tilde{h} \in \tilde{\mathcal{H}}$, where $\tilde{\mathcal{H}}$ is a convex set of functions from $\mathcal{X}$ to $\mathbb{R}$. Also assume there is a mapping $\phi : \tilde{\mathcal{H}} \mapsto \mathcal{H}$ that is used as a “rounding” procedure. When optimizing in $\tilde{\mathcal{H}}$, one conveniently works with a convex relaxation $\tilde{e}_{\mathcal{D}} : \tilde{\mathcal{H}} \to \mathbb{R}^+$ as surrogate for the discrete loss $e_{\mathcal{D}}$, defined as follows

$$
\tilde{e}_{\mathcal{D}}(\tilde{h}) = E_{(X,Y) \sim \mathcal{D}} \left[ \tilde{L} \left( \tilde{h}(X), Y \right) \right],
$$

where $\tilde{L} : \mathbb{R} \times \{0,1\} \mapsto \mathbb{R}^+$ is some function convex in the first argument, and satisfying

$$
1_{(\phi(\tilde{h}))(X) \neq Y} \leq c \tilde{L} \left( \tilde{h}(X), Y \right)
$$

for all $\tilde{h} \in \tilde{\mathcal{H}}$ and $X \in \mathcal{X}$, where $c > 0$ is some constant. In other words, $\tilde{L}$ upper bounds the discrete loss (up to a factor of $c$).

**Example 3.1** For example, consider the well known setting of SVMRank with the hinge loss relaxation (Herbrich et al., 2000; Joachims, 2002). Here, $\mathcal{X} \subseteq \mathbb{R}^d \times \mathbb{R}^d$ and we would like to learn an order over $\mathcal{X}$. We take $\mathcal{H}$ as the set of...
all vectors \( w \in \mathbb{R}^d \), and the rounding method \( \phi : \tilde{H} \mapsto H \) to convert \( w \) to an order over \( X \). For all \( w \in \tilde{H} \) and \( x = (u, v) \in X \), \( w(x) = \langle w, u - v \rangle \), and \( \tilde{L}(a, b) = \max\{1 - a(2b - 1), 0\} \). Using this choice, optimizing over (3.6) becomes the famous SVMRank optimization:

\[
\begin{align*}
\text{Minimize} & \quad F(w, \xi) = \sum_{u,v} \xi_{u,v} \\
\text{s.t., } & \forall u, v, Y(u, v) = 1 : (u - v) \cdot w \geq 1 - \xi_{u,v} \\
& \forall u, v : \xi_{u,v} \geq 0 \\
& \|w\| \leq c.
\end{align*}
\]

(Note: \( c \) is a regularization parameter.)

We now have a natural extension of relative regret:

\[
\tilde{\text{reg}}_{\tilde{h}}(\tilde{h}') = \tilde{\text{er}}_D(\tilde{h}') - \tilde{\text{er}}_D(\tilde{h}).
\]

By our assumptions on convexity, \( \tilde{\text{reg}}_{\tilde{h}} : \tilde{H} \mapsto \mathbb{R}^+ \) can be efficiently optimized.

We now say that \( f : \tilde{H} \mapsto \mathbb{R}^+ \) is an \((\varepsilon, \mu)\)-SRRA with respect to \( \tilde{h} \in \tilde{H} \) if for all \( \tilde{h}' \in \tilde{H} \),

\[
\left| \tilde{\text{reg}}_{\tilde{h}}(\tilde{h}') - f(\tilde{h}') \right| \leq \varepsilon \left( \text{dist}(\phi(\tilde{h}), \phi(\tilde{h}')) + \mu \right).
\]

If \( \mu = 0 \) then we simply say that \( f \) is an \( \varepsilon \)-SRRA. The following is an analogue to Corollary 3.1:

**Theorem 3.3** Let \( \tilde{h}_0, \tilde{h}_1, \tilde{h}_2, \ldots \) be a sequence of hypotheses in \( \tilde{H} \) such that for all \( i \geq 1 \), \( \tilde{h}_i = \arg\min_{\tilde{h}' \in \tilde{H}} f_{i-1}(\tilde{h}') \), where \( f_{i-1} \) is an \((\varepsilon, \mu)\)-SRRA with respect to \( \tilde{h}_{i-1} \). Then for all \( i \geq 1 \),

\[
\tilde{\text{er}}_D(h_i) = (1 + O(\varepsilon)) \tilde{\nu} + O(\varepsilon) \tilde{\text{er}}_D(h_0) + O(\varepsilon \mu),
\]

where \( \tilde{\nu} = \inf_{\tilde{h} \in \tilde{C}} \tilde{\text{er}}_D(\tilde{h}) \) and the \( O \)-notations may hide constants that depend on \( c \).

The proof is very similar to that of Corollary 3.1, and we omit the details.

### 3.4 Discussion

We presented here the key ingredients of our method of smooth relative regret approximations. The core of our method is the smoothness condition of Definition 3.1. The effectiveness, in terms of query complexity, of such smooth regret-estimators is evident in Corollary 3.1 and Algorithm 1. This is a meta-algorithm that requires two components to be instantiated: an SRRA implementation, and a ERM-like solver. We discussed implementations for these two aspects in Sections 3.2.1 and 3.3.

We think of the SRRA smoothness condition as a holistic explore–exploit condition for active learning. Furthermore, it seems the need for explore–exploit

33
tradeoffs exists when the corresponding discrepancy between the estimated error and the truth fluctuates. Our SRRA condition assures that this discrepancy is smooth with respect to distances from some pivotal intermediate solution. In many cases we can correlate such hypothesis-distances with corresponding instance subsets. The condition intuitively guides us to spread queries all over the query space (exploration) with a density inversely proportional to distances from some fixed intermediate solution. This intuition is well-demonstrated in the way that we construct the disagreement-based SRRA presented in Section 3.2.1, where we spread the queries in such a way that it incorporates exploitation near current solutions with exploration that enables identification of better “far” candidates.

Our method is concise and simple, yet rigorous, and provides guarantees that meet the state of the art; nevertheless we have not yet shown that it extends the state of the art. Moreover, the provided SRRA construction is infeasible because it requires calculations of disagreement regions with respect to any hypothesis. Thus, the construction presented in this chapter points to the potential of our method. We will prove the advantages of our method in the next two chapters, where we present specific feasible SRRA constructions that provide the best-known guarantees.
Chapter 4

Active Preference-based Ranking Using SRRAs

“Learning to rank” takes various forms in the theory and practice of learning and combinatorial optimization; in all its forms, the goal is to order a set $V$ of $n$ elements based on constraints. Preference-based constraints arise when the labels come from natural-source judgments, such as human ratings or rankings.

Perhaps the most studied form of preference-based ranking considers absolute-value constraints. In this setting, each element in $V$ is matched with a discrete numeric scale that defines ordinal type preferences; higher-value elements are preferred over lower elements. The goal is to learn how to order $V$ so as to respect the induced pairwise preferences. For example, review systems, such as, reviews of hotels, books, and restaurants use the “star quality” scale, \{1, 2, 3, 4, 5\}, where, if $u$ has a label of 5 (“very good”) and $v$ has a label of 1 (“very bad”), then any ordering that places $v$ ahead of $u$ is penalized. Note that even if the labels are noisy, the induced pairwise preferences here are always transitive, hence no combinatorial problem arises. Asking humans to judge “complex” phenomena according to a numeric scale is extremely problematic. Indeed, such judgments are plagued with calibration issues, errors, and inconsistencies (see e.g., Stewart et al., 2005); this is intuitively analogous to embedding a high dimensional space on the line.

Our work deals with a completely different setting, one in which the basic unit of information consists of preferences over pairs $u, v \in V$. Ranking items according to their comparison in pairs dates back to the classical work of Thurstone (1927). Such preferences are appealing firstly because they can encapsulate “complex” judgments, at least in the sense that labels are direct signals rather than surrogate signals, as with ordinal-scale labels. Secondly, relative judgments can be non-transitive. Thus, here the problem becomes combinatorially interesting.

We focus on this combinatorial aspect of the problem studying learning to rank from pairwise preferences (LRPP), a close relative of minimum feedback in
arc-set tournaments (MFAST)\(^1\) from the world of combinatorial optimization. In MFAST, the goal is to find a full linear order of \(V\), given all \(\binom{n}{2}\) pairwise comparisons of elements (i.e., all possible labels) for free. It turns out that MFAST is NP-hard (Alon, 2006), though Kenyon-Mathieu and Schudy (2007) show a PTAS for it. Namely, they show a degree-\(n\) polynomial time algorithm computing a solution with loss at most \((1 + \varepsilon)\) times the optimal (the degree of the polynomial may depend on \(\varepsilon\)). Several important recent works address the challenge of approximating the minimum feedback arc-set problem (Ailon et al., 2008; Braverman and Mossel, 2008; Coppersmith et al., 2010).

In terms of practicality, one of the obstacles of LRPP is in its apparent quadratic sample-complexity. From a learning theoretical point of view, the need to acquire quadratic amount of pairwise preferences is unacceptable even for moderately large sets \(V\) that arise in applications. On the other hand, uniform sub-sampling of pairs works poorly (see e.g., Section 4.2). Thus, devising a good active-learning approach is crucial to the problem’s practicability.

Here we consider a query-efficient variant of LRPP in which each preference comes with a cost, the goal being to produce a competitive solution while reducing the preference-query overhead. Other very recent works consider similar settings (Jamieson and Nowak, 2011; Ailon, 2012). Jamieson and Nowak (2011) consider a common scenario in which the alternatives can be characterized in terms of \(d\) real-valued features and the ranking obeys the structure of the Euclidean distances between such embeddings. They present an active-learning algorithm that requires, using average case analysis, as few as \(O(d \log n)\) labels in the noiseless case, and \(O(d \log^2 n)\) labels under a certain parametric noise model. Our work uses worst-case analysis and assumes an adversarial noise model. In Section 4.4 we analyze the pure combinatorial problem (not assuming any feature embeddings). In Section 4.5 we tackle the problem with linearly induced permutations over feature-space embeddings.

Ailon (2012) considers the same setting as ours. Our main result for query complexity stated in Corollary 4.1 is a slight improvement on Ailon’s and it provides another significant improvement. Ailon (2012) uses a querying method that is based on a divide-and-conquer strategy. The weakness of such a strategy can be demonstrated by considering an example in which we want to search over a restricted set of permutations (e.g., the setting of Section 4.5): While dividing and conquering, Ailon’s algorithm is doomed to search a Cartesian product of two permutation spaces (left and right), and there is no guarantee that there even exists a permutation in the restricted space that respects this division. In our querying algorithm, this limitation is lifted.

---

\(^1\)A maximization version exists as well.
4.1 Problem Definition

Let $V$ be a set of $n$ elements (alternatives). The instance space $\mathcal{X}$ is taken to be the set of all distinct pairs of elements in $V$, namely $V \times V \setminus \{(u,u) : u \in V\}$. The distribution $\mathcal{D}_X$ is uniform on $\mathcal{X}$. The label function $Y : \mathcal{X} \rightarrow \{0,1\}$ encodes a preference function satisfying $Y((u,v)) = 1 - Y((v,u))$ for all $u, v \in V$. We conventionally think of $Y((u,v)) = 1$ as a stipulation that $u$ is preferred over $v$. For convenience, we will drop the double-parentheses in what follows.

Remark 4.1 We make the above “anti-symmetric” assumption on $Y$ only for the convenience of exposition. We could provide an alternative definition without making any assumption on $Y$, where we index the elements of $V$ arbitrarily and take $X$ to be $\binom{V}{2}$, the set of unordered pairs of elements in $V$. Then, for the pair $\{v_i, v_j\}$ with $i < j$, the value $Y(\{v_i, v_j\}) = 1$ stipulates that $v_i$ is preferred over $v_j$, and it is zero otherwise.

The class of solution functions $\mathcal{H}$ we consider is all $h : \mathcal{X} \rightarrow \{0,1\}$ such that it is skew-symmetric, $h(u,v) = 1 - h(v,u)$, and transitive, $h(u,z) \leq h(u,v) + h(v,z)$, for all distinct $u, v, z \in V$. This is equivalent to the space of permutations over $V$. Hereafter in this chapter, we will replace $h, h', \ldots$ with $\pi, \sigma, \ldots$; we also use notation $u \prec_\pi v$ as a predicate equivalent to $\pi(u,v) = 1$; and with a slight abuse of notation we will let $\pi(u)$ denote the rank of $u$ in the permutation induced by $\pi$.

Endowing $\mathcal{X}$ with the uniform measure and denoting the number of all ordered pairs by $N = n(n-1)$, we have

$$\text{er}_D(\pi) = N^{-1} \sum_{u,v \in \mathcal{X}} \mathbf{1}_{\pi(u,v) \neq Y(u,v)}. \quad (4.1)$$

The distance $\text{dist}(\pi, \sigma)$ turns out to be (up to normalization) the well-known Kendall-$\tau$ distance:

$$\text{dist}(\pi, \sigma) = N^{-1} \sum_{u \neq v} \mathbf{1}_{\pi(u,v) \neq \sigma(u,v)}. \quad (4.2)$$

Another well-known distance over permutation that we will consider is Spearman’s Footrule distance:

$$d_{\text{SF}}(\pi, \sigma) = \sum_{u \in V} |\pi(u) - \sigma(u)|, \quad (4.3)$$

which is easier to manipulate than the Kendall-$\tau$ distance. A classical result by Diaconis and Graham (1977) connects these two distances and makes it possible to interchange one with the other

$$N \text{dist}(\pi, \sigma) \leq d_{\text{SF}}(\pi, \sigma) \leq 2N \text{dist}(\pi, \sigma). \quad (4.4)$$
4.2 The Necessity of Active Learning

A natural question in this context is why the optimal hypothesis cannot be approximated by sampling preference-pairs uniformly at random. In other words, is it sufficient for our setting to apply a passive learning method? Do we really need to apply the more sophisticated active-learning machinery?

Let us start to tackle this concern from the perspective of VC learning theory (see e.g., Vapnik, 1995). VC theory tells us that if the VC dimension of $\mathcal{H}$ is $d$ and we sample $m > n$ pairs uniformly at random, denoted by $S_m$, then with a probability of at least $1 - \delta$, the (unbiased) empirical estimation, $m^{-1} \sum_{(u,v) \in S_m} 1_{\pi(u,v) \neq Y(u,v)}$ deviates from its expectation $\text{err}_D(\pi, Y)$ by no more than $O\left(\sqrt{\frac{d \log m + \log(1/\delta)}{m}}\right)$.

However, it is known that the VC dimension of $\mathcal{H}$ is $n - 1$ (e.g., Radinsky and Ailon, 2011), which is simply because for any set of $n$ pairs there always exists a labeling $Y(\cdot)$ that defines preference cycles. In other words, the set of permutations $\mathcal{H}$ cannot shatter $n$ pairs. On the other hand, the maximal set of pairs $(u, v) \in \mathcal{X}$ such that each alternative is incident exactly in a single pair is shattered in $\mathcal{H}$.

The VC bound becomes $O\left(\sqrt{n \log m + \log(1/\delta)}\right)$. If we want to achieve an additive error of $\varepsilon$ with a probability of at least $1 - \exp(-n)$, then we will have to sample $O(\varepsilon^{-2} n \log n)$ pairs uniformly at random with repetitions and optimize the empirical estimation over $\mathcal{H}$.

Recall that Spearman’s footrule and Kendall-$\tau$ Distances are of the same order (Equation 4.4). An additive error $\varepsilon$ means that each alternative $u$ moves on average $\varepsilon n$ indices away from its (optimal) rank in $\pi^*$. In practice, we usually like to achieve a constant average misplacement. Thus, we require an additive error of $\varepsilon/n$ that gives rise to a quadratic sample complexity, which basically means that we have to query the whole instance space $\mathcal{X}$.

Let us now consider a different argument that motivates the use of active learning. Assume that the problem is realizable, that is, the noise rate of the hypothesis class is zero. The only way we can achieve a zero-error permutation using empirical risk minimization (ERM) strategy is when the sample contains all $n - 1$ consecutive pairs of the optimal permutation $\pi^*$. A standard application of concentration bounds around the mean tells us that the probability of such an event is exponentially small when sampling pairs uniformly at random (with or even without replacements). For example, take $\{X_i\}_{i=1}^m$ to be i.i.d. random variables indicating whether a sampled pair is a consecutive in $\pi^*$ or not. Clearly, each $X_i$ is a Bernulli random variable with mean $p = 1/N$. Applying Chernoff bound, we get that $\mathbf{P}\left[\sum_{i} X_i > (1 + t) m N^{-1}\right] \leq \exp\{-ct^2\}$ (for some global constant $c$). Observe that when $m = o(N)$, we must take $t = \Omega(n)$, and the probability that we actually hit the set of consecutive pairs is exponentially
low. We note that Ailon (2012) used arguments similar to ours.

4.3 Disagreement Coefficient Arguments Are Not Sufficient for Effective Active Learning

The former section motivated a preference for active- over passive-learning for LRPP. In Chapter 3 we presented a disagreement-based SRRA construction that assumes finite disagreement coefficient and VC dimension only and guarantees a sample complexity that matches the state of the art (when using these assumptions only). We will show below that one cannot achieve a useful active-learning algorithm by using disagreement coefficient arguments only.

The uniform disagreement coefficient of $\mathcal{H}$ is by Definition 2.2

$$\theta = \sup_{\pi \in \mathcal{H}} \sup_{r > 0} \frac{N^{-1} |\text{DIS}(B(\pi, r))|}{r}.$$  

It is easy to show that $\theta$ is $\Omega(n)$ (as it has been shown by Ailon, 2012). Notice that if we start from some permutation $\pi$ and swap the positions of any two elements $u, v \in V$, then we obtain a permutation of distance that is at most $O(1/n)$ away from $\pi$, as depicted in Figure 4.1. Hence, the disagreement region of the ball of radius $O(1/n)$ around $\pi$ is the entire space $\mathcal{X}$. Plugging in the corresponding values, we get $\theta \geq N^{-1}N = \Omega(n)$.

Recall that the VC dimension of $\mathcal{H}$ is $n - 1$. Using Corollary 3.2, we conclude that we would need $\Omega(n^2)$ preference labels to obtain an $(\epsilon, \mu)$-SRRA for any meaningful pair $(\epsilon, \mu)$. This is uninformative because the cardinality of $\mathcal{X}$ is $\Theta(n^2)$. A similar bound is obtained with any known active-learning bound using disagreement coefficient and VC dimension bounds only.

A slight improvement on this negative result can be obtained using the refined definition of disagreement coefficients of Remark 2.1: namely, by replacing $\sup_{r > 0}$ with $\sup_{r \geq \nu}$ in the definition of $\theta$. Observe that the uniform disagreement coefficient, as well as the disagreement coefficient at the optimal solution $\pi^*$ becomes

$$\theta_{\pi^*} = \frac{N^{-1} |\text{DIS}(B(\pi^*, r))|}{r}.$$  

Due to symmetry, the uniform disagreement coefficient here equals $\theta_{\pi}$ for any $\pi \in \mathcal{H}$.
\[ \theta = \theta_\pi = O(1/\nu), \text{ if } \nu \geq \frac{1}{n}, \text{ improving the query complexity bound to } O(n\nu^{-1}). \]

If \( \nu \) tends to \( n^{-1} \) from above, this becomes a quadratic (in \( n \)) query complexity in the limit.

### 4.4 Better SRRA for LRPP

The former section demonstrated that the general disagreement-based SRRA construction of Chapter 3 provides uninformative guarantees when applied to LRPP (when \( \nu \) is “low”). Thus, it is necessary to seek a better SRRA construction-scheme.

Before starting, let us remove the dependency on \( \mu \) in the definition of \((\varepsilon, \mu)\)-SRRA to indicate that the flavor of this setting is different from that of the general one. Following from our discussion of \( \nu \) in the former section, taking \( \mu = 1/n \) should be sufficient; thus, we will simply use the term \( \varepsilon \)-SRRA here.

Consider the following idea for creating an \( \varepsilon \)-SRRA for LRPP with respect to some fixed \( \pi \in \mathcal{H} \). Each \( u \in V \) will define a disjoint partition of \( V \), such that each subset consists of elements that induce a similar order of “point-wise” error contribution when inverting their \( \pi \)-induced relative order with \( u \). Following the guidance of the \( \varepsilon \)-SRRA definition, we will sample from the partition subsets to ensure that our sampling becomes denser as this potential point-wise error becomes low, as depicted in Figure 4.2. Below we describe the construction in detail.

Let \( p \) be a sampling parameter as defined in (4.5). For all \( u \in V \) and for all \( i = 0, 1, \ldots, \lceil \log n \rceil \), let \( I_{u,i} \) denote the set of elements \( v \) such that \((2^i - 1)p < |\pi(u) - \pi(v)| < 2^{i+1}p \) (recall that \( \pi(u) \) is the position of \( u \) in \( \pi \)). From this set, choose a random sequence of \( p \) elements \( R_{u,i} = (v_{u,i,1}, v_{u,i,2}, \ldots, v_{u,i,p}) \), each chosen uniformly and independently from \( I_{u,i} \). We define the sample size parameter to be

\[ p = O \left( \varepsilon^{-3} \log^3 n \right). \]  

This choice is derived from the machinery we will use to prove sample size guarantees.

**Remark 4.2** A variant of this sampling scheme is as follows: for each pair \((u, v)\), add it to the query-set with probability proportional to \( \min\{1, p/|\pi(u) - \pi(v)|\} \). A similar scheme can be found in Ailon et al. (2007), Halevy and Kushilevitz (2007), and Ailon (2012), but the strong properties proven here were not known.

For distinct \( u, v \in V \) and a permutation \( \sigma \in \mathcal{H} \), let \( \text{cost}_{u,v}(\sigma) \) denote the contribution of the pair \( u, v \) to \( \text{er}_D(\sigma) \) namely, \( \text{cost}_{u,v}(\sigma) = N^{-1}1_{\sigma(u,v) \neq Y(u,v)} \). Let \( \text{reg}_{u,v|\sigma} \) denote the contribution of \( \{u, v\} \in \mathcal{X} \) to \( \text{reg}_\pi(\sigma) \), that is,

\[ \text{reg}_{u,v|\sigma} = 2 (\text{cost}_{u,v}(\sigma) - \text{cost}_{u,v}(\pi)) \]  

(4.6)
Figure 4.2: Depicting the core element of our SRRA for the LRPP construction. Element $u$ defines a partition $\{I_{u,i}\}$ over $V$. Each partition subset is depicted as a dark “disc.” Observe that all the alternatives in a disc share a “similar” magnitude of point-wise (footrule) error if an alternative switches places with $u$. From each disc we sample i.i.d. $p$ elements. We do so for every $u \in V$ and define a corresponding unbiased estimator for $\text{reg}_\pi$.

