THE CLASS COVER PROBLEM AND ITS APPLICATIONS IN PATTERN RECOGNITION

by

Jason G. DeVinney

A dissertation submitted to The Johns Hopkins University in conformity with the requirements for the degree of Doctor of Philosophy

Baltimore, Maryland

January 2003

Abstract

We present a new variation of the class cover problem introduced by Adam Cannon and Lenore Cowen. The class cover problem (CCP) is a special case of the classic set cover problem, but motivated by statistical learning theory. We study both the theory and applications of the CCP. We introduce class cover catch digraphs which are a type of proximity graph. We demonstrate that there is a one to one relation between solutions of the CCP and dominating sets in a class cover catch digraph (CCCD). Both deterministic and randomized models of CCCDs are studied. We also present applications of the CCP to statistical learning theory, specifically supervised classification and clustering. We use solutions to the CCP in the creation of learning algorithms. Finally we present a chapter of performance results for our CCP-based classifiers on simulated and experimental data.

Adviser: Carey Priebe

Readers: Carey Priebe, John Wierman

Acknowledgements

"If I have seen farther, it is by standing on the shoulders of giants" - Issac Newton

I am sure that I have not performed work worthy of this statement by Issac Newton, but its meaning is not lost on me - I couldn't have made it here without the help of others! I have been very fortunate throughout my education to have a number of excellent teachers. From elementary school to graduate school, I would like to thank all of the teachers who inspired me, supported me and fueled my enthusiasm for learning. In particular, I would like to thank my advisor Carey Priebe. I always felt like a colleague rather than a student when working with you. That was appreciated more than I can say. I always felt comfortable asking questions and suggesting new ideas in our brainstorming sessions. You were also a great running partner. I will always remember the many runs behind the zoo in Druid Hill park (especially the runs at five in the morning). I would also like to thank Dave Marchette. We had fun times at the white-board and your sense of humor (and willingness to laugh at Carey sometimes) was most welcome! The Tuesday morning meetings played a big part in some of the research presented in this dissertation. Thank you also to Diego Socolinsky for the time spent working on various problems and the miles in park. I would like to thank John Wierman for his time spent working with me on the class cover problem and his advice on mathematical writing. Thank you everyone in Whitehead 104, Amy, Kay, MaryBeth, Sandy and Sharon, for all of your help through the years. And finally I have to thank everyone who supported me at the National Security Agency, especially Keith Bruso, Matt Gaston and Bert Head. Your went out of your way so that I never had to worry about anything but math, and that was a very good thing!

I also must thank my family and friends for all of their support. My friends always asked about my progress and kept me motivated. To my partner in crime Nevin, thanks for the many

hours of computer training, commiserating, debating, and computer gaming. To Mom and Pop, your support was invaluable. You were always ready and willing to listen to my latest success or setbacks in graduate school (and undergraduate school for that matter!). And last but certainly not least, I must thank most of all, my wife helen. I know that life with this Ph.D. student was probably something just this side of bearable, but you managed to do it and be extremely supportive at the same time. I can never thank you enough for supporting my opportunity to pursue my dream and I only hope I can return the favor someday. You are as good as they come and my absolute favorite.

Contents

List of Tables

List of Figures

Part I

Introduction

Chapter 1

Preliminaries

1.1 Preliminaries

As the title implies, the main focus of this dissertation is on the class cover problem and it applications. Study of the class cover problem was initiated and strongly influenced by the study of classification [10, 15, 24]. For clarity of presentation we will concentrate mainly on binary classification, a special case of supervised classification. The binary classification problem is to create a zero/one or "yes/no" decision rule based on examples. An example might be a set of measurements like height, weight, blood pressure, etc. Along with each example (each set of measurements, for instance) is a yes or no tag denoting whether or not the subject belongs to a certain class, for example the class of people with diabetes. This set of examples is called a *training set*. The goal of binary classification is, given a training set, to develop a rule or *classifier* which maps the space of examples to $\{0, 1\}$ or "yes or no" and minimizes the probability of error. For example, if we observe a set of measurements from a patient with diabetes, we would like our classifier to classify this patient as having diabetes. Applications of the binary classification problem to fields such as computer vision, data mining, and intrusion detection have been extremely successful.

Binary classification involves partitioning the space of the training set into regions corresponding to class zero and class one. The region for class one (or zero) should consist of points which are *likely* to belong to class one (or zero) given the observed training data. One approach to this task is to examine the distances between a point in question and the training set points.

This simple notion of examining interpoint distances is surprisingly powerful [25] and is the basis for our class cover problem based classifiers. We present the class cover problem below as a deterministic problem. The applications to classification, as well as a more detailed description of the classification problem, appear in Chapter 4.

A *dissimilarity space* (Ω, d) is made up of a set Ω and a dissimilarity measure d on that set. A dissimilarity measure on a set Ω is a function $d : \Omega \times \Omega \to \mathbb{R}_+$ such that $\forall x, y \in \Omega,\ d(x,y) = d(y,x) > d(x,x) = 0$. We define a *open ball* in Ω under a dissimilarity d with center $c \in \Omega$ and radius $r \in \mathbb{R}_+$ as $B_d(c,r) := \{x \in \Omega : d(x,c) < r\}$. A *closed ball* with center $c \in \Omega$ and radius $r \in \mathbb{R}_+$ is defined as $B_d[c,r] := \{x \in \Omega : d(x,c) \leq r\}$.

In a dissimilarity space (Ω, d), consider two finite non-empty sets $\mathcal{X}, \mathcal{Y} \subseteq \Omega$ with a distinction of *target* class given to one of the sets. For $\alpha, \beta \in \mathbb{Z}$ we define an α, β cover to be a collection of open balls that contain at least $|\mathcal{X}| - \alpha$ of the target class and at most β of the non-target class. In its most general form, the class cover problem (CCP) is to find a set of open balls (a set C of centers and an associated set R of radii) that minimizes some real valued function f (for example cardinality) and is an α, β cover for some given α and β. For a target class $\mathcal{X}$, the general CCP is formulated as follows:

$$\begin{aligned} \inf \quad & f(C,R) \\ \text{such that} \quad & \left| \mathcal{X} \cap \bigcup_{c_i \in C} B(c_i, r_i) \right| \geq |\mathcal{X}| - \alpha \\ & \left| \mathcal{Y} \cap \bigcup_{c_i \in C} B(c_i, r_i) \right| \leq \beta \\ & \alpha, \beta \in \mathbb{N}. \end{aligned} \tag{1.1}$$

The CCP is therefore a special case of the classic set cover problem [1] with constraints on the type of covering sets that are considered. We write the list $(\Omega, d, \mathcal{X}, \mathcal{Y}, \alpha, \beta)$ to denote an instance of the CCP. Note that there is always a solution for any instance of CCP with $\mathcal{X} \cap \mathcal{Y} = \emptyset$. A collection of balls centered at each target class point with sufficiently small radius so as to not cover any non-target class points will satisfy the constraints for any α and β.

We call a cover *proper* if it contains all of the target class. We say a cover is *pure* if it contains no elements from the non-target class. A pure cover is obtained by setting the β parameter to zero and a proper cover is obtained by setting the α parameter to zero. There are several interesting special cases of the class cover problem. A CCP is called *constrained* if the covering balls are restricted to be centered at target class points. This requires the addition of the constraint $C \subseteq \mathcal{X}$ to formulation 1.1. We will call the CCP *homogeneous* if we force all covering balls to have the same radius. This merely adds the constraint $r_i = r_j \; \forall i, j : r_i, r_j \in R$. If the function $f(C, R)$ in formulation 1.1 is replaced by the cardinality function $|C|$, we will say the problem is a *standard* CCP. This thesis will concentrate mainly on the standard constrained CCP.

1.2 Motivation and Previous Work

The class cover problem is an interesting problem involving discrete mathematics, computational geometry and optimization. We also demonstrate applications in statistical learning theory which involve probability and statistics. The CCP has its origins in the work on approximate distance clustering (ADC) of Cowen, Priebe and Cannon [32, 5, 8]. In [8], Cowen and Priebe demonstrate a technique for clustering high-dimensional data. Their method, which relies only on the interpoint distances between observations, provides a dimension reduction in which clustering [8, 32] or classification [5] can be performed. Implicit, although never mentioned in their work, is the notion of attempting to cover a set with a collection of balls. They present a randomized algorithm for choosing the data points to be centers, or *witness* elements. The class cover problem was introduced by Cannon and Cowen [4]. They investigate the standard constrained homogeneous CCP and present a polynomial time approximation algorithm for its solution. Marchette and Priebe [27] use ideas from ADC and incorporate the class cover problem into supervised semi-parametric classification and lay down the framework for the ideas in this dissertation. Their goal is to use balls centered at data points to approximate the support of a distribution.

This thesis introduces a family of digraphs called class cover catch digraphs (CCCD). CCCDs are a new family of neighborhood or proximity graphs [21]. Other variations of neighborhood graphs are relative neighborhood graphs [42], Gabriel graphs [16], β-Skeletons, sphere of influence graphs, and sphere of attraction graphs [29]. All of the preceding families of graphs have potential applications in data exploration and classification. The list of applications includes pattern recognition, computer vision, and spatial analysis.

CCCDs are also related to the well studied family of intersection graphs. McKee and McMorris [28] give a thorough review of this field. Here is a short list of graphs similar in nature to CCCD's: intersection graphs, interval graphs, catch digraphs, and sphere digraphs [26].

1.3 Summary of Results

This dissertation is divided into two main parts, theory, and applications. In the theoretical part we focus on the standard constrained CCP with pure and proper covers. In Chapter 2 we show that we can represent an instance of this CCP with a directed graph called a class cover catch digraph. A solution to the CCP uniquely corresponds to a dominating set in the representing class cover catch digraph. In Sections 2.1 and 2.2 we demonstrate some basic properties of class cover catch digraphs and characterize a special class. We also investigate some properties of dominating sets on class cover catch digraphs in Section 2.3. Results in this section are also found in [12]. In Section 2.4 we present some facts about the CCP in one dimension.

In Chapter 3 we introduce the idea of a random CCP. The randomness comes from drawing the target and non-target class points from two distributions. We study the random variable corresponding to the number of balls in a minimum cardinality cover. The derivation of a probability mass function and a strong law for this random variable are shown for a simple one-dimensional case. Studying randomized CCPs moves us closer to studying the applications of the CCP to statistical learning theory. Results in this chapter are motivated by results in [14] and [33].

In the applied part of this dissertation we consider the constrained CCP. In Chapter 4 we discuss the applications of the class cover problem to statistical learning theory. Our main application is in two-class classification. We demonstrate several methods for building classifiers using the CCP, namely the Naive, α, β and random walk CCP classifiers. We present preliminary results on the random walk CCP classifier in [13]. In Chapter 5 we present a methodology for using the CCP in unsupervised classification or clustering.

In Chapter 6 we present the results of our classifiers and other competing classifiers on simulated as well as experimental data. We simulate data from two models in two, three, and five dimensions. We also present one examples of experimental data to be classified and finally we present results on synthetic data based on the experimental data set.

Part II

Theory

Chapter 2

Standard Constrained CCP

This chapter will investigate properties of the standard constrained CCP with pure and proper covers. We will refer to this problem as the CCP1. Because only pure and proper covers are considered under the CCP1, we denote an instance of the CCP1 with target class $\mathcal{X}$ as $(\Omega, d, \mathcal{X}, \mathcal{Y})$. The CCP1 problem $(\Omega, d, \mathcal{X}, \mathcal{Y})$ is formulated as

$$\begin{aligned} \min \quad & |C| \\ \text{such that} \quad & \mathcal{X} \cap \bigcup_{c_i \in C} B(c_i, r_i) = \mathcal{X} \\ & \mathcal{Y} \cap \bigcup_{c_i \in C} B(c_i, r_i) = \emptyset \\ & C \subseteq \mathcal{X}. \end{aligned} \tag{2.1}$$

2.1 Class Cover Catch Digraphs

Consider the collection of balls $\{B_i\}$ where $B_i = \{z \in \Omega : d(z, x_i) < \min_{y \in \mathcal{Y}} d(x_i, y)\}$. Note that B_i is the largest open ball centered at x_i that does not contain a non-target class point. A solution to CCP1 is a set of balls (a set of centers and associated radii) centered at target class points whose union contains all of the target class and none of the non-target class. Now consider some solution $P = (C_P, R_P)$ to the CCP1. This solution contains at least one ball V centered at some target class point x_i. If we replace V with B_i in P, the resulting set of balls is still a solution since $V \subseteq B_i$ and B_i does not contain any non-target class points. This implies that we only need to consider $\{B_i\}$ when choosing balls for our solution. That is we can reformulate CCP1 as

$$\begin{aligned} \min \quad & |J| \\ \text{such that} \quad & \mathcal{X} \cap \bigcup_{i \in J} B_i = \mathcal{X}. \end{aligned} \tag{2.2}$$

The CCP1 therefore has a decision space of size $2^{|\mathcal{X}|}$ as opposed to the potentially infinite sized decision space of the general CCP.

The fact that we only need to consider one ball per target class point simplifies the problem greatly and allows us to make a link between the CCP1 and graph theory. We begin by introducing some standard graph theoretic terms. A *graph* is a collection (V, E) of a set of *vertices* V and a set of *edges* E. Vertices are simply elements from some set and edges are subsets of V of size two. A *directed graph* or digraph is also a collection of two sets, a set of vertices and a set of *arcs*. Arcs are ordered pairs of elements of V and are thought of as directed edges. For a graph with vertex set V and edge set E, any subset of vertices $W \subseteq V$ *induces* a graph with vertex set W and edge set $E_W := \{\{u, v\} \in E : u, v \in W\}$. The definition is the same for an induced digraph replacing edges with arcs. A graph $G_1 = (V_1, E_1)$ is said to be *isomorphic* to another graph $G_2 = (V_2, E_2)$ if there exists a one to one and onto function $f : V_2 \to V_1$ such that $V_1 = \{f(v) : v \in V_2\}$ and $E_1 = \{\{f(v), f(u)\} : \{v, u\} \in E_2\}$. Again, the same definition holds true for digraphs, replacing edges with arcs.

In a graph (V, E), we say two vertices v, w are *adjacent* (denoted $v \sim w$) if $\{v, w\} \in E$. In a digraph (V, A), we say two vertices v, w are adjacent if either $(v, w) \in A$ or $(w, v) \in A$. Two vertices v and w are called *independent* if they are not adjacent. An *independent set* in a graph is a set of vertices that are pairwise independent. A *cycle* in a graph is a sequence of vertices $v_1, v_2, \ldots, v_n$ such that $v_1 \sim v_2, v_2 \sim v_3, \ldots, v_{n-1} \sim v_n$, and $v_n \sim v_1$. A *directed cycle* in a directed graph (V, A) is a sequence of vertices $v_1, v_2, \ldots, v_n$ such that $\{(v_1, v_2), (v_2, v_3), \ldots, (v_{n-1}, v_n), (v_n, v_1)\} \subseteq A$. A *simple cycle* in a digraph $D = (V, A)$ is a directed cycle $v_1, v_2, \ldots, v_n$ such that $(v_{(i+1) \bmod n}, v_{i \bmod n}) \notin A$.

A *catch digraph* of a collection of sets $\mathcal{S} = \{S_1, S_2, \ldots, S_n\}$ and corresponding base points $\mathcal{T} = \{T_1, T_2, \ldots, T_n\}$ is the digraph with vertex set $V = \{v_1, v_2, \ldots, v_n\}$ with an arc from v_i to v_j if and only if $T_j \in S_i$ (see [28]). We will call the resulting digraph the catch digraph *induced* *by* $\mathcal{S}$ and $\mathcal{T}$.

Given two sets of points $\mathcal{X}, \mathcal{Y} \subseteq \Omega$ with $\mathcal{X}$ as the target class, we will call the catch digraph D induced by the collection of B_i and their centers x_i the *class cover catch digraph* (CCCD) *induced by* $(\Omega, d, \mathcal{X}, \mathcal{Y})$. We will define $C(\Omega, d, n, m)$ to be the family of all possible unlabeled CCCDs induced by n target class points and m non-target class points in the space (Ω, d). Note that the family of CCCDs is hereditary. That is, let $D = (V, A) \in C(\Omega, d, n, m)$ and suppose $W \subset V$ with $|W| = k$. Then the digraph induced by W, $D' = (W, A_W)$, is a member of $C(\Omega, d, k, m)$.

A *dominating set* of a directed graph $D = (V, A)$ is a set of vertices $S \subset V$ such that for any $v \in V$, either $v \in S$ or $\exists w \in S : (w, v) \in A$. We might also describe a dominating set in an alternate way. For a vertex v in a digraph $D = (V, A)$, we let $N(v)$, the neighborhood of the vertex v, be the set of vertices $\{u \in V : (v, u) \in A\}$. We let $N[v] := N(v) \cup \{v\}$. For a set of vertices $S \subseteq V$ we let $N[S] := \cup_{v \in S} N[v]$. A dominating set is therefore a set $S \subseteq V$ such that $N[S] = V$. We denote the minimum cardinality of a dominating set of a digraph D by $\gamma(D)$.

By the construction of CCCDs we get the following proposition.

Proposition 1. *Consider an instance of CCP1* $(\Omega, d, \mathcal{X}, \mathcal{Y})$*. There is a one-to-one correspondence between solutions of the CCP and minimum cardinality dominating sets in the CCCD induced by* $(\Omega, d, \mathcal{X}, \mathcal{Y})$.

The dominating set problem is known to be NP-Hard for general undirected graphs [17]. Dominating set for a general digraph is also NP-Hard since any undirected graph can be represented as a digraph by replacing each edge $\{u, v\}$ in the graph with two arcs (u, v) and (v, u). Finally, dominating set for a general CCCD, a digraph with no simple cycles (see Theorem 2), is also NP-Hard since the digraph representation of an undirected graph will have no simple cycles.

Whenever we want to solve the CCP for a particular problem, instead of finding the minimum cardinality cover, we will find approximate minimum cardinality covers using a greedy algorithm [6]. The greedy algorithm is intended for finding approximately minimum size dominating sets, but we may apply it easily to the CCP. Given a CCCD $D = (V, A)$ on n vertices the greedy algorithm for finding a dominating set is implemented as follows.

Greedy Algorithm for Dominating Set

```
Input: A directed graph $D = (V, A)$.
Output: An approximately minimum cardinality dominating set.
    $C = \emptyset$, $U = V$
    while $U \neq \emptyset$
        $O_i = \{v \epsilon U : (v_i, v) \in A\}$
        $i^* = \arg\max |O_i|$
        $U = U - O_{i^*} - v_{i^*}$
        $C = C \cup v_{i^*}$
    return $C$
```

This algorithm runs in $O(n^2)$ steps and is an $O(\log n)$ approximation for the minimum dominating set for a given a CCCD on n vertices [20]. We will denote the size of a solution returned by the greedy algorithm as $\hat{\gamma}$.

In our study of the class cover problem, we can gain insight by studying CCCDs. One of the first things we may wish to achieve is a characterization of CCCDs, that is, conditions on Ω and d such that a given digraph is a CCCD. We begin to examine this issue in Theorem 1. We first define the notion of a *ball digraph*. The ball digraph under a dissimilarity d of a set of points $z_i \in \Omega$ and associated radii $r_i \in \mathbb{R}_+$ is the catch digraph induced by the collection of $B_d(z_i, r_i)$ and their centers z_i. Note that any CCCD is also a ball digraph.

Theorem 1. *If D is a ball digraph then D contains no simple cycles.*

Proof: Let D be a ball digraph induced from points in a dissimilarity space (Ω, d). Suppose for contradiction that D has a simple cycle C consisting of vertices $v_1, \ldots, v_l$. For each $i = 1, 2, \ldots, l$, there is an arc from v_i to v_{i+1} (all addition in this proof is assumed to be *modulo* l) but not an arc from v_i to v_{i-1} since C is a simple cycle. Since D is a ball digraph there are a set of points $z_i \in \Omega$ and associated radii $r_i \in \mathbb{R}_+$ such that $d(z_i, z_{i+1}) < d(z_i, z_{i-1})\ \forall i$. This is so since $B_d(z_i, r_i)$ must contain z_{i+1} but must not contain z_{i-1}. Such a set of inequalities are impossible since they are themselves cyclic (that is, they imply that $d(z_1, z_n) < d(z_{n-1}, z_n) < \ldots < d(z_2, z_3) < d(z_1, z_2) < d(z_1, z_n)$). Therefore D cannot contain a simple cycle. □

For a general dissimilarity space, the converse is not true; the lack of simple cycles is not a sufficient condition for a digraph to be a ball digraph on that dissimilarity space. For example consider the discrete metric $d_D : \Omega \times \Omega \to \{0, 1\}$ where $d_D(z_1, z_2) = 1$ if and only if $z_1 \neq z_2$.

Using this metric as a dissimilarity measure, the ball digraph $D = (V, A)$ induced by any set of distinct points $z_i \in \Omega$ and associated radii $r_i \in \mathbb{R}_+$ will have the property that all vertices have degree zero or $|V| - 1$.

2.2 Euclidean CCCDs

Consider the special case where $\Omega = \mathbb{R}^q$ and for $x, y, \in \mathbb{R}^q$, $d(x, y) = \|x - y\| = (\sum_{i=1}^{q}(x[i] - y[i])^2)^{\frac{1}{2}}$ is the L_2 metric. We will call a CCCD induced by $(\mathbb{R}^q, L_2, \mathcal{X}, \mathcal{Y})$ a *Euclidean CCCD* for $\mathcal{X}, \mathcal{Y} \subseteq \mathbb{R}^q$. Euclidean CCCDs are a special case of *sphere digraphs* (ball graphs in Euclidean space) as introduced by Maehera in [26]. A digraph D on n vertices is a Euclidean CCCD if there exists a set of n target class points and $m > 0$ non-target class points in $\mathbb{R}^q$ for some q which induce (via the Euclidean L_2 metric) a class cover catch digraph which is isomorphic to D. We characterize Euclidean CCCDs in theorem 2.

Theorem 2. *A digraph $D = (V, A)$ is a Euclidean CCCD if and only if D has no simple cycles.*

Before a proof is given, we will introduce some of the ideas used in the proof. A *relation* on a set S is a set of ordered pairs of element from S. A *partially ordered set* or poset is a set S with a relation (the partial order) P defined on it such that P is reflexive ($(x, x) \in P \ \ \forall x \in S$), antisymmetric ($(x, y) \in P$ and $x \neq y \Rightarrow (y, x) \notin P$), and transitive ($(x, y) \in P$ and $(y, z) \in P \Rightarrow (x, z) \in P$). A directed cycle in a relation P is a set of ordered pairs of the following form $\{(u, v), (v, w), (w, x), \ldots, (s, t), (t, u)\}$. The *transitive closure* of a relation P is the intersection of all transitive relations containing P. The following theorem is Theorem 6.4 in [3]

Lemma 1. *If a reflexive relation has no directed cycles, then its transitive closure is a partial order.*

A *linear order* or a total order P on a set S is a partial order such that either $(x, y) \in P$ or $(y, x) \in P \ \forall x, y \in S$. An *extension* P' of a partial order P is a relation such that $P \subseteq P'$. For example, the transitive closure of any relation is also an extension of that relation. It is clear that any partial order can be extended to a linear order by inserting necessary ordered pairs while preserving transitivity and antisymmetry [3, 40].

Finally, the proof of theorem 2 relies on a result from multidimensional scaling. An $m \times m$ matrix M is a dissimilarity matrix if for all $i, j \in \{1, 2, \ldots, m\}$, $M_{i,j} = M_{j,i}$ (symmetry),

$M_{i,j} \geq 0$, and $M_{i,i} = 0$. An $m \times m$ dissimilarity matrix M is said to be *Euclidean embeddable* if there are points in $\mathbb{R}^{m-1}$ with interpoint (Euclidean) distance matrix equal to M. It is well known that for any dissimilarity matrix B', there is a constant $c \geq 0$ such that $B' + c \cdot ee^T - c \cdot I$ is Euclidean embeddable, where e is the m-dimensional vector of ones and I is the m-dimensional identity matrix. This result is a corollary of a general condition for embeddability in a Euclidean space (see for instance [9, 11]).

Proof of Theorem 2

($\Rightarrow$) Since any Euclidean CCCD is a ball digraph, this direction is implied by Theorem 1.

($\Leftarrow$) Let $D = (V, A)$ be a ball digraph on n vertices with no simple cycles. Using D and the logic in the proof of Theorem 1 we can obtain the necessary inequalities among the $\binom{n+1}{2}$ distances among the target class points $\{x_1, \ldots, x_n\}$ and a single non-target class point $\{y\}$. That is, we wish to find the ranking of the interpoint distances among n target class points and one non-target class point such that any set of $n+1$ points (n points designated as target class and one point designated as non-target class) in $\mathbb{R}^n$ whose interpoint distances satisfy this ranking will induce a CCCD isomorphic to D. The inequalities are obtained as follows,

- $(v_i, v_j) \in A \Leftrightarrow d(x_i, y) > d(x_i, x_j)$
- $(v_i, v_j) \notin A \Leftrightarrow d(x_i, y) \leq d(x_i, x_j)$.

For convenience we will denote y by x_0 and $d(x_i, x_j)$ by $d_{i,j}$. Let $W = \{d_{i,j} : i \neq j,\ i, j \in \{0, 1, \ldots, n\}\}$ and form a relation P on W as follows.

- $d_{0,i} > d_{i,j} \iff (d_{0,i}, d_{i,j}) \in P$
- $d_{i,j} \geq d_{0,i} \iff (d_{i,j}, d_{0,i}) \in P$
- $(d_{i,j}, d_{i,j}) \in P$ for $i \neq j,\ i, j \in \{0, 1, \ldots, n\}$

Notice that P a reflexive relation. To see that P has no directed cycles, we suppose on the contrary it has a smallest directed cycle C. We know C must be of the following form $(d_{i,j}, d_{0,i}), (d_{0,i}, d_{i,k}), (d_{i,k}, d_{0,k}), \ldots, (d_{0,j}, d_{i,j})$ because the only inequalities we have are of the form $d_{0,i} > d_{i,j}$ or $d_{i,j} > d_{0,i}$. Notice that the directed path $(d_{i,j}, d_{0,i}), (d_{0,i}, d_{i,k})$ implies that $(x_i, x_k) \in A$ and $(x_i, x_j) \notin A$. Using this argument along C implies G contains a simple cycle, which cannot be by our hypothesis. Therefore by Lemma 1 the transitive closure of P represents a partial order on the interpoint distances.

We now extend the partial order on the interpoint distances of the $n+1$ points to a total order. This linear order is an extension of the original relation P. We then create a dissimilarity matrix M with $M_{i,j} = d_{i,j}$ and assigning a value of k to the k^{th} smallest interpoint distance in our total order. We create a matrix M' (Euclidean embeddable) by adding an appropriate constant to all off-diagonal entries of M. The addition of a constant to each off-diagonal entry in M preserves the ranking on the interpoint distances, thus we have embedded points such that $d(x_i, x_0) > d(x_i, x_j) \Leftrightarrow (v_i, v_j) \in A$ and $d(x_i, x_0) < d(x_i, x_j) \Leftrightarrow (v_i, v_j) \notin A$. We have therefore shown the existence of a set of points in $\mathbb{R}^n$ that induce a CCCD isomorphic to D. □

Corollary 1. *If a digraph is a CCCD then it is a Euclidean CCCD.*

Proof. Let D be a CCCD. D is clearly a ball digraph and by Theorem 1, D contains no simple cycles. Therefore by Theorem 2, D is a Euclidean CCCD. □

Theorem 2 gives the condition for a digraph to be in $C(\mathbb{R}^n, L_2, n, 1)$. Another interesting question for a CCCD D on n vertices is what is the smallest q such that $D \in C(\mathbb{R}^q, L_2, n, m)$ for some $m \in \mathbb{N}$? One way of answering this question is to characterize the digraphs which can be induced by points in every dimension. The *adjacency matrix* of a digraph $D = (V, A)$ with $|V| = n$ is an $n \times n$ matrix M whose i, j entry $(M_{i,j})$ is one if $(v_i, v_j) \in A$ and zero otherwise. The *augmented* adjacency matrix of a digraph is the adjacency matrix of the digraph with ones along the diagonal. We say an augmented adjacency matrix has the consecutive ones property for rows and columns if within any row or column in the matrix all ones appear consecutively.

The next conjecture concerns CCCDs in one dimension. Such CCCDs are explored in depth in section 2.4. We will show that a one dimensional CCCD is made up of multiple components, representing the CCCDs for the target class points in between two consecutive non-target class points (see theorem 2). It follows that we need only characterize these smaller CCCDs which are formed from two non-target class points a, b and n target class points falling in the interval (a, b). We call such a graph a $C^*(n)$ CCCD.

