Optimization in Reproducing kernel Hilbert Spaces of Spike Trains*

António R. C. Paiva[†], Il Park[‡], and José C. Príncipe[§]

last revised: February 3, 2010

Abstract

This paper presents a framework based on reproducing kernel Hilbert spaces (RKHS) for optimization with spike trains. To establish the RKHS for optimization we start by introducing kernels for spike trains. It is shown that spike train kernels can be built from ideas of kernel methods, or from the intensity functions underlying the spike trains. However, the later approach shall be the main focus of this study. We introduce the memoryless cross-intensity (mCI) kernel as an example of an inner product of spike trains, which defines the RKHS bottom-up as an inner product of intensity functions. Being defined in terms of the intensity functions, this approach towards defining spike train kernels has the advantage that points in the RKHS incorporate a statistical description of the spike trains, and the statistical model is explicitly stated. Some properties of the mCI kernel and the RKHS it induces will be given to show that this RKHS has the necessary structure for optimization. The issue of estimation from data is also addressed. We finalize with an example of optimization in the RKHS by deriving an algorithm for principal component analysis (PCA) of spike trains.

*Chapter 1 in W. Chaovalitwongse et al. (eds.), *Computational Neuroscience*, Springer Optimization and Its Applications, vol. 38, ISBN: 978-0-387-88629-9, Springer 2010

[†]António R. C. Paiva, Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611, USA, arpaiva@cnel.ufl.edu

[‡]Il Park, Pruitt Family Department of Biomedical Engineering, University of Florida, Gainesville, FL 32611, USA, memming@cnel.ufl.edu

[§]José C. Príncipe, Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611, USA, principe@cnel.ufl.edu

1 Introduction

A spike train $s \in \mathcal{S}(\mathcal{T})$ is a sequence of ordered spike times $s = \{t_m \in \mathcal{T} : m = 1, \ldots, N\}$ corresponding to the time instants in the interval $\mathcal{T} = [0, T]$ at which a neuron fires. In a different perspective, spike trains are realizations of stochastic point processes. Spike trains can be observed whenever studying either real or artificial neurons. In neurophysiological studies, spike trains result from the activity of multiple neurons in single-unit recordings by ignoring the stereotypical shape of action potentials [1]. And, more recently, there has also been a great interest in using spike trains for biologically inspired computation paradigms such as the liquid-state machine (LSM) [2, 3] or spiking neural networks (SNN) [4, 3]. Regardless of the nature of the process giving rise to the spike trains, the ultimate goal is to filter or classify the spike trains to manipulate or extract the encoded information.

Filtering, eigendecomposition, clustering and classification are often formulated in terms of a criterion to be optimized. However, formulation of a criterion and/or optimization directly with spike trains is not a straightforward task. The most widely used approach is to bin the spike trains, obtained by segmenting the spike train in small intervals and counting the number of spikes within each interval [1]. The advantage of this approach is that the randomness in time is mapped to randomness in amplitude of a discrete-time random process, and therefore our usual statistical signal processing and machine learning techniques can be applied. It is known that if the bin size is large compared to the average inter-spike interval this transformation provides a rough estimate of the instantaneous rate. However, the discretization of time introduced by binning leads to low resolution.

The caveats associated with binned spike trains have motivated alternative methodologies involving the spike times directly. For example, to deal with the problem of classification, Victor and Purpura [5, 6] defined a distance metric between spike trains resembling the edit distance in computer science. An alternative distance measure was proposed by van Rossum [7]. Using spike train distances for classification simplifies the problem to that of finding a threshold value. However, for more general problems the range of applications that can be solved directly using distances is limited since these metrics do not lend themselves to optimization. The reason is that although distances are useful concepts in classification and pattern analysis they do not provide a general framework for statistical signal processing and machine learning. Recent attempts were also made to develop a mathematical theory from simple principles [8, 9], such as the definition of an inner product and an associated kernel, but these developments are mainly associated with the earlier proposed distance measures [6, 7].