Notice that the notation discards the dependency on $\pi$ because it is assumed to be fixed. The factor 2 is used because $\text{cost}_{u,v} \equiv \text{cost}_{v,u}$.

Our estimator $f(\sigma)$ of $\text{reg}_\pi(\sigma) = \text{er}_D(\sigma) - \text{er}_D(\pi)$ is defined as

$$f(\sigma) = \frac{1}{2} \sum_{u \in V} \sum_{i=0}^{\lfloor \log n \rfloor} \frac{|I_{u,i}|}{p} \sum_{t=1}^{p} \text{reg}_{u,v_{u,i,t}|\sigma}.$$  \hspace{1cm} (4.7)

Clearly, $f(\sigma)$ is an unbiased estimator of $\text{reg}_\pi(\sigma)$ for any $\sigma$; our goal is to prove that $f(\sigma)$ is an $\varepsilon$-SRRA.

**Theorem 4.1** With a probability of at least $1 - n^{-3}$, the function $f$ is an $\varepsilon$-SRRA with respect to $\pi$.

**Proof** The main idea is to decompose the difference $|f(\sigma) - \text{reg}_\pi(\sigma)|$ vis-à-vis
corresponding pieces of \( \text{dist}(\sigma, \pi) \). The first half of the proof is devoted to the definition of such distance “pieces.” Then, using counting and standard deviation–bound arguments, we show that the decomposition is, with high probability, an \( \varepsilon \)-SRRA.

Let us start with a few definitions. Recall that for any \( \pi \in \mathcal{H} \) and \( u \in V \), \( \pi(u) \) denotes the position of \( u \) in the unique permutation that \( \pi \) defines. For example, \( \pi(u) = 1 \) if \( u \) beats all other alternatives: \( \pi(u, v) = 1 \) for all \( v \neq u \). Similarly, \( \pi(u) = n \) if \( u \) is beaten by all other alternatives. For any permutation \( \sigma \in \mathcal{H} \), we define the corresponding profile of \( \sigma \) as the vector:

\[
\text{prof}(\sigma) = (\sigma(u_1) - \pi(u_1), \sigma(u_2) - \pi(u_2), \ldots, \sigma(u_n) - \pi(u_n)).
\]

Note that \( \|\text{prof}(\sigma)\|_1 \) is \( d_{SF}(\sigma, \pi) \), the Spearman footrule distance between \( \sigma \) and \( \pi \).

For a subset \( V' \) of \( V \), we let \( \text{prof}(\sigma)[V'] \) denote the restriction of the vector \( \text{prof}(\sigma) \) to \( V' \). In other words, the vector obtained by zeroing in \( \text{prof}(\sigma) \) all coordinates \( v \not\in V' \).

Now fix \( \sigma \in \mathcal{H} \) and two distinct \( u, v \in V \). Assume \( u, v \) is an inversion in \( \sigma \) with respect to \( \pi \) and that \( |\pi(u) - \pi(v)| = b \) for some integer \( b \). Then, either \( |\pi(u) - \sigma(u)| \geq b/2 \) or \( |\pi(v) - \sigma(v)| \geq b/2 \). We will “charge” the inversion to \( \arg\max_{z \in \{u, v\}} \{|\pi(z) - \sigma(z)|\} \).

For any \( u \in V \), let \( \text{charge}_\sigma(u) \) denote the set of elements \( v \in V \) such that \( (u, v) \) is an inversion in \( \sigma \) with respect to \( \pi \), which is charged to \( u \) based on the rule above. The function \( \text{reg}_\pi(\sigma) \) can now be written as

\[
\text{reg}_\pi(\sigma) = \sum_{u \in V} \sum_{v \in \text{charge}_\sigma(u)} \text{reg}_{u, v|\sigma},
\]

where \( \text{reg}_{u, v|\sigma} \) is defined in Equation (4.6). Indeed, any pair that is not inverted contributes nothing to the difference. Similarly, our estimator \( f(\sigma) \) can be written as

\[
f(\sigma) = \sum_{u \in U} \sum_{i=0}^{\lfloor \log n \rfloor} \frac{|I_{u, i}|}{p} \sum_{t=1}^{p} \text{reg}_{u, v_{u, i, t}|\sigma} \cdot 1_{v_{u, i, t} \in \text{charge}_\sigma(u)}.
\]

Observe that above we dropped the factor \( 1/2 \) because we count each pair \( \{u, v\} \) only once.

For any even integer \( M \), let \( U_{\sigma, M} \) denote the set of all elements \( u \in V \) such that

\[
M/2 < |\pi(u) - \sigma(u)| \leq M.
\]

Let \( U_{\sigma, \leq M} \) denote:

\[
\bigcup_{M' \leq M} U_{\sigma, M'}.
\]

---

\(^3\)For the sake of definition, assume an arbitrary indexing such that \( V = \{u_i : i = 1, \ldots, n\} \).

\(^4\)By breaking ties using some canonical rule, such as charge to the greater of \( u, v \) viewed as integers.
Hereafter we shall remove the subscript \( \pi \), because it is held fixed. Consider the following restrictions of \( \text{reg}(\sigma) \) and \( f(\sigma) \):

\[
\text{reg}(\sigma, M) = \sum_{u \in U_{\sigma, M}} \sum_{v \in \text{charge}_\sigma(u)} \text{reg}_{u,v|\sigma}, \tag{4.9}
\]

\[
f(\sigma, M) = \sum_{u \in U_{\sigma, M}} \left( \sum_{i=0}^{\lceil \log n \rceil} \sum_{t=1}^p \left| I_{u,i,t} \right| \left( \text{reg}_{u,v_{u,i,t}|\sigma} \cdot 1_{v_{u,i,t} \in \text{charge}_\sigma(u)} \right) \right). \tag{4.10}
\]

Clearly, also here, \( f(\sigma, M) \) is an unbiased estimator of \( \text{reg}(\sigma, M) \). Let \( T_{\sigma, M} \) denote the set of all elements \( u \in V \) such that \( |\pi(u) - \sigma(u)| \leq \varepsilon M \). We further split the expressions in (4.9)–(4.10) as follows:

\[
\text{reg}(\sigma, M) = A(\sigma, M) + B(\sigma, M), \quad f(\sigma, M) = \hat{A}(\sigma, M) + \hat{B}(\sigma, M), \tag{4.11}
\]

where,

\[
A(\sigma, M) = \sum_{u \in U_{\sigma, M}} \sum_{v \in \text{charge}_\sigma(u) \cap T_{\sigma, M}} \text{reg}_{u,v|\sigma}, \tag{4.12}
\]

\[
\hat{A}(\sigma, M) = \sum_{u \in U_{\sigma, M}} \left( \sum_{i=0}^{\lceil \log n \rceil} \sum_{t=1}^p \left| I_{u,i,t} \right| \left( \text{reg}_{u,v_{u,i,t}|\sigma} \cdot 1_{v_{u,i,t} \in \text{charge}_\sigma(u) \cap T_{\sigma, M}} \right) \right). \tag{4.13}
\]

We use \( \overline{\cdot} \) to denote set complement in \( V \), and \( B(\sigma, M), \hat{B}(\sigma, M) \) are analogous with \( T_{\sigma, M} \) instead of \( T_{\sigma, M} \), as follows:

\[
B(\sigma, M) = \sum_{u \in U_{\sigma, M}} \sum_{v \in \text{charge}_\sigma(u) \cap \overline{T_{\sigma, M}}} \text{reg}_{u,v|\sigma}, \tag{4.14}
\]

\[
\hat{B}(\sigma, M) = \sum_{u \in U_{\sigma, M}} \left( \sum_{i=0}^{\lceil \log n \rceil} \sum_{t=1}^p \left| I_{u,i,t} \right| \left( \text{reg}_{u,v_{u,i,t}|\sigma} \cdot 1_{v_{u,i,t} \in \text{charge}_\sigma(u) \cap \overline{T_{\sigma, M}}} \right) \right). \tag{4.15}
\]

We now estimate the deviation of \( \hat{A}(\sigma, M) \) from \( A(\sigma, M) \). Fix \( M \) and notice that the expression \( A(\sigma, M) \) is completely determined by non-zero elements of the vector \( \text{prof}(\sigma)[U_{\sigma, \leq M} \cap \overline{T_{\sigma, M}}] \). Let \( J_{\sigma,M} \) denote the number of non-zeros in \( \text{prof}(\sigma)[U_{\sigma, M}] \). Each non-zero coordinate of \( \text{prof}(\sigma)[U_{\sigma, \leq M} \cap \overline{T_{\sigma, M}}] \) is bounded below by \( \varepsilon M \) in absolute value by definition. Let \( P(d, M) \) denote the number of possibilities for the vector \( \text{prof}(\sigma)[\overline{T_{\sigma,M}}] \) for \( \sigma \) running over all permutations satisfying \( d_{\text{SF}}(\sigma, \pi) = d \). We claim that

\[
P(d, M) \leq n^{2d/(\varepsilon M)}. \tag{4.16}
\]

Indeed, there can be at most \( d/(\varepsilon M) \) non-zeros in \( \text{prof}(\sigma)[\overline{T_{\sigma,M}}] \), and each non-zero coordinate can trivially take at most \( n \) values. The bound follows.
Clearly, $\hat{J}(\sigma, M) = \sum_{u \in U_{\sigma, M}} X_{u, i, t}$. For any $u \in V$, let $i_u = \arg\max_i \{ |I_{u, i}| \leq 4M \}$ and observe that by our charging scheme, $X_{u, i, t} = 0$ almost surely, for all $i > i_u$ and $t = 1 \ldots p$. Also observe that for all $u, i, t$, $|X_{u, i, t}| \leq 2N^{-1}|I_{u, i}|/p \leq 2^{i+1}/p$ almost surely. For a random variable $X$, we denote by $\|X\|_\infty$ the infimum over numbers $\alpha$ such that $X \leq \alpha$ almost surely. We conclude:

$$\sum_{u \in U_{\sigma, M}} \sum_{i=0}^{i_u} \sum_{t=1}^{p} \|X_{u, i, t}\|_\infty^2 \leq \sum_{u \in U_{\sigma, M}} \sum_{i=0}^{i_u} N^{-2}p2^{2i+2}/p^2 \leq c_2p^{-1}N^{-2}JM^2$$

for some global constant $c_2 > 0$. (We used a bound on the sum of a geometric series.) Using the Hoeffding bound (see Appendix A), we conclude that the probability that $\hat{A}(\sigma, M)$ deviates from its expected value of $A(\sigma, M)$ by more than some $s > 0$ is at most $\exp\{-s^2p/(2c_2JM^2N^{-2})\}$. We also conclude that the probability that $A(\sigma, M)$ deviates from its expected value by more than $\varepsilon d/(N \log n)$ is at most $\exp\{-c_1\varepsilon^2d^2p/(JM^2\log^2 n)\}$ for some global constant $c_1 > 0$. Hence, by taking $p = O(\varepsilon^{-3}d^{-1}M\log^3 n)$, by union bounding over all $P(d, M)$ possibilities for $\text{prof}(\hat{T}_{\sigma, M})$, with a probability of at least $1 - n^{-7}$ simultaneously for all $\sigma$ satisfying $J_{\sigma, M} = J$ and $d_{SF}(\sigma, \pi) = d$,

$$|A(\sigma, M) - \hat{A}(\sigma, M)| \leq \varepsilon d/(N \log n) \ .$$

(4.17)

But note that, trivially, $JM \leq d$; hence, our choice of $p$ in (4.5) is satisfactory. Finally, by union bounding over the $O(n^3 \log n)$ possibilities for the values of $J, d$, and $M = 1, 2, 4, \ldots$ we conclude that (4.17) holds for all permutations $\sigma$ simultaneously, with a probability of at least $1 - n^{-3}$.

Consider now $B(\sigma, M)$ and $B(\sigma, M)$ which we will need to further decompose as demonstrated next. For $u \in U_{\sigma, M}$, we define a disjoint cover $(T^1_{u, \sigma, M}, T^2_{u, \sigma, M})$ of $\text{charge}_\sigma(u) \cap T_{\sigma, M}$ as follow: If $\pi(u) < \sigma(u)$, then

$$T^1_{u, \sigma, M} = \{ v \in T_{\sigma, M} : \pi(u) + \varepsilon M < \pi(v) < \sigma(u) - \varepsilon M \} .$$

Otherwise,

$$T^1_{u, \sigma, M} = \{ v \in T_{\sigma, M} : \sigma(u) + \varepsilon M < \pi(v) < \pi(u) - \varepsilon M \} .$$

Note that by definition, $T^1_{u, \sigma, M} \subseteq \text{charge}_\sigma(u)$. The set $T^2_{u, \sigma, M}$ is thus taken to be

$$T^2_{u, \sigma, M} = (\text{charge}_\sigma(u) \cap T_{\sigma, M}) \setminus T^1_{u, \sigma, M} .$$
The expressions $B(\sigma, M), \hat{B}(\sigma, M)$ now decompose as $B^1(\sigma, M) + B^2(\sigma, M)$ and $\hat{B}^1(\sigma, M) + \hat{B}^2(\sigma, M)$, respectively, as follows:

$$B^1(\sigma, M) = \sum_{u \in U_{\sigma,M}} \sum_{v \in T_{u,\sigma,M}^1} \text{reg}_{u,v|\sigma}$$

$$B^2(\sigma, M) = \sum_{u \in U_{\sigma,M}} \sum_{v \in T_{u,\sigma,M}^2} \text{reg}_{u,v|\sigma}$$  \hspace{1cm} (4.18)

$$\hat{B}^1(\sigma, M) = \sum_{u \in U_{\sigma,M}} \sum_{i=0}^{[\log n]} \left[ \frac{|I_{u,i}|}{p} \sum_{t=1}^{p} \text{reg}_{u,v_{u,i,t}|\sigma} \cdot 1_{v_{u,i,t} \in T_1^{\sigma}(u,\sigma,M)} \right]$$

$$\hat{B}^2(\sigma, M) = \sum_{u \in U_{\sigma,M}} \sum_{i=0}^{[\log n]} \left[ \frac{|I_{u,i}|}{p} \sum_{t=1}^{p} \text{reg}_{u,v_{u,i,t}|\sigma} \cdot 1_{v_{u,i,t} \in T_2^{\sigma}(u,\sigma,M)} \right].$$  \hspace{1cm} (4.20)

Now notice that $B^1(\sigma, M)$ can be uniquely determined from $\text{prof}(\sigma)[\hat{T}_{\sigma,M}]$. Indeed, in order to identify $T_{u,\sigma,M}^1$ for some $u \in U_{\sigma,M}$, it suffices to identify zeros in a subset of coordinates of $\text{prof}(\sigma)[\hat{T}_{\sigma,M}]$, where the subset depends only on $\text{prof}(\sigma)[\{u\}]$. Additionally, the value of $\text{cost}_{u,v}(\sigma) - \text{cost}_{u,v}(\pi)$ can be “read” from $\text{prof}(\sigma)[\hat{T}_{\sigma,M}]$ (and, of course, $Y(u, v)$) if $v \in T_{u,\sigma,M}^1$. Hence, a Hoeffding bound and a union bound similar to the one used for bounding $|\hat{A}(\sigma, M) - A(\sigma, M)|$ can be used to bound (with high probability) the difference $|B^1(\sigma, M) - B^1(\sigma, M)|$ uniformly for all $\sigma$ and $M = 1, 2, 4, ...$, as well.

Bounding $|\hat{B}^2(\sigma, M) - B^2(\sigma, M)|$ can be performed with the following simple claim.

**Claim 4.2** For $u \in V$ and an integer $q$, we say that the sampling is successful at $(u, q)$ if the random variable

$$\left| \left\{ (i, t) : \pi(v_{u,i,t}) \in [\pi(u) + (1 - \varepsilon)q, \pi(u) + (1 + \varepsilon)q] \cup [\pi(u) - (1 + \varepsilon)q, \pi(u) - (1 - \varepsilon)q] \right\} \right|$$

is at most twice its expected value. We say that the sampling is successful if it is successful at all $u \in V$ and $q \leq n$. If the sampling is successful, then uniformly for all $\sigma$ and all $M = 1, 2, 4, ...$,

$$|\hat{B}^2(\sigma, M) - B^2(\sigma, M)| = O(\varepsilon J_{\sigma,M} M / N).$$

The sampling is successful with probability at least $1 - n^{-3}$ if $p = O(\varepsilon^{-1} \log n)$.

The last assertion in the claim follows from Chernoff bounds (see Appendix A). Note that our bound (4.5) on $p$ is satisfactory, in virtue of the claim.

Summing up the errors $|\hat{A}(\sigma, M) - A(\sigma, M)|, |\hat{B}(\sigma, M) - B(\sigma, M)|$ over all $M$ gives us the following assertion: With probability at least $1 - n^{-2}$, uniformly for all $\sigma$,

$$|f(\sigma) - \text{reg}_\pi(\sigma)| \leq \varepsilon N^{-1} d_{SF}(\pi, \sigma) \leq 2 \varepsilon \text{dist}(\pi, \sigma),$$

45
where the last inequality is by Diaconis and Graham (1977). This concludes the proof.

In the following Algorithm 2 our specific $\varepsilon$-SRRA construction for LRPP is summarized. Note that by the choice of the sample size $p$, the number of preference-queries required for computing $f$ is $O(\varepsilon^{-3}n \log^3 n)$. Although we only provide the required magnitude of the required sample size, observe that each step of the construction is computationally efficient. This allows us to empirically experiment with our ideas, as will be seen in the following sections.

**Algorithm 2** SRRA for LRPP

**Require:** $V$, $\mathcal{H}$, a pivot $\pi \in \mathcal{H}$, and estimation parameter $\varepsilon \in (0, 1/5)$

$p \leftarrow O(\varepsilon^{-3} \log^3 n)$

for $u \in V$ do

  for $i = 0, 1, \ldots, \lceil \log n \rceil$ do

    $I_{u,i} \leftarrow \{v : (2^i - 1)p < |\pi(u) - \pi(v)| < 2^{i+1}p\}$

  for $t = 1, \ldots, p$ do

    $v_{u,i,t} \leftarrow$ a uniformly and independently sampled alternative from $I_{u,i}$

  end for

end for

return $f : \mathcal{H} \rightarrow \mathbb{R}$, defined by

$$f(\sigma) = \sum_{u \in V} \sum_{i=0}^{\lceil \log n \rceil} \frac{I_{u,i}}{p} \sum_{t=1}^{p} \left( \text{cost}_{u,v_{u,i,t}}(\sigma) - \text{cost}_{u,v_{u,i,t}}(\pi) \right) ,$$

We can now define a specific active learning algorithm for LRPP by plugging the $\varepsilon$-SRRA constructions, defined in Algorithm 2, into our method’s meta-algorithm which is defined in Corollary 3.1. This provides the following bound on the number of preference-queries.

**Corollary 4.1** There exists an active-learning algorithm for obtaining a solution $\pi \in \mathcal{H}$ for LRPP with $\text{er}_D(\pi) \leq (1 + O(\varepsilon))\nu$ with total query complexity of $O(\varepsilon^{-3}n \log^5 n)$. The algorithm succeeds with probability at least $1 - n^{-2}$.

Corollary 4.1 tells us that the SRRA method provides a solution of cost $(1 + \varepsilon)\nu$ with query complexity that is slightly above linear in $n$ (for constant $\varepsilon$), regardless of the magnitude of $\nu$. By comparison, we saw in Section 4.3 that any known active-learning result that uses bounded disagreement coefficient and VC dimension arguments (only) guarantees a query complexity of $O(n^{\nu^{-1}})$, tending to the pool size of $n(n-1)$ as $\mu$ becomes small. Note that $\nu = o(1)$ is quite realistic for this problem. For example, consider the following noise model: A
ground truth permutation $\pi^*$ exists, $Y(u, v)$ is obtained as a human response to the question of preference between $u$ and $v$ with respect to $\pi^*$, and the human errs with a probability proportional to $|\pi^*(u) - \pi^*(v)|^{-\rho}$. That is to say, closer pairs of items in the ground truth permutation are more prone to confuse a human labeler. The resulting noise is $\nu = n^{-\rho}$ for some $\rho > 0$. (Note however that our work does not assume Bayesian noise, and we present this scenario for purposes of illustration only.)

In terms of query complexity, it turns out that our bound provides only a slight improvement on the divide-and-conquer active-learning algorithm for LRPP of Ailon (2012). Specifically, we improve the dependency on $\varepsilon$ from $\varepsilon^{-6}$ to $\varepsilon^{-3}$. Although our method provides only a minor improvement, it still defines the current state-of-the-art for query-efficient LRPP; more importantly, though, it defines the first query-efficient LRPP algorithm that is applicable over an arbitrary set of permutations, $\mathcal{H} \subseteq \mathcal{V}$. We utilize this fact in the following section, instantiating $\varepsilon$-SRRAs for the set of permutations induced by hyper-planes in $\mathbb{R}^d$.

### 4.5 LRPP over Linearly Induced Permutations in Constant-Dimensional Feature Space

A special class of interest is known as LRPP over linearly induced permutations in constant-dimensional feature space. We use the same definition of $\mathcal{X}$ as in Section 4.1, except that now each point $v \in V$ is associated with a feature vector, which we denote using bold face: $v \in \mathbb{R}^d$. The hypothesis class $\mathcal{H}$ now consists only of permutations $\pi$ such that there exists a vector $w_\pi \in \mathbb{R}^d$ satisfying

$$\pi(u, v) = \mathbf{1}_{\langle w_\pi, u - v \rangle > 0}.$$  \hfill (4.22)

(Where $\langle \cdot, \cdot \rangle$ denotes the inner-product functional.)

Observe that $\pi \in \mathcal{H}$ is indeed a permutation. First, $\pi$ is pairwise consistent, $\mathbf{1}_{\langle w_\pi, u - v \rangle > 0} = 1 - \mathbf{1}_{\langle w_\pi, v - u \rangle > 0}$, because $\langle w_\pi, u - v \rangle = -\langle w_\pi, v - u \rangle$. Second, $\pi$ is transitive because if $\pi(u, v) = \pi(z, u) = 1$ for some $u, v, z \in V$, meaning that $\langle w_\pi, u - v \rangle > 0$ and $\langle w_\pi, z - u \rangle > 0$, then from the bilinearity of dot-product we have $\langle w_\pi, z - v \rangle = \langle w_\pi, z - u + v - u \rangle = \langle w_\pi, z - u \rangle - \langle w_\pi, u - v \rangle \geq \langle w_\pi, z - u \rangle > 0$.

We will apply the powerful mathematical theory of geometrical arrangements, which has been extensively researched with respect to computational geometry (see, e.g., Chapter 8 of de Berg et al., 2008, for further details), whereby intersections of $n$ hyper-planes in $\mathbb{R}^d$ induce the cell complex. Here we will examine the dual “space” in which $\mathcal{X}$ is a set of hyperplanes in $\mathbb{R}^d$, and permutations in $\mathcal{H}$ are unique cells in the corresponding arrangement (Cell subdivision is induced by intersections of these hyperplanes).