Conjecture 1. *A digraph D with n vertices is a $C^*(n)$ digraph if and only if there is some labeling of its vertices such that its augmented adjacency matrix M has the following properties:*

1. *The consecutive ones property for rows and columns,*

2. $M_{i,1} + M_{i,n} \geq 1 \quad \forall i \in \{1, 2, \ldots, n\}$,

3. *The middle ones square property. Let us define the middle ones square property here. Call a vertex v_i a* right *or* left *vertex if $M_{i,1} = 1$ or $M_{i,n} = 1$ respectively. Call a left or right vertex v_i a* reach *vertex if there exists a right or left (respectively) vertex v_j such that $M_{i,j} = 1$ and $M_{j,i} = 0$. The middle ones square property says that if v_i and v_j are reach vertices, then it must be the case that $M_{i,j} = M_{j,i} = 1$.*

In our efforts to better understand Euclidean CCCDs we would like to understand something about their structure in each dimension. This conjecture represents our attempt to characterize CCCDs in one dimension, a task which seems deceptively simple. While these properties seem to be necessary, we have been unable to prove their sufficiency.

2.3 Independent and Dominating Sets in CCCDs

For a digraph $D = (V, A)$, let $\alpha(D) \subseteq V$ be the size of the largest independent set and $\gamma(D) \subseteq V$ be the size of the smallest dominating set.

Theorem 3. *For any CCCD, $\alpha(D) \geq \gamma(D)$*

Proof. Let $D = (V, A)$ be a CCCD with $|V| = n$. Then by Corollary 1, D is a Euclidean CCCD ($D \in C(\mathbb{R}^n, L_2, n, 1)$), thus there are sets $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^n$ (with $|\mathcal{X}| = n$ and $\mathcal{Y} = \{0\}$) which induce a digraph isomorphic to D. We will find an independent dominating set of size $\hat{\gamma}(D)$ in D. This will show for any such digraph $\alpha(D) \geq \hat{\gamma}(D) \geq \gamma(D)$. The following *greedy radius algorithm* run on $\mathcal{X}$ and $\mathcal{Y}$ and corresponding CCCD finds an independent dominating set. The greedy radius algorithm is similar to the standard greedy algorithm for the set covering problem [20].

```
Greedy Radius Algorithm
Input: $\mathcal{X}, \mathcal{Y} \subseteq \mathbb{R}^q$ with target class $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$,
    $\mathcal{Y} = \{0\}$ and induced CCCD $D = (V, A)$.
Output: An independent dominating set for $D$.
    $K = \emptyset$, $C = V$
    while $C \neq \emptyset$
        $i^* = \arg\max\{\|x_i\| : v_i \in C\}$
        $O_{i*} = \{v \in C : (v_{i^*}, v) \in A\}$
        $C = (C - O_{i^*}) - \{v_{i^*}\}$
```

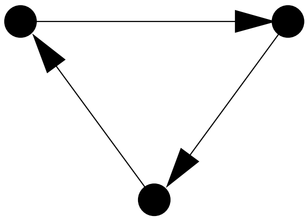


Figure 2.1: Directed graph D with $\alpha(D) < \gamma(D)$

```
    $K = K \cup \{v_{i^*}\}$
return $K$
```

To see that the set K is independent, consider two points v_i and v_j in K. Without loss of generality, suppose the algorithm chose v_i before v_j, implying $r_i \geq r_j$. It is obvious that $(v_i, v_j) \notin A$ since the algorithm only chooses points which have not been covered. Also, $(v_j, v_i) \notin A$ since $||x_i - x_j|| \geq r_i \geq r_j$. The set E is a dominating set since $C = \emptyset$ at the conclusion of the algorithm and points are removed from C only after they are covered by some point in K. □

Note that for an undirected graph G, it is always true that $\alpha(G) \geq \gamma(G)$ since a maximal independent set is always a dominating set. However, this result is not true for a general digraph. For example, consider the digraph in Figure 2.3. The largest independent set has size one, while the smallest dominating set has size two. Note also that the digraph in Figure 2.3 is a simple cycle.

In $\mathbb{R}^q$, using the Euclidean metric, define a ***kissing set*** as a set of centers of non-intersecting hyper-spheres with radius one, whose boundaries intersect the boundary of a hyper-sphere of radius one centered at the origin. The ***kissing number***, $\tau(q)$, is the size of the largest possible kissing set in $\mathbb{R}^q$ [7]. For the following lemma, let $\angle(a, b, c)$ represent the angle formed by the line segments ab and bc.

Lemma 2. *A set K of points in $\{x \in \mathbb{R}^q : ||x|| = 2\}$ is a kissing set in $\mathbb{R}^q$ if and only if for any two points $a, b \in K$, $\theta = \angle(a, \{0\}, b) \geq \frac{\pi}{3}$*

Proof. ($\Rightarrow$) We know that $||a|| = ||b|| = 2$ and $||a - b|| \geq 2$ (since the hyper-spheres centered at

a and b are non-intersecting). Thus using the Law of Cosines,

$$\begin{aligned} cos(\theta) &= \frac{||a||^2 + ||b||^2 - ||a-b||^2}{2||a||||b||} \\ &\leq \frac{1}{2}, \end{aligned}$$

which implies $\theta \geq \frac{\pi}{3}$.

($\Leftarrow$) The above can be reversed to show the converse. □

Lemma 3. *For a $C(\mathbb{R}^q, L_2, n, 1)$ digraph, $D = (V, A)$, if v_i, v_j are independent vertices and x_i, x_j the corresponding points in $\mathcal{X} \subseteq \mathbb{R}^q$, then the angle $\phi = \angle(x_i, \{0\}, x_j) \geq \frac{\pi}{3}$.*

Proof. Let $\phi = \angle(x_i, \{0\}, x_j)$ and without loss of generality let $||x_i|| \geq ||x_j||$. Using the Law of Cosines we have

$$||x_i - x_j||^2 = ||x_i||^2 + ||x_j||^2 - 2||x_i||||x_j||cos(\phi),$$

which implies

$$2||x_i||||x_j||cos(\phi) \leq ||x_j||^2$$

since $||x_i - x_j|| \geq ||x_i||$ (by our assumption of independence). Finally we get,

$$\begin{aligned} cos(\phi) &\leq \frac{||x_j||}{2||x_i||} \\ &\leq \frac{1}{2} \end{aligned}$$

which implies that $\phi \geq \frac{\pi}{3}$. □

Theorem 4. *For a $C(\mathbb{R}^q, L_2, n, 1)$ digraph D, $\alpha(D) \leq \tau(q)$*

Proof. Given a $C(\mathbb{R}^q, L_2, n, 1)$ digraph $D = (V, A)$, we will construct a kissing set in $\mathbb{R}^q$ of size $\alpha(D)$. Let $\mathcal{X}, \mathcal{Y}$ be sets of points in $\mathbb{R}^q$ (with $|\mathcal{X}| = n$ and $\mathcal{Y} = \{0\}$) which induce a digraph isomorphic to D. The existence of these points is guaranteed by Theorem 2. Let $S \subset V$ be an independent set in D and let $\mathcal{S} \subset \mathcal{X}$ be corresponding points in $\mathcal{X}$. For each $x_i \in \mathcal{S}$ define a new point $z_i = \frac{2x_i}{||x_i||}$ (this is the radial projection of each point onto the hyper-sphere of radius two centered at the origin). By Lemmas 2 and 3, the z_i's form a kissing set. □

We show that this bound is tight. Given a kissing set of size $\tau(q)$ in $\mathbb{R}^q$, we will construct an edgeless $C(\mathbb{R}^q, L_2, \tau(q), 1)$ digraph. Let $\mathcal{Y} = \{0\}$ and let $\mathcal{X}$ be the $\tau(q)$ points in the kissing set. For any pair (x_i, x_j) it must be the case that $\|x_i - x_j\| \geq 2$ since the open spheres of radius one centered at these points do not intersect. Let $D = (V, A)$ be the class cover catch digraph induced by these sets. Since the radius of each B_i is 2 it follows that $x_i \notin B_j \ \forall i \neq j$ which implies that $A = \emptyset$. Therefore $\alpha(D) = |V| = \tau(q)$.

The previous results involving CCCDs generated from multiple target class points and a single non-target class point are used to achieve an upper bound for the size of a solution of the class cover problem in higher dimensions. For a set of points $S = \{s_1, s_2, \ldots, s_n\}$ in any dissimilarity space (Ω, d), the *Voronoi region* or *Voronoi polygon* for a point s_i is given by

$$V(s_i) := \{x \in \Omega : d(s_i, x) \leq \min_j d(s_j, x)\}.$$

The collection of Voronoi regions for each element in S is called the *Voronoi diagram* generated by S [30].

Corollary 2. *For $D \in C(\mathbb{R}^q, L_2, n, m)$, $\gamma(D) \leq m \cdot \tau(q)$.*

Proof. Let $\mathcal{X}, \mathcal{Y} \subset \mathbb{R}^q$ $(|\mathcal{X}| = n, |\mathcal{Y}| = m)$ be sets which induce a digraph isomorphic to D. We partition $\mathbb{R}^q$ into the Voronoi regions $V(y_i)$ for each point $y_i \in \mathcal{Y}$. We may now bound the cardinality of the solution to each instance of CCP $(\mathbb{R}^q, L_2, \mathcal{X} \cap V(y_i), \{y_i\})$ $(i = 1, 2, \ldots, m)$ by $\tau(q)$ by Theorems 3 and 4. The result follows. □

2.4 One-Dimensional CCCDs

We consider the special case where $\Omega = \mathbb{R}$ and d is the Euclidean metric. This section will provide the groundwork so that we may fully analyze a randomized one-dimensional CCP in Chapter 3. Suppose $\mathcal{X}, \mathcal{Y}$ are finite subsets of $\mathbb{R}$ with cardinality n and m respectively. Let $y_{(i:m)}$ be the i^{th} largest element of $\mathcal{Y}$. Consider the collection of $m + 1$ intervals based on $\mathcal{Y}$

$$-\infty =: y_{(0:m)} < y_{(1:m)} \leq y_{(2:m)} \leq \cdots \leq y_{(m:m)} < y_{(m+1:m)} := +\infty;$$

$I_j := (y_{(j-1:m)}, y_{(j:m)})$ for $j = 1, \cdots, m+1$. Let $\mathcal{X}_j = I_j \cap \mathcal{X}$ and $\mathcal{Y}_j = \{y_{(j-1:m)}, y_{(j:m)}\}$. Note the following fact which allows much of our analysis in one-dimension.

Proposition 2. *Let $\mathcal{X}, \mathcal{Y} \subseteq \mathbb{R}$. If $x_i \in \mathcal{X}_i$ and $x_j \in \mathcal{X}_j$ $(i \neq j)$ then v_i is not adjacent to v_j in the CCCD induced by $\mathcal{X}, \mathcal{Y}$.*

Proof. Clearly there must be at least one non-target class point y^* between x_i and x_j since $x_i \in \mathcal{X}_i$ and $x_j \in \mathcal{X}_j$. Therefore any ball (which is an interval in one-dimension) centered at x_i could not contain x_j without containing y^*. Similarly, any ball centered at x_j could not contain x_i without containing y^*. □

Proposition 2 implies that a one-dimensional CCCD, D, is made up of $m+1$ disconnected subgraphs D_j, each of which may be null or may itself be disconnected. Define $n_j := |\mathcal{X}_j|$, and let $\gamma(D_j)$ denote the cardinality of a minimum dominating set for the CCCD induced by $\mathcal{X}_j$, $\mathcal{Y}_j$. We then have the following important formula

$$\gamma(D) = \sum_{j=1}^{m+1} \gamma(D_j). \tag{2.3}$$

Thus the study of $\gamma(D)$ is carried out via the investigation of the simpler $\gamma(D_j)$.

Lemma 4. *For any one-dimensional CCCD D and for $j = 1, m+1$ we have $\gamma(D_j) = 1\{n_j > 0\}$, where $1\{\cdot\}$ is the indicator function.*

Proof. Clearly if $n_j = 0$ then $\gamma(D_j) = 0$. Consider $j = 1$ and the case $n_1 \geq 1$. Define $B_1 := B(\min(\mathcal{X}_1), y_{(1:m)} - \min(\mathcal{X}_1))$, the largest pure open ball centered at the leftmost observation in I_1. Then $\mathcal{X}_1 \subset B_1$ and $\mathcal{Y}_1 \cap B_1 = \emptyset$, and hence $\gamma(D_1) = 1$. The case $j = m+1$ follows similarly. □

For $j = 2, \cdots, m$ we now show that $\gamma(D_j)$ takes values in $\{0, 1, 2\}$. Let

$$I_j^* := \left(\frac{\max(\mathcal{X}_j) + y_{(j-1:m)}}{2}, \frac{\min(\mathcal{X}_j) + y_{(j:m)}}{2}\right) \subset I_j.$$

Lemma 5. *For $j = 2, \cdots, m$,* $\gamma(D_j) = \begin{cases} 0 & \textit{if } n_j = 0 \\ 1 & \textit{if } I_j^* \cap \mathcal{X}_j \neq \emptyset \\ 2 & \textit{otherwise.} \end{cases}$

Proof. Again, if $n_j = 0$ then $\gamma(D_j) = 0$. Suppose now that $n_j \geq 1$. Let $X_j^- := \max_{x \in \mathcal{X}_j}\{x \leq \frac{y_{(j-1:m)} + y_{(j:m)}}{2}\}$ and $X_j^+ := \min_{x \in \mathcal{X}_j}\{x \geq \frac{y_{(j-1:m)} + y_{(j:m)}}{2}\}$. Note that $X_j^- \leq X_j^+$ if both exist.

At least one of X_j^-, X_j^+ must exist since $n_j \geq 1$. Let $B_j^- := B(X_j^-, X_j^- - y_{(j-1:m)})$ and $B_j^+ := B(X_j^+, y_{(j:m)} - X_j^+)$. ($B_j^-$(respectively B_j^+)$= \emptyset$ if $X_j^-(X_j^+)$ does not exist.) Since $\mathcal{X}_j \subset (B_j^- \cup B_j^+)$ and $\mathcal{Y}_j \cap (B_j^- \cup B_j^+) = \emptyset$, it follows that $\gamma(D_j) \leq 2$. Finally, observe that $\gamma(D_j) = 1 \iff$ there exists $x \in \mathcal{X}_j$ such that (i) $x - \min(X_j) < y_{(j:m)} - x$ and (ii) $\max(X_j) - x < x - y_{(j-1:m)}$, and (i) and (ii) hold if and only if $x \in I_j^\star$. □

We get the following corollary as an immediate consequence of Lemmas 4 and 5.

Corollary 3. *Let D be in $C(\mathbb{R}, L_2, n, m)$ with $n > 0$. Then $1 \leq \gamma(D) \leq \min(n, 2m)$.*

Finally we note the optimality of the greedy algorithm for the one-dimensional CCP.

Theorem 5. *The greedy algorithm is optimal for any one-dimensional CCCD D. That is $\hat{\gamma}(D) = \gamma(D)$.*

Proof. It is sufficient to consider the sub-digraph $D_j = (V_j, A_j)$ since D consists of connected components D_j. If $\gamma(D_j) = 1$ then $V_j^* := \{v : (v, w) \in A_j \ \ \forall w \in V_j - \{v\}\}$ is not empty. The greedy algorithm will choose some element from V_j^* and then terminate. If $\gamma(D_j) = 2$ then the first vertex chosen by the greedy algorithm will cover all vertices representing points to the left or right of the midpoint of the interval I_j. The second vertex chosen will cover all the remaining vertices since we know there is a point whose covering ball covers the remaining points to the right or left respectively of the midpoint of I_j. □

Chapter 3

Randomized Version

In the previous chapters we presented a deterministic model of the CCP. Since we will apply the CCP to real world data it is useful to consider a randomized CCP in which target and non-target class points are observations drawn from two distributions F_X and F_Y. Once again we focus on the CCP1 and assume the data are elements of some Euclidean space. We will also assume that the distributions are continuous implying that the probability of drawing the same data point more than once is zero. This assumption is necessary since in the definition of the CCP, we require $\mathcal{X}, \mathcal{Y}$ to be finite and disjoint sets. The CCCD induced by such a random drawing of points is called a random CCCD. The sample space of all possible CCCDs induced by n points drawn from F_X and m points drawn from F_Y will be denoted $R(F_X, n, F_Y, m)$. A random variable of interest for any random CCCD D drawn uniformly from $R(F_X, n, F_Y, m)$ is $\Gamma(D)$, which is the size of a minimum dominating set in D.

3.1 Distribution Results in One Dimension

We begin our study of randomized CCCDs by investigating the one-dimensional case. Please refer to section 2.4 for related notation and results. In the randomized case we define n and m one-dimensional random variables ($X = \{X_1, X_2, \ldots, X_n\}$ and $Y = \{Y_1, Y_2, \ldots, Y_m\}$) drawn independently from distributions F_X and F_Y respectively. The i^{th} *order statistic* of a set of one-dimensional random variables $U_1, U_2, \ldots, U_l$ is the i^{th} largest observation and is denoted

$U_{(i:l)}$. We note the $m+1$ intervals

$$-\infty =: Y_{(0:m)} < Y_{(1:m)} \leq Y_{(2:m)} \leq \cdots \leq Y_{(m:m)} < Y_{(m+1:m)} := +\infty.$$

$I_j := (Y_{(j-1:m)}, Y_{(j:m)})$ for $j = 1, \cdots, m+1$. Let $\mathcal{X}_j = I_j \cap X$ and $\mathcal{Y}_j = \{Y_{(j-1:m)}, Y_{(j:m)}\}$. Define the random variable $N_j := |\mathcal{X}_j|$. We begin by deriving the probability mass function for $\Gamma(\cdot)$ for a special family of CCCDs.

Let D be a random CCCD drawn uniformly from $R(F_X, n, F_Y, m)$ for some distributions F_X, F_Y on $\mathbb{R}$. Using Equation 2.3 we see

$$P[\Gamma(D) = k] = P[\sum_{i=1}^{m+1} \Gamma(D_i) = k]. \tag{3.1}$$

It is evident that if we would like to study the distribution of $\Gamma(D)$ it will be sufficient to study the distribution of $\Gamma(D_i)$ for $i = 1, 2, \ldots, m+1$.

From Lemma 4, for any distributions F_X and F_Y, we immediately get

$$P[\Gamma(D_i) = 1] = P[N_i > 0] = 1 - P[N_i = 0] = 1 - P[\Gamma(D_i) = 0] \tag{3.2}$$

for $i = 1, m+1$. Now we consider the D_i for $i = 2, 3, \ldots, m$. Notice that each of these graphs is formed from a set of two non-target class points $\{a, b\}$ and N_i target class points distributed on the interval (a, b). Suppose the distributions F_X and F_Y have density functions f_X and f_Y. Then the distribution of the N_i target class points is given by the conditional density $f_X(x|x \in (a, b))$ which we denote as $f_{a,b}(x)$. From Lemma 5 we get the following lemma.

Lemma 6. *For $i = 2, 3, \ldots, m$, and given that $Y_{(i-1:m)} = a$, $Y_{(i:m)} = b$,*

$$P[\Gamma(D_i) = 0 | N_i = k] = 1\{k = 0\}, \tag{3.3}$$

$$P[\Gamma(D_i) = 1 | N_i = k] = 1\{k > 0\}(1 - P[\Gamma(D_i) = 2 | N_i = k]), \tag{3.4}$$

$$P[\Gamma(D_i) = 2 | N_i = k] =$$

$$\int_a^{\frac{a+b}{2}} \int_T^b k(k-1) f_{a,b}(x_1) f_{a,b}(x_k) \left[\int_{x_1}^{x_k} f_{a,b}(z)dz - \int_{\frac{x_k+a}{2}}^{\frac{x_1+b}{2}} f_{a,b}(z)dz \right]^{k-2} dx_k dx_1 \tag{3.5}$$

where $T = \max\{\frac{x_1+b}{2}, 2x_1 - a\}$.

Proof. Let $\min\{\mathcal{X}_i\} = X_{(1)}$ and $\max\{\mathcal{X}_i\} = X_{(n)}$. Obviously $\Gamma(D_i) = 0$ if and only if $N_i = 0$ (D_i is the empty graph). From Lemma 5 we know that if $N_i > 0$ then $\Gamma(D_i) \in \{1, 2\}$. Therefore $P[\Gamma(D_i) = 2|N_i = k] = 1\{k > 0\}(1 - P[\Gamma(D_i) = 2|N_i = k])$. We obtain for $k > 0$,

$$\begin{aligned} P[\Gamma(D_i) = 2|N_i = k] &= P[\mathcal{X}_i \cap I_i^* = \emptyset|N_i = k] \\ &= \int_a^{\frac{a+b}{2}} \int_T^b P[\mathcal{X}_i \cap I_i^* = \emptyset|N_i = k, X_{(1)} = x_1, X_{(k)} = x_k] g(x_1, x_k) dx_k dx_1 \end{aligned}$$

where $g(x_1, x_k)$ is the the joint probability density function for $X_{(1)}$ and $X_{(k)}$

$$g(x_1, x_k) = k(k-1)\left[\int_{x_1}^{x_k} f_{a,b}(z)dz\right]^{k-2} f_{a,b}(x_1) f_{a,b}(x_k)$$

and

$$\begin{aligned} P[\mathcal{X}_i \cap I_i^* = \emptyset|N_i = k, X_{(1)} = x_1, X_{(k)} = x_k] &= [P[y \in I_i - I_i^*|X_{(1)} = x_1, X_{(k)} = x_k]]^{k-2} \\ &= \left[1 - \frac{\int_{\frac{x_k+a}{2}}^{\frac{x_1+b}{2}} f_{a,b}(z)dz}{\int_{x_1}^{x_k} f_{a,b}(z)dz}\right]^{k-2}. \end{aligned}$$

□

We now consider the special case where $F_X = F_Y = \text{Uniform}(c, d)$ (or $\text{U}(c, d)$) for $-\infty < c < d < \infty$. The uniform distribution simplifies equation (3.5) considerably. If we condition on $Y_{(i:m)} = a$ and $Y_{(i+1:m)} = b$ for any $i \in \{2, 3, \ldots, m\}$, then $f_{a,b}(x) = \frac{1}{b-a}$ for any $c \leq a < b \leq d$ and $x \in (a, b)$. To simplify notation we let $\Gamma_{n,m}$ represent $\Gamma(D)$ for D drawn uniformly from $R(U(c, d), n, U(c, d), m)$. Also let $\Gamma_{n,m}^i$ represent $\Gamma(D_i)$ for such a D. Equation 3.5 becomes (for

$k \geq 0)$

$$
\begin{aligned}
P[\Gamma^i_{n,m} = 2 | N_i = k] &= \int_a^{\frac{a+b}{2}} \int_T^b \frac{k(k-1)}{(b-a)^2} \left(\frac{3(x_k - x_1)}{2(b-a)} + \frac{(a-b)}{2(b-a)} \right)^{k-2} dx_k dx_1 \\
&= \frac{k(k-1)}{(b-a)^k} \Bigg[\int\limits_a^{\frac{2a+b}{3}} \int\limits_{(b+x_1)/2}^{b} \left(\frac{3(x_k - x_1)}{2} + \frac{(a-b)}{2} \right)^{k-2} dx_k dx_1 \\
&\quad + \int\limits_{\frac{2a+b}{3}}^{\frac{a+b}{2}} \int\limits_{2x_1 - a}^{b} \left(\frac{3(x_k - x_1)}{2} + \frac{(a-b)}{2} \right)^{k-2} dx_k dx_1 \Bigg] \\
&= \frac{1}{9} \frac{1}{4^{k-1}} (-2 - 2^k + 4^k) + \frac{1}{9} \frac{1}{4^{k-1}} (2^k - 2) \\
&= \frac{4}{9} - \frac{4}{9} \frac{1}{4^{k-1}}.
\end{aligned}
\tag{3.6}
$$

We have just proven part (iii) of the following lemma (parts (i) and (ii) follow directly from Lemma 6).

Lemma 7. *Let $\mathcal{X}$ and $\mathcal{Y}$ be distributed uniformly on $[0,1]$ with $\mathcal{X}$ as the target class. For $i \in \{2, 3, \ldots, m\}$*

(i) $P[\Gamma^i_{n,m} = 0] = P[N_i = 0]$

(ii) $\kappa(k) := P[\Gamma^i_{n,m} = 1 | N_i = k] = 1\{k > 0\}(\frac{5}{9} + \frac{4}{9}\frac{1}{4^{k-1}})$

(iii) $P[\Gamma^i_{n,m} = 2 | N_i = k] = 1\{k > 0\}(1 - \kappa(k))$

Notice that the probability distribution of $\Gamma^i_{n,m}$ is independent of the location and size of the interval I_i. This is another consequence of the uniform distribution and is what will allow the derivation of the distribution of $\Gamma_{n,m}$.

Let Z_m denote the set of non-negative integers less than m; $Z_m := \{0, \cdots, m-1\}$. Define

$$
\Delta^S_{z,b} := \{(z_1, \cdots, z_b) : \sum_{i=1}^{b} z_i = z \ ; \ z_i \in S \ \forall i\}.
$$

Theorem 6. *The probability mass function for the random variable $\Gamma_{n,m}(D)$ is given by*

$$
\begin{aligned}
P[\Gamma_{n,m} = d] =& \frac{n!m!}{(n+m)!} \quad \times \\
& \sum_{\vec{n} \in \Delta^{Z_{n+1}}_{n,m+1}} \sum_{\vec{d} \in \Delta^{Z_3}_{d,m+1}} \alpha(d[1], n[1]) \cdot \alpha(d[m+1], n[m+1]) \prod_{j=2}^{m} \beta(d[j], n[j])
\end{aligned}
$$

where

$$\alpha(d,n) = \max(1\{n=d=0\}, 1\{n \geq d = 1\})$$

and

$$\beta(d,n) = \max(1\{n=d=0\}, 1\{n \geq d \geq 1\}) \cdot \kappa(n)^{1\{d=1\}} \cdot (1-\kappa(n))^{1\{d=2\}}.$$

Proof. For $\Gamma_{n,m} = \sum_{j=1}^{m+1} \Gamma_{n,m}^{j} = d$ we must have $\Gamma_{n,m}^{i} = d[i]$ for all $i \in \{1, 2, \ldots, m+1\}$ for some vector $\vec{d} = (d[1], \cdots, d[m+1])$ such that $\sum_{j=1}^{m+1} d[j] = d$. Also if $N = \{N_1, N_2, \ldots, N_{m+1}\}$ there must be a vector $\vec{n}$ such that $N_j = n[j]$ and $\sum_{j=1}^{m+1} n[j] = n$. $\Delta_{n,m+1}^{Z_{n+1}}$ is precisely the collection of $\vec{n}$ which can occur and, since the individual $d[j]$ can take values only in $\{0,1,2\}$, $\Delta_{d,m+1}^{Z_3}$ is precisely the collection of $\vec{d}$ which can occur. Therefore we have

$$\begin{aligned} P[\Gamma_{n,m} = d] &= \sum_{\vec{n} \in \Delta_{n,m+1}^{Z_{n+1}}} \sum_{\vec{d} \in \Delta_{d,m+1}^{Z_3}} P[\vec{n}] \prod_{j=1}^{m+1} P[\Gamma_{n,m}^{j} = d[j] | N = \vec{n}] \\ &= \sum_{\vec{n} \in \Delta_{n,m+1}^{Z_{n+1}}} \sum_{\vec{d} \in \Delta_{d,m+1}^{Z_3}} P[\vec{n}] \prod_{j=2}^{m} P[\Gamma_{n,m}^{j} = d[j] | N = \vec{n}] \\ &\quad \cdot P[\Gamma_{n,m}^{1} = d[1] | \vec{n}] \cdot P[\Gamma_{n,m}^{m+1} = d[m+1] | N = \vec{n}] \end{aligned}$$

where we have used the conditional independence of the $\Gamma_{n,m}^{i}$ given N. The form of the final expression is due to the fact that we need to treat the end intervals I_1 and I_{m+1} separately. Certain pairs $(n[j], d[j])$ are incompatible, such as $n[j] = 0$ and $d[j] > 0$; the indicator functions in the statement of Theorem 6 eliminate incompatible pairs from the summation. For the end intervals I_1 and I_{m+1}, the α terms yield a value of unity if the $(n[j], d[j])$ pair is compatible. The β terms are derived from compatibility considerations and the result of Lemma 7. The desired result is obtained by noting that each $\vec{n} \in \Delta_{n,m+1}^{Z_{n+1}}$ has probability $\frac{1}{\binom{n+m}{n}}$ of occuring. □

While the expected value of $\Gamma_{n,m}$ can be obtained from Theorem 6 we present a simpler derivation.

