The Cauchy-Schwarz Divergence and Parzen Windowing: Connections to Graph Theory and Mercer Kernels

Robert Jenssen[a1], Jose C. Principe[b], Deniz Erdogmus[c], and Torbjørn Eltoft[a]

[a] *Department of Physics, University of Tromsø, N-9037 Tromsø, Norway*

[b] *Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL., 32611, USA*

[c] *Department of Computer Science and Engineering, Oregon Graduate Institute, OHSU, Portland, OR., 97006, USA*

Abstract

This paper contributes a tutorial level discussion of some interesting properties of the recent Cauchy-Schwarz (CS) divergence measure between probability density functions. This measure brings together elements from several different machine learning fields, namely information theory, graph theory and Mercer kernel - and spectral theory. These connections are revealed when estimating the CS divergence non-parametrically using the Parzen window technique for density estimation. An important consequence of these connections is that they enhance our understanding of the different machine learning schemes relative to each other.

Key words: Cauchy-Schwarz divergence, Parzen windowing, information theory, graph cut, Mercer kernel theory, spectral methods.

1 Introduction

Recently, a new scheme for statistically based machine learning has emerged, coined *information theoretic learning* (ITL) [1]. The starting point is a data set that globally conveys information about a real-world event. The goal in ITL is to capture, or learn, this information in the form of the parameters of an adaptive system.

[1] Tel. (+47) 776 46493, Fax. (+47) 776 45580,
Email: robertj@phys.uit.no, Web: www.phys.uit.no/∼robertj

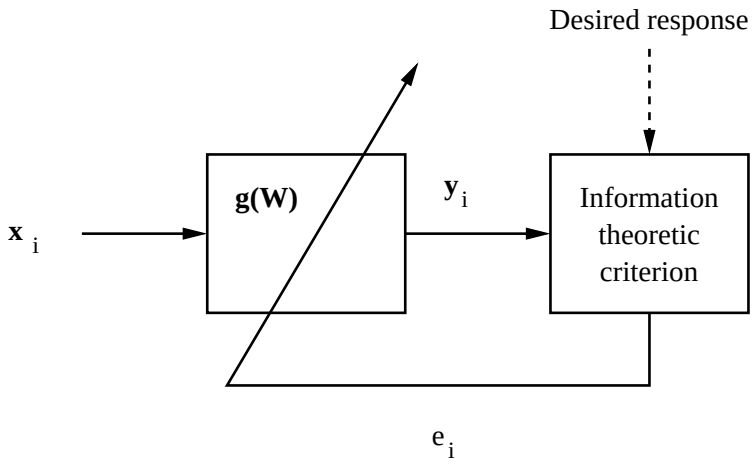


Fig. 1. Illustration of ITL setup.

This is done using *information theoretic cost functions* as learning criteria. As opposed to the traditional mean squared error criterion, information theoretic cost functions take into account statistical dependencies beyond correlations. This is important in many problems in machine learning, such as blind source separation and independent component analysis, blind equalization and deconvolution, subspace projections, dimensionality reduction and manifold learning, feature extraction, classification and clustering.

Figure 1 shows a schematic illustration of ITL. Typically, ITL is an iterative process, where the data exemplar $\mathbf{x}_i$ is presented to the system at iteration i, and the output is given by $\mathbf{y}_i = \mathbf{g}(\mathbf{W})\mathbf{x}_i$. The function $\mathbf{g}(\mathbf{W})$ represents a possibly non-linear data transformation, and the goal is to perform a specific task, according to an information theoretic criterion. The system may be guided by a correction term e_i, and the system may receive external input in the form of a desired response.

The ITL scheme implicitly requires probability density functions (pdfs) to be estimated, in order to evaluate the information theoretic criterion. Since it is oftentimes desirable to make as few assumptions as possible about the structure of the pfds in question, Principe et al. [1] argued that Parzen windowing is the appropriate density estimation technique. Combined with information theoretic criteria based on Renyi's quadratic entropy, ITL has been applied with great success on several supervised and unsupervised learning schemes, see for example [2–9,6,10].

The choice of using information theoretic criteria based on Renyi's quadratic entropy as opposed to measures based on for example Shannon's entropy was not arbitrary. One important reason for introducing these new cost functions into machine learning was that they may be estimated without making any approximations besides the Parzen windowing itself. One important quantity which was proposed was the so-called Cauchy-Schwarz (CS) pdf divergence. The CS divergence is a measure of the "distance" between two probability density functions, $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$. This

measure is given by

$$D_{CS}(p_1, p_2) = -\log \frac{\int p_1(\mathbf{x}) p_2(\mathbf{x}) d\mathbf{x}}{\sqrt{\int p_1^2(\mathbf{x}) d\mathbf{x} \int p_2^2(\mathbf{x}) d\mathbf{x}}}. \quad (1)$$

This is a symmetric measure, such that $0 \leq D_{CS} < \infty$, where the minimum is obtained if and only if $p_1(\mathbf{x}) = p_2(\mathbf{x})$.

This paper provides a tutorial level discussion of some interesting properties of the Cauchy-Schwarz divergence. It turns out that when Parzen windowing is used to estimate the CS divergence, the resulting cost function can be interpreted both in terms of graph theory and Mercer kernel - and spectral theory. Figure 2 illustrates these connections, which we will discuss further in the following sections. Hence, these links indicate that graph theoretic measures and Mercer kernel based measures are closely related to information theory and non-parametric density estimation. Graph theory and Mercer kernel based theory have been important parts of machine learning research in recent years.

Graph theory [11] has been used for decades in various scientific fields for many purposes. In the last decade, it has also been introduced to the field of computer vision and machine learning, by optimizing the so-called *graph cut* [12] The graph cut provides a measure of the cost of partitioning a graph into two subgraphs.

Mercer kernel based methods [13–16] have been dominating in machine learning and pattern recognition since the introduction of the support vector machine [17–20]. Here, the main idea is to implicitly map the data points into a potentially infinite dimensional non-linear feature space using Mercer kernels. In the Mercer kernel feature space, it is more likely to obtain linearly separable data, and linear machine learning techniques may be used.

Quite recently, yet another machine learning field has received significant attention, namely the spectral methods [13]. Spectral methods refer to techniques using the eigenvalues (spectrum) and eigenvectors of certain data matrices to perform the machine learning tasks. See for example [21–25]. Kernel PCA [26] is one prominent example of a spectral method. It basically performs a principal component analysis (PCA) approximation to Mercer kernel feature spaces.

The remainder of the paper is organized as follows. In section 2 the Parzen window technique for density estimation is discussed. In section 3, it is shown how the CS divergence may be estimated non-parametrically using the Parzen window method. Furthermore, in section 4, the connection to graph theory is discussed, and in section 5 the connection to Mercer kernel - and spectral methods is discussed. Section 6 discusses an extension of the CS divergence. We make our concluding remarks in section 7.

