Article type: Focus Article

Class Cover Catch Digraphs **Article ID**

David Marchette

Naval Surface Warfare Center

Keywords

graph theory, classification, class cover problem, dominating set

Abstract

The class cover problem is one of finding a small number of sets covering (containing) points from one class without covering any points from a second class. The class cover catch digraph provides a solution to the class cover problem, which can be extended to a nonparametric classifier, similar in flavor to a reduced nearest neighbor classifier. This article describes the class cover catch digraph and its application to classification.

Random graphs occur in many pattern recognition problems. This article describes a particular type of random graph, the class cover catch digraph, which is a type of vertex random graph. Here, the graph is defined by a set of points and sets associated to the vertices. These points represent the training data in a classification problem, and the sets represent the "coverage" or support of one of the classes.

Using the sets, a simple classifier can be defined according to which of the sets contain a new observation. Instead of using all the training data, as in a standard nearest neighbor classifier, the graphs provide a simple method to reduce the complexity of the problem, resulting in a kind of reduced nearest neighbor classifier. We consider several variations of the resulting classifier, and also discuss some of the theoretical results that are known about the graphs.

The Class Cover Problem

The class cover problem, first considered by Cannon and Cowen [2004] (see also DeVinney [2003]), is defined as follows. Given observations from two classes, $\mathcal{X} = \{x_1, \ldots, x_n\}$ and $\mathcal{Y} = \{y_1, \ldots, y_m\}$ what is the smallest number of open balls which cover (contain) all of the elements of $\mathcal{X}$ and none of $\mathcal{Y}$? We further restrict the problem by constraining the balls to be centered at elements of $\mathcal{X}$. We refer to this as the constrained class cover problem. The points in $\mathcal{X}$ are referred to as from the *target class*, the others are *non-target*. An example is illustrated in Figure 1. The problem has a very nice solution in terms of a classic graph theoretic problem. First, we will need some terminology.

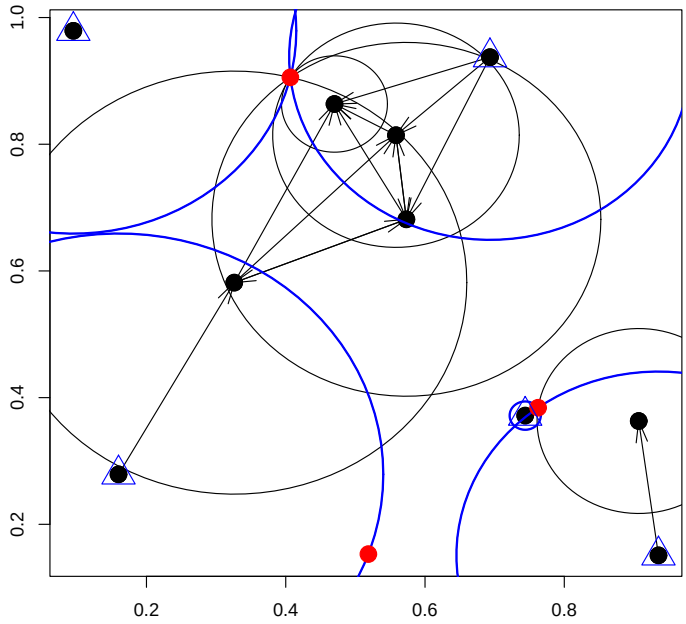

Figure 1: A class cover problem with $n = 10$ (black points) and $m = 3$ (red points). The catch digraph which solves the constrained class cover problem is shown, with it's defining set of balls. A minimal set of balls covering all the black points and none of the red points is shown in blue.

A *graph* (also called a *simple graph*) is a pair (V, E), where V is the set of vertices, $V = \{v_1, \ldots, v_n\}$, and E is the set of edges, $E = \{\{v_i, v_j\}\} \subset V \times V$, unordered pairs of distinct vertices. We will concern ourselves in this work with directed graphs, in which the edges (also called arcs or arrows) are ordered, and we will denote the directed graph (V, D). Such directed graphs are also called *digraphs*. We will use the shorthand notation vw to indicate that $\{v, w\} \in D$. Throughout the rest of this article, we assume all graphs are directed.