Theorem 7.

$$E[\Gamma_{n,m}] \quad = \quad \frac{2n}{n+m} + \frac{n!m(m-1)}{(n+m)!} \sum_{i=1}^{n} \frac{(n+m-i-1)!}{(n-i)!} \cdot (2 - \kappa(i))$$

where $\kappa(i)$ is given by Lemma 7.

Proof. The expected value of $\Gamma_{n,m}$ is given by

$$\begin{aligned} E[\Gamma_{n,m}] &= \sum_{j=1}^{m+1} E[\Gamma_{n,m}^j] \\ &= P[X_{(1:n)} < Y_{(1:m)}] + \sum_{j=2}^{m}\sum_{i=1}^{n} P[N_j = i] E[\Gamma_{n,m}^j | N_j = i] + P[X_{(n:n)} > Y_{(m:m)}] \end{aligned}$$

Using Lemma 7, for $j = 2, 3, \ldots, m$ we have,

$$E[\Gamma_{n,m}^j | N_j = i] = \kappa(i) + 2(1 - \kappa(i)) = 2 - \kappa(i).$$

Also since the N_j are identically distributed,

$$P[N_j = i] = \frac{\binom{n+m-i-1}{n-i}}{\binom{n+m}{n}}.$$

For $j = 1, m+1$ we have $\Gamma_{n,m}^j = 1\{N_j > 0\}$ and $P[N_j > 0] = \frac{n}{n+m}$. Thus,

$$\begin{aligned} E[\Gamma_{n,m}] &= \frac{2n}{n+m} + \sum_{j=2}^{m}\sum_{i=1}^{n} \frac{\binom{n+m-i-1}{n-i}}{\binom{n+m}{n}} \cdot (2 - \kappa(i)) \\ &= \frac{2n}{n+m} + (m-1)\sum_{i=1}^{n} \frac{\binom{n+m-i-1}{n-i}}{\binom{n+m}{n}} \cdot (2 - \kappa(i)) \\ &= \frac{2n}{n+m} + \frac{n!m(m-1)}{(n+m)!} \sum_{i=1}^{n} \frac{(n+m-i-1)!}{(n-i)!} \cdot (2 - \kappa(i)) \end{aligned}$$

□

3.2 Limiting Results in One Dimension

In this section we explore some limiting behaviors of $\Gamma_{n,m}$. We are interested in determining if there is an almost sure limit of $\frac{\Gamma_{\lfloor an \rfloor, n}}{n}$ for any $a \in \mathbb{R}_+$ where $\lfloor r \rfloor$ is the largest integer no larger than r. Before proceeding, we perform a calculation to make sure $E[\frac{\Gamma_{\lfloor an \rfloor, n}}{n}]$ converges as

$n \to \infty$. Using Theorem 7 we obtain

$$\begin{aligned}
\lim_{n\to\infty} E\left[\frac{\Gamma_{\lfloor an\rfloor,n}}{n}\right] &= \lim_{n\to\infty} \frac{2\lfloor an\rfloor}{n(\lfloor an\rfloor+n)} + \frac{(\lfloor an\rfloor)!(n-1)}{(\lfloor an\rfloor+n)!}\sum_{i=1}^{\lfloor an\rfloor}\frac{(\lfloor an\rfloor+n-i-1)!}{(\lfloor an\rfloor-i)!}(2-\kappa(i)) \\
&= \lim_{n\to\infty}(n-1)\sum_{i=1}^{\lfloor an\rfloor}\frac{(\lfloor an\rfloor)!(\lfloor an\rfloor+n-i-1)!}{(\lfloor an\rfloor-i)!(\lfloor an\rfloor+n)!}(2-\kappa(i)) \\
&= \lim_{n\to\infty}\sum_{i=1}^{\lfloor an\rfloor}\left(\frac{n-1}{\lfloor an\rfloor+n}\cdot\prod_{j=0}^{i-1}\left(\frac{\lfloor an\rfloor-j}{\lfloor an\rfloor+n-(j+1)}\right)(2-\kappa(i))\right)
\end{aligned}$$

and letting $f_n(i) := \frac{n-1}{\lfloor an\rfloor+n}\cdot\prod_{j=0}^{i-1}\frac{\lfloor an\rfloor-j}{\lfloor an\rfloor+n-(j+1)}$ we have

$$= \lim_{n\to\infty}\frac{13}{9}\sum_{i=1}^{\lfloor an\rfloor}f_n(i) - \frac{4}{9}\sum_{i=1}^{\lfloor an\rfloor}f_n(i)\frac{1}{4^{i-1}}. \tag{3.7}$$

We now pause to show an upper bound for $f_n(i)$.

$$\begin{aligned}
f_n(i) &= \frac{n-1}{\lfloor an\rfloor+n}\cdot\prod_{j=0}^{i-1}\frac{\lfloor an\rfloor-j}{\lfloor an\rfloor+n-(j+1)} \\
&= \frac{n-1}{an-\delta+n}\cdot\prod_{j=0}^{i-1}\frac{an-\delta-j}{an-\delta+n-(j+1)} \qquad \text{for some } \delta\in[0,1) \\
&\le \frac{n-1}{an-\delta+n}\cdot\prod_{j=0}^{i-1}\frac{an-j}{an+n-(j+1)} \qquad \text{for } n\ge 2 \text{ by Claim A}
\end{aligned}$$

and since the term corresponding to $j=0$ is larger than any other term in the product,

$$\begin{aligned}
&< \frac{n-1}{an-\delta+n}\cdot\left(\frac{an}{an+n-1}\right)^i \\
&= \frac{n-1}{n(a+1-\frac{\delta}{n})}\cdot\left(\frac{a}{a+1-\frac{1}{n}}\right)^i \\
&\le \frac{1}{a}\left(\frac{a}{a+\frac{1}{2}}\right)^{i+1}.
\end{aligned}$$

If we let $g(i) := \frac{1}{a}\left(\frac{a}{a+\frac{1}{2}}\right)^{i+1}$ we see that $\lim_{n\to\infty}\sum_{i=1}^{\lfloor an\rfloor} g(i)$ is finite for any $a > 0$. We may now apply the dominated convergence theorem [41] to equation (3.7).

$$\begin{aligned}
\lim_{n\to\infty} E\left[\frac{\Gamma_{\lfloor an\rfloor,n}}{n}\right] &= \lim_{n\to\infty} \frac{13}{9}\sum_{i=1}^{\lfloor an\rfloor} f_n(i) - \frac{4}{9}\sum_{i=1}^{\lfloor an\rfloor} f_n(i)\frac{1}{4^{i-1}} \\
&= \frac{13}{9}\sum_{i=1}^{\infty} \lim_{n\to\infty} f_n(i) - \frac{4}{9}\sum_{i=1}^{\infty} \lim_{n\to\infty} f_n(i)\frac{1}{4^{i-1}} \\
&= \frac{13}{9(a+1)}\sum_{i=1}^{\infty}\left(\frac{a}{a+1}\right)^i - \frac{16}{9(a+1)}\sum_{i=1}^{\infty}\left(\frac{a}{a+1}\right)^i\frac{1}{4^i} \\
&= \frac{13}{9(a+1)}\left(\frac{1}{1-\frac{a}{a+1}} - 1\right) - \frac{16}{9(a+1)}\left(\frac{1}{1-\frac{a}{4(a+1)}} - 1\right) \\
&= \frac{a(13a+12)}{3(a+1)(3a+4)}
\end{aligned}$$

We now show our main result

Theorem 8. *For $a \in (0,\infty)$,*

$$\lim_{n\to\infty}\frac{\Gamma_{\lfloor an\rfloor,n}}{n} = \frac{a(13a+12)}{3(a+1)(3a+4)} \quad a.s.$$

where a.s. means almost surely*.*

Before proving this result, we pause to consider its meaning. For $i \in \{1, m+1\}$ we call $\Gamma^i_{n,m}$ the *external components* of $\Gamma_{n,m}$ and for $i \in \{2,3,\ldots,m\}$ we call $\Gamma^i_{n,m}$ the *internal components* of $\Gamma_{n,m}$. Notice from Equation (3.2) and Lemma (7) that $\Gamma^i_{n,m}$ depend only on N_i for all $i \in \{1,2,\ldots,m\}$. Since both $\mathcal{X}$ and $\mathcal{Y}$ points are uniformly distributed on $[0,1]$, the N_i random variables are identically distributed. This fact along with equations (3.2) and (3.6) imply that the external components are identically distributed and the internal components are also identically distributed.

Since there are only two external components and their value is bounded above by one,

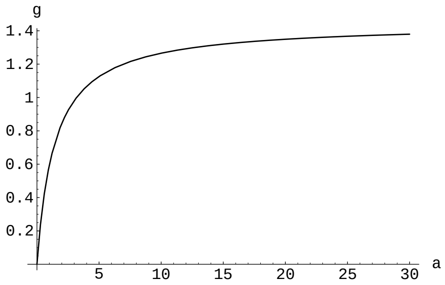


Figure 3.1: A plot of $\frac{a(13a+12)}{3(a+1)(3a+4)}$

Theorem 8 is a Strong Law of Large numbers for the internal components.

$$\begin{aligned}
\lim_{m\to\infty}\frac{\Gamma_{n,m}}{m} &= \lim_{m\to\infty}\sum_{i=1}^{m+1}\frac{\Gamma^i_{n,m}}{m} \\
&= \lim_{m\to\infty}\sum_{i=2}^{m}\frac{\Gamma^i_{n,m}}{m} + \lim_{m\to\infty}\frac{\Gamma^1_{n,m}+\Gamma^{m+1}_{n,m}}{m} \\
&= \lim_{m\to\infty}\sum_{i=2}^{m}\frac{\Gamma^i_{n,m}}{m-1}\cdot\frac{m-1}{m} \\
&= \lim_{m\to\infty}\sum_{i=2}^{m}\frac{\Gamma^i_{n,m}}{m-1}.
\end{aligned}$$

Note that the limiting expression in Theorem 8 is an increasing function of a, the ratio of numbers of target class and non-target class points. Let $g(a) := \frac{a(13a+12)}{3(a+1)(3a+4)}$. Figure 3.1 displays a plot of the function $g(a)$. As $a \to 0$, the $g(a)$ converges to zero, reflecting the fact that most intervals between sucessive $\mathcal{Y}$ points will contain no $\mathcal{X}$ point. More interestingly, as $a \to \infty$, the $g(a)$ converges to $\frac{13}{9}$. This corresponds to each interval between sucessive $\mathcal{Y}$ points containing a large number of $\mathcal{X}$ points. By Lemma 7, the probability that one ball covers all points in this interval is near $\frac{5}{9}$ and the probability that two balls are needed is near $\frac{4}{9}$, resulting in an expected value near $\frac{13}{9}$. Alternatively, one may consider normalizing by the number of $\mathcal{X}$ points. In this case we see $lim_{n\to\infty}\frac{\Gamma_{\lfloor an\rfloor,n}}{\lfloor an\rfloor} = \frac{13a+12}{3(a+1)(3a+4)}$ $a.s.$ The limiting expression now converges to zero as $a \to \infty$ and to one as $a \to 0$. The quantity $\frac{\Gamma_{\lfloor an\rfloor,n}}{\lfloor an\rfloor}$ gives a measure of the reduction in complexity resulting from using the dominating set as a representation for the entire target class.

3.2.1 Proof Sketch

While Theorem 8 is a strong law of large numbers for the internal components, unfortunately the $\Gamma_{n,m}^i$ are not independent random variables. Due to this dependence we cannot apply the standard strong law of large numbers. Attempts to compute higher moments have resulted in complicated expressions which have not been useful in establishing convergence. We circumvent this problem with an approach that establishes the strong law of large numbers for the cardinality of a solution of a CCP in a Poisson process setting. We then transfer the result back to the original setting.

For a Poisson process Q we let Q_i denote the time of the ith arrival and $Q(t)$ (for $t \in \mathbb{R}_+$) denote the number of arrivals in Q before time t. Consider two one-dimensional Poisson processes, S and T, with common rate λ with $0 < \lambda < \infty$. Points of S play the role of target class points and points in T play the role of non-target class. We let C_i be the number of S points in (T_{i-1}, T_i) (where $T_0 = 0$) and Ψ_i be the minimum number of covering balls needed to cover the C_i points of S in (T_{i-1}, T_i).

We consider Γ_n', defined as the solution to the CCP on the points of S and T in $(0, T_{n+1})$. This CCP has exactly n non-target class points and a random number, $N_n = S(T_{n+1})$, of target class points. It has an advantage over our original CCP; the Ψ_i (analogous to the internal and external components) are independent random variables, allowing the application of the standard Strong Law of Large Numbers. A simple calculation in this Poisson process setting evaluates the limit as $\frac{a(13a+12)}{3(a+1)(3a+4)}$.

Using the conditional uniformity property of Poisson processes (see section 3.2.2), and a standard density transformation result, we see that the N_n points of S and the n points of T have the same distribution as the order statistics of $N_n + n$ observations of a uniform distribution on $(0, T_{n+1})$. Rescaling the interval $(0, T_{n+1})$ to $(0, 1)$ does not change the value of Γ_n'. For each n, we correct the number of target class points as follows: If $N_n < \lfloor an \rfloor$, we add $\lfloor an \rfloor - N_n$ points $S_1', S_2', \ldots, S_{\lfloor an \rfloor - N_n}'$ which are uniformly distributed on $(0, 1)$, mutally independent and independent of the Poisson processes. If $N_n > \lfloor an \rfloor$, we choose a random subset of $N_n - \lfloor an \rfloor$ points from $\{S_1, S_2, \ldots, S_{N_n}\}$ with all subsets of size $N_n - \lfloor an \rfloor$ equally likely, to remove from consideration. We then calculate a revised CCP solution with cardinality Γ_n on this corrected set of points. The random variable Γ_n in the Poisson process setting has an identical distribution with $\Gamma_{\lfloor an \rfloor, n}$ in the original setting.

Adding or removing a target class point affects the solution of the one-dimensional CCP by at most one. The number of points to be added or removed, $N_n - \lfloor an \rfloor$, form a random walk that arises naturally from the Poisson processes S and N. The fluctuations in the random walk are sufficiently small that the effect of adding or removing points is negligible in the limit. Combining these ideas, we show that the almost sure limits of $\frac{\Gamma'_n}{n}$ and $\frac{\Gamma_n}{n}$ are identical.

To transfer the result from Γ_n in the modified Poisson process setting to $\Gamma_{\lfloor an \rfloor, n}$ in the original setting, there is one additional complication to overcome. While the marginal distributions of Γ_n and $\Gamma_{\lfloor an \rfloor, n}$ are identical for each n, the joint distributions of $\{\Gamma_n;\ n = 1, 2, \ldots\}$ and $\{\Gamma_{\lfloor an \rfloor, n};\ n = 1, 2, \ldots\}$ are not, due to the adding or removing of different sets of points for each n. However, if care is taken to demonstrate complete covergence for Γ_n, we obtain complete covergence (and therefore almost sure covergence) for $\Gamma_{\lfloor an \rfloor, n}$.

3.2.2 Poisson Representation

For proving a limiting result, it is useful to convert the model from one in which uniformly distributed points are added to a fixed interval, into a model where the limit corresponds to increasing the length of the interval. As mentioned above, we use a correspondence between uniformly distributed points and a Poisson process.

We rely upon two standard distributional results. First, from an undergraduate-level density transformation exercise, if $Q_1, Q_2, \ldots, Q_{n+1}$ are independent and identically distributed random variables with an Exponential distribution with parameter λ, then

$$\left(\frac{Q_1}{\sum_{i=1}^{n+1} Q_i}, \frac{Q_1 + Q_2}{\sum_{i=1}^{n+1} Q_i}, \frac{Q_1 + Q_2 + Q_3}{\sum_{i=1}^{n+1} Q_i}, \ldots, \frac{\sum_{i=1}^{n} Q_i}{\sum_{i=1}^{n+1} Q_i} \right)$$

has the same joint distribution as the order statistics of n independent Uniform[0,1] random variables. Secondly, recall the "conditional uniformity" property of Poisson processes: If Q is a Poisson process and $Q(t) = n$, then the n points of Q in $[0, t]$ conditionally have the same distribution as the order statistics of n independent Uniform$(0, t)$ random variables.

We consider the T process on $(0, T_{n+1})$. By the first property above, the first n points in the T process may be considered to be uniformly distributed on $(0, T_{n+1})$. At time T_{n+1}, there is a random number of S points, N_n. If we further condition on $N_n = m$, then both S and T points are uniformly distributed on $(0, T_{n+1})$. By rescaling the interval, the class cover problem on $S_1, S_2, \dots, S_m$ and $T_1, T_2, \dots, T_n$ is equivalent to the CCP on $\frac{S_1}{T_{n+1}}, \frac{S_2}{T_{n+1}}, \dots, \frac{S_m}{T_{n+1}}$ and $\frac{T_1}{T_{n+1}}, \frac{T_2}{T_{n+1}}, \dots, \frac{T_n}{T_{n+1}}$. Therefore, in the case where we stop the T process at T_{n+1} and condition on $N = m$, the size of the solution to the CCP on the S and T points has the same distribution as $\Gamma_{m,n}$.

3.2.3 Expected Value of Internal Components

For now, we return to observing the Poisson processes in $(0, T_{n+1})$. Recall that we let Γ'_n represent the size of a solution to the CCP on $S_1, \dots, S_{N_n}$ and $T_1, \dots, T_n$. Also C_i is the number of S points in (T_{i-1}, T_i) and Ψ_i is the minimum number of covering balls needed to cover the C_i points of S in (T_{i-1}, T_i). We proceed by showing $E[\Psi_i] = E[\Psi_2] = \frac{a(13a+12)}{3(a+1)(3a+4)}$ $\forall i \in \{2, \dots n\}$. By conditional uniformity and Lemma 7 we see that the distribution of Ψ_i depends only on the value of C_i. By the lack of memory property of the exponential distribution, $C_i = Z_i - 1$ where Z_i is a geometric random variable with parameter $p = \frac{1}{a+1}$. Therefore since the C_i are identically distributed, Ψ_i for $i \in \{2, \dots, m\}$ are also identically distributed. We now calculate $E[\Psi_2]$.

$$P[\Psi_2 = 0] = P[C_2 = 0] = \frac{1}{a+1}$$

and

$$\begin{aligned}
P[\Psi_2 = 1] &= \sum_{k=1}^{\infty} P[\Psi_2 = 1 | C_2 = k] P[C_2 = k] \\
&= \sum_{k=1}^{\infty} \left[\frac{5}{9} + \left(\frac{4}{9} \right) 4^{1-k} \right] \frac{a^k}{(a+1)^{k+1}} \\
&= \frac{5a}{9(a+1)^2} \sum_{k=0}^{\infty} \left(\frac{a}{a+1} \right)^k + \frac{4a}{9(a+1)^2} \sum_{k=0}^{\infty} \left(\frac{a}{4(a+1)} \right)^k \\
&= \frac{5a}{9(a+1)} + \frac{16a}{9(a+1)(3a+4)} \\
&= \frac{5a^2 + 12a}{3(a+1)(3a+4)},
\end{aligned}$$

and by subtraction,

$$P[\Psi_2 = 2] = 1 - P[\Psi_2 = 0] - P[\Psi_2 = 1] = \frac{12a^2}{3(a+1)(3a+4)},$$

from which we obtain

$$E[\Psi_2] = \frac{a(13a+12)}{3(a+1)(3a+4)} = g(a). \tag{3.8}$$

3.2.4 Complete Convergence of Γ'_n

A sequence of random variables L_i is said to have complete convergence to a random variable L if for every $\epsilon > 0$

$$\sum_{i=1}^{\infty} P\left[|L_i - L| \geq \epsilon\right] < \infty.$$

We now show complete covergence (and thus almost sure convergence [23]) of $\frac{\Gamma'_n}{n}$ to $E[\Psi_2] = \frac{a(13a+12)}{3(a+1)(3a+4)}$.

$$\begin{aligned}\sum_{n=1}^{\infty} P\left[\left|\frac{\Gamma'_n}{n} - E[\Psi_2]\right| \geq \epsilon\right] &= \sum_{n=1}^{\infty} P\left[\left|\sum_{i=0}^{n} \Psi_i - E[\Psi_2]n\right| \geq n\epsilon\right] \\ &= \sum_{n=1}^{\infty} P\left[\left|\Psi_1 + \Psi_{n+1} - E[\Psi_2] + \sum_{i=2}^{n}(\Psi_i - E[\Psi_2])\right| \geq n\epsilon\right] \\ &\leq \sum_{n=1}^{\infty} P\left[\left|\Psi_1 + \Psi_{n+1} - E[\Psi_2]\right| + \left|\sum_{i=2}^{n}(\Psi_i - E[\Psi_2])\right| \geq n\epsilon\right]\end{aligned}$$

which, using the fact that $0 \leq \Psi_i \leq 1$ for $i \in \{1, n+1\}$ and $0 \leq E[\Psi_2] \leq \frac{13}{9}$,

$$\leq \sum_{n=1}^{\infty} P\left[\left|\sum_{i=2}^{n}(\Psi_i - E[\Psi_2])\right| \geq n\epsilon - 2\right]$$

and then using the fourth moment version of Markov's inequality,

$$\leq \sum_{n=1}^{\infty} \frac{E[|\sum_{i=2}^{n}(\Psi_i - E[\Psi_2])|^4]}{(n\epsilon - 2)^4}. \tag{3.9}$$

We bound the numerator in the preceding equation by expanding the fourth power of the sum. We use the fact that $E[\Psi_i - E[\Psi_2]] = 0$ for $i = 2, \ldots, n$.

$$\begin{aligned} E\left[\left|\sum_{i=2}^{n}(\Psi_i - E[\Psi_2])\right|^4\right] &= E\left[\sum_{i=2}^{n}(\Psi_i - E[\Psi_2])^4\right] \\ &\quad + E\left[\sum_{i=2}^{n}\sum_{j=2}^{i-1} 2(\Psi_i - E[\Psi_2])^2(\Psi_j - E[\Psi_2])^2\right] \\ &= (n-1)E[(\Psi_2 - E[\Psi_2])^4] \\ &\quad + (n-1)(n-2)E\left[(\Psi_2 - E[\Psi_2])^2\right]^2 \end{aligned}$$

since the Ψ_i are independent and identically distributed and since $E[(\Psi_2 - E[\Psi_2])^2] < 4$, we have

$$\leq Cn^2 \qquad \text{for some } C \in \mathbb{R}.$$

Now we may finish showing complete convergence since

$$\begin{aligned} \sum_{n=1}^{\infty} P\left[\left|\frac{\Gamma'_n}{n} - E[\Psi_2]\right| \geq \epsilon\right] &\leq \sum_{n=1}^{\infty} \frac{E[|\sum_{i=2}^{n}(\Psi_i - E[\Psi_2])|^4]}{(n\epsilon - 2)^4} \qquad \text{by (3.9)} \\ &\leq \sum_{i=2}^{n} \frac{Cn^2}{(n\epsilon - 2)^4} \\ &< \infty. \end{aligned}$$

Thus we have shown the complete covergence of $\frac{\Gamma'_n}{n}$ to $E[\Psi_2] = \frac{a(13a+12)}{3(a+1)(3a+4)}$. This is similar to our desired result, but we need to correct the number of S points in such a way that the corrected set has the correct distribution.

3.2.5 Adding and Deleting Points

We would like to prove convergence results about $\Gamma_{\lfloor an \rfloor, n}$ working with our current result about Γ'_n. To make this connection, we add or remove the necessary number of S points (exactly $|N_n - \lfloor an \rfloor|$) in a uniformly random way and then show that $|N_n - \lfloor an \rfloor|$ is not asymptotically large enough to change the limit. Note that once we condition on N_n to determine the number of points to be added or deleted, the S points will be uniformly distributed on (T_0, T_{n+1}). We then add or delete (as appropriate) $|N_n - \lfloor an \rfloor|$ S points in a uniform way. The remaining $\lfloor an \rfloor$ points of S are therefore uniformly distributed. Let the random variable Γ_n represent the size of a solution to the class cover problem on this new corrected set of points. Note that $\Gamma_{\lfloor an \rfloor, n}$ has the same distribution as Γ_n.

We now study the fluctuations of $|N_n - \lfloor an \rfloor|$. As before, let C_i denote the number of S points in $[T_{i-1}, T_i]$. Note that, as mentioned in the calculation of $E[\Psi_2]$, by the lack of memory property of the exponential distribution, $C_i = Z_i - 1$ where Z_i has geometric distribution with parameter $\frac{1}{a+1}$. We define a random variable $E_n := |N_n - \lfloor an \rfloor|$. We will show a bound on the probability of large deviations of E_n. If $N_n - \lfloor an \rfloor < 0$, we must add E_n points to S, while if $N_n - \lfloor an \rfloor > 0$, we must delete E_n points from S. We use Chernoff's bounds [36] to obtain exponential probability bounds on the number of points added or removed. For $0 < \epsilon \leq 1$,

$$\begin{aligned} P[E_n \geq \epsilon n] &= P[|N_n - \lfloor an \rfloor| \geq \epsilon n] \\ &= P[N_n - \lfloor an \rfloor \geq n\epsilon] + P[\lfloor an \rfloor - N_n \geq n\epsilon] \end{aligned}$$

and

$$\begin{aligned} P[N_n - \lfloor an \rfloor \geq \epsilon n] &= P[N_n - an + \delta \geq \epsilon n] \quad \text{for } 0 \leq \delta < 1 \\ &= P[N_n - an + \delta - \epsilon n \geq 0]. \end{aligned}$$

Let $M_L(t)$ denote the moment generating function of a random variable L.

$$P[N_n - an + \delta - \epsilon n \geq 0] \leq M_{N_n}(t) M_{-an+\delta-\epsilon n}(t)$$

by Chernoff's bound for $0 < t < \ln \frac{a+1}{a}$

$$\begin{aligned}
&= (M_{C_i}(t))^{n+1} e^{t(-an+\delta-\epsilon n)} \\
&= (M_{Z_i}(t) M_{-1}(t))^{n+1} e^{t(-an+\delta-\epsilon n)} \\
&= \left(\frac{e^t}{1+a-ae^t} e^{-t} \right)^{n+1} e^{t(-an+\delta-\epsilon n)} \\
&= \left((1+a-ae^t) e^{t(a+\epsilon)} \right)^{-n} \frac{e^{t\delta}}{1+a-ae^t}
\end{aligned}$$

and letting $\alpha_1(t) := (1+a-ae^t)e^{t(a+\epsilon)}$

$$= \alpha_1(t)^{-n} \frac{e^{t\delta}}{1+a-ae^t}.$$

Since $\alpha_1(0) = 1$ and $\alpha_1'(0) = \epsilon > 0$ we see that we can always choose a value t_1 such that $\alpha_1(t_1) > 1$. Thus

$$P[N_n - \lfloor an \rfloor \geq \epsilon n] \leq c_1 \alpha_1(t_1)^{-n} \tag{3.10}$$

where c_1 is a positive real constant. Similarly we can show that

$$\begin{aligned}
P[\lfloor an \rfloor - N_n \geq \epsilon n] &= P[an - \delta - N_n \geq \epsilon n] \\
&\leq P[N_n + \epsilon n - an \leq 0] \\
&\leq M_{N_n}(t) M_{\epsilon n - an}(t) \qquad \forall t < 0 \\
&= \left((1+a-ae^t) e^{t(a-\epsilon)} \right)^{-n} \frac{1}{1+a-ae^t}
\end{aligned}$$

and, letting $\alpha_2(t) := (1+a-ae^t)e^{t(a-\epsilon)}$, we obtain

$$= \alpha_2(t)^{-n} \frac{1}{1+a-ae^t}.$$

Since $\alpha_2(0) = 1$ and $\alpha_2'(0) = -\epsilon < 0$ we see that we can always choose a value t_2 such that $\alpha_2(t_2) > 1$. Thus

$$P[\lfloor an \rfloor - N_n \geq \epsilon n] \leq c_2 \alpha_2(t_2)^{-n} \tag{3.11}$$

for some positive real constant c_2. Combining equations (3.10) and (3.11) gives the desired result,

$$P[E_n \geq \epsilon n] \leq c_3 c_4^{-n} \tag{3.12}$$

for $c_3 > 0$ and $c_4 > 1$.

3.2.6 The Effect of Adding or Deleting Points

We also observe that the addition or deletion of one target class point changes the cardinality of the solution to the one dimensional CCP by at most one. We use the notation of section 2.4 in the proof of the following Lemma.