Each pair of alternatives $(u, v) \in \mathcal{X}$ is geometrically viewed as the following halfspace $H_{u, v} = \{x : \langle x, u - v \rangle > 0\}$, whose (closure) supporting hyperplane is
Let $\mathcal{H}$ be the collection of these $\binom{n}{2}$ hyperplanes

$$\{h_{u,v}: (u,v) \in \mathcal{X}\}.$$  

Note that $h_{u,v} = h_{v,u}$, and thus it matches the unordered pair

$$\{u,v\}.$$  

We will assume there exists an arbitrary indexing on $V$ and will identify a pair with $h_{u,v}$ iff $u < v$. (See Remark 4.1.) The collection $\mathcal{H}$ corresponds to the maximal dimensional cells in the underlying arrangement $\mathcal{A}(\mathcal{H})$. Thus, from now on, we call $\mathcal{A}(\mathcal{H})$ the permutation arrangement and naturally identify full-dimensional cells with their induced permutations. We denote by $C_{\pi} \subseteq \mathbb{R}^d$ the unique cell corresponding to a permutation $\pi \in \mathcal{H}$. Figure 4.3 depicts the elements of this construction when the embedding is in the planar $\mathbb{R}^2$.

Figure 4.3: Arrangement and duality depicted.

Here we will use the disagreement-based SRRA constructions defined in Section 3.2. It turns out that due to the geometrical structure, this SRRA construction provides non-trivial sample complexity guarantees, contrary to what we showed for the combinatorial setting of Section 4.1.

**Bounding the VC dimension and disagreement coefficient.** Using standard tools from combinatorial geometry, the VC dimension of $\mathcal{H}$ is at most $d - 1$. Roughly speaking, this property follows from the fact that in an arrangement of $m$ hyperplanes in $d$-space, each of which meets the origin, the overall number of cells is at most $O(m^{d-1})$. See de Berg et al. (Chapter 8, 2008).

As for the uniform disagreement coefficient, we show below that it is bounded by $O(n)$. Let $\pi \in \mathcal{H}$ be a permutation with a corresponding cell $C_{\pi}$ in $\mathcal{A}(\mathcal{H})$. The ball $B(\pi, r)$ is the geometric closure of the union of all cells corresponding to “realizable” permutations $\sigma$ satisfying $\text{dist}(\sigma, \pi) \leq r$. The corresponding disagreement region $\text{DIS}(B(\pi, r))$ corresponds to the set of ordered pairs (halfspaces) intersecting $B(\pi, r)$. Next, we show:

**Proposition 4.1** The measure of $\text{DIS}(B(\pi, r))$ in $\mathcal{D}_X$ is at most $8rn$. 

48
Proof By Diaconis and Graham (1977), Spearman’s Footrule distance between any two permutations $\pi$ and $\sigma$ is at most twice $N\text{dist}(\pi, \sigma)$, where $N = n(n - 1)$. Hence, if $\text{dist}(\pi, \sigma)$ is $r$, then any element $u$ could only swap locations with a set of elements located up to $2rN$ positions away to the right or left. This yields a total of $4rN$ “swap-candidates” for each $u$. Thus, at most $4rNn$ inversions are possible. Note that each inversion corresponds to a hyperplane (unordered pair) that we cross; thus, the total number of ordered pairs is at most $8rNn$. The probability measure of this set is at most $8rn$, because we assign equal probability of $N^{-1}$ for each possible pair in $X$. The result follows.

Following from (Proposition 4.1), where the disagreement coefficient $\theta$ is always bounded by $O(n)$, we establish our bound. We now invoke Corollary 3.2 with $\mu = O(1/n^2)$ (which is tantamount to $\mu = 0$ for this problem, because $|X| = O(n^2)$ and we are using the uniform measure). We conclude:

**Theorem 4.3** An $\varepsilon$-SRRA for LRPP in linearly induced permutations in $d$ dimensional feature space can be constructed with respect to any $\pi \in \mathcal{H}$, with probability at least $1 - \delta$, using at most $O \left( n^2 \log^2 n + n \varepsilon^{-2} \log(\delta^{-1} \log n) \right)$ label-queries.

Combining Theorem 4.3, and the iterative algorithm described in Corollary 3.1, we get the following:

**Corollary 4.2** There exists an algorithm for obtaining a solution $\pi \in \mathcal{H}$ for LRPP in linearly induced permutations in $d$ dimensional feature space with $\varepsilon_D(\pi) \leq (1 + O(\varepsilon)) \nu$ with total query complexity of

$$O \left( \varepsilon^{-2} n d \log^3 n + n \varepsilon^{-2} \log^2 n \left( \log(\delta^{-1} \log n) \right) \right) .$$  \hspace{1cm} (4.23)

The algorithm succeeds with probability at least $1 - \delta$.

We compare this bound to that of Corollary 4.1. For the sake of comparison, assume $\delta = n^{-2}$, so that (4.23) takes the simpler form of

$$O \left( \varepsilon^{-2} n d \log^3 n / \log(1/\varepsilon) \right) .$$

This bound is better than that of Corollary 4.1, as long as the feature space dimension $d$ is $O(\varepsilon^{-2} \log^2 n)$. For larger dimensions, Corollary 4.1 gives a better bound. It would be interesting to obtain a smoother interpolation between the geometric structure arising from the feature space and the combinatorial structure arising from the permutations. We refer the reader to Jamieson and Nowak (2011) for a recent result with improved query complexity under certain Bayesian noise assumptions.
4.6 Heuristics for Optimizing the SRRAs

So far we have neglected the optimization step of Algorithm 1, which seeks the minimizer of $f_i(\sigma)$ over $H$, where $f_i$ is an $\varepsilon$-SRA with respect to the intermediate solution $\pi_i$. This amounts to optimization of an NP-Hard problem, called minimum feedback arc-set in sparse DiGraphs. Thus, there is no hope of devising an exact solver; instead, we replace this step with various heuristics discussed below. We will empirically experiment with these solvers in the next section.

Our first heuristic is a simple greedy hill-climbing solver that starts with a seed solution and repeatedly searches a single-element move that improves the sparse cost. A single-element move is a result of removing a single alternative from the permutation and reinserting it into a different position. If such a move is found, it is performed; otherwise, the process is halted and the resulting permutation is returned as a local optimum. When applied as part of the optimization step of Algorithm 1, we take the seed permutation to be the previous SRA's iteration solution. We refer to this solver as CLIMB.

The second solver we considered is a weighted sparse variant of the SVMRank convex relaxation. We presented this solver in Example 3.1. Recall that such convex relaxation requires $V$ to be embedded in $\mathbb{R}^d$. Therefore, we matched each alternative $u \in V$ with a (random) feature vector $u \in \mathbb{R}^d$, where, $d \geq 2n$ and each coordinate $u_i$ is chosen independently uniformly at random from the range $(0, 2)$. Using standard arguments in computational geometry, our choice of $d$ assures us that the entire set of $n!$ permutations can be almost surely obtained by ordering the alternatives according to a linear score function $u \mapsto w \cdot u$. (Note that these random features are used just to relax a combinatorially hard optimization problem – they do not describe the elements in $V$ in any way.)

Recall that the SRA construction of Algorithm 2 induces a weighted, sparse collection of labeled pairs. Accordingly, a weighted, sparse variant of SVMRank can be considered. Instead of considering all pairs $u, v$ in the convex relaxed-cost function (3.7), we only consider those pairs that were sampled in the SRA construction (Algorithm 2) with the corresponding weight. This approach has been described in detail in Ailon et al. (2012). We denote Algorithm 1 in which Line 2 uses this relaxation to optimize for the next solution by COMBI-SVM.

Our last suggested solver is a simple combination of the other two. First we obtain a solution using the convex relaxation defined by COMBI-SVM; then we use that solution as a warm start for CLIMB. Therefore, we denote this solver COMBI-SVM+CLIMB.

4.7 Empirical Proof of Concept

Due to the importance of the problem of ranking in many applications, it is imperative to test the theoretical guarantees described above. In this section,
we empirically examine our ideas on a few synthetic and real-world sets of data, but since datasets for the preferences setting we examined are not, as far as we know, publicly available, we constructed our own benchmark data. The design and construction of such data is not trivial, mainly due to the effort necessary to define diverse problems, and due to the preferences’ quadratic overhead. In view of these constraints, as well as the constraints on our resources, our data are limited in scale and size. Finally, we compare our solutions with the basic random-sampling baseline rather than competing algorithms which were lacking. Thus, we refer to our tests as a “proof of concept.”

4.7.1 Synthetic Experiments

Datasets. We define nine synthetic problems. In each problem the set $V$ contains $n = 100$ alternatives. Each alternative $u \in V$ is embedded in $\mathbb{R}^d$ as a feature vector $\mathbf{u}$, with $d = 200$. The features are randomly $i.i.d.$ generated by setting each feature as a number chosen uniformly at random in the range $(0, 2)$.

In order to generate the “ground truth” labeling function $Y : \mathcal{X} \mapsto \{0, 1\}$, we first chose a random coefficient vector $\mathbf{w} \in \mathbb{R}^d$ such that $\|\mathbf{w}\|_2 = 1$ and ordered the elements of $V$ according to the value $\mathbf{w} \cdot \mathbf{u}$. The underlying noiseless preference function $Y_{\text{perfect}} : \mathcal{X} \mapsto \{0, 1\}$ is defined by

$$Y_{\text{perfect}}(u, v) = 1 \iff \mathbf{w} \cdot \mathbf{u} > \mathbf{w} \cdot \mathbf{v}.$$  
(As expected, no ties were obtained.) We generated nine different noisy versions of $Y_{\text{perfect}}$, using two different noise generation methods with varying parameters.

In the first method, for each $u, v \in V$, we flip the value of $Y_{\text{perfect}}(u, v)$ with probability $\beta |\text{pos}(u) - \text{pos}(v)|^{-\alpha}$, where $\beta > 0$ and $0 < \alpha < 1$ are hyperparameters and $\text{pos}(x)$ is the position of $x \in V$ (“rank”) in the ordering induced by the perfect permutation. We call this model distance decaying noise and denote the corresponding noisy labeling function by $Y_{\text{dist}(\beta,\alpha)}$. We experimented with six different values of $(\beta,\alpha)$, namely $(\beta,\alpha) \in \{(0.1, 1), (0.2, 1), (0.5, 0.5), (0.3, 2), (0.5, 2), (0.7, 2)\}$. The matrices $Y_{\text{dist}(\beta,\alpha)}$ for the chosen values is depicted in Figure 4.4 (a)–(f). Note that the noise pattern for $(\beta,\alpha) = (0.5, 0.5)$ is very close to uniform noise.

The structured confusions model, STRUCT, has a different noise structure. The inversions are condensed in a randomly located fixed-size rectangle, as seen in Figure 4.4 (g), (h) and (i). In order to make the noise level of this and the above model comparable, the area of the error rectangle for $Y_{\text{STRUCT}(\gamma)}$ is the same as the amount of noise obtained in $Y_{\text{dist}(\beta=\gamma,1)}$.

Results We ran a version of Algorithm 1; at each iteration we generated a biased sample of labels (an SRRA) using Algorithm 2 with $m = 1$, which is the minimal possible value. Experimenting with other values of $m$ yielded similar
Figure 4.4: Synthetic model-based preferences. We permuted the rows and columns on the basis of the \( \mathbf{w} \)-induced order, resulting in a \( Y_{\text{perfect}} \) noiseless ranking, which contrary to the upper-right and lower-left triangles depicted here, would have appeared as perfect solid-dark and solid-light triangles, respectively.

results. We ran 15 iterations of the algorithm, measuring the error rate of the hypothesis obtained at each step. Note that we altered Algorithm 1 as follows: instead of taking the next hypothesis \( h_{i+1} \) to be the minimizer of \( f_i(h') \) over \( h' \in \mathcal{H} \), we chose it to be the minimizer of \( f_0(h') + f_1(h') + \cdots + f_i(h') \). In other words, we used all the samples that had been drawn up to that point before optimizing for the next hypothesis.

We computed the average of 10 such executions and plotted the average curve accompanied by standard deviation brackets. For comparison, we ran an experiment in which at each interaction, instead of generating a biased sample (using Algorithm 2), we generated a sample of the same size using uniformly at-random drawn pairs instead. This allowed us to compare the active-learning scheme with standard (passive) learning. At each iteration of Algorithm 1, we used COMBI-SVM (but here, we did not need to draw a random feature vector — we used the same vectors drawn in the generation \( Y_{\text{perfect}} \)).
Figure 4.5: Results for synthetic datasets averaged over 10 runs, accompanied by standard error of the mean.

Figure 4.5 depicts the learning curves corresponding to the noise models studied. Observe, that as expected from the theory (see Section 4.4), the SRRA method is an obvious winner in low noise conditions; while under high noise conditions, the RAND curve representing the uniform sample (passive learning) is better.

An obvious criticism of our experiment is that for the data generated using the distance-proportional noise, the sampling of noise locations in $Y_{\text{dist}(\beta,\alpha)}$ is suspiciously similar to the method by which the SRRA sampler (Algorithm 2) samples labels. First, note that this is true only when the hypothesis $h$ serving as
argument to Algorithm 2 is very close to $Y_{\text{perfect}}$; this is not the case, however, in the first iterations of Algorithm 1. Second, notice that the STRUCT noise model also exhibits better learning curves for small noise.

### 4.7.2 Real Data Experiments

**The Food-Dish Datasets.** We constructed three datasets based on a set of real-world alternatives and a full collection of preferences obtained by soliciting human responses via crowdsourcing services.

All three problems shared the same set $V$ of 50 alternatives. Each alternative was a food dish. These natural, real-world objects were gathered from a recipe web-site. The goal was to order the alternatives based on the three label functions $Y_{\text{COST}}, Y_{\text{WINE}}, Y_{\text{DATE}}$ that were generated by soliciting responses to the following questions on pairs of alternatives:

(i) **COST:** Which dish costs more (assuming that the dishes appear on the menu of the same restaurant)?

(ii) **WINE:** Which dish goes better with red wine?

(iii) **DATE:** Which dish is more appropriate to order on a first date?

The datasets were labeled using Amazon’s Mechanical-Turk—system for crowdsourcing. Mechanical Turk is a marketplace website where requesters, such as us, submit HITs (Human Intelligence Tasks) and workers respond to those HITS (see, e.g., Figure 4.6). A HIT may contain a simple question, the response to which can be used as a label. In our case, a HIT was created for each of the three preference questions above, for each of the 1225 possible pairs of alternatives from the ground set of 50 dish images. Each HIT was assigned to five independent workers. The value of $Y_{\text{cost}}, Y_{\text{WINE}},$ and $Y_{\text{DATE}}$ at $(u, v)$ was taken to be the majority vote over the five corresponding responses.

To decrease the heterogeneity of the responding population, we required that all workers be located in the US. To reduce dependencies between responses from the same worker, we prevented the same worker from responding to successive HITs sharing the same alternative.

Figure 4.7 depicts the responses $Y_{\text{cost}}, Y_{\text{WINE}},$ and $Y_{\text{DATE}}$ (after taking the majority for each pair). Observe that for cost, we obtain a cleaner preference matrix than for WINE, in the sense that the data are almost sortable. Similarly, DATE data seem least sortable. This corresponds to our intuition that the cost criterion

---

5 http://www.epicurious.com/
6 https://www.mturk.com/mturk/welcome
7 Each worker has a unique Mechanical-Turk ID. In addition, Amazon attempts to verify a one-to-one correspondence between users and worker IDs.
8 We could have equally well taken an average and used fractional preferences.
Both dishes are served at the same restaurant. Which one costs more?

Guidelines:
- Select exactly a single dish: either Left -or- Right
- Whenever both dishes are in the same degree for the task then you may choose any of the pair
  - Note: Please make sure you do not submit more than 100 HITs for this task
  - This is a restriction of our research, please respect it
- Feel free to add any valuable comment.

Please provide any comments you may have below, we appreciate your input!

---

**Figure 4.6:** Example for an Amazon Turk HIT that provides a label for the COST dataset.

is the least subjective, and DATE is the most subjective of the three questions asked. Note our effort to re-order the rows and columns in the corresponding pictures, so that the top-right portion appears almost solid dark (corresponding to 1’s), and the bottom-left almost solid bright (corresponding to 0’s). We cannot do this perfectly because this is NP-Hard, so we used various heuristics and have no proof of optimality. This was done for illustration purposes only.

### 4.7.3 The Combinatorial Experiment

We performed an experiment similar to that performed with the synthetic data. Unlike the synthetic data case, however, we do not assume features for the data here; the entire structure is combinatorial, not geometric.

We compared the active-learning SRRA approach with the standard (passive)
learning in which labels are revealed for pairs chosen uniformly at random. In one experiment we used \textsc{Combi-SVM+Climb} for both sampling techniques (depicted in the first row of Figure 4.8), and in the other we used \textsc{Climb} (depicted in the second row). One sees that active-learning is better in most parts of the learning curve for cost (i.e., left column). When using \textsc{Climb}, the advantage of SRRA is more significant. This is quite an interesting phenomenon, which tells us that the SRRA-active-sampling technique helps avoid local minima in the greedy optimization heuristic. For \textsc{Wine}, SRRA seems to be better than random sampling that uses \textsc{Climb} in certain parts of the learning curve, and worse than random sampling that uses \textsc{Combi-Svm}. For \textsc{Date}, random sampling is better than active sampling that uses both solvers: this is not surprising given that the noise level there, is the largest.

The Geometric Experiment

The main goal of this work was to test active sampling for the purpose of learning a combinatorial permutation. A more realistic scenario is one in which a practitioner tries to order alternatives using a set of attributes that reasonably describe these alternatives (i.e., features). Above, we used feature vectors (both for the synthetic and real-world data), but they were used to relax the combinatorial optimization difficulties, not to describe the objects.

Conveniently, the food-dish data also contains corresponding textual recipes. We generated bag-of-word-based features using stemming, stop-words elimination, and some standard contextual normalization of quantities and time expressions. Figure 4.9 depicts an example of a recipe text.

We again compared SRRA active learning with uniform sampling, using SVM-Rank with the bag-of-word features as a solver ($\text{SVM}$) in each iteration of Algorithm 1.

The comparison is shown in the third row of Figure 4.8. For cost (left column),
SRRA significantly beats random sampling after viewing at most 300 sample pairs, and thereafter performs comparably. For WINE (middle column), SRRA beats uniform random sampling after viewing at most 200 sample pairs, then falls behind. For DATE (right column), uniform random sampling is better throughout. Again, this fits the theoretical result, predicting better performance for SRRA active learning in the case of low noise.

**On the Improvements We Obtained.** The biggest improvement is obtained when using CLIMB over COST (depicted in the leftmost graph in the second row of Figure 4.8). Here, the SRRA learning curve improves random sampling curve by 5.5% on the average; In the first third of the learning phase, the advantage of SRRA is even greater: 8.3% on the average; in the second third of the learning phase, SRRA beats random sampling by an average of 4.6%; and in the last
**At a Glance**

**main ingredients:** Pear, Blue Cheese, Leafy Green, Steak;  
**type:** Quick & Easy.

**Ingredients**

3 tablespoons olive oil, divided; 1 tablespoon Sherry wine vinegar; 1 tablespoon minced shallot; 1 1/2 teaspoons honey; 2 8-to-9 ounce rib-eye steaks (each about 1 inch thick), trimmed; Coarsely cracked black pepper; 2 cups finely sliced radicchio; 1 cup (packed) mixed baby greens; 1 small ripe Comice pear, quartered, cored, thinly sliced; 1/3 cup crumbled chilled blue cheese.

**Preparation**

Whisk 2 tablespoons olive oil, Sherry wine vinegar, minced shallot, and honey in large bowl. Season dressing to taste with salt and pepper. Sprinkle steaks generously with cracked black pepper and salt. Heat 1 tablespoon oil in heavy medium skillet over medium-high heat. Add steaks; cook to desired doneness, about 3 minutes per side for medium-rare. Transfer to plates. Add last 4 ingredients to dressing; toss. Mound salad alongside steaks and serve.

**Figure 4.9:** Example of a recipe text.

third of the learning phase the advantage shrinks to 3.7% on the average. Note, however, that these nice improvements diminish when comparing the curves we produced with the SVM and COMBI-SVM + CLIMB solvers. SRRA is better using the SVM solver over COST, only at the beginning of the learning phase. However, observe that the learning phase here is completed after acquiring 500 labels. This amount of labels is twice $n \log n$, which is the best we can wish for. When combining COMBI-SVM + CLIMB, the advantage of SRRA disappears. This can be explained by the fact that the warm start we get from applying COMBI-SVM provides a good guess that diminishes the advantage we observed when applying CLIMB with arbitrary guesses.

### 4.7.4 A Note on Related Algorithms and Datasets

There are only a few active-learning methods designed specifically for our setting, and none that provide provable guarantees. The divide and conquer algorithm of Ailon (2012) is, as far as we know, the only one that is analyzed under the same worst-case agnostic-noise setting as SRRA. Its core sampling technique and relaxation, however, are very similar to those of SRRA, and thus it is expected to behave similarly to SRRA. Another well-justified algorithm was recently given by Jamieson and Nowak (2011); however, they assume a different setting that is does not lend itself to a comparison with our approach. Carterette et al. (2008) suggest a nice active sampling heuristic for comparing the rankings of two search
engines, which is a setting that is related to, yet different from ours. We tried to adapt our setting to the well-known uncertainty sampling (Tong and Koller, 2001; Schohn and Cohn, 2000; Campbell et al., 2000) over the three real datasets. The comparison did not indicate a clear winner (uncertainty wins over WINE, loses over DATE, and gains a draw over cost).

There are many public archives known as “learning to rank datasets.” To the best of our knowledge, the labels obtained in all of them are based on judgments of individual items, and not pairs. The exception is LETOR, where comparative labels are provided but, as far as we understand, are induced from individual judgments. It is challenging to design large-scale datasets using pairwise information for active learning because it is impossible to know in advance which set of alternatives the algorithms would want to access, and each set of alternatives gives rise to quadratically many possibilities. We leave for future work the construction of larger datasets, as well as a comparison with other heuristics.

4.7.5 A Note on Misconceptions and Beliefs

The lack of benchmarked empirical work in the context of ranking from pairwise preferences leaves room for various beliefs. This section seeks to corroborate a few.

Ranking by Preference Majority?

Ranking by preference majority is one of the simplest and most intuitive ranking heuristics; simply rank alternatives according to the number of pairs in which they have been preferred. We refer to this count as preference-wins.

Figure 4.10 depicts two different views of the preferences defined by the COST dataset. In (a), the preference rows (and columns) are ordered according to the preference-wins; in (b), they are ordered by applying a local-improvement heuristic over the ranking of Figure 4.10(a).