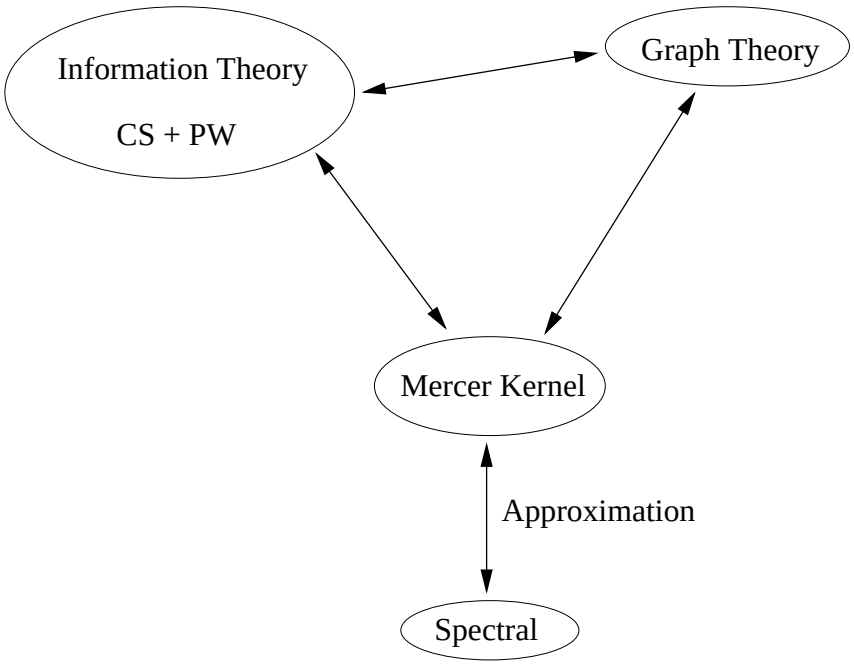


Fig. 2. The Cauchy-Schwarz (CS) divergence estimated using Parzen windowing (PW) has equivalent expressions in graph theory, Mercer kernel theory and spectral theory.

2 Parzen Windowing

In this section, we will review a well-known technique for probability density function estimation that is known as Parzen windowing, or kernel density estimation. Our review follows those given in [27–29].

There are two approaches for estimating the pdf of a random variable from its independent and identically distributed samples; parametric and non-parametric. In parametric density estimation, it is assumed that a parametric model for the pdf in question is known apriori. The task is then to estimate the model parameters from the samples, using for example the maximum likelihood principle. However, it is frequently the case that we have no apriori information about the form of the densities. In that case, it is not recommended to select a specific model for the density, because it may not describe the data samples well at all. On the contrary, it is desirable to be able to estimate the density without making any model assumptions, that is, we wish to estimate the density non-parametrically.

There are several approaches to non-parametric density estimation. One of the most well-known and widely used techniques is known as Parzen windowing [30]. Assume that we wish to estimate the density $f(\mathbf{x})$ of the process generating the d-dimensional sample $\mathbf{x}_1, \ldots, \mathbf{x}_N$. The Parzen window estimator for this distribution

is defined as

$$\hat{f}(\mathbf{x}) = \frac{1}{N}\sum_{l=1}^{N} W_{\sigma^2}(\mathbf{x}, \mathbf{x}_l). \tag{2}$$

Here, W_{σ^2} is the Parzen window, or kernel, and σ^2 controls the width of the kernel. The Parzen window must integrate to one, and is typically chosen to be a pdf itself, such as the Gaussian kernel. Hence,

$$W_{\sigma^2}(\mathbf{x}, \mathbf{x}_l) = \frac{1}{(2\pi\sigma^2)^{\frac{d}{2}}} \exp\left\{-\frac{||\mathbf{x} - \mathbf{x}_l||^2}{2\sigma^2}\right\}. \tag{3}$$

Other window functions may also be used, such as the triangle, Epanechnikov, biweight and triweight kernels [28].

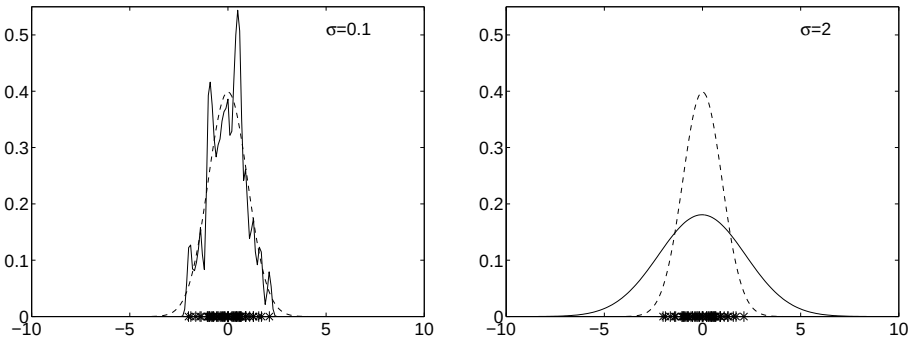
It is well-known that the most important parameter in Parzen windowing is the kernel size, given by σ, and to a lesser extent the actual form of the window used. To illustrate the dependence on σ, we have created a simple one-dimensional data set, and show in Fig. 3 the resulting pdf estimates for different σ. The 50 data samples used is generated from a standard normal density. In Fig. 3 (a), the Parzen window pdf estimate (solid line) using a kernel size $\sigma = 0.1$ is shown. The estimate is compared to a standard normal density (stapled line). Clearly, the estimate is not smooth enough, and does not approximate the true underlying density very well. In (b), we show the estimate corresponding to $\sigma = 2$. In this case the opposite effect is observed, and the estimate is clearly too smooth. Finally, in (c) we show the result obtained using $\sigma = 0.45$. In this case the estimate approximates the true density quite well. This poses the question; is there any data-driven method to determine $\sigma = 0.45$ as the appropriate kernel size?

2.1 Determining the Parzen Window Width

It is easily shown that Eq. (2) is an asymptotically unbiased and consistent estimator provided σ decays to zero at a certain rate as N tends to infinity [30]. In the finite sample case, the kernel size has to be chosen in a trade-off between estimation bias and variance. We illustrate this in the one-dimensional case. The mean integrated squared error (MISE) is the appropriate measure for analyzing $\hat{f}(x)$, where

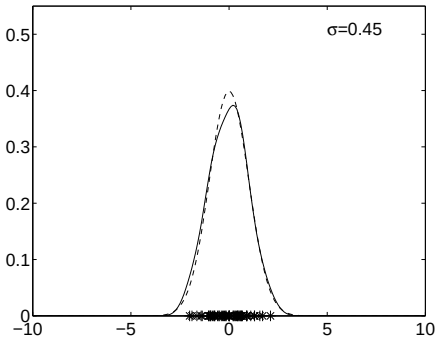
$$MISE\left\{\hat{f}(x)\right\} = \int \left[E\left\{\hat{f}(x)\right\} - f(x)\right]^2 dx + \int Var\left\{\hat{f}(x)\right\} dx, \tag{4}$$

where $E\left\{[\hat{f}(x) - f(x)]^2\right\} = \left[E\left\{\hat{f}(x)\right\} - f(x)\right]^2 + Var\left\{\hat{f}(x)\right\}$ is the mean squared error. Finding the kernel size which minimizes the MISE can be obtained by various cross-validation techniques [29]. Another straight-forward approach is to analyze



(a) Estimate for $\sigma = 0.1$.