Lemma 8. *Let $\mathcal{X}, \mathcal{Y}$ be finite subsets of $\mathbb{R}$ and consider $\mathcal{X}$ to be the target class. Let D be the CCCD induced by $\mathcal{X}, \mathcal{Y}$ and D^- be the CCCD formed from $\mathcal{X} - \{x\}, \mathcal{Y}$ where x is some element of $\mathcal{X}$. Then $|\gamma(D) - \gamma(D^-)| \leq 1$.*

Proof. Let $\mathcal{X}, \mathcal{Y}$ be finite subsets of $\Re$ with $|\mathcal{X}| = n$ and $|\mathcal{Y}| = m$. Case 1. Suppose $x < y_{(1)}$ (respectively $x > y_{(m)}$). Then only D_1 (respectively D_{m+1}) are affected by the removal of x. Also since $\gamma(D_1)$ and $\gamma(D_{m+1})$ must be either zero or 1, then it must be that $|\gamma(D) - \gamma(D^-)| \leq 1$. Case 2. Suppose $x \in (y_{(i)}, y_{(i+1)})$ for $i \in \{1, 2, \ldots, m-1\}$. Again, only D_i is affected by the removal of x. If $n_i > 1$ then $\gamma(D_i)$ may be either zero, one or two. We must rule out the case that $\gamma(D_i)$ changes from two to zero because of the removal of x. (We need not consider the case that $\gamma(D_i)$ switches from zero to two since $x \in (y_{(i)}, y_{(i+1)}) \Rightarrow n_i > 0 \Rightarrow \gamma(D_i) > 0$.) If $\gamma(D_i) = 2$ before x is removed then it must be the case that $n_i > 1$ and therefore $n_i \geq 1$ after x is removed. Therefore $\gamma(D_i)$ must be at least one after x is removed. Therefore $|\gamma(D) - \gamma(D^-)| \leq 1$. □

3.2.7 Complete Covergence of $\frac{\Gamma_{n,n}}{n}$

If we let $D_n = \Gamma_n - \Gamma'_n$, then Lemma 8 implies that $|D_n| \leq E_n$. We obtain our result as follows.

$$\begin{aligned}
\sum_{n=1}^{\infty} P\left[\left|\frac{\Gamma_{n,n}}{n} - E[\Psi_2]\right| \geq \epsilon\right] &= \sum_{n=1}^{\infty} P\left[\left|\frac{\Gamma_n}{n} - E[\Psi_2]\right| \geq \epsilon\right] \\
&= \sum_{n=1}^{\infty} P\left[\left|\frac{\Gamma'_n}{n} - E[\Psi_2] + \frac{D_n}{n}\right| \geq \epsilon\right] \\
&\leq \sum_{n=1}^{\infty} P\left[\left|\frac{\Gamma'_n}{n} - E[\Psi_2]\right| + \left|\frac{E_n}{n}\right| \geq \epsilon\right] \\
&\leq \sum_{n=1}^{\infty} P\left[\left|\frac{\Gamma'_n}{n} - E[\Psi_2]\right| \geq \frac{\epsilon}{2}\right] + \sum_{n=1}^{\infty} P\left[\left|\frac{E_n}{n}\right| \geq \frac{\epsilon}{2}\right] \\
&< \infty.
\end{aligned}$$

We have thus shown complete convergence, and therefore almost sure convergence, of $\frac{\Gamma_{n,n}}{n}$ to $E[\Psi_2]$.

Part III

Applications

Chapter 4

Classification

This chapter concentrates mainly on the applications of the constrained heterogeneous CCP to computational statistics. The main focus is on applying the CCP to statistical pattern classification.

4.1 Classification Fundamentals

Classification is a subfield of learning theory that has grown rapidly in the past century. The goal of learning theory is to enable a machine to gain knowledge from some set of examples. Computers have been used for decades to aid in the solution of difficult problems. Until recently, computers were most useful in tasks that had a known solution. That is, the computer needed to be instructed step by step on how to solve the problem at hand. For some problems, however, it is not known how to algorithmically instruct a machine to generate the correct output.

An example of such a problem is face detection; the ability to view an image and determine if there is one or more human faces within the image. Humans are very good at this task, but there is no easy "solution" to the face detection problem that we could impart to a machine. This is where learning theory comes in. We would like to "teach" a machine to learn. We have a notion of what makes something "facelike" because we have been exposed to many faces throughout our lives. This is what allows us to know we are looking at a human face even if it is one that we have never seen before. To teach a machine to detect faces in an image we will use the same method humans use. We will show the computer a collection of images and mark the images that have faces. Of course, we still need some way of instructing the computer to use these examples to gain knowledge.

Another example is credit card fraud detection. A credit card company may want to know if there is some pattern evident in its records that is indicative of a stolen credit card. In this case, the company may not be able to detect fraud efficiently by hand. They are hoping the machine will be able to find some pattern or structure that leads to better fraud detection.

Both of the preceding examples are problems in learning theory. The face detection problem is a classic supervised two class classification problem and the fraud problem is an unsupervised learning problem. In *supervised* learning, the learner is given a set of known examples in order to gain knowledge about the processes that generate the examples. This knowledge can be used for the tasks of regression, density estimation, pattern classification and others. We will focus on applying the CCP to supervised two-class classification or binary classification in this chapter. In *unsupervised* learning or *clustering* the goal is to find unknown patterns or structure in a given a set a of data. We will review a method for applying the CCP to clustering in Chapter 5. Kulkarni, Lugosi and Venkatesh's survey of classification [24] gives a more detailed look at the fundamentals of the field of learning theory. Other references include [10, 15]. We will begin our discussion by considering binary or two class classification.

Once again, we consider data in a dissimilarity space (Ω, d). In the two class case our Ω valued observations will have associated with them a Bernoulli random variable Y corresponding to the class of the observation. The probability that a general point is in class $j \in \{0,1\}$ is given by the *a priori* probability $p_j = P[Y = j]$. The distribution of an observation of known class j is given by the class conditional distribution $F_j(\cdot)$. In other words, data are drawn from the following distribution:

$$F(\cdot) = p_0 F_0(\cdot) + p_1 F_1(\cdot)$$

where $p_0 + p_1 = 1$. A random observation is generated as follows; first select a random class by observing the outcome of Y, then select a random observation X from Ω according to $F_Y(\cdot)$.

We are given as *training data* two finite, non-empty sets of class conditional Ω-valued observations, $\mathcal{X}_0$ and $\mathcal{X}_1$. The goal of classification is to design a classifier $g_{(\mathcal{X}_0,\mathcal{X}_1)} : \Omega \to \{0,1\}$, such that given an unlabeled observation X with true but unknown class label Y_X in $\{0,1\}$, the probability of misclassification $L(g) = P[g_{(\mathcal{X}_0,\mathcal{X}_1)}(X) \neq Y_X]$ is minimized. We denote $\hat{L}(g)$ as the experimental misclassification rate of a classifier g on a test set different than the training set. The optimal classifier is known as the *Bayes classifer* g^* [2, 24, 35] and is given by

$$g^*(x) = 1\{P[Y_X = 1|X = x] > P[Y_X = 0|X = x]\}. \tag{4.1}$$

Thus the optimal misclassification rate or *Bayes error rate* L^* is defined as

$$L^* := P[g^*(X) \neq Y_X].$$

We call the set where $P[Y_X = j|X = x] > P[Y_X = 1-j|X = x]$ the *discriminant region* for class $j \in \{0, 1\}$. If the densities f_0 and f_1 exist for the distributions, then an alternative definition of the discriminant region for class $j \in \{0, 1\}$ is $\{z \in \Omega : f_j(z) > f_{1-j}(z)\}$.

The construction of the Bayes classifier relies on the knowledge of the *a posteriori* probabilities $P[Y_X = j|X = x]$ $\forall x \in \Omega$ for $j \in \{0, 1\}$ or, in other words, knowledge of the class conditional distributions $F_j(x)$. In practice, these distributions are not known and we must attempt to create a classifier whose probability of misclassification is as low as possible. A classifier whose misclassification rate converges in probability (as the training sample size goes to infinity) to the Bayes error rate is called *consistent* or a consistent rule. If the convergence is almost sure, we say the classifier is *strongly consistent*. Of course consistent classifiers are attractive, but consistency alone should not rule the decision when choosing a classifier. Notions such as speed of convergence and complexity must be considered as well.

If there is reason to believe that the distributions are of a known family, for instance Normal, Exponential, etc., then a *parametric* classifier is used to estimate the parameters of the distributions. Parametric methods are very effective if the assumptions on the distributions are valid. However, it is often the case that there is little knowledge of the distributions. In such a case, it is necessary to develop classifiers which do not rely on parametric assumptions. There are two kinds of such classifiers, namely *semi-parametric* and *nonparametric*. In this section we introduce several methods for using the CCP to construct semi-parametric classifiers.

4.2 Classification with the CCP

When applying the CCP to classification, we use a generalization of the reduced nearest neighbor classifier [18, 19, 38] as a framework. The idea is to find a cover (independently) for both classes by choosing one class to be the target class, solving a CCP, and then switching the roles of target and non target class and solving a new CCP. We must also choose a *cover-dissimilarity* function $\rho : \Omega \times \mathcal{C} \to \mathbb{R}^+$ between new observations and a cover, where $\mathcal{C}$ is the space of all covers. The classifier is then

$$g(z) = \begin{cases} 0 & \text{if } \rho(z, C_0) < \rho(z, C_1), \\ 1 & \text{if } \rho(z, C_1) < \rho(z, C_0), \\ -1 & \text{otherwise} \end{cases} \tag{4.2}$$

where C_0, C_1 are the covers of class zero and one respectively and an output of -1 represents no decision.

If we use the CCP1 to find our covers, notice that all covering balls extend to the nearest non-target class point. We could shrink each ball to the farthest target class point covered by the ball. This change would affect only the classifier formed, but not the choice of cover. We formalize this by introducing the parameter τ. If we are considering a target class $\mathcal{X}_0$ and a non-target class $\mathcal{X}_1$ the final radius of a covering ball for a point x is given by

$$r_x := (1 - \tau)d(x, q_x) + \tau d(x, u_x),$$

where u_x is defined as the closest non-target point to x,

$$u_x := \arg \min_{z \in \mathcal{X}_1} d(x, z),$$

and q_x is defined as

$$q_x := \arg \max_{z \in \mathcal{X}_0} \{d(x, z) : d(x, z) < d(x, u_x)\}.$$

In other words q_x is the furthest target class point from x that is closer than the closest non-target class point u_x. Figure 4.1 provides an illustration of the effect of changing the value of τ on a covering ball. Notice that the changing τ does not change the makeup of the cover, only the size of the individual covering balls.

When choosing our covers for classification we would like to choose the pair of covers which induce the best classifier, that is the classifier with the lowest misclassification rate. The objec-

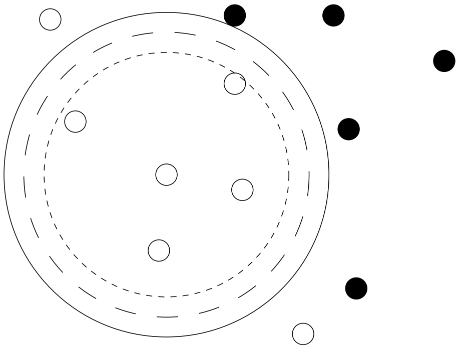


Figure 4.1: Illustration of τ. The solid line represents the case $\tau = 1$; the middle circle represents $\tau = 0.5$ and the smallest circle represents $\tau = 0.0001$.

tive function in this situation is

$$\begin{aligned} \min \quad & \hat{L}(g_{C_0,C_1}(z)) \\ \text{such that} \quad & C_0, C_1 \text{ satisfy some CCP.} \end{aligned} \tag{4.3}$$

This is in contrast to the CCP we studied in Chapter 2 where our objective was to find the cover with the smallest cardinality. Unfortunately this optimization presents a combinatorial explosion of possible covers and is intractable. We will instead try to model the discriminant regions of the two classes with the covers as found by the class cover problem.

To estimate the discriminant regions with our CCP method, points in the discriminant region for class j must be closer (under the cover dissimilarity) to the cover for class j than the cover for class $1-j$. We make the assumption that any point in the cover for class j but not in the cover for class $1-j$ should be classified as class j. It is therefore important that the cover dissimilarity function ρ we use to measure distance to a cover has the following property for two covers C_0, C_1

$$z \in C_j \cap C_{1-j}^c \Rightarrow \rho(z, C_j) < \rho(z, C_{1-j}) \qquad \text{for } j \in \{0, 1\}. \tag{4.4}$$

This assumption gives us the intuition that the covers should attempt to model the discriminant regions as accurately and efficiently as possible with the given training data.

We also need to consider two more cases. For the case $z \in C_j \cap C_{1-j}$, it is important to carefully consider the decision we make since we are likely to see such points often. For the case $z \in C_j^c \cap C_{1-j}^c$, it is not as important since it is likely any regions not in either cover are

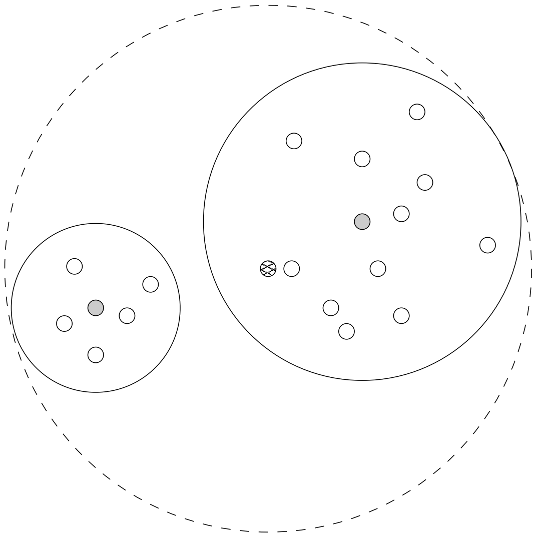


Figure 4.2: Comparing covering balls

relatively far from the majority of the data and have a low probability of occurring. Thus the correct classification or misclassification of these points will not greatly affect the performance of our classifier.

In our first two classifiers we attempt to minimize the cardinality of the cover in the construction of the classifier. Of course the minimum cardinality cover has the smallest complexity (among covers). Any objective function should keep cover complexity as low as possible to avoid overfitting. Unfortunately, finding the minimum cardinality cover is a difficult task with no known efficient (polynomial) solution. As demonstrated in Chapter 2, finding a solution to a CCP is equivalent to the dominating set problem on the representative CCCD. In practice, instead of finding an exact solution to the CCP we will find an approximate solution using the greedy algorithm presented in chapter 2.

When attempting to approximate complex discriminant regions using balls, smallest cardinality is not, in general, the best objective function. This is because the minimum cardinality cover is often made up of large radius balls that are not representative of the points they cover. To demonstrate our meaning of "representative", we would say the smallest ball containing a set of points S is a good representative of S (among balls). A poor representative is a ball centered at the point of S furthest from the mean of points in S. Figure 4.2 illustrates this point. The solid balls in the figure are excellent representatives of the points they cover, while the dashed ball is not. It is our goal that each ball in a cover be a good representative of the points it covers since each ball is attempting to model some part of the discriminant region. A minimum cardinality cover performs well, but performance could be improved by considering other objectives. We have considered statistical depth and we use a combination of point density and classifier performance in section 4.6.

Priebe et al. [34] investigate applications of the constrained heterogeneous CCP using α, β covers. We will review this method in sections 4.4 and 4.5. The focus of section 4.6 extends that work with data adaptive allowances for impureness and improperness. This is done by observing the local neighborhood while choosing the radius for the potential covering ball. We investigate classifiers which are similar to CCP classifiers in the next section.

4.3 Similar classifiers

While the CCP based classifiers are new, there are several classifiers which are similar in nature. Of course the reduced nearest neighbor classifier is similar since it provides the framework for our CCP based classifiers. Support vector machines are also similar to CCP based classifiers. Their relation is described in section 6.1.3. The most closely related classifier is the Reduced Coloumb Energy (RCE) Networks [15]. Classification using an RCE is done according to the following rule

$$g_{RCE}(z) := \begin{cases} 0 & \text{if } z \in \left\{ \bigcup_{x \in \mathcal{X}_0} B_x \cap \left(\bigcup_{x \in \mathcal{X}_1} B_x \right)^c \right\} \\ 1 & \text{if } z \in \left\{ \bigcup_{x \in \mathcal{X}_1} B_x \cap \left(\bigcup_{x \in \mathcal{X}_0} B_x \right)^c \right\} \\ -1 & \text{otherwise.} \end{cases}$$

where B_x is the largest ball centered at x that does not contain a member of the opposite class. The CCP based classifiers presented here utilize the same basic ideas as the RCE networks. The naive classifier presented in Section 4.4 is a natural extension of RCE networks incorporating the class cover problem.

4.4 Preclassifier

A naive or preclassifier is built using a minimum cardinality (or approximately minimum cardinality) pure and proper cover from each class. By switching the role of target class between $\mathcal{X}_0$ and $\mathcal{X}_1$, two different instances of the CCP can be solved, resulting in two covers C_0 and C_1.

For a cover C we define a simple cover-dissimilarity function as

$$\rho_N(z, C) = \begin{cases} 0 & \text{if } z \in \bigcup_{B \in C} B \\ 1 & \text{otherwise} \end{cases} \tag{4.5}$$

Given two covers C_0, C_1, the above cover-dissimilarity function in the nearest neighbor framework of (4.2) creates the following simple classifier $g : \Omega \to \{-1, 0, 1\}$ where,

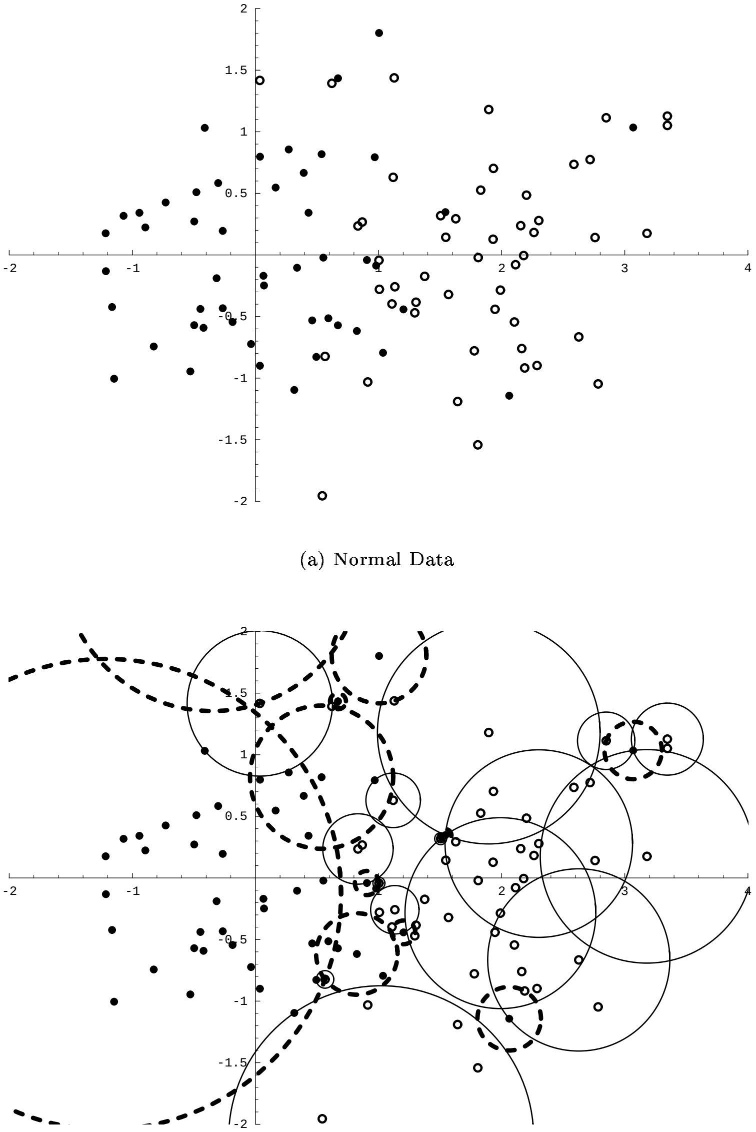
$$g_{pre}(z) = \begin{cases} 0 & z \in C_0 \cap C_1^c \\ 1 & z \in C_1 \cap C_0^c \\ -1 & \text{otherwise} \end{cases} \tag{4.6}$$

This cover-dissimilarity function makes no decision for (possibly) large regions in Ω. This may or may not be a drawback depending on the application. We can remedy this with a scaled cover-dissimilarity function. For a cover C made up of balls B_i with centers x_i and radius r_i, we define a new cover-dissimilarity function as

$$\rho_S(z, C) = \min_{\{i : B_i \in C\}} \frac{d(z, x_i)}{r_i} \tag{4.7}$$

Both dissimilarity functions in (4.5) and (4.7) have the property in (4.4).

Another drawback of the pre-classifier is its tendency to overfit. When trying to approximate the discriminant region for one class (finding a cover), it might be better to allow our covering balls to contain a few "contaminating" points from the other class. We might also allow our cover to miss a few "outlying" target class points. Ideally, these points would be points that fall in the opposing class' discriminant region. Figure 4.3 illustrates this point. Figure 4.3(a) shows data drawn from two distributions; the solid black disks (class zero) are 50 observations from a normal distribution with mean at $(0, 0)$ and the small circles (class one) are 50 observations from a normal distribution with mean $(2, 0)$. Both normal distributions have the identity covariance matrix. We will see this data set again in Chapter 6 and it will be called Model 2. In Figure 4.3(b) we see pure proper covers for both classes obtained using the greedy algorithm. The dashed and solid balls are covering balls for class zero and one respectively. The Bayes optimal



(a) Normal Data

(b) Pure proper cover

Figure 4.3: Two class Normal data and pure proper cover

rule for data drawn from these two distributions is

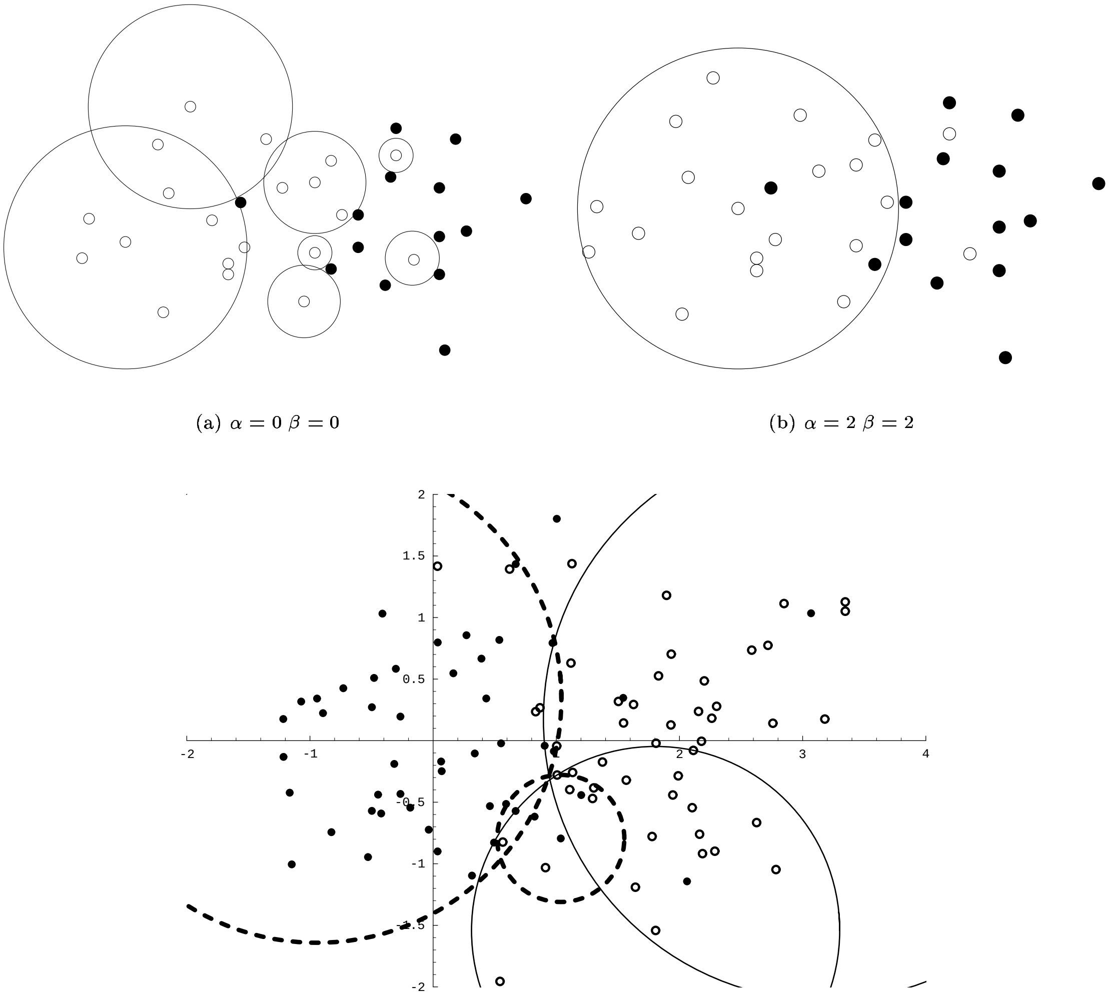
$$g_B(z) = \begin{cases} 0 & \text{if } z[1] < 1, \\ 1 & \text{otherwise.} \end{cases}$$

We would therefore like any cover for class zero to be made up of balls centered at points with first coordinate less than one. Notice that since we are requiring proper covers, as the number of training points increases there will be an increasing number of covering balls centered at points with first coordinate greater than one. We also notice that the cover for class one is made unnecessarily complex because it must avoid the class zero points. We begin to address these observations with the α, β CCP.

4.5 α, β CCP

In this section we present the CCP which involves impure and improper covers; see also [34]. Let α and β be nonnegative integers. Without loss of generality, in this section we will assume that $\mathcal{X}_0$ is the target class. Ideally we would like to require our cover for $\mathcal{X}_0$ to miss at most α points from $\mathcal{X}_0$ and to contain at most β points from $\mathcal{X}_1$. This problem is certainly NP-Hard for $\beta \geq 0$ since now we must consider $\beta + 1$ balls centered at each target class point and find a smallest subset of balls (a cover) so that their union contains at least $|\mathcal{X}_0 - \alpha|$ target class points and at most β non-target class points. It is not even clear how to efficiently find an approximate solution to this problem. We simplify the situation by redefining β to be the number of non-target class points in each covering ball. We define the covering ball B_i^β at each target class point $x_i \in \mathcal{X}_0$ as $\{x \in \Omega : d(x_i, x) < d_\beta(x_i, \mathcal{X}_1)\}$ where $d_\beta(x_i, \mathcal{X}_1)$ is the $\beta + 1$th smallest distance from x_i to elements in $\mathcal{X}_1$. Now a minimum cardinality cover for the target class is the smallest collection of balls B_i^β such that at least $|\mathcal{X}_0| - \alpha$ target class points are covered. Again, because of the difficulty of finding a minimum cardinality cover, we will find approximate solutions using the greedy algorithm of Section 4.2. Clearly we can achieve a pure and proper cover by setting $\alpha = \beta = 0$. Below (Figure 4.4) are illustrations of a pure and proper cover and an α, β cover. We also demonstrate α, β covers on the normal data of Figure 4.3(a) in Figure 4.4(c). In both cases the α, β covers are more simple and also more representative than the pure proper covers.

This technique greatly improves on the pre-classifier. Careful choice of the parameters α and



(a) $\alpha = 0$ $\beta = 0$

(b) $\alpha = 2$ $\beta = 2$

(c) α, β covers of normal data

Figure 4.4: Figure 4.4(a) shows a pure and proper cover of the black points. Figure 4.4(b) shows a cover with $\alpha = 1$ $\beta = 1$. Figure 4.4(c) shows covers using $\alpha = 5$ and $\beta = 5$ for both classes (compare with Figure 4.3(b)).

β for each class can lead to significant improvement in classifier performance as demonstrated in Chapter 6. Another improvement is reduced classifier complexity. For example, the pure proper covers shown in Figure 4.3(b) contain 10 balls for class zero and 13 balls for class one. The α, β covers in Figure 4.4(c) are each made of only two balls.

While the α, β CCP technique is an improvement, its assumption that every covering ball should contain β non-target class points is suboptimal. Ideally, a covering ball for the target class should cover only non-target class points that fall in the discriminant region of the target class. For example, observe that the small dashed ball in Figure 4.4(c) extends well into the discriminant region for the other class. This happens because the ball was forced to capture five non-target class points.