Comparing the two rankings in terms of Kendall-tau cost, the preference-majority ranking of Figure 4.10(a) entails a cost of 109, while the corresponding local-improvement ranking has a cost overhead of 78. This is a cost reduction of roughly 30%.

A similar phenomena is observed in the other two datasets, indicating that the ranking by preference-majority rule is simply not the best choice.

Cycles of Preferences

The combinatorial hardness of the discussed ranking problem is due to the existence of (preference) cycles within the graph corresponding to $W$; taken without any cycles, this is merely a sorting problem. From a practitioner’s perspective,
there is a common belief that natural sources inherently define such cycles, for example, due to human irrationality.

Thus, it is fair to ask whether our crowdsource datasets contain cycles, and what is their magnitude. Table 4.1 provides an indication of the number of cycles in each of the natural source datasets. The number of cycles is indicated by the number and size of the strongly connected components (scc) in the graph. An scc with \( n \) vertices contains \( 2^n - (n + 1) \) cycles.

Observe the correlation between the number of cycles and the subjectivity-magnitude of the preferences. The largest number of cycles occurs in the subjective-dataset DATE, and the lowest number occurs in the absolute value dataset ABS-VAL.

Table 4.1: The number of cycles in the different crowdsource datasets. We present the number and size of strongly connected components (scc’s).

<table>
<thead>
<tr>
<th>Data set</th>
<th>#scc</th>
<th>Size of max scc</th>
</tr>
</thead>
<tbody>
<tr>
<td>COST</td>
<td>2</td>
<td>43</td>
</tr>
<tr>
<td>WINE</td>
<td>6</td>
<td>45</td>
</tr>
<tr>
<td>DATE</td>
<td>1</td>
<td>50</td>
</tr>
<tr>
<td>ABS-VAL</td>
<td>50</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 4.10: Ranking by preference-majority versus local-improvement. The dark color illustrates that the row alternative is preferred over that of the corresponding column. The REF ranking corresponds to row indices, where the top and bottom rows are most and least preferred, respectively.
4.8 Discussion

The main concern of this chapter was the query-efficient variant of learning to rank from pairwise preferences. We presented a specific SRRA construction that defines the state-of-the-art query complexity bounds for this problem (when noise is low). Attempting to solve the problem with uniform sampling or any known method that uses the disagreement coefficient and VC dimension arguments only, tends to fail (in the worst-case setting).

In addition to the theoretical results, we described several approaches for implementing our solution. We cannot hope for an exact solution for the ERM optimization step of our SRRA meta-algorithm (Algorithm 1). Thus, we devised several heuristic solvers for this optimization problem. We view the resulting implementations as relaxed versions of the (theoretically justified) SRRA constructions.

The lack of publicly available benchmark datasets designed for LRPP led us to generate such a benchmark. Our benchmark consisted of several synthetic and three real-world datasets. The design and construction of the real-world datasets were conducted via crowd sourcing. An empirical evaluation of our SRRA implementations over these datasets supports the theory that our method has advantages when the inherent noise is low. The advantage is marginal with respect to the uniform sampling baseline; nevertheless, it is significant. We assume that our results reflect the small magnitude of the datasets we used and the presence of constants that are hidden in the big-O notation in our analysis. In Chapter 6, we discuss a heuristic that integrates exploration and exploitation without implementing SRRA, yet resembling it in character. There, we show that a good “coverage” of hypothesis space yields exceptional empirical results. Thus, this provides optimistic empirical support for our ideas overall.
Chapter 5

SRRA for Clustering with Side Information

Clustering with side information is a fairly new variant of clustering described independently by both Demiriz et al. (1999) and Ben-Dor et al. (1999). In the machine-learning community it is also widely known as semi-supervised clustering (see, e.g., Basu, 2005). This variant defines a family of settings that add a certain form of supervision to the originally fully unsupervised clustering setting. The nature of the available feedback that provides the side information defines the specific setting.

The most natural and commonly discussed forms of supervision are single-item labels (see, e.g., Demiriz et al., 1999), and pairwise constraints (see, e.g., Ben-Dor et al., 1999). In the single-item-labels setting, the feedback matches items with those cluster indices to which they should (or should not) belong. While this form of supervision is fairly close to that provided in supervised learning, it differs here in that no statistical assumptions are usually not made in the process that provides the labels (e.g., some cluster index labels may even be absent). The pairwise constraint setting provides “locally” flavored feedback that states whether a pair of elements either “must” or “cannot” link together. We focus on this form of supervision.

In machine learning, there are two main approaches for utilizing pairwise side information. In the first approach, this information is used to fine-tune, or to learn a distance function, which is then passed on to any standard clustering algorithm, such as $k$-means or $k$-medians (see, e.g., Klein et al., 2002; Xing et al., 2002; Cohn et al., 2000; Balcan et al., 2009b; Shamir and Tishby, 2011; Voevodski et al., 2012). The second approach, which is more related to our work, modifies the clustering algorithm’s objective to incorporate the pairwise constraints (see, e.g., Basu, 2005; Eriksson et al., 2011; Cesa-Bianchi et al., 2012). Basu (2005), whose thesis, which also serves as a comprehensive survey, has championed this approach in conjunction with $k$-means and hidden Markov random field clustering algorithms.
Our work is closely related to a combinatorial-optimization theoretical-setting known as correlation clustering. This setting deals with clustering of a finite set of elements. Full knowledge of all “must”- and “cannot”-link pairwise constraints is available (for free); however, similarities are noisy in the sense that they may induce non-transitivity noise (analogous to that which we discuss in Chapter 4). In other words, the optimal solution has a non-zero cost. The goal is to eradicate this noise.

Correlation clustering was defined by Bansal et al. (2004), and Shamir et al. (2004) under the name cluster editing. The problem is NP-Hard (see, e.g., Charikar et al., 2005). Constant factor approximations are known for various minimization versions of this problems (Charikar and Wirth, 2004; Ailon et al., 2008). A PTAS is known for a minimization version in which the number of clusters is fixed to be $k$ (Giotis and Guruswami, 2006), as in our setting.

Here, we consider a query-efficient variant of correlation clustering. Each pairwise information bit comes at a cost and must be treated frugally. The goal, then, is to cleanup the non-transitivity noise while minimizing the information overhead.

From the theoretical standpoint of computational complexity, since the underlying combinatorial problem is NP-Hard, a practitioner would have to resolve heuristics in order to deal with the underlying optimization problems, if they assume worst-case scenarios. The query complexity standpoint, however, is far from understood. We believe that our results, though highly non-trivial, do not yet close the book on the question.

Our work isolates the use of information coming from pairwise clustering constraints and separates it from the geometry of the problem. In future work, it would be interesting to analyze our framework in conjunction with the geometric structure of the input. Interestingly, Eriksson et al. (2011) studies active learning for clustering using the geometric input structure. Unlike our setting, though, they assume either no noise or parametric noise.

Our problem’s instance space could be thought of as edges of a graph (for other work on graph structure active learning, see, e.g., Cesa-Bianchi et al. (2012)). The main difference between our settings is that Cesa-Bianchi et al. assume that the graph structure is given in advance (yet the edge labels are hidden). We work over full graphs and thus cannot utilize structural information.

We are aware of no previous work that proves guarantees of the type we achieve in our setting.

The chapter is organized as follows. We start by defining the active-learning problem. We then explain why the disagreement-based methods, such as the one discussed in Section 3.2.1, would fail to provide meaningful guarantees. Then we describe a specific SRRA construction and analyze its guarantees. The chapter concludes with a discussion of a hierarchical extension of the problem.
5.1 Problem Definition

We define the problem of pool-based active clustering with side information using the following notation: Let $V$ be a set of points of size $n$, and $k$ a given number; the instance space is the set of distinct pairs of elements $\mathcal{X} = V \times V \setminus \{(u, u) \in V\}$; as before, we assume the labeling function is binary and deterministic. The label $Y((u, v)) = 1$ means that $u$ and $v$ should be clustered together, and $Y((u, v)) = 0$ means the opposite. Assume that $Y((u, v)) = Y((v, u))$ for all $u, v$. (Equivalently, assume that $\mathcal{X}$ contains only unordered distinct pairs without any constraint on $Y$. For notational convenience we preferred to define $\mathcal{X}$ as the set of ordered distinct pairs.)

The hypothesis class $\mathcal{H}$ is the set of equivalence relations over $V$ with at most $k$ equivalence classes. More precisely, every $h \in \mathcal{H}$ is identified with a disjoint cover $V_1, \ldots, V_k$ of $V$ (some $V_i$’s possible empty), with $h((u, v)) = 1$ if and only if $u, v \in V_j$ for some $j$. As usual, $Y$ may induce a non-transitive relation (e.g., we could have $Y((u, v)) = Y((v, z)) = 1$, and $Y((u, z)) = 0$). In what follows, we will drop the double parentheses. Also, we will abuse notation by viewing $h$ as both an equivalence relation and a disjoint cover $\{V_1, \ldots, V_k\}$ of $V$. We take $\mathcal{D}$ to be the uniform measure on $\mathcal{X}$. The error of $h \in \mathcal{H}$ is given as

$$\text{er}_{\mathcal{D}}(h) = N^{-1} \sum_{(u,v) \in \mathcal{X}} \mathbf{1}_{h(u,v) \neq Y(u,v)},$$

where $N = |\mathcal{X}| = n(n-1)$. Recall that $\nu$ denotes the hypothesis class noise rate $\inf_{h \in \mathcal{H}} \text{er}_{\mathcal{D}}(h)$. We define $\text{cost}_{u,v}(h)$ to be the contribution $N^{-1}\mathbf{1}_{h(u,v) \neq Y(u,v)}$ of $(u, v) \in \mathcal{X}$ to $\text{er}_{\mathcal{D}}$. The distance $\text{dist}(h, h')$ is given as

$$\text{dist}(h, h') = N^{-1} \sum_{(u,v) \in \mathcal{X}} \mathbf{1}_{h(u,v) \neq h'(u,v)}.$$

Viewing the problem as a graph, $(V, Y)$. We depict a “must link” label $Y(u) = 1$ as thick-directed on both sides of the edge, and a “cannot link” label as a dashed edge. A solution $h \in \mathcal{H}$ for the case $k = 2$ is depicted as the disjoint cover of $V$ corresponding to the dark squares.

Figure 5.1: Depicting the problem as a graph.

Sometimes it is convenient to examine the problem by using the graph $G = (V, Y)$. Here we think of the elements $V$ as nodes and consider an edge
iff $Y(u, v) = 1$. (Another possibility is to define $G$ as a full edge-weighted graph.)

The error of $h$, a disjoint cover of $V$, is the sum of edges that cross partition-subsets plus the sum of absent edges within each subset. For example, consider the graph and clustering $h$ depicted in Figure 5.1. The cost of this clustering is $\text{er}_D(h) = 1/N = 1/20$. Observe that any solution incurs a non-zero error. Thus, the depicted partition is optimal.

The active learner is given $V$, $k$, and an oracle access to $Y(\cdot)$. Recall that the corresponding correlation clustering’s PTAS is not query efficient: It requires knowledge of $Y$ on all $\mathcal{X}$. Here, the learner begins only with the knowledge of $V$ and has to pay in order for the edges $Y$ to be revealed. What we are interested in is the ERM aspects of the problem, that is, the query complexity required for achieving low error. As before, we study the query complexity required for achieving a $(1+\varepsilon)$ approximation for the noise-rate $\nu$. From a learning theoretical perspective, we want to find the best $k$-clustering that explains $V$, using as few queries as possible into $\mathcal{X}$.

### 5.2 Passive Learning Is Not Useful

It is not difficult to find examples for which sampling pairwise constraints uniformly at random will fail to provide meaningful results. First, notice that VC bounds are useless for our context. We note that the VC dimension of $\mathcal{H}$ is $\Theta(n)$, as we prove in the next section. Using arguments similar to those of Section 4.2, we get that VC theory arguments tell us that to achieve an additive error of $\varepsilon$ with a probability of at least $1 - \exp(-n)$, we would have to sample $O(\varepsilon^{-2}n \log n)$ pairs uniformly at random with repetitions (and optimize the empirical estimation over $\mathcal{H}$). However, an additive error of $\varepsilon$ means that the estimated clustering disagrees with the optimal solution on $\varepsilon N$ constraints. Thus, an item is on average either separated or in conflict with $\Theta(\varepsilon n)$ more elements than in the optimal clustering. In practice, we would like to keep this error constant and will have to require an additive error of $\varepsilon/n$. In such a case, the corresponding VC sample bound turns out to be larger than $N$.

As a second example, consider the case in which $V = V' \cup \{u\}$: the set $V'$ can be perfectly $k$-clustered, and $u$ must link only with a single element $v' \in V'$ (i.e., $Y(u, v') = 1$ and $Y(u, v) = 0$ for all $V' \setminus \{v'\}$). Clearly, $\nu = N^{-1}$ here. The only way we can ensure a $(1 + \varepsilon)\nu$ competitive solution is to establish that the sample contains $(u, v')$. The probability that we will “hit” this edge when sampling uniformly at random (e.g., with repetitions) with sample size $o(N)$ is exponentially low, if we use, e.g., Chernoff bound, described in Appendix A.
5.3 Disagreement Coefficient Arguments Are Not Sufficient for Effective Active Learning

Let us try to solve the problem using disagreement coefficient \( \theta \) and VC bound \( d \) arguments only by applying, for example, the SRRRA method with the disagreement-SRRA construction suggested in Section 3.2.1.

We argue that the uniform disagreement coefficient of \( \mathcal{H} \) is \( \Omega(n) \). Pick any \( h \in \mathcal{H} \) with corresponding partitioning \( \{V_1, \ldots, V_k\} \). Consider the partition obtained by moving an element \( u \in V \) from its current part \( V_j \) to some other part \( V_{j'} \) for \( j' \neq j \). In other words, consider the clustering \( h' \in \mathcal{H} \) given by \( \{V_j \cup \{u\}, V_j \setminus \{u\}\} \cup \bigcup_{i \notin \{j,j'\}} \{V_i\} \). Observe that \( \text{dist}(h, h') = O(1/\max\{|V_i|\}) \), which for a fixed \( k = o(n) \) matches \( O(1/n) \). On the other hand, for any \( v \in V \) and for any \( u \in V \), there is a choice of \( j' \) such that \( h \) and \( h' \) obtained as above would disagree on \( (u, v) \in \mathcal{X} \). Hence, \( \mathbb{P}_{\mathcal{D}_X} [\text{DIS}(\mathcal{B}(h, O(1/n)))] = 1 \), and we have \( \theta = \sup_{h \in \mathcal{H}} \sup_{r > 0} (\mathbb{P}_{\mathcal{D}_X} [\text{DIS}(\mathcal{B}(h, O(1/n)))] / r) \geq O(n) \).

It is also apparent that the VC dimension of \( \mathcal{H} \) is \( \Theta(n) \). Assume without loss of generality that \( n \) is even and there exists an arbitrary indexing such that \( V = (u_1, u_2, u_3, \ldots, u_n) \). Clearly, the set \( \{(u_1, u_2), (u_3, u_4), \ldots, (u_{n-1}, u_n)\} \) is shattered in \( \mathcal{H} \) (as long as \( k \geq 2 \), of course).\(^1\) On the other hand, for any set \( S \subseteq \mathcal{X} \) of size \( n \) there exists a labeling \( Y(\cdot) \) that defines an undirected cycle on the elements of \( V \). Clearly, the edges of a cycle cannot be shattered by functions in \( \mathcal{H} \) because if \( h(u_1, u_2) = h(u_2, u_3) = \cdots = h(u_{\ell-1}, u_\ell) = 1 \) for \( h \in \mathcal{H} \), then also \( h(u_1, u_\ell) = 1 \).

Using Corollary 3.2, we conclude that we need \( \Omega(n^2) \) preference labels to obtain an \((\varepsilon, \mu)\)-SRRRA for any meaningful pair \((\varepsilon, \mu)\). This is useless, because the cardinality of \( \mathcal{X} \) is \( O(n^2) \). Similarly to the results in the discussion at the end of Section 4.3, this can be improved by using Remark 2.1 to \( \Omega(n\nu^{-1}) \), which tends to be quadratic in \( n \) as \( \nu \) becomes smaller. Next, we show how to construct more useful SRRAs for the problem, for an arbitrarily small \( \nu \).

5.4 Better SRRRA for Semi-Supervised \( k \)-Clustering

Fix \( h \in \mathcal{H} \), with \( h = \{V_1, \ldots, V_k\} \) (we allow empty \( V_i \) subsets). Order the \( V_i \) subsets with respect to their sizes so that \( |V_1| \geq |V_2| \geq \cdots \geq |V_k| \). We construct an \( \varepsilon \)-SRRRA with respect to \( h \) as follows. For each cluster \( V_i \in h \) and for each element \( u \in V_i \) we draw \( k - i + 1 \) independent samples \( S_{ui}, S_{u(i+1)}, \ldots, S_{uk} \) as follows. Each sample \( S_{uj} \) is a subset of \( V_j \) of size \( q \) (to be defined below), chosen

\(^1\)Notice that this set is a perfect matching: Any vertex is incident exactly to one edge within the set. Also observe that \( \mathcal{H} \) shatters any perfect matching.
uniformly with repetitions from $V_j$, where
\[ q = c_2 \max \{ \varepsilon^{-2}k^2, \varepsilon^{-3}k \} \log n \tag{5.1} \]
for some global $c_2 > 0$. Note that the collection of pairs \( \{(u, v) \in X : v \in S_{ui} \text{ for some } i\} \) is, roughly speaking, biased in such a way that pairs containing elements in smaller clusters (with respect to $h$) are more likely to be selected.

We define our estimator $f$ to be for any $h' \in \mathcal{H}$,
\[ f(h') = \sum_{i=1}^{k} \frac{|V_i|}{q} \sum_{u \in V_i} \sum_{v \in S_{ui}} f_{u,v}(h') + 2 \sum_{i=1}^{k} \sum_{u \in V_i} \sum_{j=i+1}^{k} \frac{|V_j|}{q} \sum_{v \in S_{uj}} f_{u,v}(h') , \tag{5.2} \]
where $f_{u,v}(h') = \text{cost}_{u,v}(h') - \text{cost}_{u,v}(h)$ and $\text{cost}_{u,v}(h) = N^{-1}1_{h(u,v) \neq Y(u,v)}$. Note that the summations over $S_{ui}$ above take into account a multiplicity of elements in the multiset $S_{ui}$.

**Theorem 5.1** With a probability of at least $1 - n^{-3}$, the function $f$ is an $\varepsilon$-SRRRA with respect to $h$.

### 5.4.1 Setup

Consider another $k$-clustering $h' \in \mathcal{H}$ with corresponding partitioning \( \{V'_1, \ldots, V'_k\} \) of $V$. We can write $\text{dist}(h,h')$ as
\[ \text{dist}(h,h') = \sum_{(u,v) \in X} \text{dist}_{u,v}(h,h') \]
where $\text{dist}_{u,v}(h,h') = N^{-1}(1_{h'(u,v)=1}1_{h(u,v)=0} + 1_{h(u,v)=1}1_{h'(u,v)=0})$.

Let $n_i$ denote $|V_i|$ and recall that $n_1 \geq n_2 \geq \cdots \geq n_k$. In what follows, we remove the subscript in $\text{reg}_h$ and rename it $\text{reg}$ ($h$ is held fixed). The function $\text{reg}(h')$ will now be written as:
\[ \text{reg}(h') = \sum_{i=1}^{k} \sum_{u \in V_i} \left( \sum_{v \in V_i \setminus \{u\}} \text{reg}_{u,v}(h') + 2 \sum_{j=i+1}^{k} \sum_{v \in V_j} \text{reg}_{u,v}(h') \right) , \tag{5.3} \]
where
\[ \text{reg}_{u,v}(h') = \text{cost}_{u,v}(h') - \text{cost}_{u,v}(h) . \]

Clearly, for each $h'$, it holds that $f(h')$ from (5.2) is an unbiased estimator of $\text{reg}(h')$. We now analyze its error. For each $i, j \in [k]$, let $V_{ij}$ denote $V_i \cap V'_j$. This captures exactly the set of elements in the $i$th cluster in $h$ and the $j$th cluster in $h'$. The distance $\text{dist}(h,h')$ can be written as follows:
\[ \text{dist}(h,h') = N^{-1} \left( \sum_{i=1}^{k} \sum_{j=1}^{k} |V_{ij} \times (V_i \setminus V_{ij})| + 2 \sum_{j=1}^{k} \sum_{1 \leq i_1 < i_2 \leq k} |V_{i_1j} \times V_{i_2j}| \right) . \tag{5.4} \]
We call each Cartesian set product in (5.4) a \textit{distance-contributing rectangle}. Note that unless a pair \((u, v)\) appears in one of the distance-contributing rectangles, we have \(\text{reg}_{u,v}(h') = f_{u,v}(h') = 0\). Hence, we can decompose \(\text{reg}(h')\) and \(f(h')\) in correspondence with the distance-contributing rectangles, as follows:

\[
\text{reg}(h') = \sum_{i=1}^{k} \sum_{j=1}^{k} G_{i,j}(h') + 2 \sum_{j=1}^{k} \sum_{1 \leq i_1 < i_2 \leq k} G_{i_1,i_2,j}(h') \tag{5.5}
\]

\[
f(h') = \sum_{i=1}^{k} \sum_{j=1}^{k} F_{i,j}(h') + 2 \sum_{j=1}^{k} \sum_{1 \leq i_1 < i_2 \leq k} F_{i_1,i_2,j}(h') \tag{5.6}
\]

where

\[
G_{i,j}(h') = \sum_{u \in V_{ij}} \sum_{v \in V_i \setminus V_j} \text{reg}_{u,v}(h') \tag{5.7}
\]

\[
F_{i,j}(h') = \frac{|V_i|}{q} \sum_{u \in V_{ij}} \sum_{v \in (V_i \setminus V_j) \cup S_{ui}} f_{u,v}(h') \tag{5.8}
\]

\[
G_{i_1,i_2,j}(h') = \sum_{u \in V_{i_1j}} \sum_{v \in V_{i_2j}} \text{reg}_{u,v}(h') \tag{5.9}
\]

\[
F_{i_1,i_2,j}(h') = \frac{|V_{i_2}|}{q} \sum_{u \in V_{i_1j}} \sum_{v \in (V_{i_2j} \cap S_{ui_2})} f_{u,v}(h') \tag{5.10}
\]

(Note that \(S_{ui}\) are multisets, and the inner sums in (5.8) and (5.10) may count elements multiple times.)

\section*{5.4.2 Simple Case}

\textbf{Lemma 5.1} With a probability of at least \(1 - n^{-3}\), the following holds simultaneously for all \(h' \in \mathcal{H}\) and all \(i, j \in [k]\):

\[
|G_{i,j}(h') - F_{i,j}(h')| \leq \varepsilon N^{-1} \cdot |V_{ij} \times (V_i \setminus V_j)|. \tag{5.11}
\]

\textbf{Proof} The predicate (5.11) (for a given \(i, j\)) depends only on the set \(V_{ij} = V_i \cap V_j\). Given a subset \(B \subseteq V_i\), we say that \(h' (i, j)\)-realizes \(B\) if \(V_{ij} = B\).