The open neighborhood of a vertex $v \in V$, denoted $N(v)$ is the set of vertices with edges from v:

$$N(v) = \{w \in V | vw \in D\}.$$

The closed neighborhood is $N[v] = N(v) \cup \{v\}$. A *dominating set* is a subset $\mathcal{D}$ of the vertices such that every vertex $v \in V$ is either in $\mathcal{D}$ or there exists a $w \in \mathcal{D}$ with $\{w, v\} \in D$. We say that such a vertex is *dominated* by w. Thus, $\mathcal{D}$ is any set of vertices such that

$$\bigcup_{w \in \mathcal{D}} N[w] = V.$$

A dominating set of minimal cardinality is referred to as a *minimal dominating set* and the cardinality of such a set is denoted by γ.

In this article we describe a method for solving the class cover problem using dominating sets of a particular class of directed graphs. Using these, we show that they can be used to define a novel method of classification, and we give some pointers to the research into the statistics behind these class cover catch digraphs. While the methods described in this article can be extended to any metric spaces, we will restrict our discussion to $\mathbb{R}^d$ and Euclidean distance $d(x, y)$.

The Class Cover Catch Digraph

A *catch digraph* is defined in terms of points and sets. Suppose we are given a vertex set V such that to each vertex $v \in V$ is associated both a point $x_v \in \mathbb{R}^d$ and a set $S_v \subset \mathbb{R}^d$. We will usually assume that S_v is an open ball and x_v is its center, but for the general case this is not necessary. The catch digraph defined by these is the digraph (V, D) where $D = \{\{v, w\} \in V \times V | x_w \in S_v\}$. We say that x_w is "caught" by the set S_v, or equivalently, w is caught by v. The class cover catch digraph, first defined in Priebe et al. [2001] (see also DeVinney and Wierman [2006]), is defined as follows: given observations $\mathcal{X} = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$ from class 1 and $\mathcal{Y} = \{y_1, \ldots, y_m\} \subset \mathbb{R}^d$ from class 2, associate to each x_i the open ball

$$B(x_i; \mathcal{Y}) = \{x \in \mathbb{R}^d | d(x, x_i) < \min_{y \in \mathcal{Y}} d(x, y)\}.$$

The *class cover catch digraph* (CCCD) for $\mathcal{X}, \mathcal{Y}$ is the catch digraph defined by the $x \in \mathcal{X}$ and their associated B_x. We will write this as cccd$(\mathcal{X}, \mathcal{Y})$, or simply refer to it as "the CCCD" if $\mathcal{X}$ and $\mathcal{Y}$ are clear. The CCCD for the problem depicted in Figure 1 is shown as arrows between the x_i, with their corresponding balls depicted.

We will use the following notation. We assume that the sets $\mathcal{X}, \mathcal{Y} \subset \mathbb{R}^d$ are given. Let $B(x, r)$ be the open ball of radius r centered at the point x.

$$r_x = \min_{y \in \mathcal{Y}} (d(x, y)) \quad (1)$$

$$B_x = B(x, r_x) \quad (2)$$

Note that the CCCD is not symmetric: if we switch the rolls of $\mathcal{X}$ and $\mathcal{Y}$ we obtain very different graphs. This will become important for classification, discussed below.

The solution to the constrained class cover problem can now be reduced to the problem of finding a minimal dominating set of the corresponding CCCD. This reduction is not without its own problems. The selection of a minimal dominating set for an arbitrary graph or digraph is hard. Fortunately, a greedy algorithm is available and simple to implement:

1. Set $C = \emptyset$, $D = \emptyset$.
2. While $C \neq V$ do:
   (a) Select
   $$v \in \arg \min_{v \in V \setminus D} |N[v] \setminus C|,$$
   a vertex covering the largest number of uncovered vertices.
   (b) Set $D = D \cup \{v\}$ and $C = C \cup N[v]$.
3. Return D.