A second drawback of the α, β cover method is that the parameters α and β are chosen by the user. While the parameters α and β have a physical interpretation, it is often difficult to justify the choice of a particular set of parameters when the class conditional distributions are unknown. What we would like is some way to let each covering ball determine its own radius based on its local neighborhood. For example, if we increase the radius of a covering ball and the result is to capture one new non-target class point and ten new target class points, then we might say that change in radius is worthwhile. In this way we are accomplishing two tasks. We are adaptively choosing the α and β parameters in the sense mentioned in the beginning of section 4.5; that is, α is the number of target class points a cover is allowed to miss and β is the number of non-target class points a cover may contain. We are also choosing the radius for each ball in a more intelligent manner. We attempt to formalize this idea in the next section.

4.6 Random Walk CCP

We propose a new adaptive strategy for choosing the radii for covering balls. Our intent is to have the same effect as the α and β parameters (sensitivity to contamination and outliers), while behaving in a more local manner. Instead of choosing global parameters α and β, we allow each ball to choose its own radius based on the local density of target and non target class points. This will be done by observing a special random walk for each target class point. For this reason we call this classifier the random walk classifier or RW CCP classifier. We will also choose our covers in a slightly different way. Instead of choosing a minimum cardinality cover, we will attempt to find a cover that is a good representative of the points it covers. We will do this in a greedy fashion, choosing the best ball to add to the cover at each iteration. Finally, we will use a generalization of the cover dissimilarity function in (4.7) as the cover dissimilarity function.

4.6.1 Choosing Radii

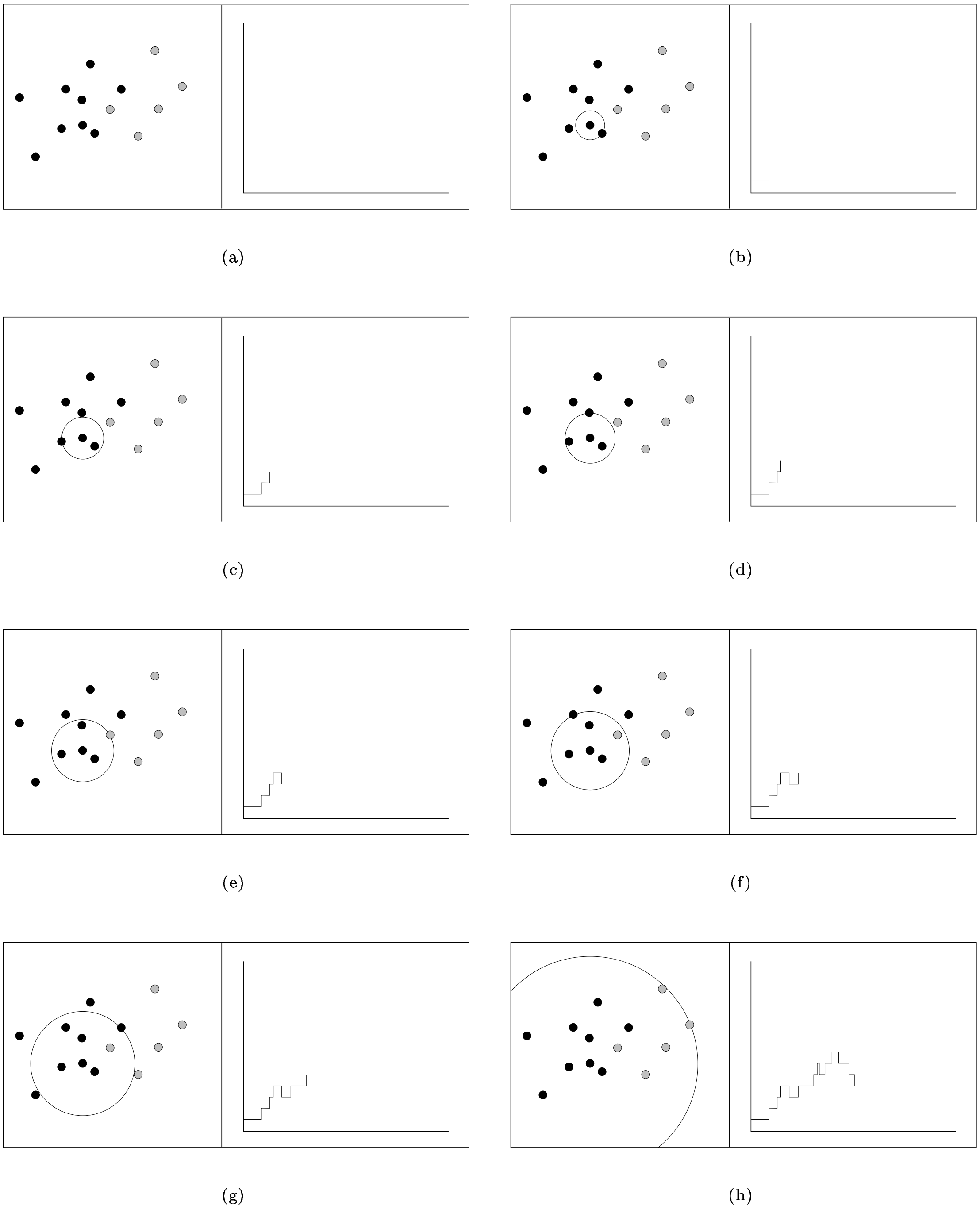
For each point x_i in the target class $\mathcal{X}_0$, we will examine a random walk R_{x_i} which is defined as follows. For any $r \in \mathbb{R}_+$ let

$$R_{x_i}(r) = |\{x \in \mathcal{X}_0 : d(x_i, x) \leq r\}| - |\{x \in \mathcal{X}_1 : d(x_i, x) \leq r\}|.$$

A way of visualizing this as a random walk is to think of a ball of radius r centered at x_i. As r increases from zero, the ball will encounter points from $\mathcal{X}_0$ and $\mathcal{X}_1$. Each time the ball encounters a target class point or non-target class point, the random walk goes up by one or down by one respectively. See Figure 4.5 for an illustration of this. A large positive value of $R_{x_i}(r)$ indicates a relatively high local density of target to non-target class points in a ball of radius r around x_i. If there are an unequal number of target class and non-target class points (consider unequal priors on the data) then we change the definition of the random walk. Suppose $|\mathcal{X}_0| = n_0$ and $|\mathcal{X}_1| = n_1$. A more general definition for the random walk is

$$R_{x_i}(r) = \frac{n_1}{n_0}|\{x \in \mathcal{X}_0 : d(x_i, x) \leq r\}| - |\{x \in \mathcal{X}_1 : d(x_i, x) \leq r\}|.$$

Once we have the random walk for some target class point x_i, we will use it to choose a radius for the ball B_i. But how shall we do this? We will argue for one possible way. Before we can do so, we must understand the goal or purpose of each individual covering ball. What follows here is a rather casual discussion designed to understand the intuition behind our methodology. Let us suppose that the training data are independent observations drawn from the class conditional distributions F_0 and F_1. We also assume that the densities (f_0 and f_1) for these distributions exist and are continuous. Let D_j be the discriminant region for class $j \in \{0, 1\}$. To approximate D_j, each covering ball B_i centered at a point x_i in class $j \in \{0, 1\}$ should be the largest ball such that $B_i \cap D_{1-j} = \emptyset$. Because our training samples are finite, it is impossible to determine



(a)

(b)

(c)

(d)

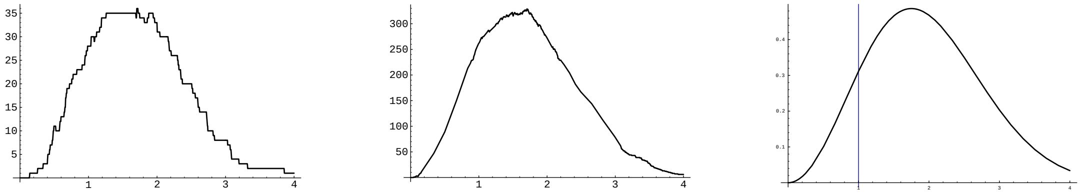
(e)

(f)

(g)

(h)

Figure 4.5: Snapshots of a random walk.



(a) $R_{(0,0)}(r)$ (50 points)

(b) $R_{(0,0)}(r)$ (500 points)

(c) $R^*_{(0,0)}(r)$

Figure 4.6: Comparison of random walks with ideal curve (Normal data)

the exact largest radius for B_i so that $B_i \cap D_{1-j} = \emptyset$. We will instead attempt to find a radius $r^*_{x_i}$ for a point x_i such that B_i covers as many target class points as possible while keeping the intersection $B_i \cap D_{1-j}$ small. In this way, our cover will become an approximation of the discriminant region D_j for each class.

Consider a point $x \in \mathcal{X}_0$. Let us first consider a simple case in which $f_0(z) = a$ and $f_1(z) = b$ for $a, b \in [0, \infty)$ with $a > b$ for all z in a disc D with radius r_D centered at x. If we observe n observations from each class, then as n approaches infinity, the curve $\frac{R_x(r)}{n}$ approaches

$$\frac{R^*_x(r)}{n} := \int_{B(x,r)} f_0(z)dz - \int_{B(x,r)} f_1(z)dz \tag{4.8}$$

for all $r \leq r_D$. For example, in two dimensions this function evaluates to $(a-b)\pi r^2$. A possible conclusion is that if we observe the function $R_x(r)$ behaving like Equation (4.8) on some interval $(0, r_m)$ we may assume that the ball $B(x, r_m)$ is contained in the discriminant region for $\mathcal{X}_0$.

Of course we must prepare for distributions which are not constant, but the previous example will act as a guide in considering more complicated situations. For example, Figure 4.6 shows the plot of $R_{(0,0)}(r)$ for data from Model 2 introduced in section 4.4.

In Figure 4.6(a) and 4.6(b), we see the plot of $R_{(0,0)}(r)$ when 50 and 500 points respectively have been drawn from each class. Figure 4.6(c) shows $R^*_{(0,0)}(r)$ as derived from equation (4.8). For any target class point, we would like to choose the ball of maximum radius that does not intersect the discriminant region of the other class. In Model 2, this is a ball of radius one for the point located at $(0,0)$. The line $r = 1$ is displayed in Figure 4.6(c). Ideally, we would like to characterize the behavior of $R_x(r)$ at the point when the ball first intersects the discriminant region for the non-target class. Unfortunately this task is difficult if not impossible. Figure 4.7

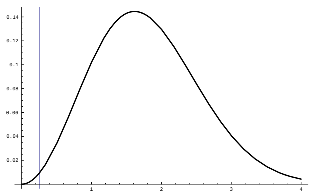


Figure 4.7: $R^*_{(0.75,0)}(r)$

(showing a plot of $R^*_{(0.75,0)}(r)$) as compared to Figure 4.6(c) illustrates this. Both curves have approximately the same shape (on different scales), yet the ideal radius for a ball centered at $(0.75, 0)$ is 0.25. The shapes of the two curves seem to be very different at these points.

So how should we determine the radius for a point? One possible method is to look for the "knee" of the curve. Recall from the simple example with constant densities that the $R_x(r)$ curve will increase like r^d (where d is the dimension of the data) in the discriminant region for the target class. Vaguely stated, in the case of non-constant densities, we would like to expand a covering ball as long as $R_x(r)$ is increasing fast enough. The word "enough" implies that the "knee" of the curve is open to interpretation.

Since $R_x(r)$ curves are step functions, we cannot work easily with derivatives, which may be helpful in finding the knee. We will instead find the knee with the following formula

$$r^*_x = \arg\max_r \{R_x(r) - P_x(r)\}$$

where $P_x(r)$ is an increasing penalty function that biases toward choosing radii smaller than $\arg\max_r\{R_x(r)\}$. The choice of smaller radii as opposed to larger radii has two advantages; we can more accurately approximate the discriminant region with smaller balls, and our balls have a higher probability of lying completely in the target class' discriminant region. The choice of $P_x(r)$ has a large influence on the make-up of the cover. Our standard choice of $P_x(r)$ is a line through the origin with slope $s(x)$ where $s(x)$ is defined

$$s(x) := \delta \cdot \frac{\max_r\{R_x(r)\}}{r} \tag{4.9}$$

where $\delta \in [0, 1]$. The parameter δ indirectly influences the size of covering balls in the cover. For example, setting $\delta = 0$ is equivalent to setting $r^*_x = \arg\max_r R_x(r)$. Somewhat surprisingly, this choice of δ seems to work best in practice. Any radius larger than $\arg\max_r\{R_x(r)\}$ implies that the covering ball intersects with the discriminant region of the non-target class and thus is not considered. We have concluded that this approach to finding the knee of the curve warrants further investigation.

Finally, we note that the covering balls used in the RW-CCP are closed balls. This is because the optimal radius r^*, as chosen by the method outlined here, is always the exact distance between the center and some target class point since the function $R_x(r)$ only increases at target class points. Since the boundary of every covering ball is passing through a target class point we will include this point by making the covering balls closed.

4.6.2 Finding the Cover

Ideally we would like to find the cover which maximizes the performance of our classifier. Because of the combinatorial explosion of possibilities an exhaustive search is unreasonable. Instead we choose our cover greedily. That is, we will choose our cover one ball at a time, each ball attempting to improve the current classifier as much as possible. Instead of checking the performance of our classifier at each stage we will instead use a closely related surrogate test. To determine which ball to add next to the cover we will favor covering balls which most improve our preclassifier. That is we will favor balls with a high number of (as yet uncovered) target class points and a low number of non target class points. We will also consider the radius of a ball. If two covering balls contain the same number of target and non-target class points but have different radii, we will choose the ball with the smaller radii. This is because the smaller ball may be more representative of the points it is covering.

We begin by assigning a score, T_x, to each potential covering ball of radius r_x^* for a target class point x. We imagine $R_x(r_x^*)$ as a raw score since it represents a measure of the difference of the number of target class points and non-target class points in the covering ball. Again, because of the local nature of our CCP methodology, it will be advantageous to favor smaller covering balls over larger balls (as mentioned in section 4.6.1). This is achieved by imposing on $R_x(r^*)$ an increasing penalty function $p_x(r)$ that increases with radius. This penalty also has the effect of favoring balls which are good representatives of the points that they cover. For the ball centered at x we assign a score $T_x = R_x(r_x^*) - p_x(r_x^*)$ and we choose the ball with maximum score to add to the cover. We have found that a linear function serves as an effective penalty function. As we will see in the next paragraph, scores as well as radii for balls will be recomputed after a ball is added to the cover. The slope of the penalty function $p_x(r)$ should be dependent on the number of points still uncovered by the current cover. If at some point in the process of finding a cover n_u target class points remain to be covered we use the function

$$p_x(r) = \frac{n_u \cdot r}{2d_m}$$

as our penalty function where d_m is the largest distance from x to any other target class point.

After a ball is added to a cover, any points covered by that ball are disregarded and we recompute radii for each uncovered point and choose a new ball to add to the cover based on newly computed scores. We continue adding balls in this way until all target class points are covered. The algorithm for finding the covers for both classes is presented below.

Random Walk CCP Classifier Construction

Input: Training sets $\mathcal{X}_0, \mathcal{X}_1$ in dissimilarity space (Ω, d).

Output: Covers for $\mathcal{X}_0$ and $\mathcal{X}_1$.

```
For $j = 0, 1$
    Set $C_j = \mathcal{X}_j, C_{1-j} = \mathcal{X}_{1-j}, S = \emptyset$.
    While $C_j \neq \emptyset$.
        Compute radii $r_x^*$ for each $x \in \mathcal{X}_j$ as in Section 4.6.1.
        Compute scores $T_x$ for each $x \in \mathcal{X}_j$ as above.
        $x^* = \arg\max\{T_x : x \in \mathcal{X}_j\}$
        $S_j = S_j \cup \{x^*\}$.
        $C_j = C_j - \{B_{x^*} \cap \mathcal{X}_j\}, \quad C_{1-j} = C_{1-j} - \{B_{x^*} \cap \mathcal{X}_{1-j}\}$
return $S_0, S_1$.
```

The adaptive procedure for finding the covering ball radii makes it necessary to recompute the radii for all unchosen balls after a new ball is chosen for the cover. This is because of the difficulty in choosing the radius mentioned in the previous section. The assumption is that a ball chosen for the cover is a good representative of the points that it covers. Once these points are removed from consideration, it may be easier to find radii for some of the remaining points, that is, to represent some of the remaining points with balls. This idea is well illustrated with the following example.

Suppose the target class is uniformly distributed on the square $[\frac{1}{4}, \frac{3}{4}] \times [\frac{1}{4}, \frac{3}{4}]$ and the non target class is uniformly distributed on the unit square in two dimensions. Figure 4.8(a) shows 200 random observations from each class. Figures 4.8(b) and 4.8(c) show the $R_{(0.5,0.5)}(r)$ and $R_{(0.3,0.7)}(r)$ curves respectively. Notice if we choose $\delta = 0$ that the radius of the ball centered at $(\frac{1}{2}, \frac{1}{2})$ is ~ 0.27 and the radius of the ball centered at $(0.3, 0.7)$ is ~ 0.55. The covering ball centered at $(0.3, 0.7)$ is not representative of the target class points it is covering and is not considered because it has a relatively low score (~ 10). The covering ball centered at $(\frac{1}{2}, \frac{1}{2})$ is chosen first since it has the largest score (~ 100).

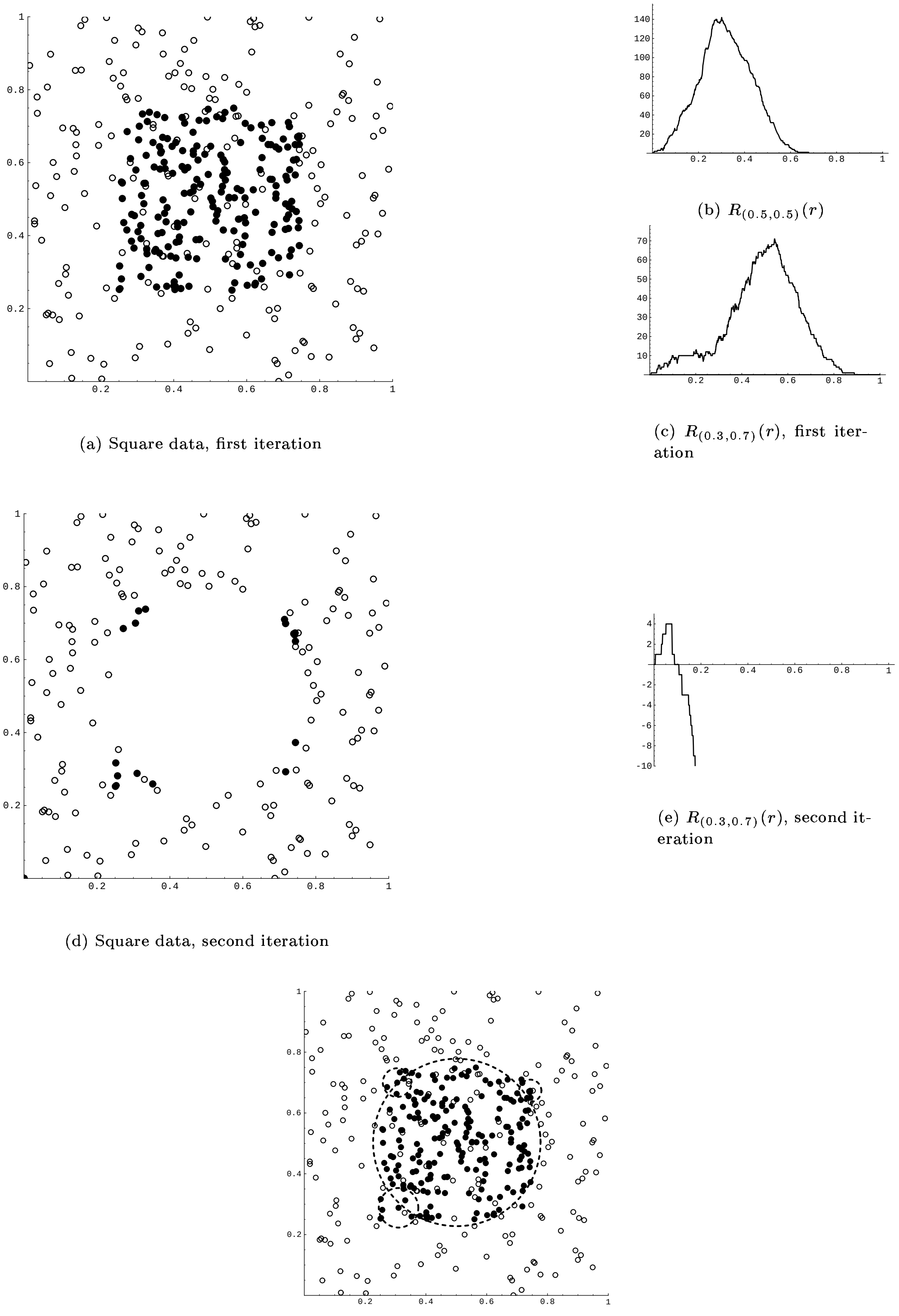
In the second iteration we remove points covered by the ball centered at $(\frac{1}{2}, \frac{1}{2})$ (See Figure 4.8(d)) and recompute radii and scores for uncovered target class points. Figure 4.8(e) shows the new random walk for the point $(0.3, 0.7)$. The radius of covering ball for the point $(0.3, 0.7)$ is a much more reasonable ~ 0.047. This covering ball is chosen next since it has the highest score (~ 4) among covering balls centered at the remaining uncovered points. Figure 4.8(f) shows the final cover for the target class.

This system of choosing the radius for each covering ball and then choosing a ball based on the score of that ball makes up for our difficulty in choosing the ideal radius for every covering ball. The algorithm starts out by trying to crudely model the discriminant region for the target class. It then uses smaller balls to fill in the uncovered areas of the discriminant region. Of course, this last step is the most difficult part since there is very little data in these areas.

We note finally a modification to the algorithm that seems to improve performance. Usually after a few iterations, the algorithm begins finding balls containing only a few target class points. These points may or may not be in the discriminant region for the target class. Because of the small number of points, we decide to not include these balls in our cover. We find that a threshold of $\log(n)$ works well in practice, where n is the number of target class points in our training set.

4.6.3 The Classifier

Once we have the cover for both classes, we choose a cover-dissimilarity function ρ to describe distance to a cover and then use the classifier as defined in equation (4.2) to perform the classification. The cover-dissimilarity function we have the most success with is a generalization



(a) Square data, first iteration

(b) $R_{(0.5,0.5)}(r)$

(c) $R_{(0.3,0.7)}(r)$, first iteration

(d) Square data, second iteration

(e) $R_{(0.3,0.7)}(r)$, second iteration

(f) Cover of square data

Figure 4.8: Recomputing the radii

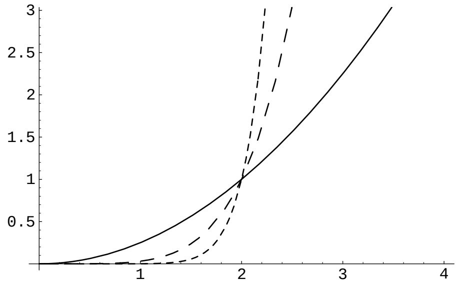


Figure 4.9: Comparing plots of $\pi_i(z)$ for a ball with $r_i = 2$ and $T_{x_i} = 2, 5, 10$ for the solid, medium and heavy dashed lines respectively.

of the scaled function in equation (4.7). For a cover C made up of balls B_i with centers x_i and radius r_i,

$$\rho_{GS}(z, C) := \min_{\{i:B_i \in C\}} \left[\frac{d(z, x_i)}{r_i}\right]^{T_{x_i}^{\epsilon}}. \tag{4.10}$$

The parameter ϵ is a control parameter in $[0, 1]$. This parameter controls how closely the classification regions of the classifier will follow the boundary of the high scoring covering balls. For instance, let $\epsilon = 1$ and consider a ball with a large score. Let $\pi_i(z) := \left[\frac{d(z,x_i)}{r_i}\right]^{T_{x_i}^{\epsilon}}$. As $T_x \to \infty$, $\pi_i(z) \to 0$ for any z such that $d(z, x_i) < r_i$ and $\pi_i(z) \to \infty$ for any z such that $d(z, x_i) > r_i$. Thus any point inside such a ball will tend to be classified as the same class as the point that is the center of the ball. Figure 4.9 illustrates this point. The three plotted lines represent $\pi_i(z)$ for balls with various scores.

Consider the situation in Figure 4.10. Suppose the light and dark circles are covering balls from different classes (say class zero and one respectively) and that the score of the light ball is larger than that of the dark ball. If $\epsilon = 0$ then we see that the score of the balls is disregarded. The solid vertical line represents points that are equally close to both covers when $\epsilon = 0$. However, increasing the value of ϵ draws the classification region toward the higher scoring ball. This is consistent with the logic that a ball with a higher score should be favored in the region of intersection with a ball of lower score.

Figure 4.11 shows the classification regions for two data sets with two different values of ϵ. Figures 4.11(b) and 4.11(c) show classification regions for classifiers built on training data from figure 4.6(a) using the same cover shown in figure 4.11(a). The boundary of the discriminant region is shown by the black square. Notice that in figure 4.11(b) with $\epsilon = 0$, the classification region is oddly shaped compared to the classification region in figure 4.11(c) which follows the

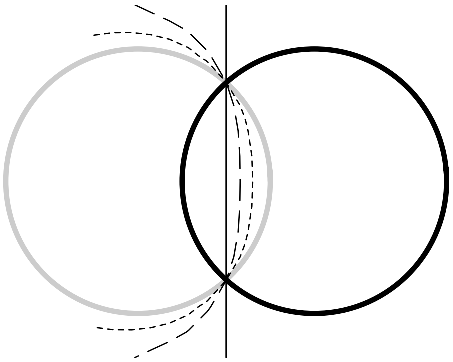


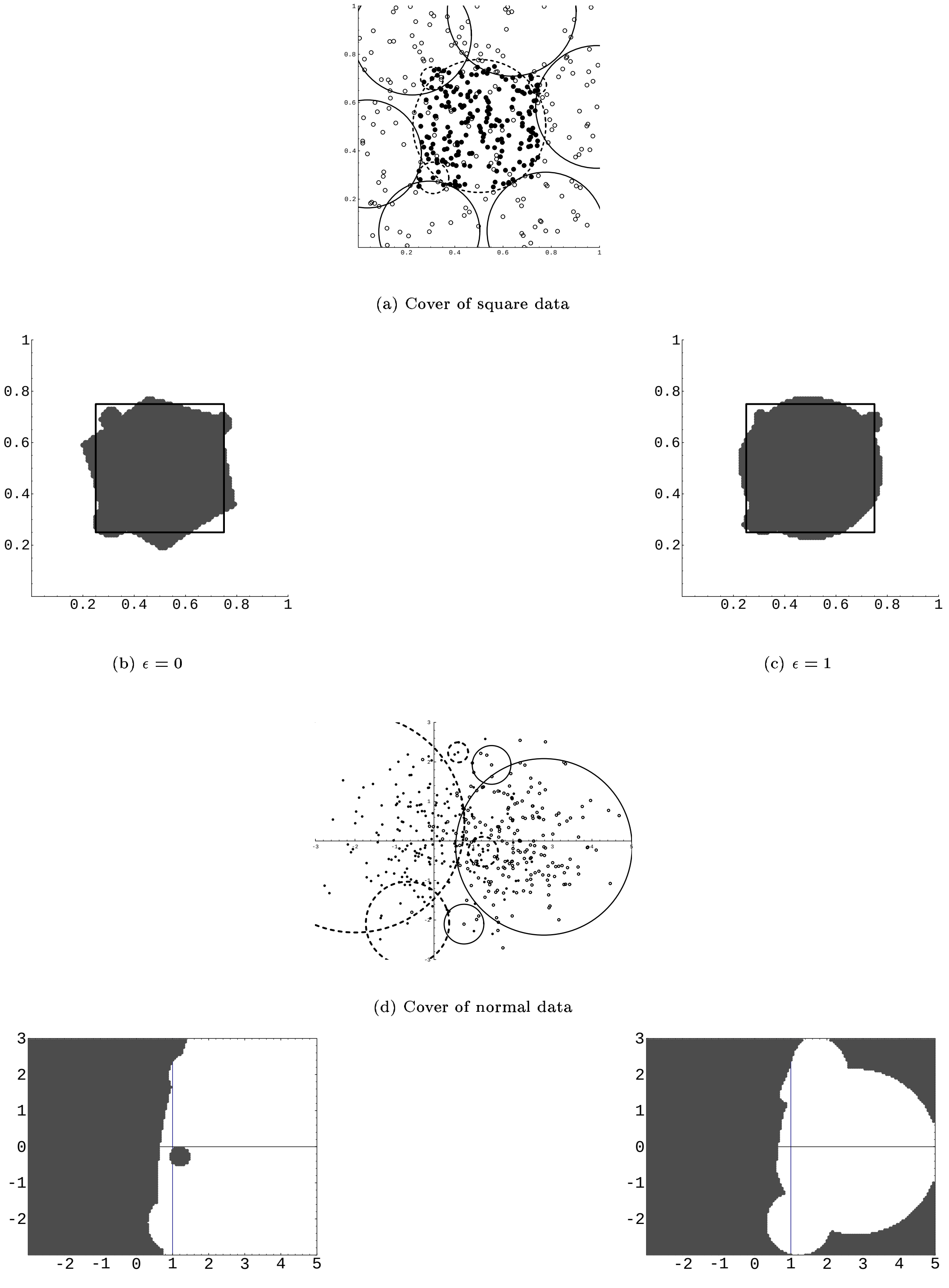
Figure 4.10: The solid vertical line corresponds to $\epsilon = 0$, the medium dashed line to $\epsilon = 0.5$ and the dashed line to $\epsilon = 1$.

cover for the data which is distributed on the smaller square. Figures 4.11(e) and 4.11(f) show classification regions built on the data of Model 2 (introduced in section 4.4) shown in figure 4.3(a) using the cover in figure 4.11(d). In this case, the discriminant boundary is the vertical line passing through the x-axis at one.