Now fix \(i, j\), and \(B \subseteq V_i\). Assume \(h' (i, j)\)-realizes \(B\). Let \(\beta = |B|\) and \(\gamma = |V_i|\). Consider the random variable \(F_{i,j}(h')\). Think of the sample \(S_{ui}\) as a sequence \(S_{ui}(1), \ldots, S_{ui}(q)\), where each \(S_{ui}(s)\) is chosen uniformly at random from \(V_i\) for \(s = 1, \ldots, q\). We can now rewrite \(F_{i,j}(h')\) as follows:

\[
F_{i,j}(h') = \frac{\gamma}{q} \sum_{u \in B} \sum_{s=1}^{q} Z(S_{ui}(s))',
\]
where

\[ Z(v) = \begin{cases} f_{u,v}(h') & v \in V_i \setminus V_{ij} \\ 0 & \text{otherwise} \end{cases} \]

For all \( s = 1, \ldots q \) the random variable \( Z(S_{ui}(s)) \) is bounded by \( 2N^{-1} \) almost surely, and its moments satisfy:

\[
E[Z(S_{ui}(s))] = \frac{1}{\gamma} \sum_{v \in (V_i \setminus V_{ij})} f_{u,v}(h'), \\
E\left[ Z(S_{ui}(s))^2 \right] \leq \frac{4N^{-2}(\gamma - \beta)}{\gamma}. \tag{5.12}
\]

We are now ready to apply Bernstein inequality using the normalized version of \( Z(v) \). We have

\[
P[|F_{i,j}(h') - G_{i,j}(h')| \geq t] \leq \exp \left\{ -\frac{t^2}{2} \sum_{i=1}^q \sum_{u \in V_{ij}} \frac{\gamma^2}{q^2} \left( \frac{4N^{-2}(\gamma - \beta)}{\gamma} - \frac{1}{\gamma} \sum_{v \in (V_i \setminus V_{ij})} f_{u,v}(h') \right) + \frac{Mt}{3} \right\}
\]

\[
\leq \exp \left\{ -\frac{t^2}{2} \sum_{i=1}^q \sum_{u \in V_{ij}} \frac{\gamma^2}{q^2} \frac{4N^{-2}(\gamma - \beta)}{\gamma} + \frac{Mt}{3} \right\}
\]

\[
\leq \exp \left\{ -\frac{t^2}{2} \frac{4N^{-2}\beta(\gamma - \beta)\gamma}{q} + \frac{Mt}{3} \right\}
\]

\[
\leq \exp \left\{ -\frac{t^2}{2} \frac{4N^{-2}\beta(\gamma - \beta)\gamma}{q} + \frac{2N^{-1}t}{3q} \right\}.
\]

From this we conclude that for any \( t \leq 6N^{-1}\beta(\gamma - \beta) \),

\[
P[|F_{i,j}(h') - G_{i,j}(h')| \geq t] \leq \exp \left\{ -\frac{qt^2}{16\beta(\gamma - \beta)N^{-2}} \right\}.
\]

Plugging in \( t = \varepsilon N^{-1}\beta(\gamma - \beta) \), we conclude

\[
P[|F_{i,j}(h') - G_{i,j}(h')| \geq \varepsilon N^{-1}\beta(\gamma - \beta)] \leq \exp \left\{ -\frac{q\varepsilon^2\beta(\gamma - \beta)}{16\gamma} \right\}.
\]

Now note that the number of possible sets \( B \subseteq V_i \) of size \( \beta \) is at most \( n^{\min\{\beta, \gamma - \beta\}} \). Using union bound and recalling our choice of \( q \), the lemma follows. \( \blacksquare \)
5.4.3 More Involved Case

Proving the following is more involved.

Lemma 5.2 With a probability of at least $1 - n^{-3}$, the following holds uniformly for all $h' \in \mathcal{H}$ and for all $i_1, i_2, j \in [k]$ with $i_1 < i_2$:

$$|F_{i_1, i_2, j}(h') - G_{i_1, i_2, j}(h')| \leq \varepsilon N^{-1} \max \left\{ \frac{|V_{i_1} \times (V_{i_2} \setminus V_{i_2,j})|}{k}, \frac{|V_{i_2} \times (V_{i_1} \setminus V_{i_2,j})|}{k} \right\} \tag{5.13}$$

Proof The predicate (5.13) (for a given $i_1, i_2, j$) depends only on the sets $V_{i_1} = V_{i_1} \cap V_{i_1}'$ and $V_{i_2,j} = V_{i_2} \cap V_{i_2}'$. Given subsets $B_1 \subseteq V_{i_1}$ and $B_2 \subseteq V_{i_2}$, we say that $h'(i_1, i_2, j)$-realizes $(B_1, B_2)$ if $V_{i_1} = B_1$ and $V_{i_2} = B_2$.

We now fix $i_1 < i_2, j$ and $B_1 \subseteq V_{i_1}, B_2 \subseteq V_{i_2}$. Assume $h'(i_1, i_2, j)$-realizes $(B_1, B_2)$. For brevity, denote $\beta_i = |B_i|$ and $\gamma_i = |V_i|$ for $i = 1, 2$. Using the Bernstein inequality as in Lemma 5.1, we conclude the following two inequalities:

$$\mathbb{P} \left[ |G_{i_1, i_2, j}(h') - F_{i_1, i_2, j}(h')| > t \right] \leq \exp \left\{ -\frac{c_3 t^2 q}{\beta_1 \beta_2 \gamma_2 N^{-2}} \right\} \tag{5.14}$$

for any $t$ in the range $[0, N^{-1} \beta_1 \beta_2]$ and some global $c_3 > 0$. For $t$ in the range $(N^{-1} \beta_1 \beta_2, \infty)$ and some global $c_4$ we have

$$\mathbb{P} \left[ |G_{i_1, i_2, j}(h') - F_{i_1, i_2, j}(h')| > t \right] \leq \exp \left\{ -\frac{c_4 t q}{\gamma_2 N^{-1}} \right\}. \tag{5.15}$$

Consider the following three cases.

1. $\beta_1 \beta_2 \geq \max\{\beta_1(\gamma_1 - \beta_1)/k, \beta_2(\gamma_2 - \beta_2)/k\}$. Hence, $\beta_1 \geq (\gamma_2 - \beta_2)/k, \beta_2 \geq (\gamma_1 - \beta_1)/k$. In this case, we can plug $t = \varepsilon N^{-1} \beta_1 \beta_2$ in (5.14) to get

$$\mathbb{P} \left[ |G_{i_1, i_2, j}(h') - F_{i_1, i_2, j}(h')| > \varepsilon N^{-1} \beta_1 \beta_2 \right] \leq \exp \left\{ -\frac{c_3 \varepsilon^2 \beta_2 q}{\gamma_2} \right\}. \tag{5.16}$$

Consider two subcases: (i) If $\beta_2 \geq \gamma_2/2$, then the RHS of (5.16) is at most $\exp \left\{ -\frac{c_3 \varepsilon^2 \beta_2 q}{2k} \right\}$. The number of possible subsets $B_1, B_2$ of sizes $\beta_1, \beta_2$, respectively, is clearly at most $n^{\beta_1 + (\gamma_2 - \beta_2)} \leq n^{\beta_1 + k \beta_2}$. Therefore, as long as $q = O(\varepsilon^{-2} k^2 \log n)$, then with a probability of at least $1 - n^{-6}$, this case is taken care of in the following sense: Simultaneously for all $j, i_1 < i_2$, all possible $\beta_1 \leq \gamma_1 = |V_{i_1}|, \beta_2 \leq \gamma_2 = |V_{i_2}|$ satisfying the assumptions and for all $B_1 \subseteq V_{i_1}, B_2 \subseteq V_{i_2}$ of sizes $\beta_1, \beta_2$, respectively, and for all $h' (i_1, i_2, j)$-realizing $(B_1, B_2)$ we have that $|G_{i_1, i_2, j}(h') - F_{i_1, i_2, j}(h')| \leq \varepsilon \beta_1 \beta_2$.

(ii) If $\beta_2 < \gamma_2/2$, then by our assumption, $\beta_1 \geq \gamma_2/2k$. Hence the RHS of (5.16) is at most $\exp \left\{ -\frac{c_3 \varepsilon^2 \beta_2 q}{2k} \right\}$. The number of sets $B_1, B_2$ of sizes $\beta_1, \beta_2$
respectively is clearly at most \( n^{(\gamma_1 - \beta_1) + \beta_2} \leq n^{\beta_2(1+k)} \). Therefore, as long as \( q = O(\varepsilon^{-2}k^2 \log n) \), then with a probability of at least \( 1 - n^{-6} \), this case is taken care of in the following sense: Simultaneously for all \( j, i_1 < i_2 \), all possible \( \beta_1 < \gamma_1 = |V_{i_1}|, \beta_2 < \gamma_2 = |V_{i_2}| \) satisfying the assumptions and for all \( B_1 \subseteq V_{i_1}, B_2 \subseteq V_{i_2} \) of sizes \( \beta_1, \beta_2 \), respectively, and for all \( h' (i_1, i_2, j) \)-realizing \((B_1, B_2)\), we have that \( |G_{i_1, i_2, j}(h') - F_{i_1, i_2, j}(h')| \leq \varepsilon \beta_1 \beta_2 \). The requirement \( q = O(\varepsilon^{-2}k^2 \log n) \) is satisfied by our choice, Equation (5.1).

2. \( \beta_2 (\gamma_2 - \beta_2)/k \geq \max\{\beta_1 \beta_2, \beta_1 (\gamma_1 - \beta_1)/k\} \). We consider two subcases.

(a) \( \varepsilon \beta_2 (\gamma_2 - \beta_2)/k \leq \beta_1 \beta_2 \). Using (5.14), we get

\[
\Pr[|G_{i_1, i_2, j}(h') - F_{i_1, i_2, j}(h')| > \varepsilon N^{-1} \beta_2 (\gamma_2 - \beta_2)/k] \leq \exp \left\{ -\frac{c_3 \varepsilon^2 \beta_2 (\gamma_2 - \beta_2)^2 q}{k^2 \beta_1 \gamma_2} \right\}
\]

(5.17)

Again, consider two subcases. (i) \( \beta_2 \leq \gamma_2/2 \). In this case we conclude from Equation (5.17)

\[
\Pr[|G_{i_1, i_2, j}(h') - F_{i_1, i_2, j}(h')| > \varepsilon N^{-1} \beta_2 (\gamma_2 - \beta_2)/k] \leq \exp \left\{ -\frac{c_3 \varepsilon^2 \beta_2 \gamma_2 q}{4k^2 \beta_1} \right\}
\]

(5.18)

Now, note that by our assumption

\[
\beta_1 \leq (\gamma_2 - \beta_2)/k \leq \gamma_2/k \leq \gamma_1/k , \quad (5.19)
\]

the last inequality is in virtue of our assumption \( \gamma_1 \geq \gamma_2 \). Also by assumption,

\[
\beta_1 \leq \beta_2 (\gamma_2 - \beta_2)/(\gamma_1 - \beta_1) \leq \beta_2 \gamma_2/(\gamma_1 - \beta_1) . \quad (5.20)
\]

Plugging (5.19) in the RHS of (5.20), we conclude that

\[
\beta_1 \leq \beta_2 \gamma_2/(\gamma_1 (1 - 1/k)) \leq 2 \beta_2 \gamma_2/\gamma_1 \leq 2 \beta_2 .
\]

From here we conclude that the RHS of (5.18) is at most \( \exp \left\{ -\frac{c_3 \varepsilon^2 q}{8k^2} \right\} \).

The number of sets \( B_1, B_2 \) of sizes \( \beta_1, \beta_2 \), respectively, is clearly at most \( n^{\beta_1 + \beta_2} \leq n^{2 \beta_2 + \beta_2} \leq n^{3 \beta_2} \). Hence, as long as \( q = O(\varepsilon^{-2}k^2 \log n) \) (satisfied by our assumption), with a probability of at least \( 1 - n^{-6} \) simultaneously for all \( j, i_1 < i_2 \), all possible \( \beta_1 \leq \gamma_1 = |V_{i_1}|, \beta_2 \leq \gamma_2 = |V_{i_2}| \) satisfying the assumptions and for all \( B_1 \subseteq V_{i_1}, B_2 \subseteq V_{i_2} \) of sizes \( \beta_1, \beta_2 \), respectively, and for all \( h' (i_1, i_2, j) \)-realizing \((B_1, B_2)\), we have that \( |G_{i_1, i_2, j}(h') - F_{i_1, i_2, j}(h')| \leq \varepsilon \beta_2 (\gamma_2 - \beta_2)/k \). In the second subcase
(ii) $\beta_2 > \gamma_2/2$; the RHS of (5.17) is at most $\exp \left\{ -\frac{2c_3 \varepsilon^2 (\gamma_2 - \beta_2)^2 q}{k^2 \beta_1} \right\}$.

By our assumption, $(\gamma_2 - \beta_2)/(k\beta_1) \geq 1$; hence, this is at most $\exp \left\{ -\frac{2c_3 \varepsilon^2 (\gamma_2 - \beta_2)}{k} \right\}$. The number of sets $B_1, B_2$ of sizes $\beta_1, \beta_2$, respectively, is clearly at most $n^{\beta_1 + (\gamma_2 - \beta_2)} \leq n^{\gamma_2 - \beta_2} \leq n^2 (\gamma_2 - \beta_2)$.

Therefore, as long as $q = O(\varepsilon^{-2} k \log n)$ (satisfied by our assumption), then with a probability of at least $1 - n^{-6}$, using a similar counting and union bound argument as above, this case is taken care of in the sense that: $|G_{i_1, i_2, j}(h') - G_{i_1, i_2, j}(h')| \leq \varepsilon \beta_2 (2 - \beta_2) / k$.

(b) $\varepsilon \beta_2 (2 - \beta_2) / k > \beta_1 \beta_2$. We now use (5.15) to conclude

$$P[|G_{i_1, i_2, j}(h') - F_{i_1, i_2, j}(h')| > \varepsilon N^{-1} \beta_2 (2 - \beta_2) / k] \leq \exp \left\{ -\frac{c_4 \varepsilon \beta_2 (2 - \beta_2) q}{k^2 \beta_2} \right\}.$$  

(5.21)

Again, we consider the cases (i) $\beta_2 \leq \gamma_2/2$ and (ii) $\beta_2 > \gamma_2/2$ as above. In (i), we get that the RHS of (5.21) is at most $\exp \left\{ -\frac{c_4 \varepsilon \beta_2 q}{2k^2} \right\}$. Now notice that by our assumptions,

$$\beta_1 < \varepsilon (2 - \beta_2) / k \leq \gamma_2 / 2 \leq \gamma_1 / 2.$$  

(5.22)

Also by our assumptions, $\beta_1 < \beta_2 (2 - \beta_2) / (\gamma_1 - 1)$, which by (5.22) is at most $2\beta_2 \gamma_2 / \gamma_1 \leq 2\beta_2$. Hence, the number of possibilities for $B_1, B_2$ is at most $n^{\beta_1 + \beta_2} \leq n^{3\beta_2}$. In (ii), we get that the RHS of (5.21) is at most $\exp \left\{ -\frac{c_4 \varepsilon (2 - \beta_2) q}{2k} \right\}$, and the number of possibilities for $B_1, B_2$ is at most $n^{\beta_1 + (\gamma_2 - \beta_2)}$, which is bounded by $n^2 (\gamma_2 - \beta_2)$, following from our assumptions. For both (i) and (ii), taking $q = O(\varepsilon^{-1} k \log n)$ ensures that with a probability of at least $1 - n^{-6}$, using a similar counting and union bounding argument as above, case (b) is taken care of in the sense that: $|G_{i_1, i_2, j}(h') - F_{i_1, i_2, j}(h')| \leq \varepsilon N^{-1} \beta_2 (2 - \beta_2) / k$.

3. $\beta_1 (\gamma_1 - 1)/k \geq \max\{\beta_1 \beta_2, \beta_2 (2 - \beta_2)/k\}$. We consider two subcases.

(a) $\varepsilon \beta_1 (\gamma_1 - 1)/k \leq \beta_1 \beta_2$. Using (5.14), we get

$$P[|G_{i_1, i_2, j}(h') - F_{i_1, i_2, j}(h')| > \varepsilon N^{-1} \beta_1 (\gamma_1 - 1)/k] \leq \exp \left\{ -\frac{c_3 \varepsilon^2 \beta_1 (\gamma_1 - 1)^2 q}{k^2 \beta_2 \gamma_2} \right\}.$$  

(5.23)

As before, consider case in which (i) $\beta_2 \leq \gamma_2/2$ and (ii) $\beta_2 > \gamma_2/2$. For case (i), we use the fact that $\beta_1 (\gamma_1 - 1) \geq \beta_2 (\gamma_2 - 2)$ by assumption and notice that the RHS of (5.23) is at most $\exp \left\{ -\frac{c_3 \varepsilon^2 \beta_2 (\gamma_2 - \beta_2) (\gamma_1 - 1)^2 q}{k^2 \beta_2 \gamma_2} \right\}$. Hence, this is at most $\exp \left\{ -\frac{c_3 \varepsilon^2 (\gamma_1 - 1)^2 q}{2k^2} \right\}$. The number of possibilities of $B_1, B_2$ of sizes $\beta_1, \beta_2$ is clearly at most $n^{(\gamma_1 - 1) + \beta_2} \leq n^{(\gamma_1 - 1) + \beta_2} / k \leq n^2 (\gamma_1 - 1)$.
From this, we conclude that $q = O(\varepsilon^{-2}k^2 \log n)$ is sufficient for this case. For case (ii), we bound the RHS of (5.23) by

$$\exp \left\{ - \frac{c_3 \varepsilon^2 \beta_1 (\gamma_1 - \beta_1)^2 q}{2k^2 \beta_2^2} \right\}. $$

Using the assumption that $(\gamma_1 - \beta_1)/\beta_2 \geq k$, the latter expression is upper bounded by $\exp \left\{ - \frac{c_3 \varepsilon^2 \beta_1 q}{2} \right\}$. Again, by our assumptions,

$$\beta_1 \geq \beta_2 (\gamma_2 - \beta_2)/(\gamma_1 - \beta_1) \geq (\varepsilon (\gamma_1 - \beta_1)/k)(\gamma_2 - \beta_2)/(\gamma_1 - \beta_1) = \varepsilon (\gamma_2 - \beta_2)/k. \tag{5.24}$$

The number of possibilities of $B_1, B_2$ of sizes $\beta_1, \beta_2$ is clearly at most $n^{\beta_1 + (\gamma_2 - \beta_2)}$, which by (5.24) is bounded by $n^{\beta_1 + k \beta_1/\varepsilon} \leq n^{2k \beta_1/\varepsilon}$. From this we conclude that as long as $q = O(\varepsilon^{-3}k \log n)$ (satisfied by our choice), this case is taken care of in the sense repeatedly explained above.

(b) $\varepsilon \beta_1 (\gamma_1 - \beta_1)/k > \beta_1 \beta_2$. Using (5.15), we get

$$\mathbb{P} \left[ |G_{i_1,i_2,j}(h') - F_{i_1,i_2,j}(h')| > \varepsilon N^{-1} \beta_1 (\gamma_1 - \beta_1)/k \right] \leq \exp \left\{ - \frac{c_4 \varepsilon \beta_1 (\gamma_1 - \beta_1) q}{k \gamma_2} \right\}. \tag{5.25}$$

We consider two subcases, (i) $\beta_1 \leq \gamma_1/2$ and (ii) $\beta_1 > \gamma_1/2$. In case (i), we have that

$$\frac{\beta_1 (\gamma_1 - \beta_1)}{\gamma_2} = \frac{1}{2} \frac{\beta_1 (\gamma_1 - \beta_1)}{\gamma_2} + \frac{1}{2} \frac{\beta_1 (\gamma_1 - \beta_1)}{\gamma_2} \geq \frac{1}{2} \frac{\beta_1 \gamma_1}{\gamma_2} + \frac{1}{2} \frac{\beta_2 (\gamma_2 - \beta_2)}{\gamma_2} \geq \beta_1/4 + \min \{\beta_2, \gamma_2 - \beta_2\}/2. \tag{5.26}$$

(The last step used $\gamma_1 \geq \gamma_2$.) Hence, the RHS of (5.25) is bounded from above by

$$\exp \left\{ - \frac{c_4 \varepsilon q (\beta_1/4 + \min \{\beta_2, \gamma_2 - \beta_2\}/2)}{k} \right\}. \tag{5.27}$$

The number of possibilities of $B_1, B_2$ of sizes $\beta_1, \beta_2$ is clearly at most $n^{\beta_1 + \min \{\beta_2, \gamma_2 - \beta_2\}}$; hence, as long as $q = O(\varepsilon^{-1}k \log n)$ (satisfied by our choice), this case is taken care of in the sense repeatedly explained above. In case (ii), we can upper-bound the RHS of (5.25) by

$$\exp \left\{ - \frac{c_4 \varepsilon \beta_1 (\gamma_1 - \beta_1) q}{2k \gamma_2} \right\} \geq \exp \left\{ - \frac{c_4 \varepsilon (\gamma_1 - \beta_1) q}{2k} \right\}. \tag{5.28}$$

The number of possibilities of $B_1, B_2$ of sizes $\beta_1, \beta_2$ is clearly at most $n^{(\gamma_1 - \beta_1)/k}$, which, using our assumptions, is bounded from above by $n^{(\gamma_1 - \beta_1)/k} \leq n^{2(\gamma_1 - \beta_1)}$. Hence, as long as $q = O(\varepsilon^{-1}k \log n)$, this case is taken care of in the sense repeatedly explained above.
This concludes the proof of the lemma.