(b) Estimate for $\sigma = 2$.



(c) Estimate for $\sigma = 0.45$.

Fig. 3. Using the Parzen window method to estimate a density based on 50 samples drawn from a standard normal density. The Parzen window estimate is shown using a solid line. The location of each sample is indicated by the symbol $\star$. The standard normal density is shown using a stapled line. The kernel size, given by σ, has a visible effect on the resulting estimates.

the MISE asymptotically, i.e. when the number of samples N goes to infinity. The resulting expression for the asymptotic MISE (AMISE) becomes

$$AMISE\left\{\hat{f}(x)\right\} = \frac{\sigma^4 \mu_2^2(K) R(f'')}{4} + \frac{R(K)}{\sigma N}, \tag{5}$$

where K is in this case the standard normal density function, $\mu_2(K) = \int z^2 K(z) dz$, $R(f'') = \int \{f''(x)\}^2 dx$ where $f''(x) = \frac{d^2}{dx^2} f(x)$, and $R(K) = \int K(z)^2 dz$. It can be seen that the left term on the right-hand side of Eq. (5) is minimized by minimizing σ. This is the bias part. However, the right term, which is the variance part, is minimized by maximizing σ. Hence, there is an inherent bias-variance trade-off associated with the Parzen window technique for density estimation.

Note that one may obtain an explicit formula for the AMISE optimal kernel size by differentiating Eq. (5) and equating it to zero, obtaining

$$\sigma_{AMISE} = \left[\frac{R(K)}{\mu_2^2(K) R(f'') N} \right]^{\frac{1}{5}}. \tag{6}$$

One straight-forward approach is to estimate $R(f'')$ with reference to a normal density. This quantity is then plugged back into Eq. (6) to obtain an estimate for σ_{AMISE}. It can be shown that the corresponding kernel size is given by $\hat{\sigma}_{AMISE} \approx 1.06 \hat{\sigma} N^{-\frac{1}{5}}$, where $\hat{\sigma}$ is an estimate of the standard deviation of the normal density [27]. In the d-dimensional case, the normal reference rule becomes

$$\sigma_{AMISE} = \hat{\sigma} \left[\frac{4}{(2d+1)N} \right]^{\frac{1}{d+4}}, \tag{7}$$

where $\hat{\sigma}^2 = d^{-1} \sum_i \mathbf{\Sigma}_{ii}$, and $\mathbf{\Sigma}_{ii}$ are the diagonal elements of the sample covariance matrix.

In fact, when determining the Parzen window width in the simple example illustrated in Fig. 3, the normal reference rule yields $\sigma = 0.45$.

3 Cauchy-Schwarz Divergence

Measures of how close two pdfs $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$ are in some specific sense, are provided by the information theoretic divergences, such as the Kullback-Leibler divergence [31] or the Chernoff divergences [32]. In this paper, we focus on the so-called Cauchy-Schwarz divergence, recently proposed by Principe et al. [1].

Define the inner-product between two square-integrable functions $h(\mathbf{x})$ and $g(\mathbf{x})$ as $\langle h, g \rangle = \int h(\mathbf{x}) g(\mathbf{x}) d\mathbf{x}$. Then, by the Cauchy-Schwarz inequality

$$\left| \int h(\mathbf{x}) g(\mathbf{x}) d\mathbf{x} \right|^2 \leq \int |h(\mathbf{x})|^2 d\mathbf{x} \int |g(\mathbf{x})|^2 d\mathbf{x}, \tag{8}$$

with equality if and only if the two functions are linearly dependent. Now consider two pdfs, $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$, i.e. non-negative functions which integrate to one. In this case, a measure of the "distance" between the pdfs may be defined, which was named the Cauchy-Schwarz divergence [1]. We repeat the expression, as

$$D_{CS}(p_1, p_2) = -\log \frac{\int p_1(\mathbf{x}) p_2(\mathbf{x}) d\mathbf{x}}{\sqrt{\int p_1^2(\mathbf{x}) d\mathbf{x} \int p_2^2(\mathbf{x}) d\mathbf{x}}}. \tag{9}$$

As mentioned, this is a symmetric measure, such that $0 \leq D_{CS} < \infty$, where the minimum is obtained if and only if $p_1(\mathbf{x}) = p_2(\mathbf{x})$.

Let us estimate this quantity by replacing the actual pdfs by their Parzen window estimators. Let $\mathbf{x}_i,\ i = 1, \ldots, N_1$, be data points drawn from the density $p_1(\mathbf{x})$, and let $\mathbf{x}_j,\ j = 1, \ldots, N_2$, be data points drawn from $p_2(\mathbf{x})$. Then, the Parzen window estimators for these distributions are [30]

$$\hat{p}_1(\mathbf{x}) = \frac{1}{N_1} \sum_{i=1}^{N_1} W_{\sigma^2}(\mathbf{x}, \mathbf{x}_i), \quad \hat{p}_2(\mathbf{x}) = \frac{1}{N_2} \sum_{j=1}^{N_2} W_{\sigma^2}(\mathbf{x}, \mathbf{x}_j), \tag{10}$$

It can be shown that according to the convolution theorem for Gaussian functions, the following relation holds

$$\int W_{\sigma^2}(\mathbf{x}, \mathbf{x}_l) W_{\sigma^2}(\mathbf{x}, \mathbf{x}_{l'}) d\mathbf{x} = W_{(\sqrt{2}\sigma)^2}(\mathbf{x}_l, \mathbf{x}_{l'}). \tag{11}$$

For simplicity, we will in the remainder of this paper denote $W_{(\sqrt{2}\sigma)^2}(\mathbf{x}_l, \mathbf{x}_{l'})$ by $k_{ll'}$.

Thus, when we replace the actual densities in the argument of (9) by the Parzen pdf estimators of (10), and utilize (11), we obtain

$$\begin{aligned} \int p_1(\mathbf{x}) p_2(\mathbf{x}) d\mathbf{x} &\approx \int \hat{p}_1(\mathbf{x}) \hat{p}_2(\mathbf{x}) d\mathbf{x} \\ &= \frac{1}{N_1 N_2} \sum_{i,j=1}^{N_1,N_2} \int W_{\sigma^2}(\mathbf{x}, \mathbf{x}_i) W_{\sigma^2}(\mathbf{x}, \mathbf{x}_j) d\mathbf{x} \\ &= \frac{1}{N_1 N_2} \sum_{i,j=1}^{N_1,N_2} k_{ij}, \end{aligned} \tag{12}$$

where the index i is associated with $p_1(\mathbf{x})$ and the index j is associated with $p_2(\mathbf{x})$.