4.6.4 Future Improvements

There are three important stages in the creation of the random walk CCP classifier; choosing the radii of potential covering balls, choosing a set of covering balls to be the cover and choosing the cover-dissimilarity function. For each task, we have presented a method; however, there is room for improvement in each. For example, instead of choosing each cover independently, another possibility is to choose both covers simultaneously in a greedy manner. This idea holds merit because the classifier makes its decision based on the interaction between the two covers. A more detailed description of this method and examples of its application are presented in [39].

Improvements could also be made in the choice of the cover-dissimilarity function by more careful exploitation of the curve $R_x(r)$ for each point at the center of a covering ball. The cover-dissimilarity function presented here works well for classifying points inside one or both class covers. However, improvement is possible for the classification of points outside of both covers. We might use the shape of $R_x(r)$ to gain better classifier performance. For example if $R_x(r)$ falls sharply in some interval (r^*, R) then we might guess that a point z just outside of this ball $(d(x, z) < R)$ is not likely to be the same class as x. Exactly how to implement this idea is not clear.



(a) Cover of square data

(b) $\epsilon = 0$

(c) $\epsilon = 1$

(d) Cover of normal data

(e) $\epsilon = 0$

(f) $\epsilon = 1$

Figure 4.11: Changing ϵ

Chapter 5

Clustering

5.1 Clustering

Another potential application of the ideas presented above is to unsupervised classification or clustering. In unsupervised classification, we are not given a training set of labeled observations. Instead we are asked to cluster the given data into k groups where k is an integer that may or may not be specified. The ability to effectively cluster data is helpful in discovering unknown structure in a data set.

We are given a single data set $\mathcal{X} \subset \Omega$ from a dissimilarity space (Ω, d). Our approach is to use a one-class version of the random walk CCP. That is, for each point $x \in \mathcal{X}$ we will define a random walk $C_x(r)$ by

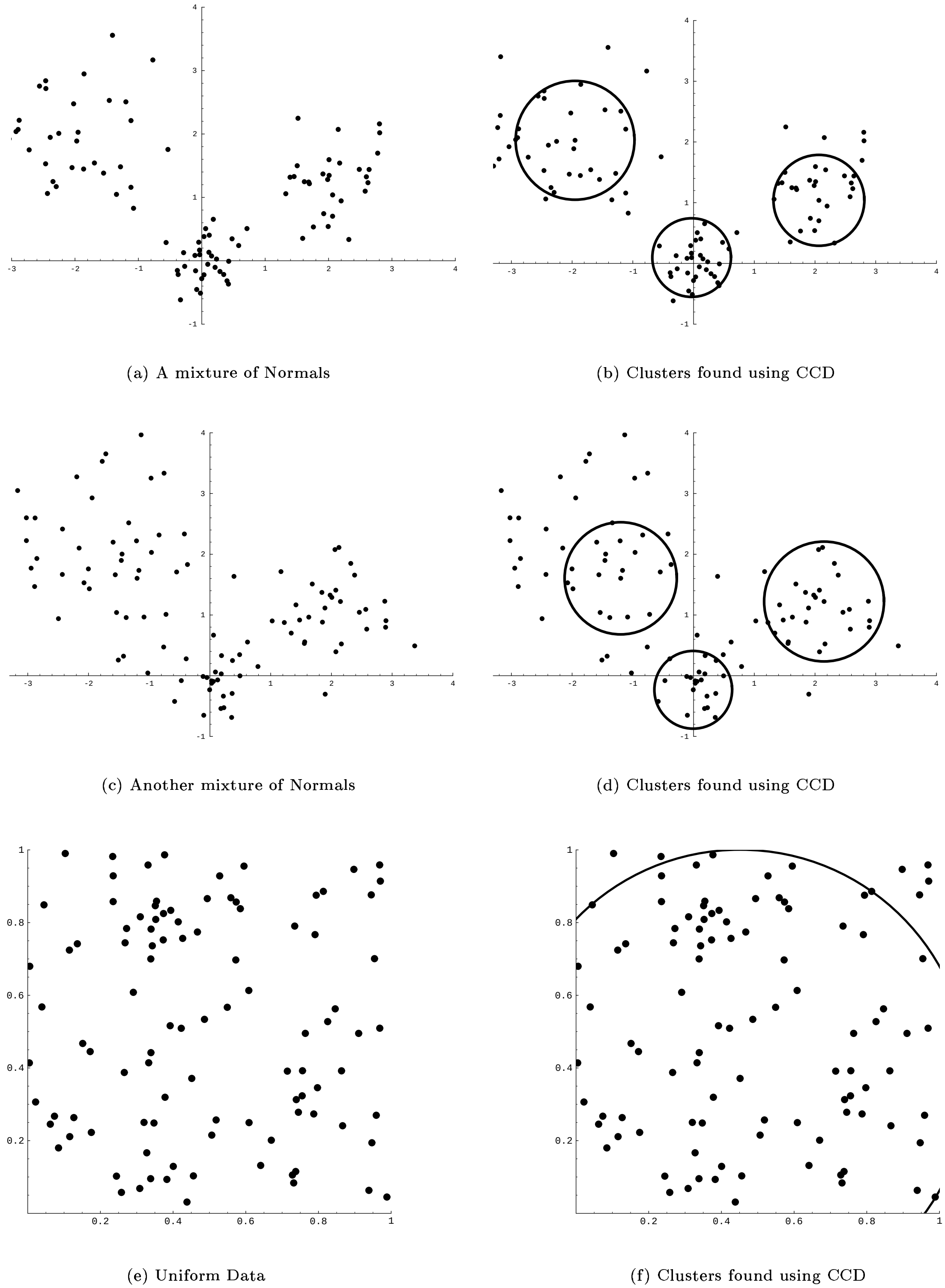
$$C_x(r) = |\{z \in \mathcal{X} : d(x, z) \leq r\}|. \tag{5.1}$$

This is very similar to the definition of $R_x(r)$ in section 4.6 except there is only one class. We picture this as a random walk that increases by one at r if the boundary of a ball centered at x intersects some point in $\mathcal{X}$ and otherwise stays at its present value.

Stated casually, we would like to place a covering ball at a point x to represent the local cluster that x belongs to. This is very similar to our goal with the random walk classifier. In this case, however, $C_x(r)$ is an increasing function, so choosing the value of r where $C_x(r)$ reaches its maximum will not work, since every ball would cover every point in $\mathcal{X}$. Our goal is to differentiate a point at or near the "center" of a cluster from a point which is not near the center of a cluster. We will use "not near the center of a cluster" as a null hypothesis. That is, we will assume that every point is not near the center of a cluster unless we see evidence that contradicts this. One way in which a point is not near the center of a cluster is if there is complete spatial randomness, that is, the points of $\mathcal{X}$ are uniformly distributed in space. The curve $C_x(r)$ for a point x among other points uniformly distributed looks like mr^d for some scalar m, where d is the dimensionality of the data. One way to choose the radius for a covering ball for a point x is to find the largest positive value of $C_x(r) - mr^d$. Once we have a covering ball for each point we can find a dominating set in some fashion and use the points covered by each ball as a separate cluster.

This method of clustering is similar to a common clustering method called k-means clustering [15]. The object of k-means clustering is to find the mean vectors $\mu_1, \mu_2, \ldots, \mu_k$ of the k clusters (assuming the data is drawn from a mixture of k probability distributions with means $\mu_1, \mu_2, \ldots, \mu_k$). In the CCP clustering method, the representative for a cluster i is the center c_i of the ball B_i containing those points. The point c_i is a an approximation to the mean of the points covered by the ball B_i since this ball was chosen as a good representative of the points it covers. One possible advantage of the CCP clustering method over k-means clustering algorithms is that it adaptively chooses the parameter k. This can be helpful in cases where k is unknown.

Figure 5.1(a) shows 100 observations drawn from a mixture of three normals. The exact distribution is $\sum_{i=1}^{3} \frac{1}{3}\Phi(\vec{\mu_i}, \mathbf{\Sigma}_i)$ where $\vec{\mu_1} = (0,0)$, $\vec{\mu_2} = (2,1)$, $\vec{\mu_3} = (-2,2)$ and $\mathbf{\Sigma}_i = (0.1 * i + 0.3)\mathbf{I}_2$. Figure 5.1(b) shows the cover of the data. Using $m = 0.2$, our method has chosen three balls (clusters) centered at $(-0.03, 0.10)$, $(2.06, 1.03)$ and $(-1.95, 2.03)$. Notice that several points have not been included in any cluster. We have chosen to implement a threshold on the number of points in a covering ball similar to our method in Section 4.6.2. Figure 5.1(c) shows 100 observation drawn from another mixture of three normals. This distribution has the same mean vectors but with larger standard deviations. The exact distribution is $\sum_{i=1}^{3} \frac{1}{3}\Phi(\vec{\mu_i}, \mathbf{\Sigma}_i)$ where $\mathbf{\Sigma}_i = (0.1 * i + 0.4)\mathbf{I}_2$. The RW-CCP method finds three clusters centered at $(-0.01, -0.23)$, $(2.15, 1.22)$ and $(-1.20, 1.60)$. We used $m = 0.07$. Finally, Figure 5.1(e) shows 100 observations of data uniformly distributed on the unit square. Figure 5.1(f) shows the cluster found by the RW-CCP method. In this case, our method has found one large cluster. It is unclear what the correct answer is for this case since one definition of a cluster is an area or non-uniform density.



(a) A mixture of Normals

(b) Clusters found using CCD

(c) Another mixture of Normals

(d) Clusters found using CCD

(e) Uniform Data

(f) Clusters found using CCD

Figure 5.1: Example of clustering

Chapter 6

Performance of CCP Classifiers

In this chapter we will investigate the performance of our CCP based classifiers, ans compare against a few well-known classifiers on several data sets, both simulated and real. We compare the performances of the nearest neighbor, and k-nearest neighbor classifiers, SVM's and three different CCP classifiers. The naive CCP is as described in section 4.4 using pure and proper covers and the cover-dissimilarity function ρ_s. The α, β CCP classifier (section 4.5) allows a non-zero choice of the α and β parameters for each class and also uses the ρ_s dissimilarity. And finally we use the random walk classifier as described in section 4.6. The Euclidean metric is used in all tests. We attempted to optimize parameter choice for all training sets for all classifiers except the nearest neighbor and naive CCP. Parameter values are reported in Appendix A. We also introduce the linear classifier when considering model 2 data. A brief description of the competing classifiers used is presented below.

6.1 Other Classifiers

6.1.1 Nearest Neighbor/k-Nearest Neighbor Classifier

The nearest neighbor is one of the simplest classification rules in existence. It's performance can be surprisingly good, however. The nearest neighbor classifier is defined as

$$g_{NN}(z) := \begin{cases} 0 & \text{if } \min_{x \in X_0} d(z,x) < \min_{y \in X_1} d(z,y) \\ 1 & \text{otherwise} \end{cases}$$

Thus the nearest neighbor classifier assigns the class label of the training observation that is closest. This classifier is not universally consistent, but its misclassification rate is guaranteed to converge to at most twice the Bayes optimal rate [15].

The k-nearest neighbor classifier assigns an unknown observation the class label which is most prevalent among its k-nearest training set neighbors. This rule is slightly more complicated than the nearest neighbor rule; however, its misclassification rate is guaranteed to converge to Bayes optimal as long as k goes to infinity along with the training sample size and $\frac{k}{n}$ goes to zero.

6.1.2 Linear Classifier

A general binary linear classifier can be expressed as

$$g(\mathbf{z}) := \begin{cases} 0 & \text{if } \mathbf{w} \cdot \mathbf{z} + \mathbf{w}_o > 0 \\ 1 & \text{if } \mathbf{w} \cdot \mathbf{z} + \mathbf{w}_o < 0. \end{cases} \tag{6.1}$$

In the case of two multivariate normal densities with equal prior probabilities and with means μ_0 and μ_1 respectively and covariance matrices $\mathbf{\Sigma}_0 = \mathbf{\Sigma}_1 = \mathbf{I}$, the Bayes optimal classifier is linear and of the form

$$g_L(\mathbf{z}) := \begin{cases} 0 & \text{if } (\mu_0 - \mu_1)^T(\mathbf{x} + \mathbf{x}_o) > 0 \\ 1 & \text{if } (\mu_0 - \mu_1)^T(\mathbf{x} + \mathbf{x}_o) < 0 \end{cases} \tag{6.2}$$

where

$$\mathbf{x}_0 = \frac{1}{2}(\mu_0 - \mu_1).$$

This implies that the optimal linear classifier is the hyperplane that is the perpendicular bisector of the line segment connecting μ_0 and μ_1. When μ_0 and μ_1 are not known we may use an approximate version of this classifier by substituting the sample means $\hat{\mu}_0$ and $\hat{\mu}_1$ for the means in Equation 6.2. We will be using the linear classifier in our Model 2 only.

6.1.3 Support Vector Machines

We say a training set is *linearly separable* if a hyperplane exists such that class zero observations are separated from class one observations. In other words, there is a vector $\mathbf{a}$ such that $\mathbf{a}^T x > 0$

if $x \in \mathcal{X}_0$ and $\mathbf{a}^T x < 0$ if $x \in \mathcal{X}_1$. The vector $\mathbf{a}$ is perpendicular to the *separating hyperplane*. The *margin* of a separating hyperplane is the minimum distance between the hyperplane and any point in the training set. We call the points of the training set closest to the optimal separating hyperplane (the separating hyperplane with the largest margin) are called the *support vectors*.

The goal of a *support vector machine* [15, 43] is to find the separating hyperplane with the largest margin. Of course not all training sets are linearly separable, but with an appropriate mapping $\phi(\cdot)$ to a higher dimension, any training set can become linearly separable. The classifier is a linear classifier in the higher dimension but can be highly non-linear in the original space. The choice of the mapping $\phi(\cdot)$ is called the *kernel function*.

The decision function for a support vector machine with kernel $\phi(\cdot)$ and support vectors $x_1, x_2, \ldots, x_N$ is

$$g_{svm}(z) := \begin{cases} 0 & \left(\sum_{i=1}^{N} a_i \phi(x, x_i) - b\right) < 0 \\ 1 & \text{otherwise} \end{cases}$$

where a_i are the coefficients for the optimal separating hyperplane. Of particular interest to our research is the *radial basis kernel function*. These kernel functions $K(|x - x_i|)$ depend only on the distance between two vectors. Support vector machines with radial basis kernel functions are similar to CCP based classifiers since the classification decision based on the distance of a given point to a set of representative points.

6.2 Simulation Data

For each model and data dimensionality, we created training sets of n observations from each class ($n \in \{50, 100, 200, 500\}$) and then performed Monte Carlo replicates on test sets of 100 new observations from each class. For each replicate, the performance of a classifier is measured by the fraction of observations misclassified (as an approximation of the misclassification rate $L(g)$). We performed Monte Carlo replicates until the standard deviation for the average performance became less than 0.003.

In each section below we examine the performance on different data sets in two, three and five dimensions of our CCP classifiers, the nearest neighbor classifier, the k-nearest neighbor classifier and support vector machines. We use the radial basis function kernel included in the SVM-light package to implement our support vector machines [22].

6.3 Model 1

In this simulation we have $F_0 = U[0,1]^d$ and $F_1 = \frac{1}{2}\,U[0.1, 0.55]^d + \frac{1}{2}\,U[0.6, 0.8]^d$ where d is the dimension of the data. The Bayes optimal decision rule for this model is

$$g_1^*(z) := \begin{cases} 0 & \text{if } z \notin \{[0.1, 0.55]^d \cup [0.6, 0.8]^d\} \\ 1 & \text{otherwise} \end{cases}$$

and the Bayes optimal error rate is given by

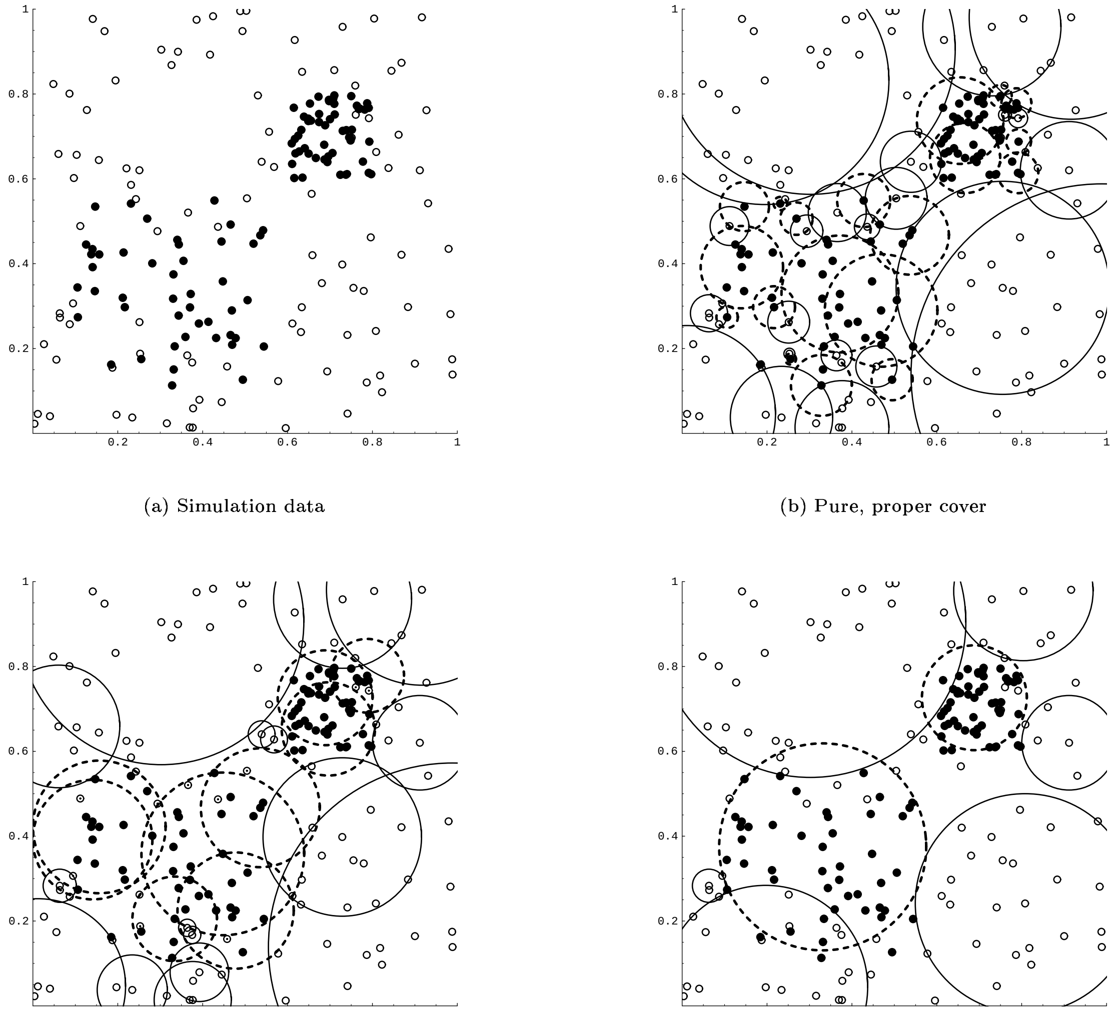
$$L^* = \frac{1}{2}\left((0.55 - 0.1)^d + (0.8 - 0.6)^d\right).$$

This model presents a special challenge to classifiers because of the sharp corners of the discriminant region. The CCP based classifiers perform well even using the Euclidean metric. The results for these classifiers would improve if we switched to the L_∞ metric.

6.3.1 Two dimensions

Figures 6.1(a) and 6.3(a) show training sets in two dimensions of size 100 and 500 points from each class respectively. The points from F_0 are represented as empty circles and those drawn from F_1 as black filled circles. The experimental approximation of the misclassification rate for each classifier is shown in Table 6.1. For this model in two dimensions we have $L^* \approx 0.121$. Notice that the α, β and RW CCP-based classifiers outperform the nearest neighbor and the optimized k-nearest neighbor classifiers. classifier complexity. Table 6.2 on page 75 shows the average number of balls in a cover for each class for each value of n.

Figures 6.1 and 6.3 show the covers of a training sets of size 100 and 500 respectively. The covering balls for class one are represented as dashed balls. Figures 6.2 and 6.4 illustrate the classification regions calculated by the various classifiers for $n = 100$ and $n = 500$ respectively. The gray regions are the regions each classifier will classify as class one. The optimal classification regions for class one are outlined in black.



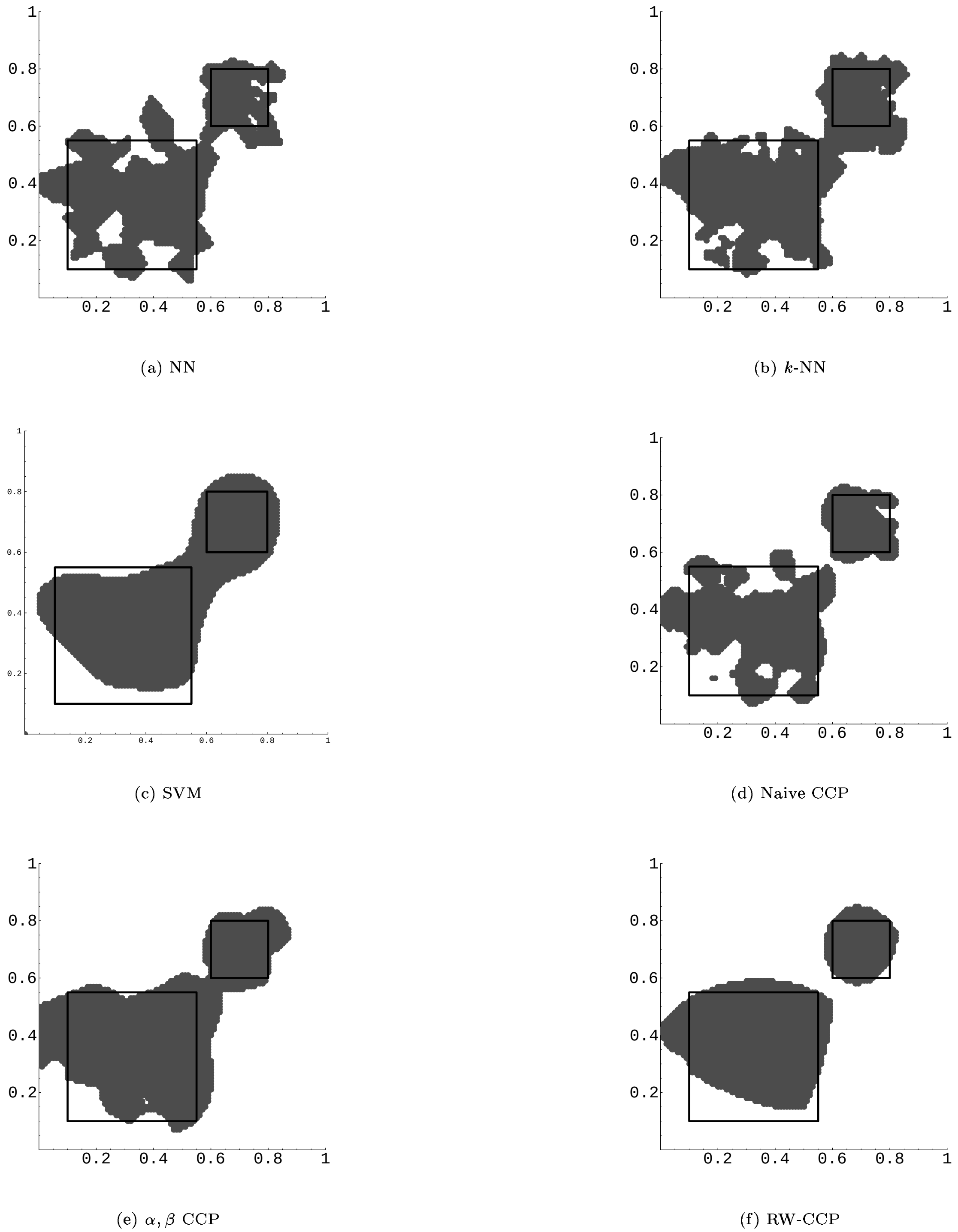
(a) Simulation data

(b) Pure, proper cover

(c) α, β cover

(d) RW cover

Figure 6.1: Covers of model one data in two dimensions (100 points).



(a) NN

(b) k-NN

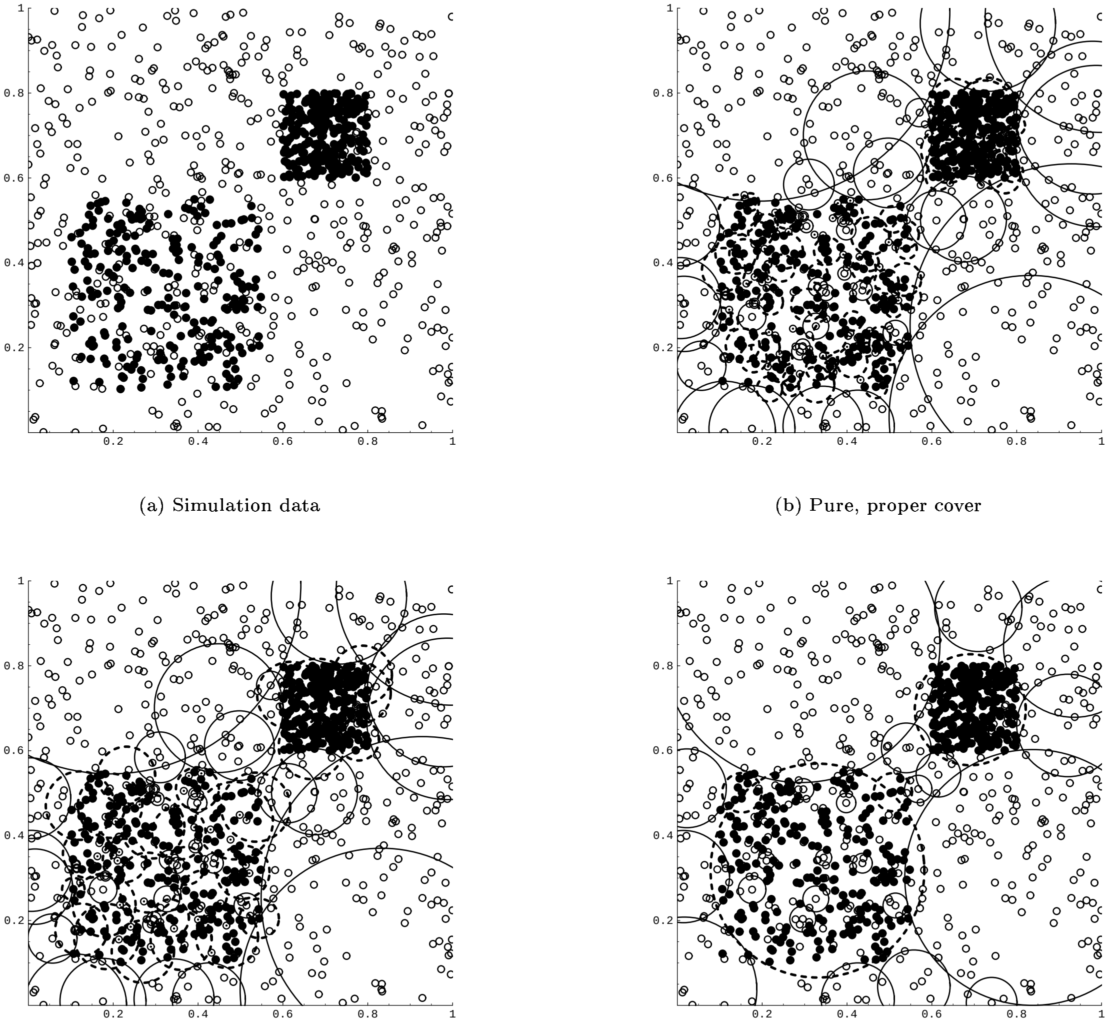
(c) SVM

(d) Naive CCP

(e) α, β CCP

(f) RW-CCP

Figure 6.2: Comparison of classification regions for model one data in two dimensions (100 points).