Note that this algorithm does not uniquely define a dominating set, since we do not specify how to select among equivalent vertices in step 2a. While it may be possible to improve the results by clever selection at this stage, this results in a more computationally complex algorithm. Instead, we select arbitrarily. Note further that the minimal dominating set is not assumed to be unique, and in fact often is not. Finally, the greedy algorithm is not guaranteed (except in very special circumstances) to produce a minimal dominating set, but in practice it does produce near-minimal dominating sets that will be adequate for the methods described below.

Statistical Analysis of the CCCD

Several results are known about the statistical properties of the CCCD, in particular about the value of γ. For real valued a, b, let $\mathcal{Y} = \{a, b\}$ and $\mathcal{X} = \{x_1, \ldots, x_n\} \subset [a, b]$ uniformly distributed on $[a, b]$. Priebe et al. [2001] showed that:

1. γ for cccd$(\mathcal{X}, \mathcal{Y})$ is 1 or 2.
2. $P[\gamma = 1] = \frac{5}{9} + \frac{4}{9} 4^{-(n-1)}$.

3. The greedy algorithm always returns a minimal dominating set.

The authors prove more general, analogous results for the case where $m > 2$, and compute the expected value of γ for the uniform case (X and Y are both uniformly and independently distributed on an interval (a, b)) and Ceyhan [2008] extends these results to non-uniform distributions.

There are also law of large numbers and central limit results available for the 1-dimensional case. See DeVinney and Wierman [2003], Wierman and Xiang [2008], Xiang and Wierman [2009]. For example, let $\mathcal{X} = \{x_1, \ldots, x_n\}$ and $\mathcal{Y} = \{y_1, \ldots, y_m\}$ be distributed uniformly, independently on $[0, 1]$, and let $m = \lfloor rn \rfloor, r \in (0, \infty)$. Then writing $\gamma = \gamma_{n,m}$ we have

$$\lim_{m \to +\infty} \frac{\gamma_{n,m}}{m} = \frac{12r + 13}{3(r + 1)(4r + 3)}$$

almost surely.

Xiang and Wierman [2009] prove that under the same conditions as the SLNN result above,

$$\frac{1}{\sqrt{m}}(\gamma - E[\gamma]) \xrightarrow{\mathcal{L}} N(0, \sigma^2),$$

and give an exact limiting value for the variance.

The higher dimensional cases are more difficult to analyze, and not much is know. However, in a series of papers starting with his dissertation, Ceyhan defined a variant of the standard CCCD that allows for some calculations. See Ceyhan [2004], Ceyhan and Priebe [2005, 2007], Ceyhan et al. [2006, 2007].

Classification

Let $W \in \mathbb{R}^d$ and and $Z \in \{1, \ldots, C\}$ be random variables. Assume we observe pairs $(w_1, z_1) \ldots, (w_n, z_n)$. The classification problem is to find a classifier $g : \mathbb{R}^d \to \{1, \ldots, C\}$ such that some given loss is minimized: for example, given a W with true value Z, minimize the probability of error

$$P[g(W) \neq Z].$$

In the following, we will restrict ourselves to the case $C = 2$, except where explicitly stated. We let X and Y represent the conditional random variables $W|Z = 1, W|Z = 2$.

Given a CCCD, say $G_{\mathcal{X},\mathcal{Y}} = \text{cccd}(\mathcal{X}, \mathcal{Y})$, we set $\mathcal{S}_{\mathcal{X}}$ to be a dominating set of $G_{\mathcal{X},\mathcal{Y}}$ (say, computed via the greedy algorithm above), and $\mathcal{B}_{\mathcal{X}}$ the set of corresponding open balls.

A simple classifier (referred to as a *preclassifier* in Priebe et al. [2003a]) can be constructed as follows:

1. Given training data $\mathcal{X} = \{x_1, \ldots, x_n\}$ and $\mathcal{Y} = \{y_1, \ldots, y_m\}$, construct $G_{\mathcal{X},\mathcal{Y}}$ and $G_{\mathcal{Y},\mathcal{X}}$.