Now we may perform an exactly similar calculation for the two quantities in the denominator of (9), yielding

$$\int p_1^2(\mathbf{x}) d\mathbf{x} \approx \frac{1}{N_1^2} \sum_{i,i'=1}^{N_1,N_1} k_{ii'}, \quad \int p_2^2(\mathbf{x}) d\mathbf{x} \approx \frac{1}{N_2^2} \sum_{j,j'=1}^{N_2,N_2} k_{jj'}. \tag{13}$$

Based on these expressions, the non-parametric sample-based estimator we obtain for the Cauchy-Schwarz pdf divergence is given by

$$\hat{D}_{CS}(p_1, p_2) = -\log \frac{\sum_{i,j=1}^{N_1,N_2} k_{ij}}{\sqrt{\sum_{i,i'=1}^{N_1,N_1} k_{ii'} \sum_{j,j'=1}^{N_2,N_2} k_{jj'}}}. \tag{14}$$

Notice that we can obtain the same expression also using non-Gaussian Parzen windows. This is shown in the Appendix.

4 Relation to Graph Theory

In this section we will introduce the graph cut. The graph cut has been an important cost function used for example in image segmentation [12]. Thereafter, we will show that the CS divergence is actually closely related to the graph cut.

4.1 The Graph Cut

A set of points, $\mathbf{x}_l, l = 1, \ldots, N$, in an arbitrary data space can be represented as a weighted undirected graph $\mathcal{G}$. Each node in the graph corresponds to a data point. The edge formed between a pair of nodes, say l and l', is weighted according to the similarity between the corresponding data points. The edge-weight is denoted $v_{ll'}$.

One way to measure the cost of partitioning the graph $\mathcal{G}$ into two subgraphs $\mathcal{G}_1$ and $\mathcal{G}_2$ is provided by the graph *cut*, defined as

$$Cut(\mathcal{G}_1, \mathcal{G}_2) = \sum_{i,j=1}^{N_1,N_2} v_{ij}, \tag{15}$$

where the index $i = 1, \ldots, N_1$, runs over the N_1 nodes of subgraph $\mathcal{G}_1$ and the index $j = 1, \ldots, N_2$, runs over the N_2 nodes of subgraph $\mathcal{G}_2$. That is, the *cut* measures the weight of the edges of $\mathcal{G}$ which have to be removed in order to create $\mathcal{G}_1$ and $\mathcal{G}_2$.

Any similarity measure can be used in order to define the edge-weights. However, oftentimes the exponential kernel is used, i.e.

$$v_{ll'} = \exp\left\{-\frac{||\mathbf{x}_l - \mathbf{x}_{l'}||^2}{2\sigma^2}\right\}, \tag{16}$$

where σ is a very important scale-parameter which the user must specify.

4.2 The CS Divergence as a Normalized Graph Cut

By comparing Eq. (15) with Eq. (12), it turns out that the CS divergence is related to the graph cut. This can be seen by considering the constants $k_{ll'}$ as edge-weights,

that is, equivalent to the weights given by $v_{ll'}$. Hence, we relate the samples corresponding to $p_1(\mathbf{x})$ with a graph $\mathcal{G}_1$, and the samples corresponding to $p_2(\mathbf{x})$ with a graph $\mathcal{G}_2$. In that case

$$\int \hat{p}_1(\mathbf{x})\hat{p}_2(\mathbf{x})d\mathbf{x} \propto \sum_{i,j=1}^{N_1,N_2} k_{ij} = Cut(\mathcal{G}_1, \mathcal{G}_2). \tag{17}$$

In graph theory, a quantity known as the *volume* of a graph is given by the sum of all the edge-weights in the graph. Hence, we may write

$$Vol(\mathcal{G}_1) = \sum_{i,i'=1}^{N_1,N_1} k_{ii'} \propto \int \hat{p}_1^2(\mathbf{x})d\mathbf{x}. \tag{18}$$

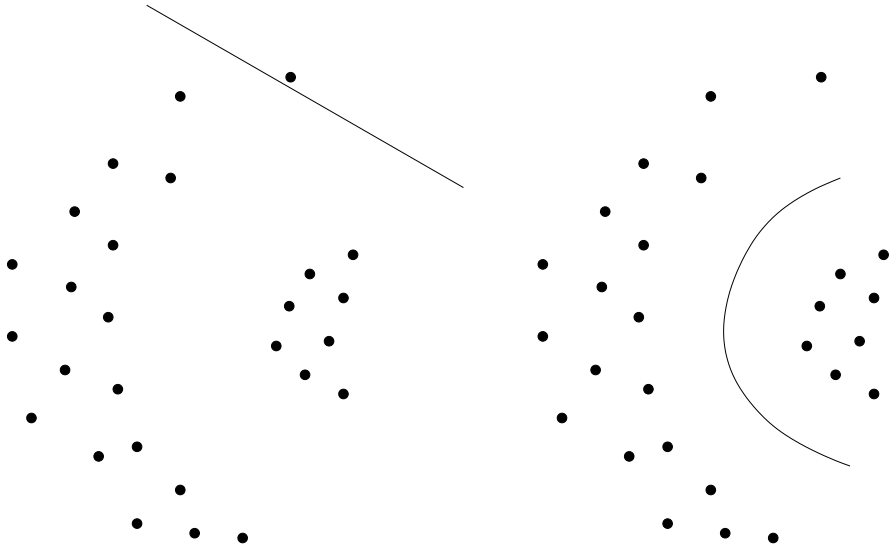
Similarly, we have $Vol(\mathcal{G}_2) = \sum_{j,j'=1}^{N_2,N_2} k_{jj'}$. The following quantity can therefore be defined, which was called the *information cut* (IC) in [33]

$$IC(\mathcal{G}_1, \mathcal{G}_2) = \frac{Cut(\mathcal{G}_1, \mathcal{G}_2)}{\sqrt{Vol(\mathcal{G}_1)Vol(\mathcal{G}_2)}}. \tag{19}$$

Of course, $\hat{D}(p_1, p_2) = -\log IC(\mathcal{G}_1, \mathcal{G}_2)$. The name *information cut* reflects the fact that a well-defined normalized version of the graph cut has been obtained from an information theoretic starting point. This means that when the CS divergence, estimated using Parzen windowing, is optimized for machine learning tasks, we are at the same time optimizing a graph theoretic quantity.

Note that much effort has been made in order to construct modifications to the cut-cost [34–37]. The reason is that the cut-cost alone has the undesirable property that it is minimized when isolating one single node in one of the subgraphs, and all the rest of the nodes in the other subgraph. Up to this point in time, all the proposed modifications to the cut-cost have been motivated by this observation, and several suggestions based on heuristics have been made. However, based on the CS divergence and Parzen windowing, we have obtained a completely new and theoretically well-defined normalization based on the subgraph volumes. This is illustrated in Fig. 4. In (a) it is shown (solid straight line) that based on the cut-cost, the optimum partitioning is obtained by isolating one single node. This is not the case when using the information cut. In this case, the normalization due to the subgraph volumes will prevent the optimum from being reached when one single node is isolated. Rather, the optimum partitioning will be obtained by splitting the graph along the solid curve shown in (b).

Moreover, the connection to Parzen windowing gives us a theoretical criterion for determining appropriate graph edge-weights, namely by using a Parzen window where the window size is determined for example by Eq. (7). This is a very important side-effect of the connection between the CS divergence and Parzen windowing and graph theory.