The framework described in this paper is different in the sense that it does not attempt to propose a distance or criterion directly. Rather, we propose to define first inner product kernel functions[1] for spike trains. These kernels induce reproducing kernel Hilbert spaces (RKHS) of functions on spike trains, which provide the needed mathematical structure to easily define and optimize criteria for a diverse range of problems. Another advantage of

[1]Throughout this document we will refer to inner products and kernels indistinguishably since they represent the same concept. However, stated more correctly, kernels denote inner products in a reproducing kernel Hilbert space of functions on the arguments of the kernel.

this approach is that many of the difficulties found in manipulating spike trains which lead to the use of binning are implicitly taken care of through the mapping to the RKHS. In this paper we exemplify the construction of an RKHS by defining an inner product of spike trains called *memoryless cross-intensity (mCI) kernel*. This spike train kernel defines the RKHS bottom-up as an inner product of intensity functions and thus incorporates a statistical description of the spike trains. As will be showed later, this particular kernel is related to the *generalized cross-correlation* (GCC) [10] but provides a more principled and broader perspective on many spike train methods reported in the literature.

For continuous and discrete random processes, RKHS theory has already been proven essential in a number of applications, such as statistical signal processing [11, 12] and detection [13, 14, 15], as well as statistical learning theory [16, 17, 18]. Indeed, Parzen showed that several statistical signal processing algorithms can be stated as optimization problems in the RKHS and easily solved [11, 12]. For instance, the cross-correlation function used throughout statistical analysis and signal processing, including the celebrated Wiener filter [19], is a valid kernel and induces an RKHS space [11]. Although frequently overlooked, RKHS theory plays a pivotal role in kernel methods [16, 17] because it is the reason for the famed kernel trick which allows for the otherwise seemingly intractable task of deriving and applying kernel techniques.

In the following, we introduce how to define spike train kernels and present some examples. A systematic approach which builds the RKHS from the ground up is followed by defining inner products for spike trains. The main advantage in this path is a general and mathematically precise methodology which, nevertheless, can easily be interpreted intuitively by analyzing the definition of the inner product or, conversely, defining the inner product to match our understanding of a given problem. In this study we present the mCI kernel as an example, since it incorporates a statistical description of the spike trains and the statistical model is clearly stated, but the ideas can be easily extended. A number of properties are proved for the mCI kernel, and the relationships between the RKHS and congruent spaces are discussed for additional insight. The issue of estimation from data is also addressed. Finally, the usefulness of an RKHS framework for optimization is demonstrated through the derivation of an algorithm for principal component analysis (PCA) of spike trains.

2 Some background on RKHS theory

In this section, some basic concepts of kernel methods and RKHS theorem necessary for the understanding of the next sections are reviewed. The notation was purposely chosen to be different from the one used later since the presentation here is meant to be as general and introductory as possible.

The fundamental result in RKHS theory is the famed *Moore-Aronszajn theorem* [20, 21]. Let K denote a generic symmetric and positive definite function of two variables defined on some space E. That is, a function $K(\cdot,\cdot) : E \times E \to \mathbb{R}$ such that it verifies:

(i) Symmetry: $K(x,y) = K(y,x), \quad \forall x, y \in E.$

(ii) Positive definiteness: for any finite number of l ($l \in \mathbb{N}$) points $x_1, x_2, \ldots, x_l \in E$ and any corresponding coefficients $c_1, c_2, \ldots, c_l \in \mathbb{R}$,

$$\sum_{m=1}^{l} \sum_{n=1}^{l} c_m c_n K(x_m, x_n) \geq 0. \tag{1}$$

These are sometimes called the Mercer conditions [22] in the kernel methods literature. Then, the Moore-Aronszajn theorem [20, 21] guaranties that there exists a unique Hilbert space $\mathcal{H}$ of real valued functions defined on E such that, for every $x \in E$,

(i) $K(x, \cdot) \in \mathcal{H}$, and

(ii) for any $f \in \mathcal{H}$

$$f(x) = \langle f(\cdot), K(x, \cdot) \rangle_{\mathcal{H}} . \tag{2}$$

The identity on equation (2) is called the *reproducing property* of K and, for this reason, $\mathcal{H}$ is said to be an RKHS with reproducing kernel K.

Two essential corollaries of the theorem just described can be observed. First, since both $K(x, \cdot)$ and $K(y, \cdot)$ are in $\mathcal{H}$, we get from the reproducing property that

$$K(x, y) = \langle K(x, \cdot), K(y, \cdot) \rangle_{\mathcal{H}} . \tag{3}$$

Hence, K evaluates the inner product in this RKHS. This identity is the *kernel trick*, well known in kernel methods, and is the main tool for computation in this space. Second, a consequence of the previous properties and which can be seen easily in the kernel trick is that, given any point $x \in E$, the representer of evaluation in the RKHS is $\Psi_x(\cdot) = K(x, \cdot)$. Notice that the *functional transformation* Ψ from the input space E into the RKHS $\mathcal{H}$ evaluated for a given x, and in general any element of the RKHS, is a real function defined on E.