2. Using the greedy algorithm (or any other method for finding small cardinality dominating sets) construct $\mathcal{B}_\mathcal{X}$ and $\mathcal{B}_\mathcal{Y}$.

3. Define the preclassifier g as follows:

$$g(z) = \begin{cases} 1 & \exists B_x \in \mathcal{B}_\mathcal{X} \text{ such that } z \in B_x, z \notin B_y \forall B_y \in \mathcal{B}_\mathcal{Y} \\ 2 & \exists B_y \in \mathcal{B}_\mathcal{Y} \text{ such that } z \in B_y, z \notin B_x \forall B_x \in \mathcal{B}_\mathcal{X} \\ \text{ambiguous} & \exists B_x \in \mathcal{B}_\mathcal{X}, B_y \in \mathcal{B}_\mathcal{Y} \text{ such that } z \in B_x \cap B_y \\ \text{outlier} & z \notin \mathcal{B}_\mathcal{X} \cup \mathcal{B}_\mathcal{X} \end{cases}$$

Thus, if we can unambiguously state that z is in the support of one class and not the other, we make the call, otherwise we make no decision. Note that this is not, strictly speaking, a classifier because of the two "no decision" cases, hence the term preclassifier. We can correct this by using the ball radii to determine how "close" z is to the training data retained in $\mathcal{S}_\mathcal{X}$ and $\mathcal{S}_\mathcal{Y}$:

$$g_{\text{cccd}}(z) = I\{\min_{x \in \mathcal{S}_\mathcal{X}} d(x,z)/r_x < \min_{y \in \mathcal{S}_\mathcal{Y}} d(y,z)/r_y\} + 1. \tag{3}$$

Here I is the indicator function. Distances are scaled by ball radii, and these scaled distances then allow us to produce a true classifier. Note that since we are using the reduced training set given by the dominating sets, we have produced a type of reduced nearest neighbor classifier (Devroye et al. [1996]).

It is easy to see how to extend this classifier to the multi-class case. Since it is defined in terms of the CCCDs, and each of these is constructed using one class as the target, we may identify the other classes as "non-target" and proceed as above, modifying Equation (3) to select the class with minimum scaled distance:

$$g_{\text{cccd}}(z) = \arg\min_{c \in C}\{\min_{x \in \mathcal{S}_{\mathcal{X}_c}} d(x,z)/r_x\}, \tag{4}$$

where $\mathcal{X}_c$ denotes the training data for class c.

The constraint that we have a pure (no points from the other class are covered) and proper (all points of the target class are covered) solution to the class cover problem can be relaxed in several ways. One, considered in Marchette and Priebe [2003], is to allow for each ball to cover a fixed number of non-target points, and to allow a fixed number of target points to go uncovered. An alternative method, discussed in DeVinney et al. [2002] and Marchette [2004] is to use a "random walk" to set the ball radii. The idea is to slowly grow the balls and keep track of how fast non-target points are covered, as compared to target points. When the ball starts to see "too many" non-target points relative to the number of target points covered, the ball stops growing. This allows balls to cover a small number of "outlier" points encroaching on the target class. Once the balls are defined, the CCCDs are constructed, and the classification proceeds as above.

Other variations and applications of the CCCD that have appeared in the literature are a one-class version of the CCCD (applied to face detection in images, Eveland et al. [2005]), detection of aggregation and segregation of spatial patterns (Ceyhan and Priebe [2005], Ceyhan et al. [2006, 2007]), clustering using the random walk version of the CCCD (Marchette [2004]) and an application to Gene expression monitoring in DNA microarrays (Priebe et al. [2003b]).

Examples

A simple three-class classification problem is depicted in Figure 2. The training data (1000 observations drawn uniformly from $[-2, 2]^2$, colored according to class) are depicted in the top left plot, with the circles defining the resultant classifier in the top right plot, and the decision regions in the bottom plot. Although the CCCD uses balls to define the regions, it has little trouble estimating the straight decision boundaries between the three classes.