(a) Graph cut. (b) Information cut.

Fig. 4. Illustrating the different graph partitioning obtained by (a) the cut-cost, (b) the information cut.

5 Relation to Mercer Kernel Theory

In this section, we will explain the idea behind Mercer kernel-based machine learning algorithms. We will then show that the CS divergence can be considered a distance measure in such a Mercer kernel feature space. We will discuss how the Mercer kernel feature space can be approximated using spectral techniques, and show that the distance measure represented by the CS divergence makes sense in such a feature space.

5.1 Mercer Kernel Theory

Mercer kernel-based learning algorithms [13–16] make use of the following idea: via a nonlinear mapping

$$\begin{aligned} \mathbf{\Phi} : R^d &\to \mathcal{F} \\ \mathbf{x} &\to \mathbf{\Phi}(\mathbf{x}) \end{aligned} \tag{20}$$

the data $\mathbf{x}_1, \ldots, \mathbf{x}_N \in R^d$ is mapped into a potentially much higher dimensional feature space $\mathcal{F}$. For a given learning problem one now considers the same learning problem in $\mathcal{F}$ instead of in R^d, working with $\mathbf{\Phi}(\mathbf{x}_1), \ldots, \mathbf{\Phi}(\mathbf{x}_N) \in \mathcal{F}$.

The learning algorithm itself is typically linear in nature, and can be expressed solely in terms of inner-product evaluations. This makes it possible to apply the

algorithm in feature space without actually carrying out the data mapping. The key ingredient is a highly effective trick for computing inner products in the feature space using *kernel functions*. One therefore *implicitly* executes the linear algorithm in kernel feature space. This property is advantageous since execution of the learning algorithm in a very high dimensional space is avoided. Because of the non-linear data mapping, the linear operation in kernel feature space corresponds to a non-linear operation in the input space.

Consider a symmetric kernel function $\rho(\mathbf{x}, \mathbf{y})$. If $\rho : \mathcal{C} \times \mathcal{C} \rightarrow R$ is a continuous kernel of a positive integral operator in a Hilbert space $L_2(\mathcal{C})$ on a compact set $\mathcal{C} \in R^d$, i.e.

$$\forall \psi \in L_2(\mathcal{C}) : \int_{\mathcal{C}} \rho(\mathbf{x}, \mathbf{y}) \psi(\mathbf{x}) \psi(\mathbf{y}) d\mathbf{x} d\mathbf{y} \geq 0. \tag{21}$$

Then there exists a space $\mathcal{F}$ and a mapping $\mathbf{\Phi} : R^d \rightarrow \mathcal{F}$, such that by Mercer's theorem [38]

$$\rho(\mathbf{x}, \mathbf{y}) = \langle \mathbf{\Phi}(\mathbf{x}), \mathbf{\Phi}(\mathbf{y}) \rangle = \sum_{l=1}^{N_{\mathcal{F}}} \lambda_l \phi_l(\mathbf{x}) \phi_l(\mathbf{y}), \tag{22}$$

where $\langle \cdot, \cdot \rangle$ denotes an inner product, the ϕ_l's are the eigenfunctions of the kernel and $N_{\mathcal{F}} \leq \infty$ [39,18]. This operation is known as the "kernel-trick", and it implicitly computes an inner-product in the kernel feature space via $\rho(\mathbf{x}, \mathbf{y})$.

A kernel which satisfies Eq. (21) is known as a Mercer kernel. The most widely used Mercer kernel is the exponential function

$$\rho(\mathbf{x}, \mathbf{y}) = \exp\left\{ -\frac{||\mathbf{x} - \mathbf{y}||^2}{2\sigma^2} \right\}, \tag{23}$$

where σ is a scale parameter which controls the width of the function. Such a kernel corresponds to an infinite-dimensional Mercer kernel feature space, since it has an infinite number of eigenfunctions.

5.2 The CS Divergence in a Mercer Kernel Feature Space

It also turns out that the CS divergence may be considered a distance measure in a Mercer kernel feature space. Assume that we use a Gaussian Parzen window when constructing the coefficients $k_{ll'}$. By comparing Eq. (11) with Eq. (23), it is clear that the $k_{ll'}$'s are also Mercer kernels. Hence, we may express the CS divergence in terms of Mercer kernel feature spaces.

Thus, $k_{ll'} = \langle \mathbf{\Phi}(\mathbf{x}_l), \mathbf{\Phi}(\mathbf{x}_{l'}) \rangle$ where $\mathbf{\Phi}$ is the mapping of the input data to a kernel feature space. Hence, we may rewrite the information cut as

$$\begin{aligned} IC &= \frac{\frac{1}{N_1 N_2} \sum_{i,j=1}^{N_1,N_2} \langle \mathbf{\Phi}(\mathbf{x}_i), \mathbf{\Phi}(\mathbf{x}_j) \rangle}{\sqrt{\frac{1}{N_1^2} \sum_{i,i'=1}^{N_1,N_1} \langle \mathbf{\Phi}(\mathbf{x}_i), \mathbf{\Phi}(\mathbf{x}_{i'}) \rangle \frac{1}{N_2^2} \sum_{j,j'=1}^{N_2,N_2} \langle \mathbf{\Phi}(\mathbf{x}_j), \mathbf{\Phi}(\mathbf{x}_{j'}) \rangle}} \\ &= \frac{\left\langle \frac{1}{N_1} \sum_{i=1}^{N_1} \mathbf{\Phi}(\mathbf{x}_i), \frac{1}{N_2} \sum_{j=1}^{N_2} \mathbf{\Phi}(\mathbf{x}_j) \right\rangle}{\sqrt{\left\langle \frac{1}{N_1} \sum_{i=1}^{N_1} \mathbf{\Phi}(\mathbf{x}_i), \frac{1}{N_1} \sum_{i'=1}^{N_1} \mathbf{\Phi}(\mathbf{x}_{i'}) \right\rangle \left\langle \frac{1}{N_2} \sum_{j=1}^{N_2} \mathbf{\Phi}(\mathbf{x}_j), \frac{1}{N_2} \sum_{j'=1}^{N_2} \mathbf{\Phi}(\mathbf{x}_{j'}) \right\rangle}} \\ &= \frac{\langle \mathbf{m}_1, \mathbf{m}_2 \rangle}{\sqrt{\langle \mathbf{m}_1, \mathbf{m}_1 \rangle \langle \mathbf{m}_2, \mathbf{m}_2 \rangle}} = \cos \angle(\mathbf{m}_1, \mathbf{m}_2), \end{aligned} \tag{24}$$

where $\mathbf{m}_1 = \frac{1}{N_1} \sum_{i=1}^{N_1} \mathbf{\Phi}(\mathbf{x}_i)$ and $\mathbf{m}_2 = \frac{1}{N_2} \sum_{j=1}^{N_2} \mathbf{\Phi}(\mathbf{x}_j)$ can be considered mean vectors of feature space data clusters corresponding to the data points associated with $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$, respectively. This means that $\hat{D}(p_1, p_2) = -\log \cos \angle(\mathbf{m}_1, \mathbf{m}_2)$. Thus, the information theoretic divergence divergence measure between pdfs that we started out with turns out to have a dual expression in a Mercer kernel feature as a measure of *the cosine of the angle between the cluster mean vectors*.