Consequently, we get the following:

**Lemma 5.3** with a probability of at least $1 - n^{-3}$, the following holds simultaneously for all $k$-clusterings $\mathcal{H}$:

$$|\text{reg}(h') - f(h')| \leq 5\varepsilon \text{dist}(h', h).$$

**Proof** By the triangle inequality,

$$|\text{reg}(h') - f(h')| \leq \sum_{i=1}^{k} \sum_{j=1}^{k} |G_{i,j}(h') - F_{i,j}(h')| + 2 \sum_{j=1}^{k} \sum_{1 \leq i_1 < i_2 \leq k} |G_{i_1,i_2,j}(h') - F_{i_1,i_2,j}(h')|$$

(5.26)

Using (5.5)-(5.6), then Lemmas 5.1-5.2 (assuming the success of a high-probability event), and rearranging the sum and finally using (5.4), we get:

$$|\text{reg}(h') - f(h')| \leq \sum_{i=1}^{k} \sum_{j=1}^{k} \varepsilon N^{-1} |V_{ij} \times (V_i \setminus V_{ij})|$$

$$+ 2\varepsilon N^{-1} \sum_{j=1}^{k} \sum_{i_1 < i_2} (|V_{i_1j} \times V_{i_2j}| + k^{-1}|V_{i_1j} \times (V_i \setminus V_{i_1j})| + k^{-1}|V_{i_2j} \times (V_i \setminus V_{i_2j})|)

\leq \sum_{i=1}^{k} \sum_{j=1}^{k} \varepsilon N^{-1} |V_{ij} \times (V_i \setminus V_{ij})| + 2\varepsilon N^{-1} \sum_{j=1}^{k} \sum_{i_1 < i_2} |V_{i_1j} \times V_{i_2j}|

+ 2\varepsilon N^{-1} \sum_{j=1}^{k} \sum_{i_1 < i_2} k^{-1}|V_{i_1j} \times (V_i \setminus V_{i_1j})| + 2\varepsilon N^{-1} \sum_{j=1}^{k} \sum_{i_1 < i_2} k^{-1}|V_{i_2j} \times (V_i \setminus V_{i_2j})|

\leq \sum_{i=1}^{k} \sum_{j=1}^{k} \varepsilon N^{-1} |V_{ij} \times (V_i \setminus V_{ij})| + 2\varepsilon N^{-1} \sum_{j=1}^{k} \sum_{i_1 < i_2} |V_{i_1j} \times V_{i_2j}|

+ 2\varepsilon N^{-1} \sum_{j=1}^{k} \sum_{i_1 = 1}^{k} kk^{-1}|V_{i_1j} \times (V_i \setminus V_{i_1j})| + 2\varepsilon N^{-1} \sum_{j=1}^{k} \sum_{i_2 = 1}^{k} k^{-1}|V_{i_2j} \times (V_i \setminus V_{i_2j})|

\leq 5\varepsilon N^{-1} \sum_{i=1}^{k} \sum_{j=1}^{k} |V_{ij} \times (V_i \setminus V_{ij})| + 2\varepsilon N^{-1} \sum_{j=1}^{k} \sum_{i_1 < i_2} |V_{i_1j} \times V_{i_2j}|

\leq 5\varepsilon \text{dist}(h, h'),$$

as required.  

\[75\]
Algorithm 3 SRRA for Semi-Supervised $k$-Clustering

**Require:** $V$, $k$, $\mathcal{H}$, a pivot $h = \{V_i\}_{i=1}^k \in \mathcal{H}$, estimation parameter $\epsilon \in (0, 1/5)$

$q \leftarrow O(\max \{\epsilon^{-2} k^2, \epsilon^{-3} k^2\} \log n)$

Index the clusters of $h$ such that $|V_1| \geq |V_2| \geq \ldots \geq |V_k|$

for $u \in V_i$, $i = 1, \ldots, k$ do

    for $j = i, \ldots, k$ do

    $S_{u,i} \leftarrow$ sample $q$ elements from $V_j$ independently uniformly at random with repetitions

end for

end for

return $f : \mathcal{H} \rightarrow \mathbb{R}$, defined by

$$f(h') = \sum_{i=1}^{k} \left[ \frac{|V_i|}{q} \sum_{u \in V_i} \sum_{v \in S_{u,i}} \left( \text{cost}_{u,v}(h') - \text{cost}_{u,v}(h) \right) \right]$$

$$+ 2 \sum_{i=1}^{k} \sum_{u \in V_i} \sum_{j=i+1}^{k} \left[ \frac{|V_j|}{q} \sum_{v \in S_{u,j}} \left( \text{cost}_{u,v}(h') - \text{cost}_{u,v}(h) \right) \right],$$

5.4.4 Conclusion: $f$ is an $\epsilon$-SRRA

We conclude that $f$ is an $\epsilon$-SRRA estimator. Its construction pseudocode is presented for convenience in Algorithm 3. Clearly, the number of label queries required for obtaining this $\epsilon$-SRRA estimator is $O(n \max \{\epsilon^{-2} k^3, \epsilon^{-3} k^2\} \log n)$. Combining Theorem 5.1 with this bound and the iterative algorithm described in Corollary 3.1 (Algorithm 1), we obtain the following:

**Corollary 5.1** There exists an active-learning algorithm for obtaining a solution $h \in \mathcal{H}$ for semi-supervised $k$-clustering with $\text{er}_D(h) \leq (1 + O(\epsilon)) \nu$ with total query complexity of $O \left( n \max \{\epsilon^{-2} k^3, \epsilon^{-3} k^2\} \log^2 n \right)$. The algorithm succeeds with probability at least $1 - n^{-2}$.

We do not believe the $\epsilon^{-3}$ factor in the corollary is tight, and we speculate that it could be reduced to $\epsilon^{-2}$ by using more advanced measure concentration tools. Note, we assume $k$ is fixed. Indeed, in practice, $k$ is often taken to be a global constant (or at most $o(n)$). Thus, the sample complexity of our active-learning method using these direct SRRA constructions is almost linear in $n$. As in the case of Corollary 4.1 and the ensuing discussion around LRPP, the result in Corollary 5.1 significantly outperforms known active-learning results that depend only on disagreement coefficient and VC dimension bounds, for small $\nu$.
5.5 Hierarchical Correlation Clustering

A natural extension of the clustering setting is concerned with clustering in the face of structural constraints. Probably the most common structure is the tree hierarchy, such as those that arise in taxonomies. Here, we will examine hierarchies corresponding to uniform-depth trees with a fixed and known depth.

A depth-$\ell$ hierarchical-clustering over a ground set $V$ is a uniform-depth tree of height $\ell$, with $V$ sitting in the leaves. The root corresponds to $V$ and its children correspond to the level-$\ell$ clustering. Each sub-tree of the root is now a depth $\ell - 1$ hierarchical clustering.

When viewed as a uniform-depth tree, a natural pseudometric arises over $V$, where the distance between two elements $u, v \in V$ is half the edge length of the path connecting their corresponding leaves. Such a metric is called an ultrametric and satisfies certain strong structural properties (see Section 5.5.1). Alternatively, any ultrametric on a set $V$ with distances in the set $0, \ldots, \ell$ defines such a tree, and in turn, a hierarchical clustering. For example, if $\ell = 2$, then the distances between element pairs are either 0, 1, or 2. If the two elements sit in the same tree leaf, meaning that they are in the same nested clusters, then it is 0. If the two elements do not sit in the same leaf, but rather descend from the same child of the root, then it is 1. If they descend from two different children of the root, it is 2. The extension to other values of $\ell$ is clear.

In this work, we also assume a limit of $k$ on the width of the sought solutions, defined as the maximal number of children of a node in the tree. In the language of hierarchical clusterings, this means that each cluster (at any level) may be further sub-clustered into at most $k$ clusters in the next level.

The label of a pair of elements $u, v \in X$ in our setting is a number in $0, \ldots, \ell$, which is a noisy stipulation of the sought distance between $u$ and $v$ in the solution hierarchical clustering. By noisy, we mean that the corresponding matrix of labels need not be a metric (let alone an ultrametric). The goal is to fit the label matrix to an ultrametric on $X$, while satisfying the usual supervised learning desiderata, namely, achieving low risk at a low query complexity.

Contrary to all former settings, the labeling function here is not binary. Still, we measure the quality of an output hierarchical clustering by its degree of proximity to the label matrix. The risk metric we choose is the average absolute value of differences between distance matrix elements. (In particular, we assume a uniform measure on the matrix elements.)

The size of the input space is quadratic in $n$; however, the logarithm of the number of hierarchical clusterings of depth $\ell$ and width $k$ is at most $\ell n \log k$. (There are at most $k^\ell$ leaves, and it suffices to assign a leaf for each $v \in V$; hence, $k^{\ell n}$ possibilities.) Accordingly, we suspect that a small sample should suffice for the purpose of obtaining a good solution.
5.5.1 Definitions and Notations

Hierarchical clusterings are in a natural one-to-one correspondence with a geometric object called ultrametric. In what follows, the two terms will be used interchangeably. As before, let $V$ be a set of $n$ items, our instance space $\mathcal{X}$ consists of all distinct pairs of elements in $V$: namely, $\mathcal{X} = (V \times V) \setminus \{(u, u) : u \in V\}$ and its size $|\mathcal{X}|$ is $N = n(n-1)$. Also, we assume $D \sim D_{\mathcal{X}}$ is the uniform measure.

An (integer) ultra-pseudometric over $V$ is a function $\tau : \mathcal{X} \rightarrow \mathbb{N}$, which is a pseudometric on $V$, and it also satisfies the following strong triangle inequality for all distinct $u, v, w, \in V$: $\tau((u, v)) \leq \max\{\tau((u, w)), \tau((w, v))\}$. (Note that the strong triangle inequality implies the usual triangle inequality.)

![Figure 5.2: An example of an ultrametric tree with $\ell = k = 3$. Let $v_i$ denote the vertex indexed with $i \in [27]$. Observe that $\tau(v_{25}, v_{27}) = 1$, $\tau(v_{22}, v_1) = 3$ is the maximal distance, and $\mathcal{V}(\mathcal{R}_{\tau, 2}) = \{C_1, C_2, C_3\}$ defines the tree’s first level.]

Let $\mathcal{U}(V)$ denote the space of ultrametrics over $V$. It is well-known (see, e.g., Ailon and Charikar, 2011) that ultrametrics have a one-to-one correspondence with a natural family of rooted trees in which each $v \in V$ is hosted in a leaf, and the edge distance between the root and any leaf is a constant, which is called the height of the tree. If $\mathcal{T}$ is such a tree, then the corresponding ultrametric $\tau_{\mathcal{T}} \in \mathcal{U}(V)$ is defined by $\tau_{\mathcal{T}}((u, v))$, equaling half the edge distance between the leaves hosting $u, v$ in the tree.

---

2By stating that “each vertex is hosted in a leaf,” we simply mean that there is a mapping from $V$ to the set of leaves, and each leaf is identified with the corresponding pre-image.
Mapping \( \tau \) to a tree \( T \) such that \( \tau = \tau_T \) is also easy: Let \( M = \max_{(u,v) \in \mathcal{X}} \tau((u,v)) \). If \( M = 0 \), then the tree contains a single node hosting all vertices. Otherwise, the relation

\[ \mathcal{R}_{\tau,i} = \{(u,v) \in \mathcal{X} : \tau((u,v)) \leq i\} \cup \{(u,u) : u \in V\} \]

is an equivalence relation (by the strong triangle inequality); the tree is constructed by creating a root node with children equaling the roots of recursively constructed trees, one for each equivalence class of \( \mathcal{R}_{\tau,M} \) (with corresponding restriction of \( \tau \) to the members of the class). In what follows, we will let \( V(\mathcal{R}_{\tau,i}) \) denote the set of equivalence classes of \( \mathcal{R}_{\tau,i} \), namely, any \( C \in V(\mathcal{R}_{\tau,i}) \) is a maximal subset of \( V \) satisfying \( (u,v) \in \mathcal{R}_{\tau,i} \) for all \( u,v \in C \).

We will also need the notion of restrictions. For a set \( C \subseteq V \), let \( X_C = (C \times C) \setminus \{(u,u) : u \in C\} \), namely, the set of distinct pairs of elements of \( C \). For \( \tau \in \mathcal{U}(V) \), let \( \tau_C \) denote the restriction of \( \tau \) to \( X_C \); clearly, \( \tau_C \in \mathcal{U}(C) \). When using restrictions, it will be implicitly understood that the underlying measure is uniform on \( X_C \).

An ultrametric \( \tau \in \mathcal{U}(V) \) has width \( k \) if the number of children of any node in the corresponding tree is at most \( k \). The depth(\( \tau \)) of an ultrametric is defined as the uniform edge distance between the root and any leaf in the corresponding tree, or alternatively, as the maximal distance between a pair of elements. Let \( \mathcal{U}_{k,\ell}(V) \) denote the space of ultrametrics on \( V \) of degree \( k \) and depth at most \( \ell \), which we call the space of \((k,\ell)\)-ultrametrics.

We call the non-hierarchical clustering described in Section 5.1 a flat \( k \)-clustering. Here, we think of a flat clustering as a function \( h : \mathcal{X} \mapsto \{0, 1\} \), such that \( h((u,v)) = 0 \) if \( \{u,v\} \subseteq V_i \) for some \( i \), and otherwise \( h((u,v)) = 1 \). Note that this is opposite to the convention used so far (defined in Section 5.1). Here \( h \) is a distance function, whereas it was formerly a similarity function. Using the distance semantic allows us to view a clustering as a pseudometric over \( V \), over the distances \( \{0, 1\} \). More specifically, when using the labeling \( Y : \mathcal{X} \mapsto \{0, 1\} \), we rewrite the error rate to be \( \text{er}_D(h, Y) = N^{-1} \sum_{(u,v) \in \mathcal{X}} |h(u,v) - Y(u,v)| \). Observe that now the set of flat clusterings over \( V \) becomes \( \mathcal{U}_{k,1}(V) \); that is, \( \mathcal{U}_{k,1}(V) \) is exactly the hypothesis class \( \mathcal{H} \) defined in Section 5.1.

**Learning \((k,\ell)\)-Ultrametrics**

Consider the following supervised learning scenario. The label function is now a symmetric function \( Y : \mathcal{X} \mapsto [\ell] \). The setting is agnostic in the sense that \( Y \) may not even be a metric (let alone an ultrametric).

We are interested in learning an ultrametric \( \tau \in \mathcal{U}_{k,\ell}(V) \). Our goal is to find a solution as close as possible to \( Y \). We choose the risk function

\[
\text{risk}_{\mathcal{D},\ell}(\tau) = N^{-1} \sum_{(u,v) \in \mathcal{X}} |\tau((u,v)) - Y((u,v))| .
\]
Note that here we apply the absolute-loss function $\ell(a - b) = |a - b|$ for any $a, b \in \mathbb{N}$. Similarly, we define the (pseudometric) $\text{dist}(\tau, \tau')$ by

$$\text{dist}(\tau, \tau') = N^{-1} \sum_{(u,v) \in X} |\tau((u,v)) - \tau'((u,v))| .$$

As in the case of flat $k$-clusterings, here again we have a supervised learning problem, except that now the label space is no longer binary (it is $[\ell]$), the hypothesis class is $U_{k,\ell}(V)$, and we measure performance with respect to absolute loss instead of zero-one loss.

Let $\tau^* = \arg\min_{\tau \in U_{k,\ell}} \text{dist}(\tau, Y)$. Let $\nu = \text{dist}(\tau^*, Y)$ be the optimal risk. Throughout, we assume the agnostic setting, namely, $\nu > 0$. The goal is to find a solution $\tau$ with small excess risk, defined as $\text{dist}(\tau, Y) - \nu$.

The query complexity is defined as the number of instances $((u,v)) \in \mathcal{X}$ for which $Y((u,v))$ is uncovered. We wish to keep this complexity measure low.

In what follows, we suppress the double-parentheses notation and write, e.g., $Y(u,v), \tau(u,v)$. It is understood that $Y$ and $\tau$ take a single argument in $\mathcal{X}$.

### 5.5.2 SRRA for Learning Shallow Ultrametrics

We start by defining a reduction from distances between ultrametrics to distances between flat (correlation) clusterings. The idea is to decompose such a distance from flat clustering distances corresponding to decomposition of tree levels. This is summarized in the next lemmata. The proof is a trivial induction over $\ell$.

**Lemma 5.4** Let $\tau, \omega$ be two $k,\ell$-ultrametrics over $V$. Denote the distance between two flat clusterings by $d_{\text{flat}}$. We have

$$\text{dist}(\tau, \omega) = \sum_{i=0}^{\ell-1} d_{\text{flat}}(\mathcal{V}(\mathcal{R}_\tau,i), \mathcal{V}(\mathcal{R}_\omega,i)) .$$

**Proof** We will apply induction over $\ell$. Clearly, this is true when $\ell = 1$ (i.e., the clustering is flat). Assume it holds for $\ell' \geq 1$, and we are in the case $\ell = \ell' + 1$.

Define $\tilde{\tau}, \tilde{\omega}$ to be $\ell-1$ height trees resulting from subtracting the leaves from $\tau$ and $\omega$. That is, the leaves of $\tilde{\tau}, \tilde{\omega}$ are $\mathcal{V}(\mathcal{R}_{\tau,0})$ and $\mathcal{V}(\mathcal{R}_{\omega,0})$ correspondingly. For any pair $(u,v) \in \mathcal{X}$, let $\text{dist}_{u,v}(\tau, \omega) = |\tau(u,v) - \omega(u,v)|$. Using the tree-analogous definition of ultrametric pairwise distance, we have:

$$\text{dist}_{u,v}(\tau, \omega) = \underbrace{\text{dist}_{u,v}(\tilde{\tau}, \tilde{\omega})}_{(i)} + N^{-1} \sum_{(u,v) \in X} \underbrace{|1_{\exists C_\tau \in \mathcal{V}(\mathcal{R}_{\tau,0}) : \{u,v\} \subseteq C_\tau} - 1_{\exists C_\omega \in \mathcal{V}(\mathcal{R}_{\omega,0}) : \{u,v\} \subseteq C_\omega}|}_{(ii)} .$$

The proof is complete by noting that $(ii)$ is the flat-clustering distance $d_{\text{flat}}(\mathcal{V}(\mathcal{R}_{\tau,0}), \mathcal{V}(\mathcal{R}_{\omega,0}))$ and by applying the inductive assumption on $(i)$. ■
Similarly, we can decompose the ultrametric risk to error rates of \( \ell \) flat clusterings. Define the pairwise risk over pair \((u, v)\) in \( X \) to be
\[
\text{risk}_{u,v}(\tau) = N^{-1}|\tau(u, v) - Y(u, v)|.
\]

It is not difficult to show that
\[
\text{risk}_{u,v}(\tau) = N^{-1}\sum_{i=0}^{\ell-1}|1_{\exists C_\tau \in \mathcal{V}(\mathcal{R}_{\tau,i}); \{u,v\} \subseteq C_\tau} - 1_{Y(u,v)<i}|.
\]

An inductive argument similar to the one described above will give us the following.

**Lemma 5.5** Let \( \tau \) be a \( k, \ell \)-ultrametric over \( V \). For any \( i \in \llbracket \ell \rrbracket \), let \( \text{er}_D(i) \) denote the flat-clustering error rate corresponding to the disjoint cover \( \mathcal{V}(\mathcal{R}_{\tau,i}) \) and the pairwise binary labeling \( 1_{Y(u,v)<i} \). We have
\[
\text{risk}_{D,\ell}(\tau) = \sum_{i=0}^{\ell-1} \text{er}_D(i)
\]

We are now ready to define the SRRA construction. Define \( f_i \) to be the flat clustering defined in Algorithm 3 with respect to the pivotal flat clustering \( \mathcal{V}(\mathcal{R}_{\tau,i}) \). The estimator is then defined to be \( f = \sum_{i=0}^{\ell-1} f_i \).

**Lemma 5.6** With a probability of at least \( 1 - n^{-2} \), we have that \( f \) is an \( \varepsilon \)-SRRA.

**Proof**
\[
|f(\omega) - \text{reg}_{\tau}(\omega)| \leq \sum_{i=0}^{\ell-1} |f_i - \text{er}_D(i)|
\]
\[
\leq \sum_{i=0}^{\ell-1} \varepsilon \text{ d}_{\text{flat}}(\mathcal{V}(\mathcal{R}_{\tau,i}), \mathcal{V}(\mathcal{R}_{\omega,i}))
\]
\[
= \varepsilon \text{ dist}(\tau, \omega).
\]

(5.27) is due to the definition of \( f \) and Lemma 5.5, and we get (5.28) from Theorem 5.1, and (5.29) from Lemma 5.4.

We conclude the following.

**Corollary 5.2** There exists an active-learning algorithm for learning \( k, \ell \)-ultrametrics with a cost of at most \((1 + \varepsilon)\nu\) and query complexity
\[
O\left(n\ell \max\left\{\varepsilon^{-2}k^{3\ell}, \varepsilon^{-3}k^{2\ell}\right\} \log^2 n\right).
\]

The algorithm succeeds with probability \( > 1 - n^{-1} \).

81
Observe that the query complexity of this decomposition is exponential in $k, \ell$. Also note that $\ell \leq \log k n$. Thus, the bound is meaningful as long as $\ell \in o(\log n)$. In practical terms, this means that this SRRA construction is good only for “shallow” hierarchies (compared to $|V|$). Yet, we stress that this is, as far as we know, the first active-learning method for learning ultrametrics with worst-case query complexity guarantees.

5.6 Discussion

We presented the query-efficient variant of the correlation clustering problem. The problem deals with active learning over equivalence relations (with a bounded number of equivalence classes). Similar to LRPP, which dealt with order relations, this setting has a finite instance space and the distribution is redundant (a distribution-free setting). We showed that attempting to solve this problem with uniform sampling and disagreement-coefficient-based methods fail to provide useful sample complexity guarantees (in the worst-case setting). In contrast, our direct SRRA construction achieves non-trivial query complexity guarantees that outline the state of the art.

The specific correlation clustering of the SRRA construction we define treats two types of disagreements: in-class, and between-class disagreements. This made the analysis a bit more complex than in former SRRA constructions. Yet, the principals remain similar. The sampling should be denser over instances in which “close” solutions are more “likely” to disagree. Indeed, we give larger “weights” to small clusterings (as defined by the pivot).

In addition, we discussed the related problem of query-efficient hierarchical correlation clustering, and more specifically learning ultrametrics. Here, we presented a simple reduction that utilizes flat clustering SRRA constructions and achieves a query complexity that is non-trivial for “shallow” hierarchies.
Chapter 6

Active Exploration for
Graph-based Learning:
Clustering with Side Information
is a Big Plus

So far, we have been establishing a method of smooth relative regret approximation. A key point in the definition of SRRA functions is the fine-grained coverage of the hypotheses space. Although this is only subtle in the definition itself (Definition 3.1), it clearly arises in all of the SRRA constructions that we have discussed this far (Chapters 3–5). Consider for example, the disagreement-based SRRA construction presented in Section 3.2.1. The way it selects queries ensures that on the one hand, it acquires labeled examples from all disagreement regions, and on the other hand, that queries are denser near the pivotal solution. This exactly accounts for integrating exploration and exploitation with respect to an intermediate solution.

While it is tempting to view SRRAs as the secret ingredient for balancing the exploration-exploitation tradeoff in active learning, it turns out that we need an intimate understanding of the relations between the hypothesis and instance spaces to actually utilize it. In Chapter 3, we gained the knowledge of disagreement regions; and in Chapters 4 and 5, we were able to break the hypothesis metric-space into small combinatorial pieces. A natural question that arises is whether we can apply the method in the absence of such knowledge?