The resulting classifier uses 15 balls to define the blue class (752 training points), 10 to define the red (117 training points), and 11 to define the black (131 training points). There were $10,000$ points used to define the decision regions, and the confusion matrix for these is given in Table 1. A total of 219 errors were made, for an error rate of about 2%. A nearest neighbor classifier produces 282 errors, so the reduced complexity (from $1,000$ points to 36) has not harmed the classifier.

Table 1 Confusion table for the simple example of Figure 2.

	True Class		
Estimated Class	1	2	3
1	7445	60	41
2	26	1135	33
3	29	30	1201

$10,000$ observations from the square.

A slightly more challenging situation is provided by the KDDCUP [1] intrusion detection dataset. This is an artificial dataset, in some sense, extracted from a simulation designed to generate data for testing intrusion detection systems. We considered a subset of the data consisting of the variables in Table 2, and classifying each record as "normal" or "abnormal" (attack or other non-normal data). For this experiment we used only the TCP packets, and selected a random training set of 1000 observations from each class, testing on the remaining $1,868,596$ observations. The results of the CCCD classifier are given in Table 3.

[1] `http://www.sigkdd.org/kddcup/index.php?section=1999&method=info`

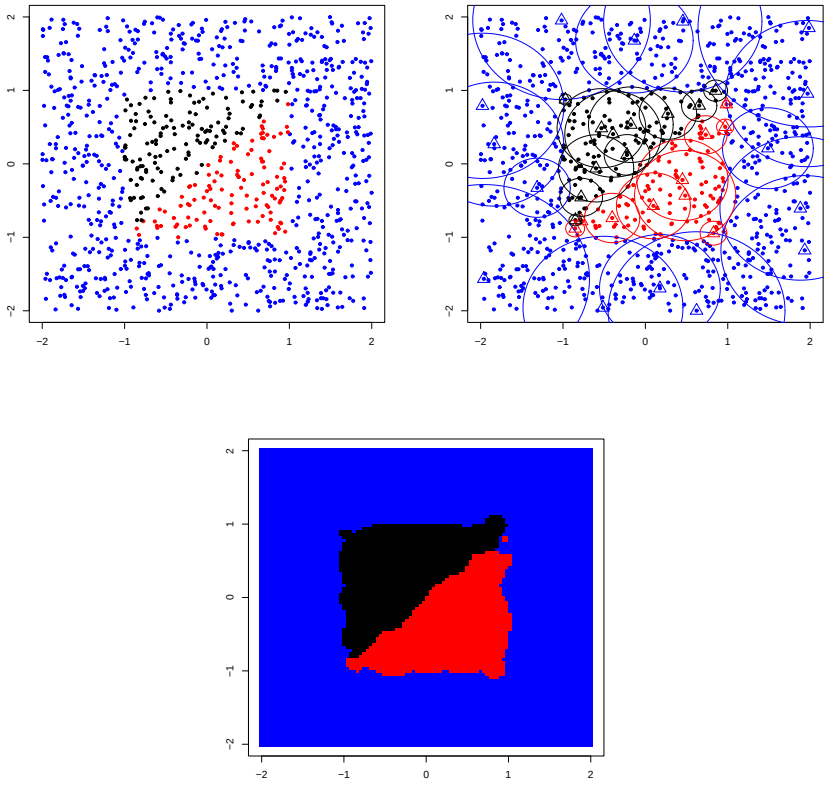

Figure 2: An example training set for a three class classification problem, and the resulting decision regions defined by the class cover catch digraph classifier of Equation (4).

Table 2 Variables used for the intrusion detection dataset.

Name	Value
Duration	Length of the session in time
Src Bytes	Number of bytes from the source
Dst Bytes	Number of bytes from the destination
Count	Number of packets in the session
Dst Hosts	Number of destination hosts
Dst Hosts Same Srv	Number of connections to same service
Dst Hosts Diff Srv	Number of connections to different service
Dst Hosts Same Port	Number of connections with same source port

The variables chosen are somewhat arbitrary, more for illustration purposes than as a serious attempt to solve the intrusion detection problem. There have been many criticisms of the data; see Marchette [2001] for more information.