Thus, when optimizing the CS divergence for machine learning tasks, we are at the same time optimizing a quantity in a Mercer kernel feature space. Moreover, a theoretical criterion for determining the Mercer kernel size can be obtained via optimal Parzen windowing. The Parzen window defines the Mercer kernel. Hence, by determining the Parzen window size using for example Eq. (7), *the Mercer kernel size is also determined*. This is a very important side-effect of the connection between the CS divergence and Mercer kernel feature spaces.

5.3 Spectral Approximation

The kernel PCA method [26] was introduced as a technique for projecting data samples represented in a Mercer kernel feature space onto the principal axes in that space. This provides a means for performing the machine learning tasks on the kernel PCA projected data. The dimensionality of the kernel PCA data is normally reduced, since in theory the mapping produces N dimensional data.

The first step in kernel PCA is to collect the inner-product evaluations in a matrix. Since the inner-product between $\mathbf{\Phi}(\mathbf{x}_l)$ and $\mathbf{\Phi}(\mathbf{x}_{l'})$ are calculated using a kernel function, which we denote $k_{ll'}$, the corresponding matrix $\mathbf{K}$ is often referred to as the kernel matrix. Hence, the kernel matrix is defined such that element ll' of $\mathbf{K}$ equals $k_{ll'}$, for $l = 1, \ldots, N$ and $l' = 1, \ldots, N$.

The kernel PCA mapping, depending on the eigenstructure of the correlation matrix in the Mercer kernel feature space, can be expressed in terms of the eigenvalues and eigenvectors of the kernel matrix. The kernel matrix can be expressed as $\mathbf{K} =$

$\mathbf{EDE}^T$, where the columns of $\mathbf{E}$ contains the eigenvectors $\mathbf{e}_i,\ i = 1, \ldots, N$, of $\mathbf{K}$, and the diagonal matrix $\mathbf{D}$ contains the corresponding eigenvalues $\lambda_i,\ i = 1, \ldots, N$, $\lambda_1 \geq, \ldots, \geq \lambda_N$. It can be shown that if the data set consists of C clusters which are "infinitely" far apart, then the kernel PCA mapping is C-dimensional. It is therefore common to reduce the dimension of the kernel PCA data set from N to C by using only the eigenvectors corresponding to the C largest eigenvalues. In that case, the C-dimensional kernel PCA data mapping is given by [26]

$$\mathbf{\Phi}(\mathbf{x}_l) \approx [\sqrt{\tilde{\lambda}_1} e_{1l}, \ldots, \sqrt{\tilde{\lambda}_C} e_{Cl}]^T,\ l = 1, \ldots, N, \tag{25}$$

where e_{il} denotes the l'th element of the i'th eigenvector. This data mapping has also been derived using different approaches [40,41]. Such a data mapping approach has been used for example in clustering [24] and in classification [42]. In that case, these machine learning methods are known as *spectral* methods, since they depend on the spectral properties of the kernel matrix.

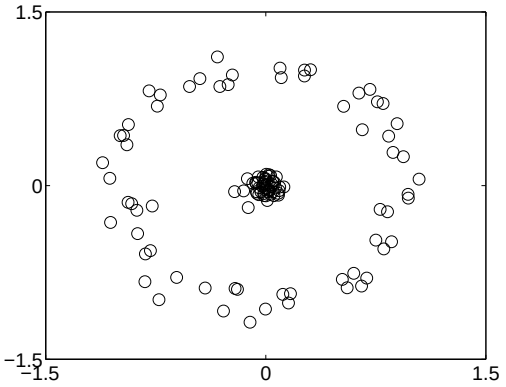
We may use the spectral approximation to the Mercer kernel feature space in order to evaluate the appropriateness of the CS divergence as a distance measure between cluster mean vectors in that space. Figure 5 (a) shows a data set consisting of two clusters. We determine the Parzen window size by Eq. (7), and create the kernel matrix $\mathbf{K}$. Thereafter, we perform a two-dimensional kernel PCA data mapping. The resulting data set is shown in Fig. 5 (b). This data set can be considered an approximation to the Mercer kernel feature space data set. Interestingly, in the spectral domain, the data is distributed along two lines radially from the origin, in two different angular directions. Hence, the mean vectors of the clusters will be in the same direction as the lines, clearly indicating that a distance measure between the clusters that is based on *angles* between mean vectors makes sense. The same effect is observed for the data set shown in Fig. 5 (c). This data set consists of three clusters, so the kernel PCA mapping is three-dimensional. Again, using Eq. (7) to determine the kernel size, we obtain the data set shown in Fig. 5 (d). Also in this case, the data is distributed along lines radially from the origin, again indicating the appropriateness of an angular distance measure in that space.

6 Extension to the Multi-PDF Case

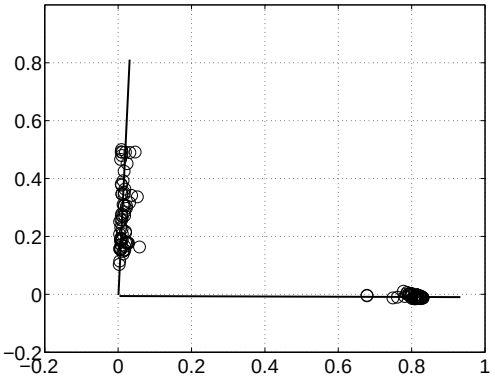
The CS divergence may also be extended, such that it measures the overall "distance" between several pdfs at the same time, as follows

$$D_{CS}(p_1, \ldots, p_C) = -\log \sum_{i=1}^{C-1} \sum_{j>i} \frac{\langle p_i, p_j \rangle}{\kappa \sqrt{\langle p_i, p_i \rangle \langle p_j, p_j \rangle}}, \tag{26}$$

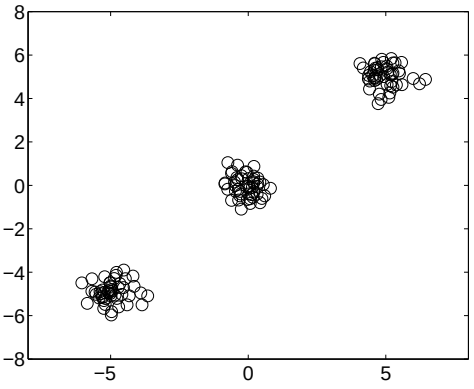
where $\kappa = \sum_{c=1}^{C-1} c$ and $0 \leq D_{CS}(p_1, \ldots, p_C) < \infty$. Note that $D_{CS} = 0$ only for $p_1(\mathbf{x}) = \ldots = p_C(\mathbf{x})$. When replacing the actual pdfs by their Parzen window



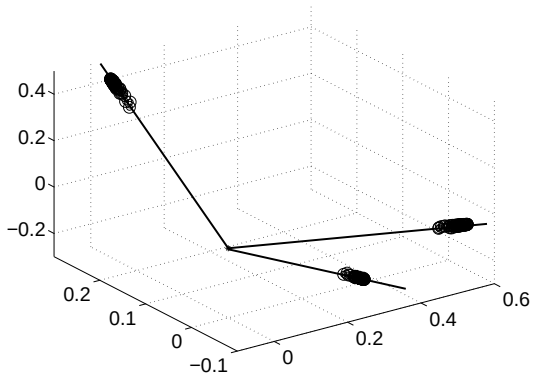
(a) Two-cluster data set.