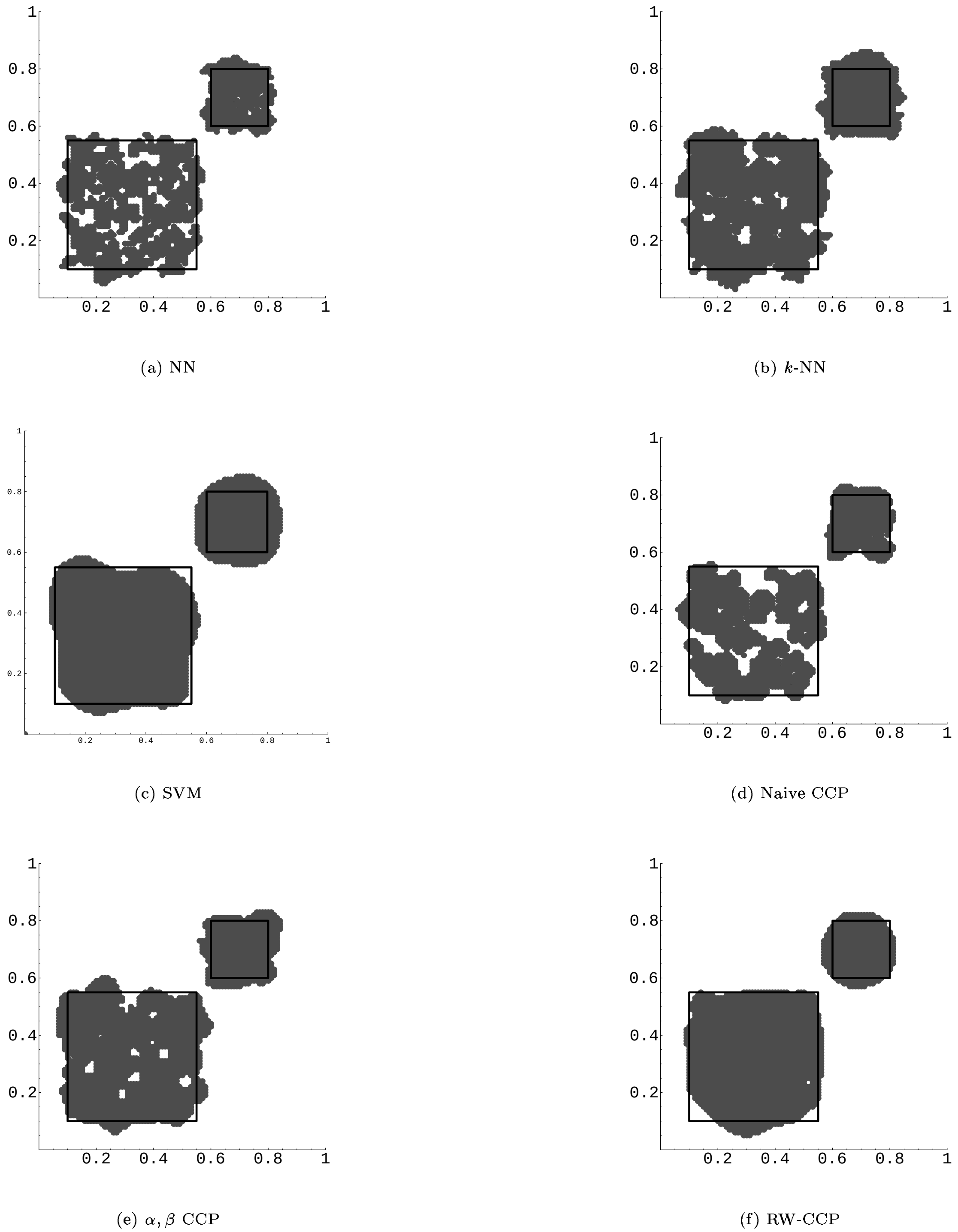
(a) Simulation data

(b) Pure, proper cover

(c) α, β cover

(d) RW cover

Figure 6.3: Covers of model one data in two dimensions (500 points).



(a) NN

(b) *k*-NN

(c) SVM

(d) Naive CCP

(e) α, β CCP

(f) RW-CCP

Figure 6.4: Comparison of classification regions for model one data in two dimensions (500 points).

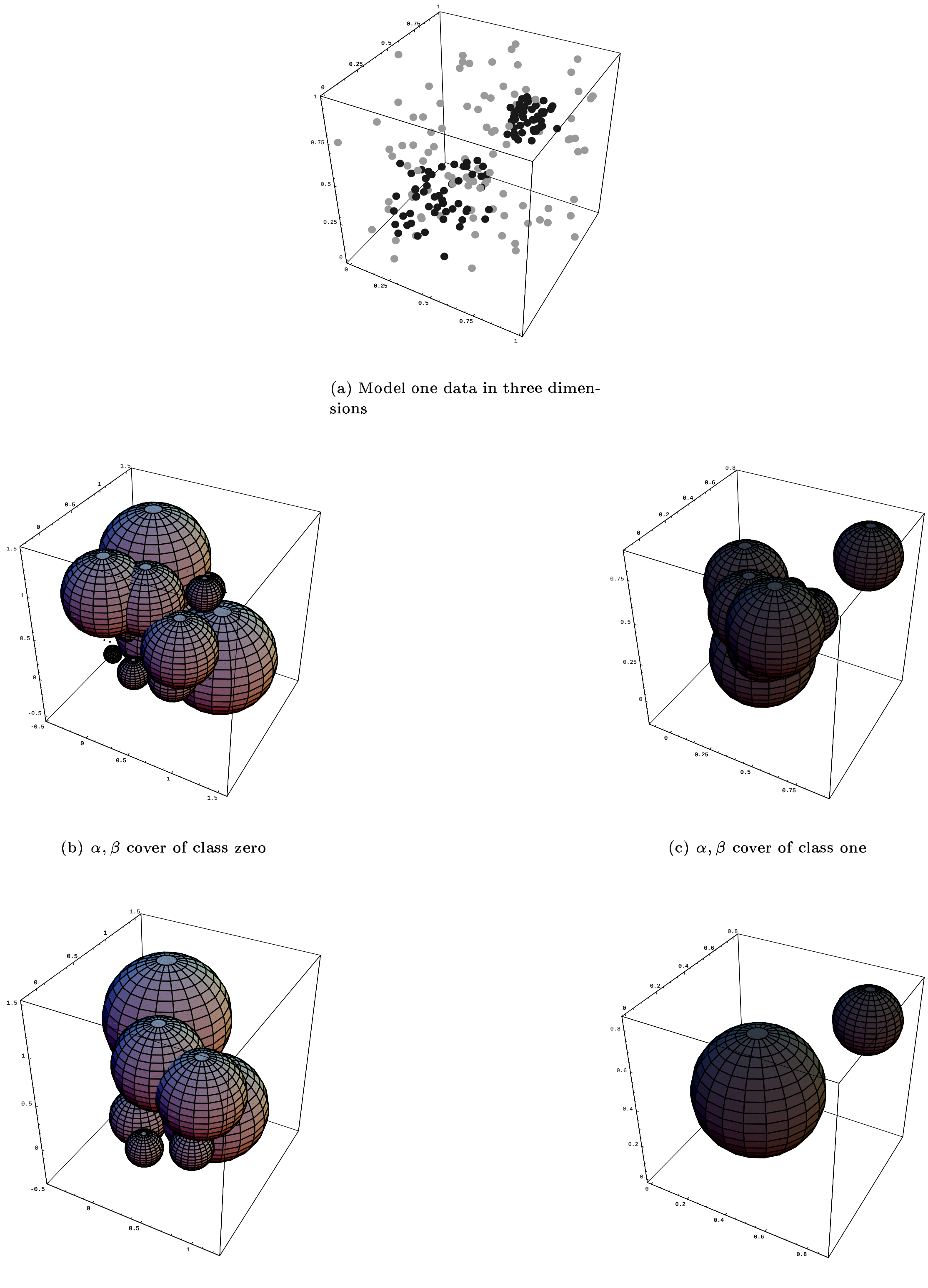
6.3.2 Three Dimensions

Figure 6.5(a) shows a sample of 100 observations from each class. Figures 6.5(b) and 6.5(c) display the class zero and class one covers (respectively) for the data as determined by the α, β CCP using $\alpha_0 = \alpha_1 = \beta_1 = 0, \beta_0 = 3$. Figures 6.5(d) and 6.5(e) show the covers for the random walk CCP for class zero and class one respectively.

The sample misclassification rates are shown in Table 6.3. In three dimensions, $L^* \approx 0.099$. Once again, the CCP classifiers outperform the nearest neighbor and k-nearest neighbor classifiers. RW and α, β CCP classifiers are competitive with SVM's in three dimension. Also note the drop in classifier complexity from the naive CCP to the RW CCP (Table 6.4).

6.3.3 Five Dimensions

Finally we present results in five dimensions where $L^* \approx 0.019$. Here we begin to see effects of choosing the Euclidean metric. The sample misclassification rates are shown in Table 6.5. Once again, the CCP classifiers outperform the nearest neighbor and k-nearest neighbor classifiers. RW and α, β CCP classifiers are competitive with SVM's in three dimension. Also note the drop in classifier complexity from the naive CCP to the RW CCP (Table 6.6).



(a) Model one data in three dimensions

(b) α, β cover of class zero

(c) α, β cover of class one

(d) RW cover of class zero

(e) RW cover of class one

Figure 6.5: Covers of model one data in three dimensions.

Training Size	NN	k-NN	SVM	Naive CCP	α, β CCP	RW-CCP
50	0.242	0.240	0.201	0.228	0.212	0.212
100	0.224	0.212	0.184	0.213	0.190	0.183
200	0.210	0.188	0.168	0.201	0.171	0.165
500	0.199	0.166	0.152	0.194	0.154	0.153

Table 6.1: Misclassification rates for model one data in two dimensions.

Training Size	Naive CCP		α, β CCP		RW-CCP	
	class 1	class 0	class 1	class 0	class 1	class 0
50	14.3	16.3	6.7	16.3	2.8	5.3
100	27.0	28.2	9.3	28.2	3.4	8.4
200	51.2	49.5	13.4	49.6	4.6	13.0
500	122.7	110.1	27.3	110.1	5.9	19.2

Table 6.2: Average cover cardinality for model one data in two dimensions.

Training Size	NN	k-NN	SVM	Naive CCP	α, β CCP	RW-CCP
50	0.187	0.187	0.132	0.158	0.148	0.140
100	0.162	0.162	0.111	0.141	0.130	0.118
200	0.144	0.138	0.096	0.128	0.114	0.104
500	0.126	0.114	0.082	0.114	0.097	0.0935

Table 6.3: Misclassification rates for model one in three dimensions.

Training Size	Naive CCP		α, β CCP		RW-CCP	
	class 1	class 0	class 1	class 0	class 1	class 0
50	9.4	14.4	6.1	14.4	2.2	4.9
100	17.0	22.8	6.9	22.8	2.5	7.0
200	31.3	37.4	11.0	37.4	3.0	10.1
500	70.9	75.3			4.7	16.7

Table 6.4: Average cover cardinality for model one data in three dimensions.

Training Size	NN	k-NN	SVM	Naive CCP	α, β CCP	RW-CCP
50	0.146	0.146	0.054	0.093	0.086	0.090
100	0.118	0.118	0.046	0.078	0.070	0.072
200	0.097	0.097	0.040	0.068	0.059	0.058
500	0.078	0.079	0.034	0.057	0.051	0.048

Table 6.5: Misclassification rates for model one in five dimensions.

Training Size	Naive CCP		α, β CCP		RW-CCP	
	class 1	class 0	class 1	class 0	class 1	class 0
50	5.1	15.9	3.4	15.8	2.1	5.5
100	8.35	22.7	3.2	22.7	2.1	7.0
200	14.5	32.7	4.6	32.7	2.3	9.6
500	30.8	55.0	7.5	55.0	2.7	13.8

Table 6.6: Average cover cardinality for model one data in five dimensions.

6.4 Model 2

In this model, F_0 is the normal distribution with mean at the origin and the identity covariance matrix and F_1 is the normal distribution with mean $(2, 0, \ldots, 0)$ and the identity covariance matrix. The Bayes optimal classification rule for this model for an unknown point $z \in \mathbb{R}^d$ is

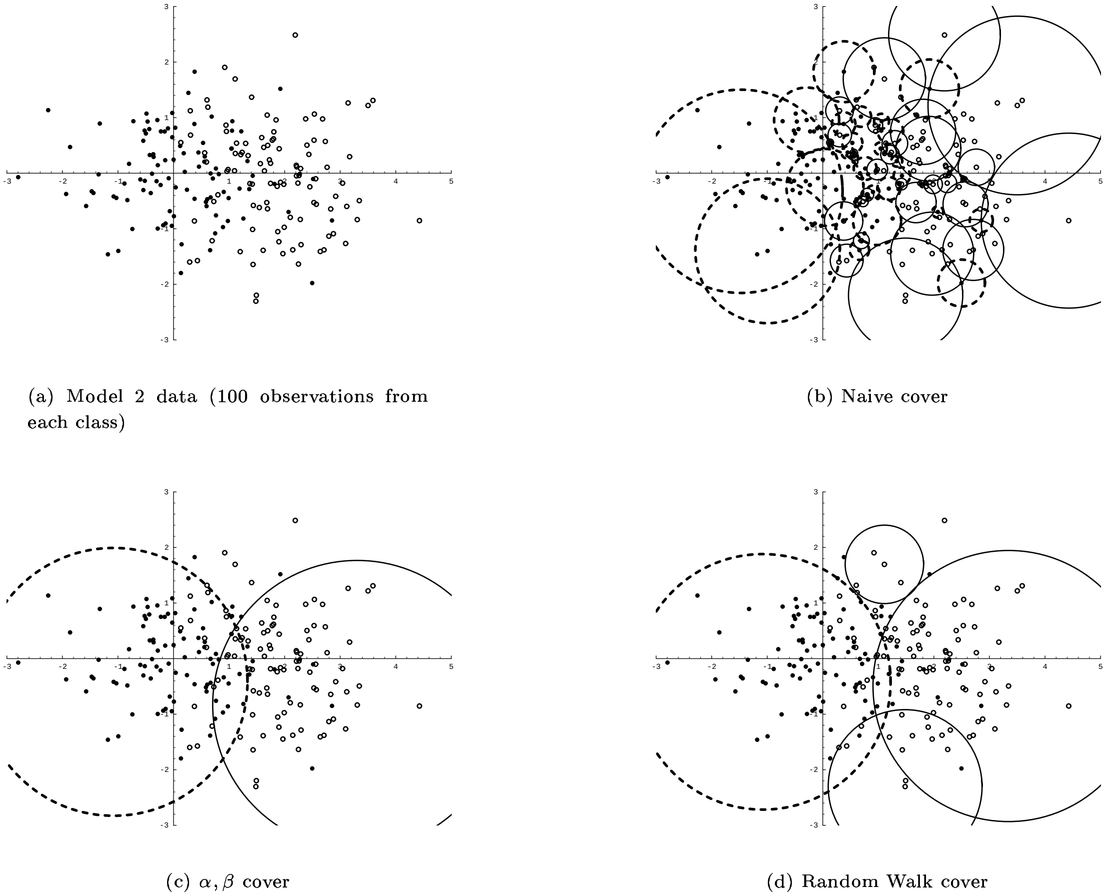
$$g^*(z) := \begin{cases} 0 & \text{if } z[1] < 1, \\ 1 & \text{otherwise.} \end{cases}$$

The linear non-parametric classifier is the most efficient classifier for this model. We compare our CCP based classifiers with the linear classifier, the nearest and k-nearest neighbor classifiers, and support vector machines. In this case we use a linear kernel function for the SVM. The linear classifier and the SVM perform very well on this data set since the optimal discriminant boundary is linear. Also the k-nearest neighbor does very well with high k-values.

6.4.1 Two Dimensions

Figure 6.6(a) shows a training set in two dimensions of size 100 points from each class. The points from F_0 are represented as empty circles and those drawn from F_1 as black filled circles. The experimental approximation of the misclassification rate for each classifier is shown in Table 6.7. The Bayes optimal error rate for this model in any dimension is approximately 0.1586. The average number of balls per cover for each CCP classifier is presented in Table 6.8. The optimal parameter choice for α, β give an average of one ball per class. This is not surprising since the classification region between two equal sized balls from different classes will be a straight line. The random walk classifier covers tend to have one large ball and several smaller balls. Unfortunately this does not provide the best cover for classification purposes. A linear discriminant boundary is most efficiently represented with one ball from each cover.

Figure 6.6 shows the covers of a training set of size 100. The covering balls for class zero are represented as dashed balls. Figure 6.7 illustrates the classification regions calculated by the various classifiers for $n = 100$. The gray regions are the regions each classifier will classify as class zero. The optimal classification region is to the left of the vertical line $x = 1$.



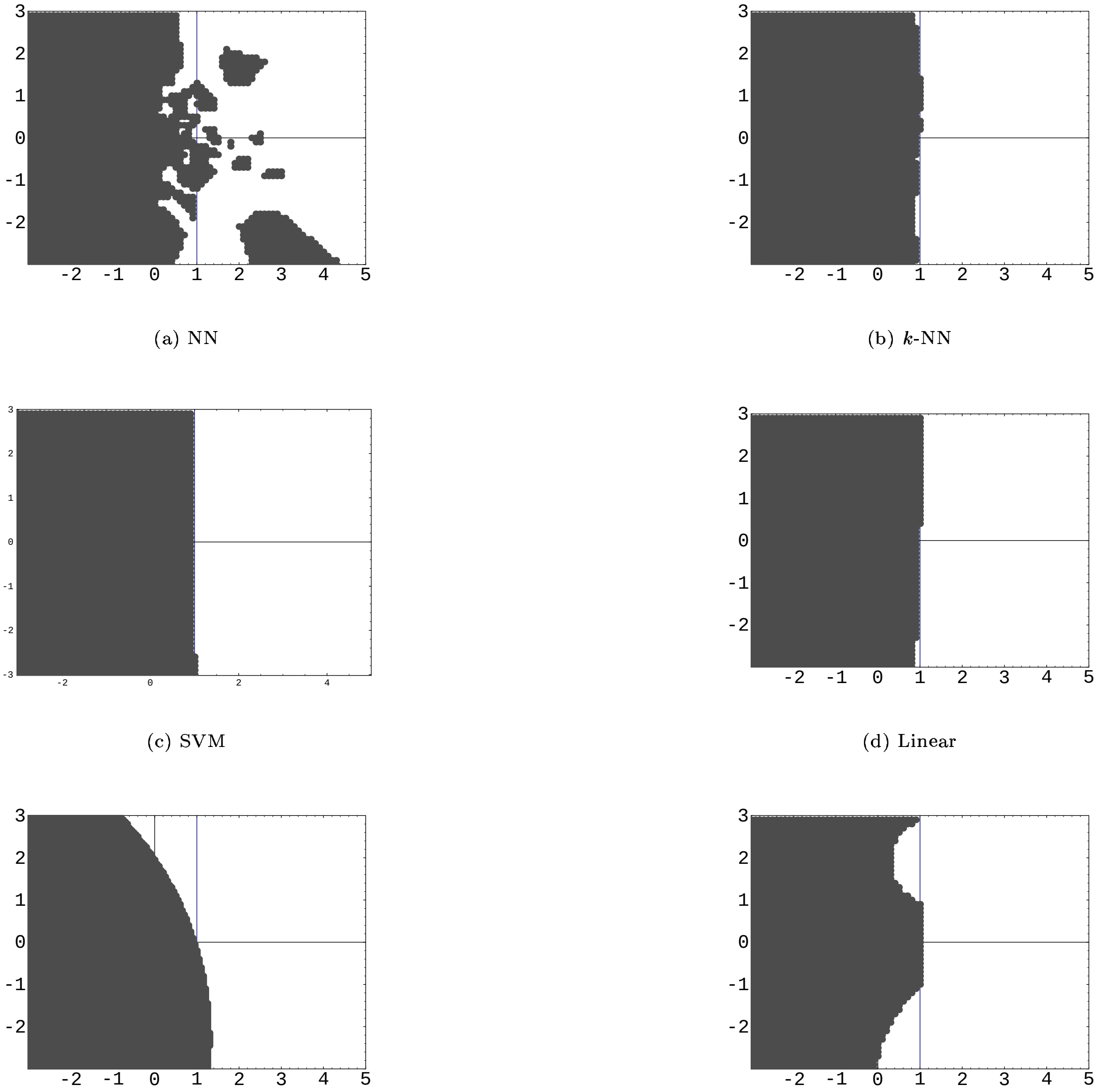
(a) Model 2 data (100 observations from each class)

(b) Naive cover

(c) α, β cover

(d) Random Walk cover

Figure 6.6: Covers of model two data in two dimensions.



(a) NN

(b) *k*-NN

(c) SVM

(d) Linear

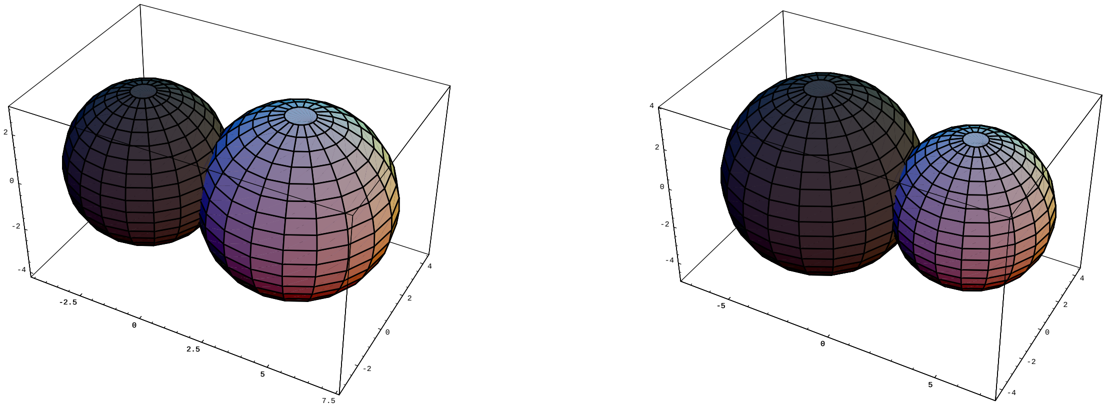
(e) α, β CCP

(f) RW-CCP

Figure 6.7: Comparison of classification regions for Model two data in two dimensions.

6.4.2 Three and Five Dimensions

We have performed the same tests as the above section for Model two in three and five dimensions. Figures 6.8 show covers for Model 2 data in three dimensions using 100 training points from each class. Both the α, β cover and the random walk cover consist of one ball per class. The sample misclassification rates are presented in Tables 6.13 and 6.11 for three and five dimensions respectively. Also Tables 6.14 and 6.12 show the average number of balls in each cover for Model two in three and five dimensions respectively.



(a) α, β cover

(b) Random Walk cover

Figure 6.8: Covers of model two data in three dimensions.

Tr. Size	Linear	NN	k-NN	SVM	Naive CCP	α, β CCP	RW-CCP
50	0.161	0.228	0.164	0.162	0.181	0.167	0.172
100	0.160	0.227	0.161	0.161	0.178	0.164	0.170
200	0.159	0.226	0.160	0.161	0.176	0.162	0.166
500	0.159	0.225	0.159	0.161	0.176	0.160	0.163

Table 6.7: Misclassification rates for model two data in two dimensions.

Training Size	Naive CCP		α, β CCP		RW-CCP	
	class 0	class 1	class 0	class 1	class 0	class 1
50	13.9	13.9	1.0	1.0	1.4	1.4
100	27.0	27.0	1.0	1.0	2.0	2.0
200	53.3	53.3	1.0	1.0	3.1	3.1
500	132.1	132.0	1.0	1.0	6.2	6.2

Table 6.8: Average cover cardinality for model two data in two dimensions.

Training Size	Linear	NN	k-NN	SVM	Naive CCP	α, β CCP	RW-CCP
50	0.163	0.235	0.166	0.192	0.218	0.172	0.179
100	0.160	0.232	0.162	0.163	0.216	0.167	0.175
200	0.159	0.229	0.160	0.162	0.215	0.164	0.170
500	0.159	0.227	0.159	0.161	0.213	0.161	0.166

Table 6.9: Misclassification rates for model two data in three dimensions.

Tr. Size	Naive CCP		α, β CCP		RW-CCP	
	class 0	class 1	class 0	class 1	class 0	class 1
50	14.1	14.1	1.0	1.0	1.6	1.6
100	26.9	26.9	1.0	1.0	2.3	2.3
200	52.8	52.8	1.0	1.0	3.7	3.6
500	129.2	129.3	1.0	1.0	7.2	7.2

Table 6.10: Average cover cardinality for model two data in three dimensions.

Training Size	Linear	NN	k-NN	SVM	Naive CCP	α, β CCP	RW-CCP
50	0.165	0.250	0.171	0.205	0.221	0.182	0.184
100	0.162	0.244	0.165	0.167	0.218	0.174	0.177
200	0.160	0.240	0.162	0.162	0.215	0.169	0.172
500	0.160	0.236	0.161	0.159	0.200	0.166	0.170

Table 6.11: Misclassification rates for model two data in five dimensions.

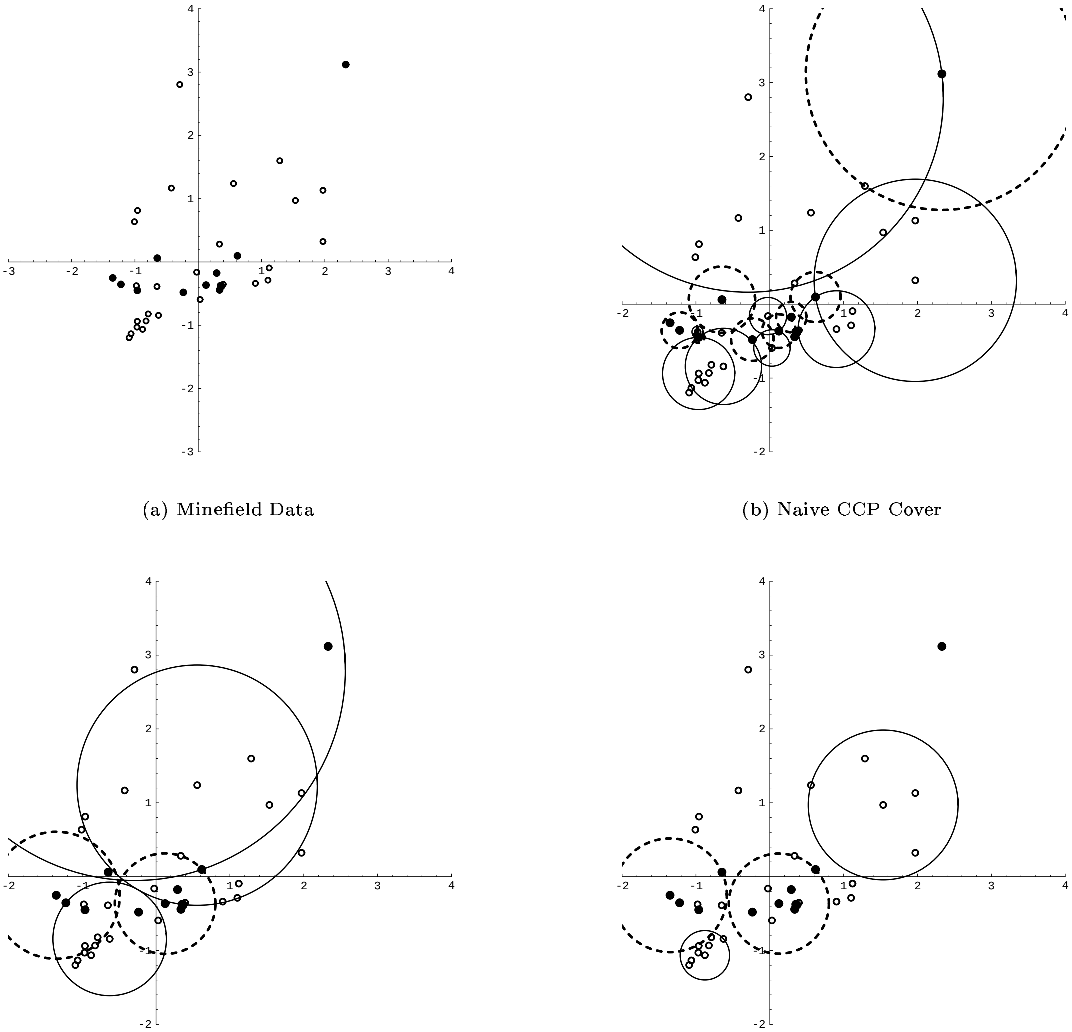
Tr. Size	Naive CCP		α, β CCP		RW-CCP	
	class 0	class 1	class 0	class 1	class 0	class 1
50	15.0	14.9	1.0	1.0	1.7	1.7
100	28.3	28.4	1.0	1.0	2.2	2.1
200	54.0	54.0	1.0	1.0	3.0	3.0
500	130.6	130.7	1.0	1.0	5.0	4.9

Table 6.12: Average cover cardinality for model two data in five dimensions.