Table 3 Confusion table for the normal/abnormal computer security dataset. Columns add to 1.0

	True Class	
Estimated Class	Normal	Abnormal
Normal	0.9945	0.0025
Abnormal	0.0055	0.9975

KDD Intrusion Dataset

The results appear quite promising. Most interesting, the CCCD classifier chose to use 9 exemplars for each class, a vast reduction in complexity over the 2000 observations in the training set. From Figure 3, several observations are possible, even for such a complicated 8-dimensional problem. First, the largest ball belongs to the "normal" class, which fits with our intuition that there are many ways that Internet sessions may be "normal", and thus these should take up a large portion of "session space". It also reassures us that there are likely to be large regions in this parameter space in which sessions are likely to be "normal".

An alternative explanation of the large radius is that its defining observation is an outlier, and thus perhaps the ball should be dropped from the classifier. To test this, we simply remove that point from the CCCD, and reclassify the test data. Removing the point results in an increase of the false alarm rate from 0.0055 to 0.0091 while decreasing the miss rate from 0.0025 to 0.0022. In this application, nearly doubling the false alarm rate is probably more serious than the tiny reduction in misses warrants, and so the ball should probably remain in the classifier. This ability to edit and adjust the classifier "by hand" seems worthy of investigation, but we are not aware of any further work in this direction, although Solka et al. [2002] presents a software system in which such experiments might easily take place.

Figure 3 also shows that large numbers of destination bytes are "normal"; this makes some sense as these tend to be transfers such as web pages, images, etc. Recall that "source" refers to the initiating computer, which typically is in the protected network in these data. Large transfers from the source are more likely to be "abnormal" as can be seen by the red curve in the top hand plot. From the bottom hand plot, we see that large numbers of different hosts or ports are indications of "abnormal", which also meets with our intuition, since activities such as these tend to be scanning attacks rather than normal activity.

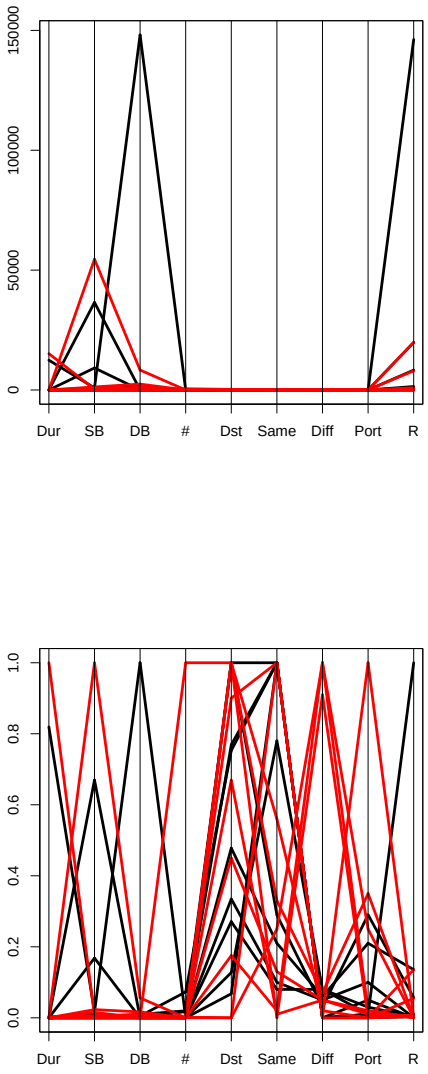

Figure 3: Solution ball centers chosen by the CCCD for each class, plotted in two parallel coordinates plots. On the top the axes are unscaled, on the bottom each axis is scaled independently. The axes are in the order of Table 2, with the ball radius appended as the last axis. Black corresponds to "normal" and red to "abnormal".

Conclusion

The class cover catch digraph provides a simple and powerful set of classifiers. It implements a type of reduced nearest neighbor classifier, using a combination of balls defined in terms of an application of the class cover problem to the training data. By using a directed graph to encode the class coverage, a graph-theoretic methodology, the minimal dominating set, can be used to reduce the complexity of the classifier. This provides a fast and efficient classification methodology. As seen on the few examples shown here, as well as those discussed in the literature, the CCCD is a useful tool for the statistician.