A quite interesting perspective to RKHS theory is provided by Parzen's work [23]. In his work, Parzen proved that for *any* symmetric and positive definite function there exists a space of Gaussian distributed random variables defined in the input space of the kernel for which this function is the covariance function [11]. Notice that, assuming stationarity and ergodicity, this space might just as well be thought of as a space of random processes. That is to say that any kernel inducing an RKHS denotes simultaneously an inner product in the RKHS and a covariance operator in another space. Furthermore, it is established that there exists an isometric inner product-preserving mapping, a *congruence*, between these two spaces. Consequently, the RKHS $\mathcal{H}$ induced by the kernel and the space of random variables where this kernel is a covariance function are said to be *congruent*. This is an important result as it sets up a correspondence between the inner product due to a kernel in the RKHS to our intuitive understanding of the covariance function and associated linear statistics. In other words, due to the congruence between the two spaces an algorithm can be derived and interpreted in any of the spaces.

3 Inner product for spike times

Denote the mth spike time in a spike train indexed by i as $t_m^i \in \mathcal{T}$, with $m \in \{1, 2, \ldots, N_i\}$ and N_i the number of spike times in the spike train. To simplify the notation, however, the spike train index will be omitted if is irrelevant for the presentation or obvious from the context.

The simplest inner product that can be defined for spike trains operates with only two spike times at a time, as observed by Carnell and Richardson [8]. In the general case, such an inner product can be defined in terms of a kernel function defined on $\mathcal{T} \times \mathcal{T}$ into the reals, with $\mathcal{T}$ the interval of spike times. Let κ denote such a kernel. Conceptually, this kernel operates in the same way as the kernels operating on data samples in kernel methods [16] and information theoretic learning [24]. Although it operates only with two spike times, it will play a major role whenever we operate with complete realizations of spike trains. Indeed, as the next sections show, the estimators for one of the kernels we define on spike trains rely on this kernel as an elemental operation for computation.

To take advantage of the framework for statistical signal processing provided by RKHS theory, κ is required to be a symmetric positive definite function. By the Moore-Aronszajn theorem [20], this ensures that an RKHS $\mathcal{H}_\kappa$ must exist for which κ is a reproducing kernel. The inner product in $\mathcal{H}_\kappa$ is given as

$$\kappa(t_m, t_n) = \langle \kappa(t_m, \cdot), \kappa(t_n, \cdot) \rangle_{\mathcal{H}_\kappa} = \langle \Phi_m, \Phi_n \rangle_{\mathcal{H}_\kappa} . \tag{4}$$

where Φ_m is the element in $\mathcal{H}_\kappa$ corresponding to t_m (that is, the transformed spike time).

Since the kernel operates directly on spike times and is, typically, undesirable to emphasize events in this space, κ is further required to be *shift-invariant*; that is, for any $\theta \in \mathbb{R}$,

$$\kappa(t_m, t_n) = \kappa(t_m + \theta, t_n + \theta), \quad \forall t_m, t_n \in \mathcal{T}. \tag{5}$$

In other words, the kernel is only sensitive to the difference of the arguments and, consequently, we may also write $\kappa(t_m, t_n) = \kappa(t_m - t_n)$.

For any symmetric, shift-invariant, and positive definite kernel, it is known that $\kappa(0) \geq |\kappa(\theta)|$.[2] This is important in establishing κ as a similarity measure between spike times. In other words, as usual, an inner product should intuitively measure some form of interdependence between spike times. However, notice that the conditions posed do not restrict this study to a single kernel. Quite on the contrary, any kernel satisfying the above requirements is theoretically valid and understood under the framework proposed here although, obviously, the practical results may vary.

An example of a family of kernels that can be used (but not limited to) are the radial basis functions [25],

$$\kappa(t_m, t_n) = \exp(-|t_m - t_n|^p), \quad t_m, t_n \in \mathcal{T}, \tag{6}$$

for any $0 < p \leq 2$. Some well known kernels, such as the widely used Gaussian and Laplacian kernels, are special cases of this family for $p = 2$ and $p = 1$, respectively.

[2] This is a direct consequence of the fact that symmetric positive definite kernels denote inner products that obey the Cauchy-Schwarz inequality.