(b) Kernel PCA mapping.



(c) Three-cluster data set.



(d) Kernel PCA mapping.

Fig. 5. Approximating a Mercer kernel feature space using kernel PCA. The CS divergence, corresponding to the cosine of the angle between cluster mean vectors in the Kernel PCA space, clearly makes sense.

estimators, it can easily be shown that

$$\begin{aligned}\hat{D}_{CS}(p_1,\ldots,p_C) &= -\log \sum_{i=1}^{C-1}\sum_{j>i}\frac{1}{\kappa} IC(\mathcal{G}_i,\mathcal{G}_j) \\ &= -\log \sum_{i=1}^{C-1}\sum_{j>i}\frac{1}{\kappa}\cos\angle(\mathbf{m}_i,\mathbf{m}_j). \end{aligned} \tag{27}$$

Hence, the Parzen window based estimator for the multi-pdf CS divergence basically measures the sum of pairwise information cuts, or equivalently the sum of cosines between cluster mean vectors in the Mercer kernel feature space.

7 Discussion

In this paper, some recent connections between the information theoretic Cauchy-Schwarz pdf divergence measure, graph theory and Mercer kernel - and spectral theory have been presented. These connections are revealed when the CS divergence is estimated using the Parzen window technique for probability density function estimation. Thus, these connections have the important consequence that they enhance our understanding of these seemingly different machine learning schemes relative to each other, since they have been shown to be equivalent in many respects. The equivalence between the CS divergence and Mercer kernel methods depends on the Parzen window satisfying the Mercer conditions. Some of the CS properties were presented in [33,43,44].

A very important side-effect the equivalence between the CS divergence and graph theory and Mercer kernel theory has, concerns the kernel size. A kernel size must be selected both in graph theory and in Mercer kernel theory. To this date, no widely accepted theoretically well-defined criterion for kernel size selection exist, neither in graph theory nor in Mercer kernel theory. However, in Parzen windowing, such a theoretical criterion for kernel size selection does exist. We have shown that the Parzen window may define the kernel function both in graph theory and in Mercer kernel theory. Therefore, in theory at least, it provides a solution for the kernel size selection problem in graph theory and Mercer kernel theory too.

The Cauchy-Schwarz pdf divergence has already been applied in several machine learning problems, especially for data clustering, for example in a hierarchical clustering procedure presented in [45]. The connections to graph theory were utilized in a clustering algorithm presented in [33] and in [46]. Several attempts have also been made to use the CS divergence in connection with Mercer kernel-based and spectral theory. A preliminary information theoretic spectral clustering algorithm was presented in [47]. A new classifier based on the so-called Laplacian matrix and the CS divergence was presented at the ICASSP 2005 conference [42], and was shown to produce very promising results. Quite recently, new algorithms for clustering have also been presented, further fusing together information theoretic ideas, non-parametric density estimation and Mercer kernel-based and spectral theory [48].

In the future, new and more powerful machine learning algorithms may be developed by further fusing toghether information theory, non-parametric density estimation, graph theory and Mercer kernel - and spectral theory.

Appendix

Eq. (9) can be rewritten as

$$D_{CS} = -\log \frac{E_{p_1}\{p_2(\mathbf{x})\}}{\sqrt{E_{p_1}\{p_1(\mathbf{x})\}E_{p_2}\{p_2(\mathbf{x})\}}}, \tag{28}$$

where $E_p\{\cdot\}$ denotes the expectation operator with respect to the density p. Using the sample mean to estimate the expectations, we obtain

$$\begin{aligned} E_{p_1}\{p_2(\mathbf{x})\} &\approx \frac{1}{N_1}\sum_{i=1}^{N_1} p_2(\mathbf{x}_i) \\ &= \frac{1}{N_1}\sum_{i=1}^{N_1}\frac{1}{N_2}\sum_{j=1}^{N_2} W(\mathbf{x}_i,\mathbf{x}_j) \\ &= \frac{1}{N_1 N_2}\sum_{i,j=1}^{N_1,N_2} W(\mathbf{x}_i,\mathbf{x}_j), \end{aligned} \tag{29}$$

where W is some (non-Gaussian) Parzen window. In a similar manner, $E_{p_1}\{p_1(\mathbf{x})\} \approx \frac{1}{N_1 N_1}\sum_{i,i'=1}^{N_1,N_1} W(\mathbf{x}_i,\mathbf{x}_{i'})$ and $E_{p_2}\{p_2(\mathbf{x})\} \approx \frac{1}{N_2 N_2}\sum_{j,j'=1}^{N_2,N_2} W(\mathbf{x}_j,\mathbf{x}_{j'})$, such that we obtain

$$IC(\mathcal{G}_1,\mathcal{G}_2) = \frac{\sum_{i,j=1}^{N_1,N_2} k_{ij}}{\sqrt{\sum_{i,i'=1}^{N_1,N_1} k_{ii'}\sum_{j,j'=1}^{N_2,N_2} k_{jj'}}}, \tag{30}$$

where we have defined $W(\mathbf{x}_l,\mathbf{x}_{l'}) = k_{ll'}$.

References

[1] J. Principe, D. Xu, and J. Fisher. Information Theoretic Learning. In *Unsupervised Adaptive Filtering*, volume I, S. Haykin (Ed.), John Wiley & Sons, New York, 2000. Chapter 7.

[2] D. Erdogmus, R. Agrawal, and J. C. Principe. A Mutual Information Extension to the Matched Filter. *Signal Processing, to appear*, 2005.

[3] M. Lazaro, I. Santamaria, D. Erdogmus, K. E. Hild II, C. Pantaleon, and J. C. Principe. Stochastic Blind Equalization Based on PDF Fitting using Parzen Estimator. *IEEE Transactions on Signal Processing*, 53(2):696–704, 2005.

[4] D. Erdogmus, K. E. Hild, Y. N. Rao, and J. C. Principe. Minimax Mutual Information Approach for Independent Component Analysis. *Neural Computation*, 16:1235–1252, 2004.

[5] D. Erdogmus and J. C. Principe. Convergence Properties and Data Efficiency of the Minimum Error-Entropy Criterion in Adaline Training. *IEEE Transactions on Signal Processing*, 51(7):1966–1978, 2003.

[6] D. Erdogmus, K. E. Hild, and J. C. Principe. Blind Source Separation using Renyi's α-Marginal Entropies. *Neurocomputing*, 49:25–38, 2002.