References

A. H. Cannon and L. J. Cowen. Approximation algorithms for the class cover problem. *Annals of Mathematics and Artificial Intelligence*, 40:215–223, 2004.

E. Ceyhan. *An Investigation of Proximity Catch Digraphs in Delaunay Tessellations*. PhD thesis, The Johns Hopkins University, 2004.

E. Ceyhan. The distribution of the domination number of class cover catch digraphs for non-uniform one-dimensional data. *Discrete Mathematics*, 308: 5376–5393, 2008.

E. Ceyhan and C. E. Priebe. The use of domination number of a random proximity catch digraph for testing spatial patterns of segregation and association. *Statistics & Probability Letters*, 73:37–50, 2005.

E. Ceyhan and C. E. Priebe. On the distribution of the domination number of a new family of parametrized random digraphs. *Model Assisted Statistics and Applications*, 1:231–255, 2007.

E. Ceyhan, C. E. Priebe, and D. J. Marchette. A new family of random graphs for testing spatial segregation. *Canadian Journal of Statistics*, 35:27–50, 2007.

E. Ceyhan, C. E. Priebe, and J. C. Wierman. Relative density of the random r-factor proximity catch digraphs for testing spatial patterns of segregation and association. *Computational Statistics and Data Analysis*, 50:1925–1964, 2006.

J. DeVinney. *The Class Cover Problem and its Applications in Pattern Recognition*. PhD thesis, The Johns Hopkins University, 2003.

J. DeVinney, C. E. Priebe, D. J. Marchette, and D. Socolinsky. Random walks and catch digraphs in classification. *Computing Science and Statistics*, 34: 107–117, 2002.

J. DeVinney and J. C. Wierman. A SLLN for a one-dimensional class cover problem. *Statistics & Probability Letters*, 59:425–435, 2003.

J. DeVinney and J. C. Wierman. A new family of proximity graphs: Class cover catch digraphs. *Discrete Applied Mathematics*, 154:1975–1982, 2006.

L. Devroye, L. Gyrofi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer, New York, 1996.

C. K. Eveland, D. A. Socolinsky, C. E. Priebe, and D. J. Marchette. A hierarchical methodology for one-class problems with skewed priors. *Journal of Classification*, 22:17–48, 2005.

D. J. Marchette. *Computer Intrusion Detection: a Statistical Viewpoint*. Springer, New York, 2001.

D. J. Marchette. *Random Graphs for Statistical Pattern Recognition*. John Wiley, New York, 2004.

D. J. Marchette and C. E. Priebe. Characterizing the scale dimension of a high dimensional classification problem. *Pattern Recognition*, 36:45–60, 2003.

C. E. Priebe, J. DeVinney, and D. J. Marchette. On the distribution of the domination number for random class cover catch digraphs. *Statistics & Probability Letters*, 55:239–246, 2001.

C. E. Priebe, D. J. Marchette, J. DeVinney, and D. Socolinsky. Classification using class cover catch digraphs. *Journal of Classification*, 20:3–23, 2003a.

C. E. Priebe, J. L. Solka, D. J. Marchette, and T. Clark. Class cover catch digraphs for latent class discovery in gene expression monitoring by DNA microarrays. *Computational Statistics and Data Analysis*, 43:621–632, 2003b.

J. L. Solka, B. T. Clark, and C. E. Priebe. A visualization framework for the analysis of hyperdimensional data. *International Journal of Image and Graphics*, 2: 145–161, 2002.

J. C. Wierman and P. Xiang. A general SLLN for the one-dimensional class cover problem. *Statistics & Probability Letters*, 78:1110–1118, 2008.

P. Xiang and J. C. Wierman. A CLT for a one-dimensional class cover problem. *Statistics & Probability Letters*, 79:223–233, 2009.

Cross-References

Graph theory Directed graph Classification