It is interesting to notice that shift-invariant kernels result in a natural norm induced by the inner product with the following property,

$$\|\Phi_m\| = \sqrt{\kappa(0)}, \quad \forall \Phi_m \in \mathcal{H}_\kappa. \tag{7}$$

Since the norm of the transformed spike times in $\mathcal{H}_\kappa$ is constant, all the spike times are mapped to the surface of an hypersphere in $\mathcal{H}_\kappa$. The set of transformed spike times is called the manifold of $\mathcal{S}(\mathcal{T})$. Moreover, this shows in a different perspective why the kernel used needs to be non-negative. Furthermore, the *geodesic distance* corresponding to the length of the smallest path contained within this manifold (in this case, the hypersphere) between two functions in this manifold, Φ_m and Φ_n, is given by

$$\begin{aligned} d(\Phi_m, \Phi_n) &= \|\Phi_m\| \arccos\left(\frac{\langle \Phi_m, \Phi_n \rangle}{\|\Phi_m\| \, \|\Phi_n\|}\right) \\ &= \sqrt{\kappa(0)} \arccos\left[\frac{\kappa(t_m, t_n)}{\kappa(0)}\right]. \end{aligned} \tag{8}$$

Put differently, from the geometry of the transformed spike times, the kernel function is proportional to the cosine of the angle between two transformed spike times in $\mathcal{H}_\kappa$. Because the kernel is non-negative, the maximum angle is $\pi/2$, which restricts the manifold of transformed spike times to a small area of the hypersphere. With the kernel inducing the above metric, the manifold of the transformed points forms a *Riemannian space*. However, this space is *not* a linear space. Fortunately, its span is obviously a linear space. In fact, it equals the RKHS associated with the kernel. Although this is not a major problem, computing with the transformed points will almost surely yield points outside of the manifold of transformed spike times. This means that such points cannot be mapped back to the input space directly. Depending on the aim of the application this may not be necessary, but if required, it may be solvable through a projection to the manifold of transformed input points.

4 Inner product for spike trains

Although any kernel verifying the conditions discussed in the previous section induces an RKHS and therefore is of interest on itself, the fact that it only operates with two spike times at a time limits its practical use. In particular, spike trains are *sets* of spike times but we have not yet addressed the problem of how to combine the kernel for all spike times. One immediate approach is to utilize the linearity of the RKHS [8]. If the mth spike time is represented in the RKHS by Φ_m, then the spike train can be represented in the RKHS as the sum of the transformed spike times,

$$\Psi = \sum_{m=1}^{N} \Phi_m. \tag{9}$$

Notice that if a spike time is represented by a given function, say, an impulse, the spike train will be a sum of time-shifted impulses centered at the spike times. Then equation (9)

implies that the mapping of the spike train into the RKHS induced by the spike time kernel is linear. Using the linearity of the RKHS it results that the inner product of spike trains is

$$\left\langle \Psi_{s_i}, \Psi_{s_j} \right\rangle_{\mathcal{H}_\kappa} = \sum_{m=1}^{N_i} \sum_{n=1}^{N_j} \left\langle \Phi_m^i, \Phi_n^j \right\rangle_{\mathcal{H}_\kappa} = \sum_{m=1}^{N_i} \sum_{n=1}^{N_j} \kappa(t_m^i, t_n^j). \tag{10}$$

It must be remarked that equation (10) is only one example of a spike train kernel from inner products on spike times. Indeed, as is commonly done in kernel methods, more complex spike train kernels can be defined utilizing the kernel on spike times as a building block equating the nonlinear relationship between the spike times. On the other hand, the main disadvantage in this approach towards spike train analysis is that the underlying model assumed for the spike train is not clearly stated. This is important in determining and understanding the potential limitations of a given spike train kernel for data analysis.

Rather than utilizing this direct approach, an alternative construction is to define first a general inner product for the spike trains from the fundamental statistical descriptors. In fact, it will be seen that the inner product for spike trains builds upon the kernel on single spike times. This bottom-up construction of the kernel for spike trains is unlike the previous approach and is rarely taken in machine learning, but it exposes additional insight on the properties of the kernel and the RKHS it induces for optimization and data analysis.

A spike train is a realization of an underlying stochastic point process [26]. In general, to completely characterize a point process, the conditional intensity function must be used. The Poisson process is a special case because it is memoryless and therefore the intensity function (or rate function) is sufficient [26, chapter 2]. Spike trains in particular have been found to be reasonably well modeled as realizations of Poisson processes [27, chapter 2]. Hence, for the remaining of this study only Poisson spike trains are considered.