6.5 Minefield Data

The experimental data set is multispectral data observations of minelike objects taken by an unmanned aerial vehicle as part of the Coastal Battlefield Reconnaissance and Analysis (COBRA) Program. There are 39 observations, of which 12 are actual mines and 27 are false alarms. The raw data is six dimensional (six spectral bands), but we consider the two dimensions most valuable to classification based on the work of Olson, Pang and Priebe [31]. Figure 6.9(a) shows a two-dimensional plot of these two features. The filled disks represent mines and the circles represent non-mines. Figure 6.9 shows the covers for the three CCP based classifiers.

Figure 6.10 shows the classification regions produced by the six different classifiers. Using the leave-one-out error rate estimate we observe that the random walk CCP and α, β CCP classifiers have the best performance of 9/39 and 8/39 incorrect respectively. The nearest neighbor, k-nearest neighbor and SVM classifiers classify 10/39 incorrectly or worse. Optimal parameter choice for the minefield data is given in Table A.7.



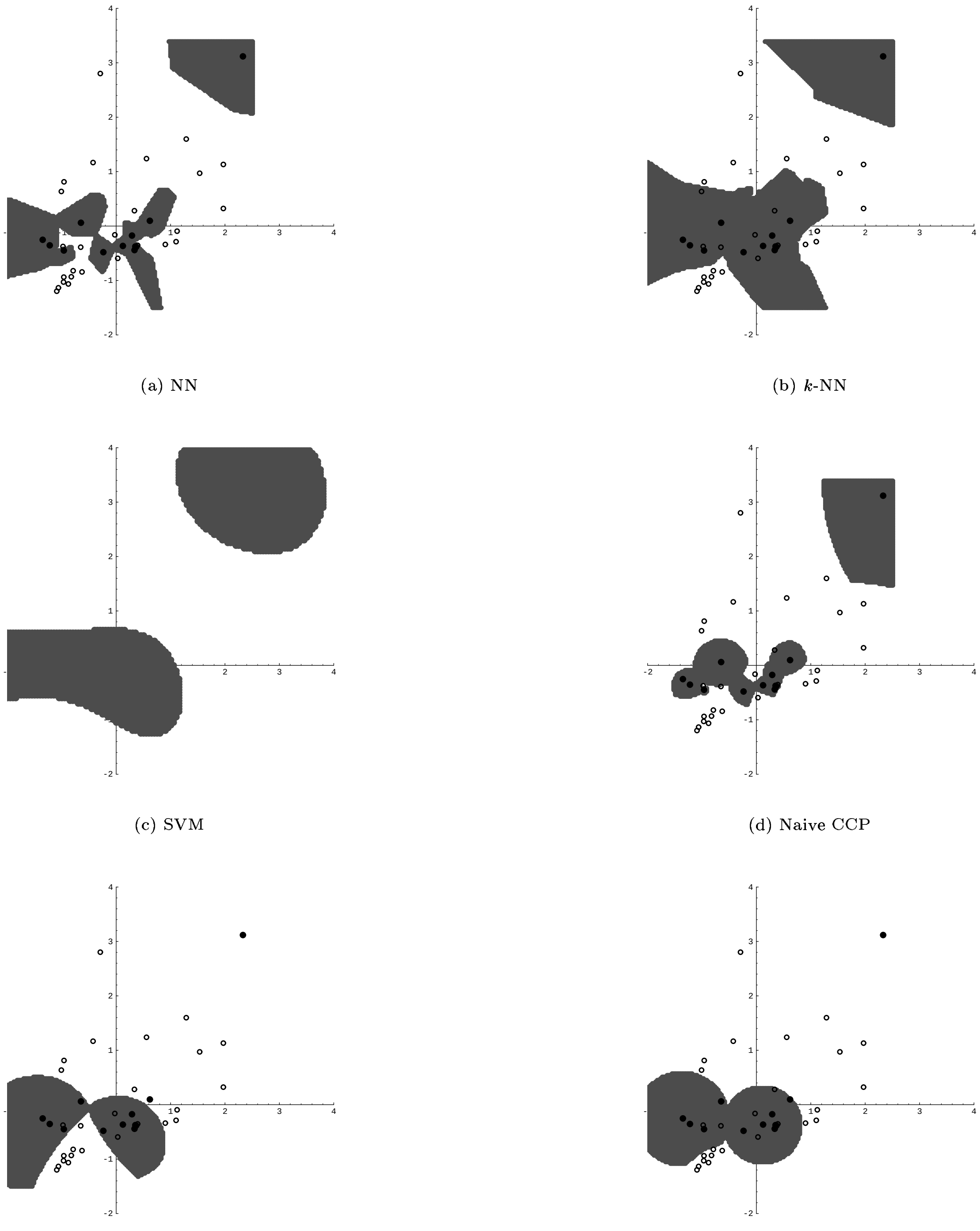
(a) Minefield Data

(b) Naive CCP Cover

(c) α, β CCP cover

(d) RW-CCP cover

Figure 6.9: Covers for minefield data.



(a) NN

(b) *k*-NN

(c) SVM

(d) Naive CCP

(e) α, β CCP

(f) RW-CCP

Figure 6.10: Comparison of classification regions for minefield data.

6.6 Synthetic Data

In this experiment we use the minefield data in the previous section to generate what is called *synthetic* data. We use the available data to estimate the class conditional densities f_0 and f_1. We perform the density estimation using a *kernel density estimation* techniques [37]. Kernel density estimates are a generalization of histograms. For a set of data $Y = \{y_1, y_2, \ldots, y_n\}$ from density f we achieve a kernel density estimate $\hat{f}$ as follows

$$\hat{f}(z) = \frac{1}{nh}\sum_{i=1}^{n}\kappa\left(\frac{y_i - z}{h}\right)$$

where $\kappa()$ is the *kernel function* and h is a smoothing factor. In our estimation of the class conditional densities, we use a kernel function of $\frac{1}{\sqrt{2\pi}}e^{\frac{x^2}{2}}$ and an h value of 0.4. To draw a random observation from our kernel density estimate we simply choose uniformly at random a point x among the observed data and then draw a point from a normal distribution centered at x with standard deviation 0.4.

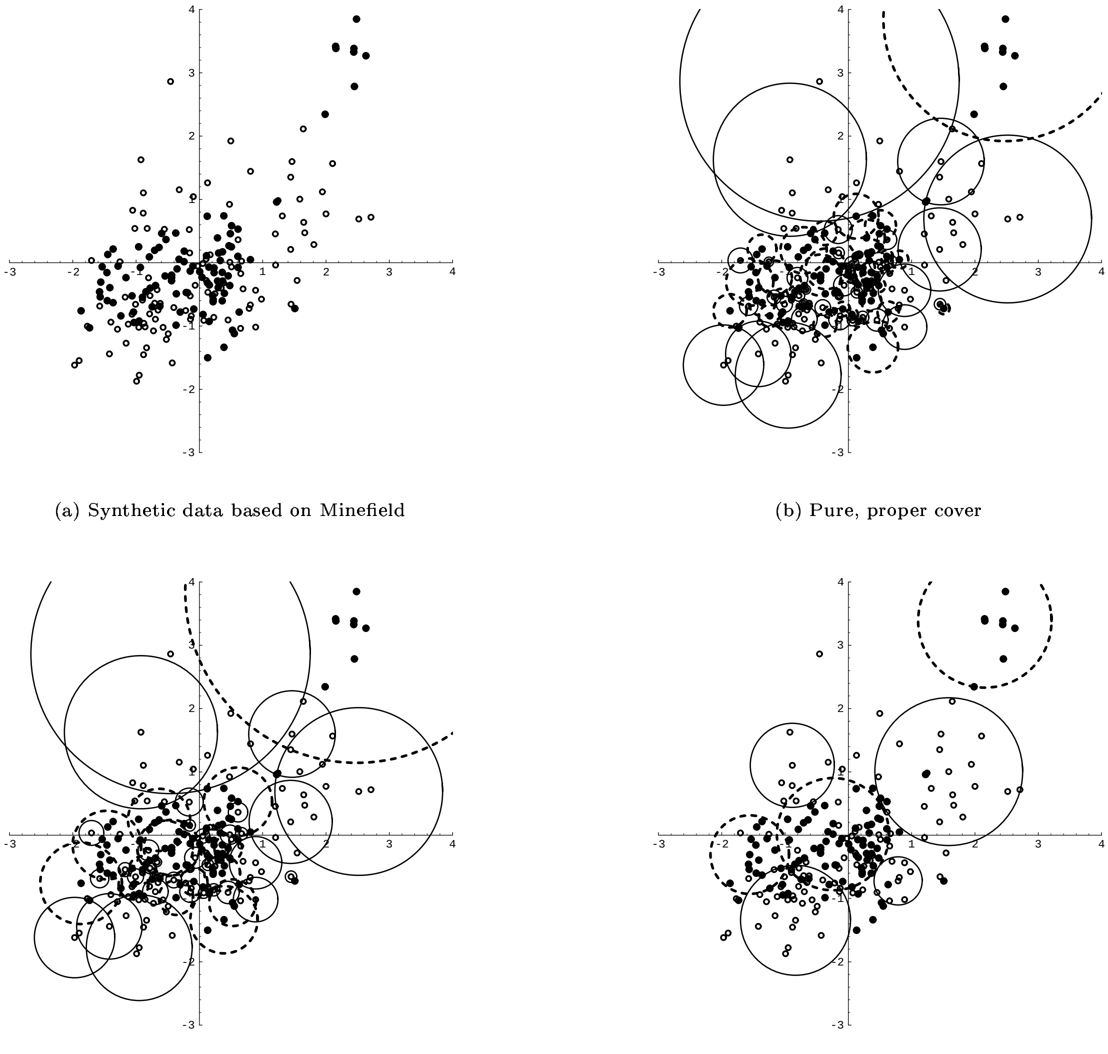
Figure 6.11(a) shows an example of 100 points from each class and Figure 6.11 shows the covers for the three CCP based classifiers. Notice the similarity of the covers in Figure 6.9 and Figure 6.11. Figure 6.12 shows the classification regions for the six classifiers.

Training Size	NN	k-NN	SVM	Naive CCP	α, β CCP	RW-CCP
50	0.345	0.319	0.284	0.339	0.316	0.326
100	0.339	0.292	0.268	0.332	0.300	0.289
200	0.337	0.272	0.261	0.329	0.292	0.273
500	0.335	0.260	0.257	0.326	0.287	0.266

Table 6.13: Misclassification rates for synthetic minefield data.

Tr. Size	Naive CCP		α, β CCP		RW-CCP	
	class 0	class 1	class 0	class 1	class 0	class 1
50	21.3	20.8	13.7	19.9	2.8	3.3
100	41.4	40.0	14.1	36.9	3.3	4.0
200	80.8	77.6	48.2	72.4	3.0	4.7
500	200.0	191.7	125.3	181.3	4.8	6.4

Table 6.14: Average cover cardinality for synthetic minefield data.



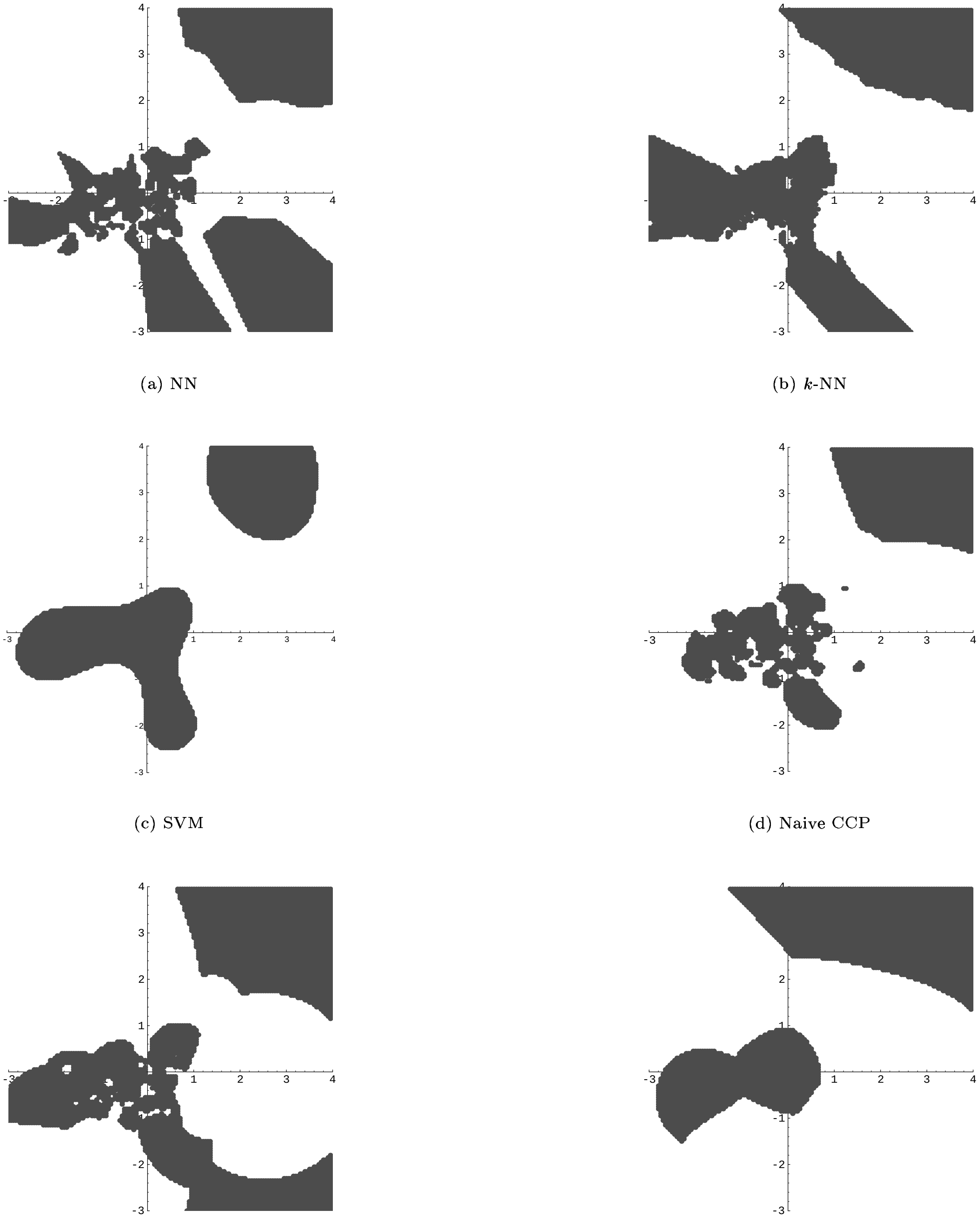
(a) Synthetic data based on Minefield

(b) Pure, proper cover

(c) α, β cover

(d) RW cover

Figure 6.11: Covers of synthetic minefield data



(a) NN

(b) k-NN

(c) SVM

(d) Naive CCP

(e) α, β CCP

(f) RW-CCP

Figure 6.12: Comparison of classification regions for synthetic minefield data.

Chapter 7

Conclusions

We have presented the class cover problem from two different angles. In Part II of this work, we investigate the theoretical properties of a specific CCP. In Chapter 2 we introduce class cover catch digraphs. We give a necessary condition for a digraph to be a CCCD and show this is also a sufficient condition for a special class of CCCDs: Euclidean CCCDs. We also present some results involving the domination number of CCCDs. There remain many unanswered questions regarding the structure of CCCD's in high dimensional Euclidean space and other dissimilarity spaces. For example, there is no conjecture on sufficient conditions for CCCDs in $\mathbb{R}^q$ for any $q > 1$. Other research directions include approximation algorithms for the domination number in CCCDs, and the study of invariants such as chromatic number, number of edges, and maximum cardinality independent sets. Another interesting topic is the study of the pair of CCCD digraphs induced by considering one class as the target class and then switching roles. In Chapter 3 we present results on the domination number in randomized CCCDs. Most of these results involve the CCP in one dimension. Work is currently underway to find similar results in higher dimensions. Random CCP results are motivated by the applications of the CCP to statistical pattern recognition.

In Part III we present applications of the CCP. We introduce three new CCP based classifiers. Each new classifier has its own strengths and weaknesses and are all generalizations of the reduced nearest neighbor classifier. The Naive CCP classifier also behaves much like the nearest neighbor classifier; it is simple to implement and tends to overfit. The α, β CCP classifier is analogous to the k-nearest neighbor classifier as it attempts to improve generalization of the

Naive CCP classifier. The random walk CCP classifier is an adaptive version of the α, β CCP classifier. This presentation of CCP based classifiers is intended to be a proof-of-concept. There is still much work that can be done to improve the performance and our understanding of these methods. This is especially true in the case of the random walk CCP classifier. Many of the methods presented here are ad hoc in nature and could possibly be replaced with more theoretically founded methods. We have also presented an application of the class cover problem to unsupervised classification or clustering. Our method adaptively chooses the number of clusters. In Chapter 6 we present the results of several experiments comparing CCP based classifiers to several popular classifiers. In these tests the performance of our classifiers was competitive.

Appendix A

Parameter selection

Training set size	k-NN	SVM	α, β CCP				
	k	g	α_0	β_0	α_1	β_1	τ
50	3	35	0	2	0	0	0.5
100	3	46	1	3	0	0	0.001
200	4	57	1	5	0	0	0.001
500	4	76	1	6	0	0	0.001

Table A.1: Optimal parameters for model one in two dimensions.

Training set size	k-NN	SVM	α, β CCP				
	k	g	α_0	β_0	α_1	β_1	τ
50	1	18	0	1	0	0	0.5
100	1	29	0	3	0	0	0.001
200	3	46	1	3	0	0	0.5
500	5	69	1	4	0	0	0.001

Table A.2: Optimal parameters for model one in three dimensions.

Training set size	k-NN	SVM	α, β CCP				
	k	g	α_0	β_0	α_1	β_1	τ
50	1	9	0	0	0	0	0.5
100	1	11	1	3	0	0	0.001
200	1	14	2	4	0	0	0.001
500	3	26	5	6	0	0	0.001

Table A.3: Optimal parameters for model one in five dimensions.

Training set size	k-NN	α, β CCP				
	k	α_0	β_0	α_1	β_1	τ
50	55	15	8	15	8	1
100	105	30	18	30	18	0.5
200	216	48	37	48	37	0.5
500	491	105	85	105	85	0.001

Table A.4: Optimal parameters for model two in two dimensions.

Training set size	k-NN	α, β CCP				
	k	α_0	β_0	α_1	β_1	τ
50	61	11	13	11	13	0.5
100	113	22	22	22	22	1
200	191	44	37	44	37	1
500	477	98	96	98	96	0.001

Table A.5: Optimal parameters for model two in three dimensions.

Training set size	k-NN	α, β CCP				
	k	α_0	β_0	α_1	β_1	τ
50	55	16	13	16	13	0.5
100	121	28	20	28	20	0.001
200	223	46	47	46	47	0.001
500	501	5	6	0	0	0.001

Table A.6: Optimal parameters for model two in five dimensions.

k-NN	α, β CCP					
	k	α_0	β_0	α_1	β_1	τ
3	1	4	2	2	1	1

Table A.7: Optimal parameters for minefield data.

Training set size	k-NN	α, β CCP				
	k	α_0	β_0	α_1	β_1	τ
50	5	4	3	3	0	0.001
100	9	4	3	3	0	0.001
200	21	6	1	5	0	0.001
500	37	8	1	10	0	0.001

Table A.8: Optimal parameters for synthetic minefield data.

Appendix B

Notation

$\mathbb{Z}$	the set of integers ($\{\ldots-2,-1,0,1,2,\ldots\}$)
$\mathbb{N}$	the set of natural numbers ($\{0,1,2,\ldots\}$)
$\mathbb{R}^q$	q dimensional real space
$\mathbb{R}_+$	$\{x \in \mathbb{R} : x \geq 0\}$
$x[i]$ for $x \in \mathbb{R}^q$	the i^{th} element in the q dimensional vector x
$\{0\}$	the origin in $\mathbb{R}^q$ for any q
$\angle x, y, z$	the angle formed by points x, y, and z.
$M_{i,j}$ for a matrix M	the element in row i column j in matrix M.
$\mathbf{x}^T$	the transpose of the vector $\mathbf{x}$
A^c for $A \subset \Omega$	the complement of the set A, in other words $\Omega - \{A\}$.
$B(x, r)$	the ball centered at x with radius r.

Bibliography

[1] D. Bertsimas and J. Tsitsiklis. *Linear Optimization*. Athena Scientific, 1997. 1.1

[2] P. Bickel and K. Doksum. *Mathematical Statistics*. Prentice Hall, second edition, 2001. 4.1

[3] K. Bogart. *Introductory Combinatorics*. Harcourt Brace Jovanovich, 1990. 2.2, 2.2

[4] A. Cannon and L. Cowen. Approximations algorithms for the class cover problem. In *6th International Symposium on Artificial Intelligence and Mathematics, 2000*, 2000. 1.2

[5] A.H. Cannon, L.J. Cowen, and C.E. Priebe. Approximate distance classification. *Computing Science and Statistics*, 30:544–549, 1998. 1.2

[6] V. Chvatal. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 1979. 2.1

[7] J. Conway and N. Sloane. *Sphere Packings, Lattices and Groups*. Springer-Verlag, 3 edition, 1999. 2.3

[8] L.J. Cowen and C.E. Priebe. Randomized nonlinear projections uncover high-dimensional structure. *Advances in Applied Mathematics*, 19:319–331, 1997. 1.2

[9] T. Cox and A. Cox. *Multidimensional Scaling*. Chapman and Hall/CRC, 2001. 2.2

[10] N. Cristianini and J. Shawe-Taylor. *Support Vector Machines*. Cambridge, 2000. 1.1, 4.1

[11] F. Critchley and B. Fichet. *Lecture Notes in Statistics: Classification and Dissimilarity Analysis*, volume 93, chapter 2. Springer-Verlag, 1994. 2.2

[12] J. DeVinney and C. Priebe. Class cover catch digraphs. 2002. Submitted for Publication. Available as Technical Report No. 633, Department of Mathematical Sciences, Johns Hopkins University, Baltimore, MD 21218-2682. 1.3

[13] J. DeVinney, C. Priebe, D. Marchette, and D. Socolinsky. Random walks and catch digraphs in classification. *Computing Science and Statistics*, 34, to appear. 1.3

[14] J. Devinney and J. Wierman. A slln for a one-dimensional class cover problem. *Statistics and Probability Letters*, to appear. 1.3

[15] R. Duda, P. Hart, and D. Stork. *Pattern Classification.* Wiley-Interscience, 2001. 1.1, 4.1, 4.3, 5.1, 6.1.1, 6.1.3

[16] K.R. Gabriel and R.R. Sokal. A new statistical approach to geographic variation analysis. *Systematic Zoology*, 1:196–212, 1969. 1.2

[17] M. Garey and D. Johnson. *Computers and Intractability.* W.H. Freeman and Co., 1979. 2.1

[18] W. Gates. The reduced nearest neighbor rule. *IEEE Transactions on Information Theory*, 18:431–433, 1972. 4.2

[19] P. Hart. The condensed nearest neighbor rule. *IEEE Transactions on Information Theory*, 14:515–516, 1968. 4.2

[20] D. Hochbaum, editor. *Approximation Algorithms for NP-Hard Problems.* PWS Publishing Co., 1997. 2.1, 2.3

[21] J.W. Jaromczyk and G.T. Toussaint. Relative neighborhood graphs and their relatives. *Proceedings of the IEEE*, 80(9):1502–1517, 1992. 1.2

[22] T. Joachims. *Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning.* MIT-Press, 1999. 6.2

[23] A. Karr. *Probability.* Springer, 1993. 3.2.4

[24] S. Kulkarni, G. Lugosi, and S. Venkatesh. Learning pattern classification - a survey. *IEEE Transactions on Information Theory*, 44(6):2178–2206, October 1998. 1.1, 4.1

[25] J. Maa, D.K. Pearl, and R. Bartoszynski. Reducing multidimensional two-sample data to one-dimensional interpoint comparisons. *Annals of Statistics*, 24:1069–1074, 1996. 1.1

[26] H. Maehara. A digraph represented by a family of boxes or spheres. *J. Graph Theory*, 8:431–439, 1984. 1.2, 2.2

[27] D.J. Marchette and C.E. Priebe. Characterizing the scale dimension of a high dimensional classification problem. *Pattern Recognition*, 36:45–60, 2003. Available as Technical Report No. 614, Department of Mathematical Sciences, Johns Hopkins University, Baltimore, MD 21218-2682. 1.2

[28] T McKee and F. McMorris. *Topics in Intersection Graph Theory.* SIAM, 1999. 1.2, 2.1

[29] F. McMorris and C. Wang. Sphere of attraction graphs. *Congressus Numerantium*, 142:149–160, 2000. 1.2

[30] A. Okabe, B. Boots, K. Sugihara, and S.N. Chiu. *Spatial Tesselations.* John Wiley and Sons, second edition, 2000. 2.3

[31] T. Olson, J.S. Pang, and C.E. Priebe. A likelihood–mpec approach to target classification. *Mathematical Programming*, to appear. Available as Technical Report No. 590, Department of Mathematical Sciences, Johns Hopkins University, Baltimore, MD 21218-2682. 6.5

[32] C. Priebe and L. Cowen. Approximate distance clustering. *Computing Science and Statistics*, 29:337–346, 1997. 1.2

[33] C. Priebe, J. DeVinney, and D. Marchette. On the distribution of the domination number for random class cover catch digraphs. *Stat. Probab. Lett.*, (55), 2002. 1.3

[34] C. Priebe, D. Marchette, J. DeVinney, and D. Socolinsky. Classification using class cover catch digraphs. 2002. Submitted for publication. Available as Technical Report No. 628, Department of Mathematical Sciences, Johns Hopkins University, Baltimore, MD 21218-2682. 4.2, 4.5

[35] B.D. Ripley. *Pattern Recognition and Neural Networks*, chapter 6.3. Cambridge, 1996. 4.1

[36] S. Ross. *A First Course in Probability.* Prentice Hall, 1998. 3.2.5

[37] B.W. Silverman. *Density Estimation for Statistics and Data Analysis.* Chapman and Hall, 1986. 6.6

[38] D.B. Skalak. Prototype selection for composite nearest neighbor classifiers. Technical report, University of Massachusetts, 1995. 4.2

[39] D. Socolinksy, J. Neuheisel, C. Priebe, D. Marchette, and J. DeVinney. Fast face detection with a boosted cccd classifier. In *Computing Science and Statistics*, to appear. 4.6.4

[40] R. Stanley. *Enumerative Combinatorics*, volume 1. Cambridge, 1997. 2.2

[41] R. Strichartz. *The Way of Analysis.* Jones and Bartlett Publishers, 1995. 3.2

[42] G.T. Toussaint. The relative neighborhood graph of a finite planar set. *Pattern Recognition*, 12:261–268, 1980. 1.2

[43] V. Vapnik. *The Nature of Statistical Learning Theory.* Springer, 1999. 6.1.3

Index

Vita

Jason DeVinney was born in Abington, Pennsylvania on May 17, 1976. He attended Gettysburg College from August 1994 to May 1998. While at Gettysburg College he majored in Mathematics and Physics and minored in Computer Science. After graduating from Gettysburg College, Jason enrolled in the Ph.D. program in the Mathematical Sciences department at Johns Hopkins University. He received his Masters of Science in Mathematical Sciences in May 2000.