This chapter considers this question. Note, however, that we focus here on testing the effectiveness of our ideas from a practitioner’s point of view. Therefore, we deal with a setting that has a few empirically good active-learning algorithms at its disposal. All active-learning algorithms that we examine here are either known, or variations on, known heuristics. Our proposed solution can be viewed as a modular approach for active (transductive graph-based) learning that ensures coverage in the “spirit” of SRRA smoothness.
As before, we focus on active classification problems within a transductive setting; however, instances here are always constraint-free and described as points in a feature space. Given a sample of unlabeled examples and a labeling budget $m$, the learner must select $m$ examples to be labeled by the teacher. The goal is to use the $m$ labeled examples to classify the rest of the points in the sample.

Despite the attractiveness of the active-transductive learning setting demonstrated in the former chapters, most of the research contributions on active learning for classification have focused on inductive models. A few studies do consider the above active-transductive model (Zhu et al., 2003b; Herbster et al., 2005; Yu et al., 2006), but these works tend to rely on graph-based algorithms, which have been used extensively in transductive settings (Chapelle et al., 2006).\footnote{Note that the semantics of the graphs here are different than what we considered in LRPP and semi-supervised clustering. This will be explained later in detail.} One of our motivations for emphasizing the importance of coverage in active learning is the observation that the known active-transductive algorithms (as well as many active-inductive algorithms) tend to suffer from excessive “self-confidence,” which can severely impair their performance. This flaw, which results in the neglect of entire areas of the input space during the early stages of the learning process, is demonstrated in Section 6.0.1. We propose a simple yet effective solution that enforces systematic exploration of the input space whenever it is necessary. Our $+\text{EXPLORE}$ method (Begleiter et al., 2008) can be viewed as a “patch” for fixing this deficiency and as a proposed modular approach for generating effective new, active-transductive algorithms that clearly outperform currently available active-transductive algorithms.

A few previous works consider the above mentioned deficiency of self-confident active-inductive learning algorithms. All of their solutions are based on ensemble methods. In Baram et al. (2004), it was demonstrated that self-confident learners fail on XOR-like problems. Their solution provides a general framework for combining a set of active-learning schemes. The framework is based on online learning algorithms for the multi-armed bandit problem. A simpler solution by Osugi et al. (2005) more directly relates the deficiency to the classical exploration–exploitation problem. They switch between a self-confident learner and an “exploration” learner whenever the induced hypothesis does not change “much.” The ensemble methods employed by Baram et al. (2004) and Osugi et al. (2005) depend on a number of hyper-parameters and there is no clear way to calibrate them. A recent, simpler idea appearing in Guo and Greiner’s work (2007) combines two active learners using a round-robin-like scheme; however, the learners employed in this solution are both self-confident. Thus, the combined algorithm tends to suffer from the above deficiency.

Our proposed $+\text{EXPLORE}$ solution is based on cluster covering. Other works have considered clustering within the context of active-inductive learning. The active method of Xu et al. (2003) queries cluster centers of the instances that lie
within the margin of the support-vector machine. The algorithm of Nguyen and Smeulders (2004) combines clustering and active learning; however, their method is not general in purpose and the switch between the baseline active learner and the clustering depends on a predefined hyper-parameter.

6.0.1 Motivating Example, and a Preview

The active-transductive algorithm of Zhu et al. (2003b) is among the few known algorithms designed for the active-transductive game (more details on this algorithm appear in Section 6.2). The starting point of the current study was our empirical evaluation of this algorithm (as well as others), which showed that it is a top performer in this setting. Our initial study also revealed a major deficiency, depicted in Figure 6.1. Recall that here we always treat instances as points in real-valued feature space ($\mathbb{R}^d$). The synthetic example in Figure 6.1(a) is a bi-

![Figure 6.1: Motivating example.](image)

nary classification problem with three (non-isotropic) Gaussians and two “outlier” points that reside between the lower Gaussians. When applied to this example, the algorithm of Zhu et al. (2003b) exhibits behavior that is summarized by the upper learning curve of Figure 6.1(b); it does not query any point within the lowest Gaussian and fails to decrease its error below 15% within the first 100 active queries. Of course, this example was carefully constructed to emphasize this bad behavior, but it does represent a re-occurring pattern we observed with many “real” datasets. The +EXPLORE method we develop here salvages this algorithm w.r.t. such learning problems, without reducing its already good performance for easier problems.

---

We set the hyper-parameter $k = 10$ in the algorithm of Zhu et al. (2003b) and noticed that the algorithm failed with other $k$ values on similar examples. The noise is essential for establishing this example, because this algorithm does handle noiseless XOR-like problems (see, e.g., Zhu et al., 2003b, Figure 2).
6.1 Problem Definition

We consider a distribution-free transductive setting (Vapnik, 1982, Chapter 10). We will use notations that are a bit different than the ones we used so far.

The distribution-free transductive setting is defined as follows. Consider a fixed set \( S_{m+u} = \{ (x_i, y_i) \}_{i=1}^{m+u} \subseteq \mathbb{R}^d \times \{ +1, -1 \} \) of \( m + u \) points, along with their binary labels. The learner is provided with the unlabeled full-sample \( X_{m+u} = \{ x_i \}_{i=1}^{m+u} \). A training set \( S_m \) consisting of \( m \) labeled points \( (x_i, y_i) \) is selected from \( S_{m+u} \) uniformly at random among all subsets of size \( m \) and is given to the learner. The test set \( X_u \) of size \( u \) containing the remaining unlabeled points is also given to the learner. The learners we consider here generate soft classification vectors \( h = (h_1, \ldots, h_{m+u}) \in \mathbb{R}^{m+u} \), where \( h_i \) is the soft label of example \( x_i \) given by the hypothesis \( h \). The algorithm outputs \( \text{sign}(h_i) \) for the actual (binary) classification of \( x_i \). In the passive transductive setting the goal of the learner is to predict the labels of the test points in \( X_u \) from \( (S_m, X_u) \), so as to minimize the transductive risk, \( \frac{1}{u} \sum_{x_i \in X_u} \ell(\text{sign}(h_i), y_i) \), w.r.t., the 0/1 loss function \( \ell \). The \( m \) training points are actively selected by the learner. The examples to be queried are selected iteratively. At each iteration the learner selects the next example to be queried and receives its label from a teacher.\(^3\) The goal of the active learner is to minimize the transductive risk over the remaining points that were not queried.\(^4\)

Graph-based Transductive Learning. In the graph-based transductive setting the learner receives an adjacency matrix over the points \( X_{m+u} \) in addition to the pool of unlabeled points. This \( (m + u) \times (m + u) \) adjacency matrix reflects the similarity between the full-sample points. A common way to establish such a graph is by connecting \( k \)-nearest neighbors with respect to Euclidean distances. See for example, our description of the adjacency matrix construction in Section 6.3.1. The similarity can be utilized additional side information for improving classification. It is important to note that this setting is different from those of LRPP and correlation clustering (Chapters 4 and 5). Here, instances correspond to vertices, and the graph structure is known (or partially known) in advance. In LRPP and correlation clustering, instances consist of edges in a graph, and the graph’s structure is revealed in tandem with the acquisition of labels.

\(^3\)There is also a “batch” form of selection of training examples, in which the learner selects the examples for the training set and then simultaneously queries all of them (e.g., see Yu et al. 2006). Batch querying is a special case of sequential querying and can be viewed as a limitation of the learner.

\(^4\)Alternate optimization criteria may consider the entire learning curve of the active learner (see, e.g., Baram et al., 2004; Melville et al., 2005).
6.2 A Review of the Graph-Based Transductive Algorithms We Use

We focus on four graph-based transductive algorithms\(^5\) by Joachims (2003), Zhu et al. (2003a), Belkin et al. (2004), and Zhou et al. (2004). These algorithms generate smooth solutions, that is, namely the soft-classification does not change much between nearby points.

Let \(y \in \{-1, 1, 0\}^{m+u}\) be a vector of known labels defined as follows: if \(x_i \in X_u\), then the \(i\)th entry in \(y\) is 0; otherwise, the \(i\)th entry in \(y\) is \(y_i\) (the true label of \(x_i\)). All four algorithms minimize the objective function
\[
\min_{h \in \mathbb{R}^{m+u}} (h^t R h + c(h - y)^t C(h - y)),
\]
where the left-hand term is a regularization term corresponding to the smoothness requirement and the right-hand term corresponds to the loss of the hypothesis \(h\). The constant \(c \in \mathbb{R}\) provides a balance between the regularization and loss terms.

The regularizer \(R\) is an \((m+u) \times (m+u)\) matrix induced by an adjacency matrix \(W\). The adjacency matrix reflects the similarity between the full-sample points. In our experiments we used the adjacency matrix corresponding to the \(k\)NN graph \(G\), which is built as described in Section 6.3.1. All four algorithms use the graph Laplacian regularizer \(L = D - W\) or its normalized version \(L_{norm} = I - D^{-\frac{1}{2}}W D^{-\frac{1}{2}}\), where \(D\) is a diagonal matrix with the \((i,i)\)th entry \(d_i = \sum_{j=1}^{m+u} w_{ij}\) and \(I\) is an identity matrix. The cost matrix \(C\) is an \((m+u) \times (m+u)\) diagonal matrix with the \((i,i)\)th entry being a misclassification cost for the \(i\)th example. All examples in the training (test) set have the same misclassification cost, denoted by \(C_{m} (C_u)\).

We now describe the four graph-based algorithms that we used: GRFM, SOFT, SGT, and CM. These algorithms differ essentially by their definition of the cost matrix \(C\). The first three algorithms use the graph Laplacian regularizer \(R = L\) and the last one (CM) uses the normalized graph Laplacian \(L_{norm}\).

The Gaussian random field model (GRFM) algorithm (Zhu et al., 2003a) sets \(C_l = \infty\) and \(C_u = 0\). Hence, this algorithm gives an infinite penalty on empirical errors and thus enforces solutions with zero training errors.

Algorithm SOFT. The algorithm of Belkin et al. (2004) also sets \(C_u = 0\). However, in this algorithm \(C_l = 1\) and hence empirical errors are allowed.\(^6\) We refer to this algorithm as SOFT.

The Spectral Graph Transducer (SGT) algorithm (Joachims, 2003) sets \(0 < C_l < \infty\), and \(C_u = 0\). However, SGT adds two constraints: \(\sum_i h_i = 0\);

---

\(^5\)The application of our active-learning scheme to other transductive algorithms is straightforward. See Zhu (2006) for a comprehensive survey of the existing transductive algorithms.

\(^6\)In addition, this algorithm uses the constraint \(\sum_i h_i = 0\), which is required for proving a risk bound.
and $\sum_i h_i^2 = m + u$, imposing solutions that minimize the ratiocut, induced in $G$ by positive and negative values in $h$.

**Algorithm CM.** Finally, the consistency method (CM) of Zhou et al. (2004) uses the normalized Laplacian regularizer $R = L_{norm}$ and sets $C_l = C_u = 1$. This value of $C_u$ forces the soft classification values of the unlabeled (test) points that are far from the labeled ones to be close to zero.

### 6.3 An Exploration-Exploitation Routine and Its Implementation

Our starting point is an active-learning algorithm $\text{ALG} = (P, Q)$ where $P$ is a passive-learning algorithm and $Q$ is a querying component. The passive learner $P$ uses a given $(S_m, X_u)$ to generate a transductive hypothesis $h$ and the querying component selects the next example to query $x \in X_u$, using $h$ and $(S_m, X_u)$. In this section we describe the $+\text{EXPLORE}$ routine, whose goal is to improve the performance of such algorithms by enforcing systematic exploration of unlabeled points. The proposed routine requires two components: an auxiliary querying function $Q_A(S_m, X_u)$ and a switching function $SW(S_m, X_u)$ that determines whether to generate the query using $Q$ or $Q_A$. Note that $Q_A$ and $SW$ do not rely on $h$ and thus do not suffer from the “self-confidence” deficiency.

Our routine can be viewed as a meta-algorithm that operates the given active learner and augments it with additional querying capabilities. The routine performs $m$ iterations corresponding to the $m$ required queries. At each iteration, the switching component defines which querying method to apply next. Upon termination, the routine uses the passive learner $P$ and the aggregated training set $S_m$ to classify the remaining test points.

The pseudocode in Figure 6.2 defines our proposed procedure. In the following sections we describe our implementation of the decision function $SW$ and auxiliary querying component $Q_A$ (Section 6.3.1). In our experiments we examined several implementations of the $Q$ and $P$ components; these are described in Sections 6.2 and 6.3.2.

#### 6.3.1 Implementation of $Q_A$ and $SW$

Our implementation of $Q_A$ and $SW$ is based on a very simple and effective method of cluster covering. At each iteration we cluster $X_u$. If there is an uncovered cluster containing no labeled points, our switching function $SW$ decides to use $Q_A$, which

---

Let $A_+(h)$ and $A_-(h)$ be the set of the indices of positive and negative components in $h$, respectively. The ratiocut is defined to be $\sum_{i \in A_+(h), j \in A_-(h)} w_{ij} \left( \frac{1}{|A_+(h)|} + \frac{1}{|A_-(h)|} \right)$.

88
Require: The unlabeled full sample $X_{m+u}$; an active learning algorithm $\text{ALG} = (P,Q)$; a switching component $\text{SW}$; and an auxiliary querying component $Q_A$.

Ensure: A classification of $X_{m+u}$.

$S_0 = \emptyset$.

for $i = 1$ to $m$ do
    $h = P(S_{i-1}, X_{m+u-i+1})$.
    if $\text{SW}(S_{i-1}, X_{m+u-i+1}) == Q$ then
        $x = Q(h, S_{i-1}, X_{m+u-i+1})$.
    else
        $x = Q_A(S_{i-1}, X_{m+u-i+1})$.
    end if
    Query the label $y$ of the $x$ point.
    $S_i = S_{i-1} \cup \{(x, y)\}$.
    $X_{m+u-i} = X_{m+u-i+1} \setminus \{x\}$.
end for
return $P(S_m, X_u)$.

Figure 6.2: +EXPLORE routine.

selects a “representative” point from the largest uncovered cluster. Otherwise, $\text{SW}$ operates the original querying function $Q$.

The clustering we perform in this implementation is semi-supervised, which means that it takes into account all available points and labels. Since the set of acquired labels grows during the active learning session, the clustering we compute is dynamically improved after each iteration.

In what follows of this section, we describe our implementation of the semi-supervised clustering. At the $i$th iteration we build a graph $G_i$ representing the current training and test sets $(S_{i-1}, X_{m+u-i+1})$. This is done in two steps. In the first step we generate a symmetric kNN graph, denoted by $G$, which represents the unlabeled full sample $X_{m+u}$. In this graph, there is an edge between two points iff one of them is among the $k$ “most similar” points to the other. We measure the similarity between $x_i$ and $x_j$ by the cosine similarity, $d(x_1, x_2) = x_1 \cdot x_2 / (||x_1|| \cdot ||x_2||)$. We note that this choice of metric is arbitrary and any metric could be used. If there exists an edge between the points $x_i$ and $x_j$, then we set its weight to be $w_{ij} = d(x_i, x_j)$; otherwise, $w_{ij} = 0$.

Starting with $G$, we then construct the graph $G_i$, which encodes all known labels (in $S_{i-1}$). Many methods have been proposed for incorporating labeled points in clustering (see Kulis et al., 2005, and the references therein). We tried several of them and obtained rather weak results in our setting. Hence, we propose a novel heuristic, which is guided by the following commonly used principles (Kulis et al., 2005):
1. Points with different labels should not in general be “similar.” Thus, we delete the edges between such points in $S_{i-1}$ (by setting their weights to zero).

2. Points with the same label can be similar. Hence, if there exists a pair of points $x_r$ and $x_s$ in $S_{i-1}$ with the same label and there is no edge $w_{rs}$ between them, we add an edge whose weight is $w_{rs} = \frac{1}{2} \left( \min_{j: w_{rj} \neq 0} \{w_{rj}\} + \min_{j: w_{sj} \neq 0} \{w_{sj}\} \right)$.

After $G_i$ is constructed, we cluster it using a graph-based (pairwise) clustering algorithm. In general, any unsupervised clustering algorithm can be used. We preferred an algorithm that includes some kind of reasonable mechanism for selecting the number of clusters. While automatic selection of the number of clusters is an ill-defined problem, there are some reasonable heuristics, such as the Eigenvector-Alignment mechanism of Zelnik-Manor and Perona (2004). In light of our familiarity and extensive experience with spectral techniques (such as those discussed in Ng et al., 2001; von Luxburg, 2007), we selected this method.

What remained to be described is our implementation of $Q_A$, the auxiliary querying component. As mentioned above, given the largest uncovered cluster, our goal is to select a representative point in this cluster. A representative point can be defined in several meaningful ways. For example, it can be referred to as the most central point in the cluster (in the sense of minimizing the maximal distance to any point). While this approach makes sense, it seems to be computationally expensive (for example, it takes cubic time in cluster size using the Floyd-Warshall algorithm to select a representative point). Therefore, we defined the representative point as the one that is most similar to its neighbors, namely, the point with the largest sum of weights of its edges. This point can be identified in quadratic time in the cluster size.

### 6.3.2 On Some Known and Some New Querying Components

In this section we describe all querying components $Q$ that were used in our experiments. Several of them are known, whereas some are new. The first active querying method that we consider is a transductive variant of the worst-case heuristic of Campbell et al. (2000). This heuristic is motivated by the following considerations. We assume that the absolute value $|h_i|$ of the soft-classification of the $i$th point is proportional to the true probability whose label $y_i$ is $\text{sign}(h_i)$, and we choose to query $x = \arg\max_{x \in X^u} \min \{(1 - h_i)^2, (-1 - h_i)^2\}$. It can be verified\(^8\) that the values of $h$ produced by the passive algorithms that we consider (see Section 6.2) are in $[-1, 1]^{n+u}$. Therefore, the solution for the above

\(^8\)If the absolute values of some set of $h_i$s exceed ±1, then cutting all of them to $\text{sign}(h_i)$ will only reduce the training error and the regularization term.
optimization problem is the most uncertain point, \( \arg\min_{x_i \in X_u} |h_i| \). We term this method “UNCERTAIN.” Some active-transductive experiments with the UNCERTAIN querying function are presented in Zhu et al. (2003b), and Herbster et al. (2005).

When operated with SVM, UNCERTAIN coincides with the minimum margin method called SIMPLE in (Tong and Koller, 2001), which queries the point with the minimal distance to the separating hyperplane. We propose a transductive variant of the SIMPLE strategy. A graph cut between positive and negative vertices, induced by \( h \), can be considered a transductive variant of the separating hyperplane. Hence, the transductive analogue, denoted by CUT, queries the unlabeled point that is closest to the cut. The distance to the cut is measured according to the edge weights, and the larger the path to the cut, the closer the point is. The ties are resolved by a random selection among the closest points to the cut. Unlike SVM, in graph-based algorithms the UNCERTAIN and CUT methods can query different points, although they mostly query points lying on the graph cut (i.e., the points \( x_i \) such that there exists \( x_j, w_{ij} \neq 0 \) and \( \text{sign}(h_i) \neq \text{sign}(h_j) \).

We define a novel querying method that queries the most “coarse” point. This coarseness corresponds to the difference between the soft classifications of the point and its neighbors. Specifically, we define the coarseness of \( x_i \) as \( \sum_{j=1}^{m+u} (h_i - h_j)^2 w_{ij} \). Here, this method is called COARSE. Intuitively, COARSE queries points residing in regions that include many close points with opposite labels.

The last two active querying methods we examined are those of Zhu et al. (2003b) and Herbster et al. (2005). The method of Zhu et al. (2003b), termed here REDUCE-RISK, queries the point that minimizes the expected transductive risk of the underlying passive algorithm. Since the true risk cannot be computed, Zhu et al. (2003b) approximated it by the overall uncertainty over the test set, \( \sum_{x_i \in X_u} |h_i| \). The naïve implementation of REDUCE-RISK is computationally intensive, since for each query it needs to run the passive classifier \( \Omega(u) \) times. They developed an efficient implementation for their passive algorithm (Zhu et al., 2003a).

The querying function of Herbster et al. (2005) queries the point that optimizes the trade-off between being uncertain and being central (namely, the distance from it to any point in the graph). This heuristic is motivated by the bound on the number of mistakes made by the underlying online algorithm of Herbster et al. (2005).

6.4 Empirical Evaluation

We empirically validated the efficiency of the +EXPLORE procedure using 14 different self-confident active-transductive algorithms and 11 standard datasets. Among these algorithms, two are known (Zhu et al., 2003b; Herbster et al., 2005) and the rest are novel.
### 6.4.1 Datasets and Experimental Setting

The comparison is made on 11 datasets: PIMA, BUPA, VOTING, TAE, IONOSPHERE, MUSH, MUSK, MONK, COIL, DIGIT, and TEXT. These datasets are used in the context of empirical validation of transductive algorithms. The first eight datasets are standard UCI datasets used by Blum and Chawla (2001); the image datasets COIL and DIGIT are used by Chapelle et al. (2006); and the 20-newsgroups’ binary sub-problem “Atheism versus Religion” TEXT was used by Zhu et al. (2003b). All datasets were shuffled and cut in half.

We ran GRFM, SOFT, SGT, and CM passive learners with the following querying components: CUT, UNCERTAIN, and COARSE. In addition, we experimented with the active-transductive algorithms of Zhu et al. (2003b) and Herbster et al. (2005). We term all these (P, Q)-combination algorithms as SELF-CONF(P, Q).

Recall that SELF-CONF(P, Q) algorithms base their query on a transductive hypothesis and thus require an initial training set consisting of two examples. Hence, we report the mean error of such learners over five initializations chosen uniformly at random. Note that our +EXPLORE procedure always starts with exploration steps and thus implies a deterministic choice of the initial training set.