Consider two spike trains, $s_i, s_j \in \mathcal{S}(\mathcal{T})$, with $i, j \in \mathbb{N}$. Denote the intensity of the underlying Poisson processes by $\lambda_{s_i}(t)$ and $\lambda_{s_j}(t)$, respectively, where $t \in \mathcal{T} = [0, T]$ denotes the time coordinate. Note that the dependence of the intensity function on t indicates that the Poisson processes considered may be inhomogeneous (i.e., non-stationary). For any practical spike train and for finite T, we have that

$$\int_{\mathcal{T}} \lambda_{s_i}^2(t) dt < \infty. \tag{11}$$

As a consequence, the intensity functions of spike trains are valid elements of $L_2(\mathcal{T}) \subset L_2$. Moreover, in this space, we can define an inner product of intensity functions as the usual inner product in L_2,

$$I(s_i, s_j) = \left\langle \lambda_{s_i}, \lambda_{s_j} \right\rangle_{L_2(\mathcal{T})} = \int_{\mathcal{T}} \lambda_{s_i}(t) \lambda_{s_j}(t) dt. \tag{12}$$

We shall refer to $I(\cdot, \cdot)$ as the memoryless cross-intensity (mCI) kernel. Notice that the mCI kernel incorporates the statistics of the processes directly, and treats seamlessly even the case of inhomogeneous Poisson processes.

Furthermore, the definition of inner product naturally induces a norm in the space of the intensity functions,

$$\|\lambda_{s_i}(\cdot)\|_{L_2(\mathcal{T})} = \sqrt{\langle \lambda_{s_i}, \lambda_{s_i} \rangle_{L_2(\mathcal{T})}} = \sqrt{\int_{\mathcal{T}} \lambda_{s_i}^2(t) dt} \tag{13}$$

which is very useful for the formulation of optimization problems.

It is insightful to compare the mCI kernel definition in equation (12) with the so-called *generalized cross-correlation* (GCC) [10],

$$\begin{aligned} C_{AB}(\theta) &= E\left\{\lambda_A(t)\lambda_B(t+\theta)\right\} \\ &= \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} \lambda_A(t)\lambda_B(t+\theta) dt. \end{aligned} \tag{14}$$

Although the GCC was proposed directly as a more general form of cross-correlation of spike trains, one verifies that the two ideas are fundamentally equivalent. Nevertheless, the path towards the definition of mCI is more principled. More importantly, this path suggests alternative spike train kernel definitions which may not require a Poisson assumption, or, if the Poisson model is assumed, extract more information in the event of deviations from the model.

5 Properties and estimation of the memoryless cross-intensity kernel

5.1 Properties

In this section some relevant properties of the mCI kernel are presented. In addition to the knowledge they provide, they are necessary for a clear understanding of the following sections.

Property 1 *The mCI kernel is a symmetric, non-negative and linear operator in the space of the intensity functions.*

Because the mCI kernel operates on elements of $L_2(\mathcal{T})$ and corresponds to the usual dot product from L_2, this property is a direct consequence of the properties inherited from L_2. More specifically, property 1 guaranties the mCI kernel is a valid inner product.

Property 2 *For any set of $n \geq 1$ spike trains, the mCI kernel matrix*

$$\boldsymbol{V} = \begin{bmatrix} I(s_1,s_1) & I(s_1,s_2) & \dots & I(s_1,s_n) \\ I(s_2,s_1) & I(s_2,s_2) & \dots & I(s_2,s_n) \\ \vdots & \vdots & \ddots & \vdots \\ I(s_n,s_1) & I(s_n,s_2) & \dots & I(s_n,s_n) \end{bmatrix},$$

is symmetric and non-negative definite.

The proof is given in the appendix. Through the work of Moore [21] and due to the Moore-Aronszajn theorem [20], the following two properties result as corollaries of property 2.

Property 3 *The mCI kernel is a symmetric and positive definite kernel. Thus, by definition, for any set of $n \geq 1$ point processes and corresponding n scalars $a_1, a_2, \ldots, a_n \in \mathbb{R}$,*

$$\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j I(s_i, s_j) \geq 0. \tag{15}$$

Property 4 *There exists an Hilbert space for which the mCI kernel is a reproducing kernel.*

Actually, property 3 can be obtained explicitly by verifying that the inequality of equation (15) is implied by equations (44) and (45) in the proof of property 2 (see the appendix).