[7] I. Santamaria, D. Erdogmus, and J. C. Principe. Entropy Minimization for Supervised Digital Communications Channel Equalization. *IEEE Transactions on Signal Processing*, 50(5):1184–1192, 2002.

[8] D. Erdogmus and J. C. Principe. Generalized Information Potential Criterion for Adaptive System Training. *IEEE Transactions on Neural Networks*, 13(5):1035–1044, 2002.

[9] D. Erdogmus and J. C. Principe. An Error-Entropy Minimization Algorithm for Supervised Training of Nonlinear Adaptive Systems. *IEEE Transactions on Signal Processing*, 50(7):1780–1786, 2002.

[10] J. C. Principe, D. Xu, Q. Zhao, and J. W. Fisher. Learning From Examples with Information Theoretic Criteria. *Journal of VLSI Signal Processing*, 26(1):61–77, 2000.

[11] F. R. K. Chung. *Spectral Graph Theory*. American Mathematial Society, 1997.

[12] Z. Wu and R. Leahy. An Optimal Graph Theoretic Approach to Data Clustering: Theory and Its Applications to Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1101–1113, 1993.

[13] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[14] K. R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf. An Introduction to Kernel-Based Learning Algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–201, 2001.

[15] F. Perez-Cruz and O. Bousquet. Kernel Methods and Their Potential Use in Signal Processing. *IEEE Signal Processing Magazine*, pages 57–65, May 2004.

[16] B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, Cambridge, 2002.

[17] C. Cortez and V. N. Vapnik. Support Vector Networks. *Machine Learning*, 20:273–297, 1995.

[18] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1995.

[19] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines*. Cambridge University Press, Cambridge, 2000.

[20] C. J. C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Knowledge Discovery and Data Mining*, 2(2):121–167, 1998.

[21] S. Roweis and L. Saul. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290:2323–2326, 2000.

[22] J. Tenenbaum, V. de Silva, and J. C. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290:2319–2323, 2000.

[23] M. Belkin and P. Niyogi. Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computation*, 15:1373–1396, 2003.

[24] A. Y. Ng, M. Jordan, and Y. Weiss. On Spectral Clustering: Analysis and an Algorithm. In *Advances in Neural Information Processing Systems, 14*, pages 849–856, MIT Press, Cambridge, 2002.

[25] Y. Weiss. Segmentation Using Eigenvectors: A Unifying View. In *Proceedings of IEEE International Conference on Computer Vision*, pages 975–982, Corfu, Greece, September 20-25, 1999.

[26] B. Schölkopf, A. J. Smola, and K. R. Müller. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*, 10:1299–1319, 1998.

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

[28] D. W. Scott. *Multivariate Density Estimation*. John Wiley & Sons, New York, 1992.

[29] M. P. Wand and M. C. Jones. *Kernel Smooting*. Chapman and Hall, London, 1995.

[30] E. Parzen. On the Estimation of a Probability Density Function and the Mode. *The Annals of Mathematical Statistics*, 32:1065–1076, 1962.

[31] S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[32] H. Chernoff. A Measure of Asymptotic Efficiency for Tests of a Hypothesis Based on a Sum of Observations. *The Annals of Mathematical Statistics*, 23:493–507, 1952.

[33] R. Jenssen, J. C. Principe, and T. Eltoft. Information Cut and Information Forces for Clustering. In *Proceedings of IEEE International Workshop on Neural Networks for Signal Processing*, pages 459–468, Toulouse, France, September 17-19, 2003.

[34] J. Shi and J. Malik. Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[35] C. H. Q. Ding, X. He, H. Zha, M. Gu, and H. D. Simon. A Min-max Cut Algorithm for Graph Partitioning and Data Clustering. In *Proceedings of IEEE International Conference on Data Mining*, pages 107–114, San Jose, USA, November 29 - December 2, 2001.

[36] Y. Gdalyahu, D. Weinshall, and M. Werman. Self-Organization in Vision: Stochastic Clustering for Image Segmentation, Perceptual Grouping, and Image Database Organization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1053–1074, 2001.

[37] J. Scanlon and N. Deo. Graph-Theoretic Algorithms for Image Segmentation. In *IEEE International Symposium on Circuits and Systems*, pages VI141–144, Orlando, Florida, 1999.

[38] J. Mercer. Functions of Positive and Negative Type and their Connection with the Theory of Integral Equations. *Philos. Trans. Roy. Soc. London*, A:415–446, 1909.

[39] K. R. Müller, A. J. Smola, G. Rätsch, B. Schölkopf, J. Kohlmorgen, and V. N. Vapnik. Predicting Time Series with Support Vector Machines. In *Proceedings of International Conference on Artificial Neural Networks - Lecture Notes in Computer Science, Springer-Verlag*, volume 1327, pages 999–1004, Berlin, 1997.

[40] C. Williams and M. Seeger. Using the Nyström Method to Speed Up Kernel Machines. In *Advances in Neural Information Processing Systems, 13*, pages 682–688, MIT Press, Cambridge, 2001.

[41] Y. Bengio, P. Vincent, and J.-F. Paiement. Spectral Clustering and Kernel PCA are Learning Eigenfunctions. Technical report, Département d'informatique et recherche opérationelle, université de Montréal, Montréal, Canada, 2003.

[42] R. Jenssen, D. Erdogmus, J. C. Principe, and T. Eltoft. The Laplacian Spectral Classifier. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 325–328, Philadelphia, USA, March 16 - 23, 2005.

[43] R. Jenssen, D. Erdogmus, J. C. Principe, and T. Eltoft. Towards a Unification of Information Theoretic Learning and Kernel Methods. In *Proceedings of IEEE Workshop on Machine Learning for Signal Processing*, pages 93–102, Sao Luis, Brazil, September 29 - October 1, 2004.

[44] R. Jenssen, D. Erdogmus, J. C. Principe, and T. Eltoft. The Laplacian PDF Distance: A Cost Function for Clustering in a Kernel Feature Space. In *Advances in Neural Information Processing Systems 17*, pages 625–632, MIT Press, Cambridge, 2005.

[45] R. Jenssen, J. C. Principe, and T. Eltoft. Cauchy-Schwartz pdf Divergence Measure for non-Parametric Clustering. In *Proceedings of IEEE Norway Section Signal Processing Symposium (cd-rom)*, Bergen, Norway, October 2-4, 2003.

[46] R. Jenssen, D. Erdogmus, K. E. Hild, J. C. Principe, and T. Eltoft. Optimizing the cauchy-schwarz pdf distance for information theoretic, non-parametric clustering. In *Proceedings of International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, page (to appear), St. Augustine, USA, November 9-11, 2005.

[47] R. Jenssen, T. Eltoft, and J. C. Principe. Information Theoretic Spectral Clustering. In *Proceedings of International Joint Conference on Neural Networks*, pages 111–116, Budapest, Hungary, July 25-29, 2004.

[48] J. C. Principe R. Jenssen, D. Erdogmus and T. Eltoft. Spectral Clustering based on the Cauchy-Schwarz Divergence and Parzen Windowing. In *International Conference on Acoustics, Speech and Signal Processing (submitted)*, May 15-19, Toulouse, France, 2006.