<table>
<thead>
<tr>
<th>Data</th>
<th>P = SGT</th>
<th>SELF-CONF(P, Q)</th>
<th>ACTIVE-TRANSDUCTIVE ALGORITHMS OF</th>
<th>ZHU ET AL.</th>
<th>HERBSTER ET AL.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIMA</td>
<td>29.8±0.4</td>
<td>31.1±1.0</td>
<td>27.5</td>
<td>28.8±0.5</td>
<td>27.2</td>
</tr>
<tr>
<td>BUPA</td>
<td>38.5±0.7</td>
<td>36.6±1.6</td>
<td>36.6</td>
<td>35.6±1.0</td>
<td>34.1</td>
</tr>
<tr>
<td>VOTING</td>
<td>5.6±1.0</td>
<td>0.6±0.5</td>
<td>0.6</td>
<td>0.5±0.2</td>
<td>0.0</td>
</tr>
<tr>
<td>TAE</td>
<td>30.8±2.7</td>
<td>22.3±2.8</td>
<td>7.7</td>
<td>15.4±2.1</td>
<td>11.5</td>
</tr>
<tr>
<td>IONOSPHERE</td>
<td>22.1±1.4</td>
<td>19.7±1.8</td>
<td>13.5</td>
<td>19.4±1.5</td>
<td>18.3</td>
</tr>
<tr>
<td>MUSH</td>
<td>6.1±1.8</td>
<td>0.4±0.0</td>
<td>3.6</td>
<td>0.6±0.4</td>
<td>0.0</td>
</tr>
<tr>
<td>MUSK</td>
<td>19.2±1.9</td>
<td>15.0±1.4</td>
<td>13.1</td>
<td>11.0±0.4</td>
<td>12.0</td>
</tr>
<tr>
<td>MONK</td>
<td>19.2±1.9</td>
<td>10.4±1.6</td>
<td>13.3</td>
<td>18.6±1.5</td>
<td>15.1</td>
</tr>
<tr>
<td>COIL</td>
<td>19.2±3.4</td>
<td>11.6±2.1</td>
<td>10.0</td>
<td>9.5±0.6</td>
<td>8.4</td>
</tr>
<tr>
<td>DIGIT</td>
<td>3.4±0.7</td>
<td>0.3±0.0</td>
<td>0.3</td>
<td>1.5±0.2</td>
<td>1.3</td>
</tr>
<tr>
<td>TEXT</td>
<td>7.8±0.7</td>
<td>5.0±0.3</td>
<td>4.5</td>
<td>10.5±0.5</td>
<td>9.4</td>
</tr>
</tbody>
</table>

Table 6.1: The error (%) of the “best” representatives of the +EXPLORE, Q, and P methods. The lowest error in each row (over all columns) appears in bold font.

---

9Some of these UCI datasets contain nominal features, which we translated into a vector of indicator bits.

10This was done to reduce the amount of running time, which took more than a month using 20 CPUs.
We report the best result in hindsight achieved over a grid of hyperparameters. In general, no parameter selection scheme exists for (both transductive and inductive) active learning. Hence the goal of this section is to explore the potential of +EXPLORE on top of SELF-CONF. The grid of \( k \in \{5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 75, 100\} \) was shared by all of the algorithms. The adjacency matrices \( W \) were built with cosine similarity. We used the following values for \( c \), the hyper-parameter that balances the regularization and loss terms: \{0.001, 0.01, 0.1, 1, 10, 100\} in SOFT and CM; \{0.1, 1, 10, 100, 1000, 3200\} in SGT\(^{11}\).

### 6.4.2 The Efficiency of +EXPLORE

Figure 6.3(a)-(c) depicts scatter plots comparing the +EXPLORE, SELF-CONF, and RAND (P) methods. The comparison comprises 154 experiments and was carried out over all datasets using all \( \text{SELF-CONF}(P, Q) \) combinations and the known active-transductive algorithms of Zhu et al. (2003b) and Herbster et al. (2005). Notice that most of these experiments correspond to new \( \text{SELF-CONF}(P, Q) \) combinations that have not been tested before. In Figure 6.3(a)-(c), the points above/below the dividing line correspond to a loss/win of the y-axis method over the x-axis. We only depict results for which there is no overlap between the corresponding mean error \( \pm \) the standard error of the mean (SEM).

![Figure 6.3: Comparing the three methods: RAND, SELF-CONF, and +EXPLORE. Each point in each axis comprises two mean error results of two methods (in the x-axis and the y-axis) over one of the datasets for a training size \( m = 50 \).](image)

Observe that the SELF-CONF methods are better than RAND only for 55 out of 81 results. When applying SELF-CONF together with +EXPLORE, the advantage over RAND is increased to 104 out of 122 results. Note that the number of significant wins over \( P \) is increased by 89\% when using +EXPLORE. This effect is confirmed by Figure 6.3(c), which depicts the clear advantage of +EXPLORE over SELF-CONF.

\(^{11}\)All other hyper-parameters of the SGT implementation were set to their default values.

93
Next in Table 6.1 we compare the “best” representative of each method. These representatives were chosen according to the Friedman rank test (Demšar, 2006) at a 95% significance level. For completeness, we also include the results of the relevant active-transductive algorithms of (Zhu et al., 2003b; Herbster et al., 2005). The comparison shows that +EXPLORE achieves the best results on 9 out of the 11 datasets.

6.4.3 The Advantage of Adaptive Exploration

We sketch a few numerical examples that indicate the usefulness of performing a dynamic exploration. This adaptive nature of exploration is crucial for establishing the advantage added by +EXPLORE to SELF-CONF algorithms. Figure 6.4(a) depicts how a bad choice of only three points at the beginning of the learning rounds dramatically affects the performance. The three dots on the error curve of +EXPLORE correspond to the exploration steps (determining the initial training set).

Figure 6.4(b-c) depicts the positive effect of performing adaptive exploration. Observe in Figure 6.4(b) how the sequence of three exploration steps, starting around $m = 33$, separates the error curves of SELF-CONF and +EXPLORE. The sequence of explorations as depicted in Figure 6.4(c) dramatically reduces the error rate from 0.2 to 0.

![Figure 6.4: The effect of dynamic exploration: Comparing the learning (error) curves of SELF-CONF with SELF-CONF+EXPLORE. Queries by exploration (using $Q_A$) are indicated by dark dots.](image)

6.5 Discussion

One way to think of +EXPLORE is as a simple yet effective enhancement procedure that repairs the self-confidence deficiency of many active-transductive algorithms; however, we prefer to view it as a surrogate for the rigorous SRRA condition. +EXPLORE is literally an implementation of the key observation at the core of the SRRA method.
We empirically tested our proposed +EXPLORE method using the known active-transductive algorithms of (Zhu et al., 2003b; Herbster et al., 2005) and 12 new algorithms. The experiments clearly indicate that our +EXPLORE enhancement improves the performance of self-confident active learners in most cases. Moreover, state-of-the-art results are achieved when applying (SGT, UNCERTAIN) and (CM, UNCERTAIN) along with the +EXPLORE method.

The +EXPLORE method is a heuristic guided by the “spirit” of the SRRA smoothness condition; yet, its clear empirical advantage provides a very good indication of the potential effectiveness of our ideas.
Appendix A

Standard Concentration to the Mean Bounds

In our proofs, we use the following concentration to the mean bounds. These rather well-known standard tools that are widely used in our context; however, we state them here for the sake of completeness.

We use the well known Hoeffding inequality (e.g., Devroye et al., 1996, Chapter 8.1, page 122).

**Theorem A.1 (Hoeffding)** Let $X_1, X_2, \ldots, X_n$ be $n$ i.i.d. random variables, and $B_1, B_2, \ldots, B_n$ a series of natural numbers. For each $i$ $E[X_i] = 0$, and $X_i \leq B_i$ almost surely. Then,

$$P \left[ \sum_{i=1}^{n} X_i > t \right] \leq \exp \left[ -\frac{t^2}{2 \sum_{i=1}^{n} B_i} \right].$$

We use the following version of Chernoff bound for the tail distribution of a sum of $0-1$ random variables not necessarily distributed equally. Such variables are known as Poisson trails (not to be confused with Poisson random variables). Bernoulli trails are special case of Poisson trials in which all random variables have the same distribution. We take the bound from (Mitzenmacher and Upfal, 2005, Theorems 4.4 and 4.5, Chapter 4.2.1).

**Theorem A.2 (Chernoff)** Let $X_1, X_2, \ldots, X_n$ be independent Poisson trails such that $Pr(X_i) = p_i$. Let $X = \sum_{i=1}^{n} X_i$ and $\mu = E[X]$. Then, for $0 < \delta \leq 1$

$$P \left[ X \geq (1 + \delta)\mu \right] \leq \exp \left[ -\frac{t^2}{3} \right],$$

and

$$P \left[ X \geq (1 - \delta)\mu \right] \leq \exp \left[ -\frac{t^2}{2} \right].$$
We take the description of Bernstein inequality from (e.g., Devroye et al., 1996, Theorem 8.2, page 124). (Recall that \( \text{Var}(X) = E(X^2) - E(X)^2 \).)

**Theorem A.3 (Bernstein)** Let \( X_1, X_2, \ldots, X_n \) be \( n \) i.i.d. random variables, with \( E[X_i] = 0 \) for all \( i \). Assume there exists a number \( M \) for which \( X_i \leq M \) almost surely. Then,

\[
P \left[ \sum_{i=1}^{n} X_i > t \right] \leq \exp \left[ -\frac{t^2}{2} \frac{1}{E[X_i^2] + Mt/3} \right].
\]
References


מחקר בנושאים עיוניים ומעשיים של לימוד

פעילה

רן בוגליטר

רן בוגליטר
מחקרי בנושאים עיוניים ומעשיים של למידה

פעילה

יתboro על מחקר

לש מלייה חלקי של הדרישות לקבלת התואר

דוקטור לפילוסופיה

רון בליטר

הוג של הסטטס ההנדסיים - מכון טכנולוגי לישראל

2013 מץ החיפה

אדר תשע"ב
המחקר נעשה בהנחיית דוקטור Nir Pais ופרופסור חבר Or Aloni.
הפקולטה להנדעי המחשב.

הכֶרֶת תָודה

אני מודה ל Tatto על המימית הספיט והנידת בשטחמותי.
תקציר

ל畢竟 מככה המי תחת המשמשות והדברים הסתכלות הנחותו. העיתון ובמיるのは התחום, יוכלול האפשרונות ועבורה של כל היכולות לשונות השכלות.

באוסף שונים משלמה של יחסים יזמותיים בבתי צהריים. לדומינציה ומטריך התאוריות רפיה ד Jagieli.

הוולברג מחלים, רכיבים הולכים בקהילה עצמאית, סימור היצואות וחפצים במענייה יפים, השלב לתכניות ומילポン במסגרות (תורדו בין סמס), יואם התנאים והשידור במענה פיננシーン, איזור תומ اﻷת

מסד תומまでの גודל, ודמיון או תומכויות של קבוצות интер crítica.

וcheon מתשואת המרכז בתווך למוביל הכלה דנ במשמעת הכלה, תחליק פעלת יצירתיות של כל הכלים או חקים כללים מאסף של פריטים. מודל הכללים الكمبيوتر שדר בבייניות ובモד למקהל המכוון. המטרות במובד זה היו, בהינתן גורם פורם וטבון, לבחון לכל מתוך אוסף למוכל של כל סיוון שיתיב או אמבר אט שופט של פריטים עדכני.

לזרעא, בוית את פסיק שלdots "לט זיסוון והדשה של הדעה מתכנית, לבחון לכל שלכל תופעת על כל התוכנית. לדומינציה, הפרטים בקרה, "זרז גראים". בודר כל מתכנים sped, "ערפי אוסף של מדרדר תכניות", ב kaldır כל קוברת "זרז גראים", סיוון של זرار זرار "חרוז", בבית סיוון בקר "ה בצורה, האוסף הכליל

ה StringSplitOptions נקראי "מא_pointה השערתיי".

בעות מועש השתוב בתווך ובקביב התווך האפשריות של זכויות היא יקר. דוגמא של למסים."זרכת" בצל התווך האפשרינקבב של די מתיונים롤ים כ솟 השטכת שרוחTorrent Führung "למקרא סיוון ושרים בצל זワーク, את תורדת של"מות השמד כימיה של המים, (קראה או שלוש הערוך). במקראים יהודים התווכן רכסי בברך כשל מושך ווליתיסים.
עלחות נגרות מומיה של מומיה תחתון. כרכי ציא מצורף זה הביא לทดสอบ הרגשות לDetach.

הלמידה המומיה שועשתת על פי כל סיווג טופ תחジ די חסן בצומת התחום החשוב המומיה. והמודלים המרוכנים של הלימודי מי כי מומחה, הלימודי הכרוניקטרויבי, הלימודי פעיל. בשני

המודלים rarımımış_cekalı 받לי הלימד, מסכים לרוגנוזダイエット המומיה, גיוס לדוגמאות לחומרים שיתויית עד ת-caption.

אותו זייא על כל בsetDisplay לשר את מימ公園 הרוגנוזダイエット העתידית עד/UU لتשחי. במודל

הלמידה הפועילה מחברת הלימד אפרתור L"אלוז‡" עם סיסו האמנים; ובו הוא ראה לברז

תעודה בשעון מע MDMA שלחר אוס מומר מומיה מוס小孩 (מקנ Strait) (מקנ Strait) (מקנ Strait) (מקנ Strait) (מקנ Strait) (מקנ Strait) (מקנ Strait) (מקנ Strait) (מקנ Strait) (מקנ Strait) (מקנ Strait) (מקנ Strait)

עבידת תיאז组装วย תוניות ועשריות של מומיה הלימה הפועילה. תכלד הלימה الفועילה

השלטת על בוחרי הרוגנוזダイエット האמנים מריק את כל החלבולם. פרט במודל הלימה הפועילה

עם די פי דיוד "רודי המדר🎊" בינוני אומד ממספי התגים שידר, לכל חוח, על די

הלימד כית תזכית השערה שטעות סיווג כוקב בחזקת בחמאת תפוח האמטנה.

למודל הלימה הפועילה יש שתי נראים מרכזים: המוקוונת, גרסה המקוונת. ברברס המוקוונת

של הלימה הפועילה בורח הלימד סדרה של הרוגנוזダイエット התיית. לכל בלע, הלימד בחר בדונש

ויתר המותחיטים זאמ לבוש את תגי האמנים של גם ולהישלים ממנה. ברברס המוקוונת,

מקבל הלימד הפועילה גישה למספי של מומר והימורים של מומיה שדר, לכל חוח, על די

ה.Theme המדר Fresno" הוא ראמיא לחשף את התגי של כל חוח מאר בבל שבל. בגודזל זה הוא כתב היסוד של

לגירס התמראバルף, אולא נגור כינית להפג את התגיות והחברות של למולד

ברברס המוקוונת.

ה디יצי במודל הלימה הפועילה חולד על בושת התוכני, אולא הוא כבד נפש ממשלתי הוהב

טוב עותני בעשור האחראון. ברארה גני מהמודל תחת התניה שבסולק המוששים כלית או

כל סיווג האמנים. זוהי התניה שלקהל תכל ששביעי (רעש". אנק, המודל מקונה

תחת התניה ב"האופשי" על סיווג הכל קלעתי ובכל פלאו שוזו מאפה. כאן הלימה

הפעילה מסקולה לחפש וב כל חשים של תוניס מצפייה ממורב התהפות את כל ההישרוע של

מסכיםים עם התגי. כל שואפים יוני תניית, מסכים לתערוב חד שגיאיות עם

ההישרוע של סק עי, הקבוצה שצמודי מכיל את סיווג המטרה. המحلول לשפוג קיימת

עד מחלנויות הקלאסי של כור מתפלל llenor [1994] מחלהוגוダイエット העוניית פורץ הדרד של

Technion - Computer Science Department - Ph.D. Thesis PHD-2013-04 - 2013
ככל ילגלג פעיל אל חיות צורת על פי חיות המדס סוללה

ניחה הביאה תחת הנחת המקרה האופטימלי היא השבוף אפלס היא אוניה מעשהית. ראשית

משוש שמפוניים של בחינת אתון את בחינת, להגמוניה לשיחת שם צורת תהליך מדויק. שיער, הבירה

SVM — של החלקת השונות נמצה שיצירת המחירים המלווה (לדגים, בחירת אלגוריתמים זה —

הקשרים את המחירים השונות לחד מפרידים למטרים לאיצטראית תוח מנזר באנימאיי סופו.) בחירה כל

המרכז האנרגטי ממקאמ בדר ככלל מוחבל, לדוגן כי לסף צרכי עיון החשוב.

נימה מאמנות יוט להAlamat בחושב את התويد השجائורים המבוססים היא מובילים כל בל הבカラー

הכל ציוט בחר בציר בלגון האצטראית. להניח את או הקביס הנחת המקרה האגונסטי. מציע כי התואב את דסגופטה או חסם התוח מדויק. כל זהה תכנית ניקר

למדות חוצים. מרד הסיבות לネタ יקארמק ההסכמה (אידקכר, [2007, קורходит] הזהה שעוי

משוסמכה לסנס כתיק של בחינת מדידת פיליס גסה נוגה ומושע של הגדרת פיתוח ארגונומיים מדויק

מהוים. פקודו או העסכמה אזומ, עזר קובע השיאור הםגדיר דכר ברדייס קובע מסיב

לתשובה התוזה בתחת במחילות השיאור, את קוס prova המגזר (אפ הסטטארワイン) עליה אי

הסכמה תויו קובוץ. המเล่ม מנסר את החושי של בחירת שאלה התווג ת복지 השמקל בבל כל

نشاطבר בחפשו את התווג של הפרטיות הנכון.

تباعות התווג הוא או מזינים גישה חזרה למדידת פיליס, ברשות המוגר, תוח הנחת

המקרה האגונסטי. ב𬸣 שאלה של מי והכת חלומות לע הפרש האמוטיבים סעונות של השערה

cלכל, מחتلك התווגות, הביסול לตน השיאור בצורתה המוסמכה כקדוק השואה. גארה מרבים

שכיסי אלגוריתמים למדידת פיליס, המבוססים על אמור אגן טעון כל, המושג קובע של מידה מודרני

האגלוריטם שאטו פינוים הוא כל יומיוס תלי בהגרדה מוסיסים של אמור צעדוות הבפרור.

הושבר ביווית מיצאתי המוסיים הממודים את האמוס (בסטארויה בקודה.)

تباعות התווג או מזינים גישה שמענייה את אמוס. חזרת של או אמוס מתאימים לכל

למדידת פיליס כלהל. אגרה מרבים אמוס או המحمام גדו משמש זה פילס עם הכובדש ה

נולד ממדג גלוב וחיידת. או אם או מושר או, הלメール, או, האמוסים הביא גל

iii

Technion - Computer Science Department - Ph.D. Thesis  PHD-2013-04 - 2013
שאנון comprar, אשר מוגדרים עבורה ביעות ספיגיות מ뎁יסי באתווט האומדן הפותחות בחור.

יתר על כן, האזור מראריא שערות ביעות אל כל האומרית לקויה פניהם עד המובס
鄅 Wohnב הייוספומריקלי—א-רונקובקעל מודר אי ההסכמה יכלו מילאיה גודל ממון ומ氪ועתי
(כשרオススメ המן). (תдвиг כי אל ההנאות הכח בסיים י慮 א ReadOnlyمصמי רואלום לכל)

היא שיתניה של שול שמרות המיתועoledאמעה שבלשכי ממידי הסיבוב של

ובועית הספיגיות שא罶אני יד גורק: יمضي סדרי מעיל העמדות בועות, רוחות
לצבירפע פך קושר בלוב. שתי העשת למעעות המ違って של אOperationExceptionים קומבינטורית.
וחי זרכ לחר ווחר עיניים ב--[חלקת פופיד] המ_DECL. ובויה הראשה עתונה מקבלס בברזה
ספור של עצומים, גוישה (ועל) לטיניגוס הסלופים העמדות בועות על עצומים. המטרה היא
ל럴מר שדר מעיל עצומים שמחיטס תוקף עדיפי. מ גם הענה ציהו הוה בברזה ספור של עצומים

וביתומיס מעיל הוה שאל עצומים. כא מתרזה היא הלהק את תובעת העצומים לתוך בברזה
וה.equalToנוגיס טאמו ואת עצומים יריימ לטך דוד בሌא קובעה ואלא אסא רעדות
לשקן ייח. בשתי העפות התויוניס משלאשה יאליגיג, התרהמאה ישל קימי פוטור שיאני

זכר חק מום (הנאה שמרות=? למקרה האוגוניסית).

בболезн لتحצל של א계약ימיסיו ויתוהה התיאורית, אנשי נועיים בועודב ובעיני הייחומי.

בעיעית שלמלד הסדרים אתני מראה ביעות קסי אמפייריה תחת התוכנה האלגוניסטים שגדור.
עבורה הגיה יטרג נ.fontSize ראות פסגות המובסיות על העמדות שאל אנהו.
הsetMax מיסיتنوعים להקלות התיאוריות השול.

בנוסח אתני וממציה שיחות כלמד פניל
כלל התומנודר בחר הנייה שלן מקנידמיי אמפייריה שיחות מנהל את הנאות התודעה
על קביעה דרבר של תנייה שמהאת סטנדארדר.
השליטה התאיתית שמןなし, מקיבלת
ברמקלPont במרכזים פניל ממסיבית על שלב הודא שאלאת התאווה שמלאת הר
"קלוקת." בוזוב הא, השישון כללות עשתה לוער כל א계약ימיסי ממיד פניל.

הנאות המרכזיות של עבדות התיה:

1. בונאה ממעיים גישה חודה לכלמד פניל שבבסיסה תנייה תוחלת מעיל אמותי יוטה

יחסית וייאו מלחוה באלגריים לכל שוסⒾמות באומדים כל.

iv
2. אנחנ מציגים מימוחו בטיעת כלל המבוססים על פקוניות, ל"זרובניק" על פקוניות. התוצאות המתקבלות היא אגלוריטם למידה פיעילה כללה חתות חסמי מינימלית. אנחנ מריאים שאלגוריתמים המתקבלים משיג את תצאות הסמי גמל המדגים חותות ייחודה אחר חוסר." - וופניק חסמי.

3. עבבר ביעית לימוד מסדרי מועד הע違反ה בוגר, אנחנ מציגים ביניהם של אומד גורם.
שישלב אגלוריתמים הלמיה הפעילה, של יישומנו, משפר את תצאות הסמי גמל המדגים
חתות ייחודה. אנחנ מראים של אגלוריתמים הלמידה פיעילה יוזם, שיעור תחת
אות חוסר, אל ישיג תצאות סבירו עבבר בעיה או (בחנה שערת העquina נמוך). בנסף
ancock ממציע את חכמת התכונת יישומיות ליעום של מיכל חוסר חוריים שיערים, מופרנעה
את הפרטיניאל של יישומנו.

4. אנחנ דגש בבעית החלוקה של מצומם לפ סיסמ בוגר, אנחנ מראים של אגלוריתמים
לימוד פעילה יוזם, המтокסס על החותות ככלי מינימלי, יוכלلاحיו, אך ורק בודל
מנד הוריאליים (בחנה שערת העquina נמוך). בנסף, אנחנ שואל מגדירים במלים
לבעית שישלב אגלוריתמים הולש ככלי מינימלי של אגלוריתמים הלמידה פיעילה שמעון הסמי
גודל ומשם, מבחרה, לביעי, אל.

5. מבלי גישה המבוססת תיאוריה, אנחנ מציגים שיטה הוריאסית בחות הניתנה של.
השיטה סותרת למד פעדל כשלים יוממתה של יוכל שירטרא שבירית התוכנה את
מתוך הדורוהוטה. כתה יוממה מקיפה של השיטה, על פאלא של שיעור התוכנה בויתר
הידיעת מודרנית שאראעד שאל נמיupiter מפורך בוגר המتركي את הבינואים התחום בויתר
הידיעת. בכה הושם ימקה נקף ליעומת של.

v