Properties 2 through 4 are equivalent in the sense that any of these properties implies the other two. The most important consequence of these properties, explicitly stated through property 4, is that the *mCI kernel induces an unique RKHS*, henceforth denoted by $\mathcal{H}_I$.

Property 5 *The mCI kernel verifies the Cauchy-Schwarz inequality,*

$$I^2(s_i, s_j) \leq I(s_i, s_i) I(s_j, s_j) \qquad \forall s_i, s_j \in \mathcal{S}(\mathcal{T}). \tag{16}$$

The proof is given in the appendix. The Cauchy-Schwarz inequality is important since the triangle inequality results as an immediate consequence and it induces a correlation coefficient-like measure very useful for matching spike trains. Indeed, the Cauchy-Schwarz inequality is the concept behind the spike train measure proposed by Schreiber et al. [28]. However, our proof in appendix verifies that all it is required is a spike train kernel inducing an RKHS, and therefore the idea by Schreiber and colleagues is easily extendible.

Property 6 *For any two point processes $s_i, s_j \in \mathcal{S}(\mathcal{T})$ the triangle inequality holds. That is,*

$$\left\|\lambda_{s_i} + \lambda_{s_j}\right\| \leq \left\|\lambda_{s_i}\right\| + \left\|\lambda_{s_j}\right\|.$$

As before, the proof is given in the appendix.

5.2 Estimation

As previous stated, spike trains are realizations of underlying point processes, but the memoryless cross-intensity kernel as presented so far is a deterministic operator on the point processes rather than on the observed spike trains. Using a well known methodology for the estimation of the intensity function we now derive an estimator for the memoryless cross-intensity kernel. One of the advantages of this route is that the conceptual construction of spike train kernel is dissociated from the problem of estimation from data. Put differently, in this way it is possible to have a clear statistical interpretation while later approaching the problem from a practical point of view. The connection between the mCI kernel and κ will now become obvious.

A well known method for intensity estimation from *a single spike train* is kernel smoothing [1, 29]. Accordingly, given a spike train s_i comprising of spike times $\{t_m^i \in \mathcal{T} : m = 1, \ldots, N_i\}$ the estimated intensity function is

$$\hat{\lambda}_{s_i}(t) = \sum_{m=1}^{N_i} h(t - t_m^i), \tag{17}$$

where h is the smoothing function. This function must be non-negative and integrate to one over the real line (just like a probability distribution function (pdf)). Commonly used smoothing functions are the Gaussian, Laplacian and α-functions, among others.

From a filtering perspective, equation (17) can be seen as a linear convolution between the filter impulse response given by $h(t)$ and the spike train given as a sum of Dirac functionals centered at the spike times. In particular, binning is nothing but a special case of this procedure in which the spike times are first quantized according to the binsize and h is a rectangular window [1]. Moreover, compared with pdf estimation with Parzen windows [30], we immediately observe that intensity estimation as shown above is directly related to the problem of pdf estimation except for a normalization term, a connection made clear by Diggle and Marron [31].

Consider spike trains $s_i, s_j \in \mathcal{S}(\mathcal{T})$ with estimated intensity functions $\hat{\lambda}_{s_i}(t)$ and $\hat{\lambda}_{s_j}(t)$ according to equation (17). Substituting the estimated intensity functions in the definition of the mCI kernel (equation (12)) yields

$$\hat{I}(s_i, s_j) = \sum_{m=1}^{N_i} \sum_{n=1}^{N_j} \kappa(t_m^i - t_n^j). \tag{18}$$

where κ is the 'kernel' obtained by the autocorrelation of the smoothing function h. Notice that ultimately the obtained estimator linearly combines and weights the contribution of a kernel operating on a pair of event coordinates. Moreover, this estimator operates directly on the event coordinates of the whole realization without loss of resolution and in a computationally efficient manner since it takes advantage of the, typically, sparse occurrence of events.

If the kernel κ is chosen such that it satisfies the requirements in section 3, then the mCI kernel corresponds to a summation of all pairwise inner products between spike times of the spike trains, evaluated by kernel on the spike time differences. Put in this way, we can now clearly see how the mCI inner product on spike trains builds upon the inner product on spike times denoted by κ and the connection to equation (10). The later approach, however, clearly states the underlying point process model.

6 Induced RKHS and congruent spaces

Some considerations about the RKHS space $\mathcal{H}_I$ induced by the mCI kernel and congruent spaces are made in this section. The relationship between $\mathcal{H}_I$ and its congruent spaces provides alternative perspectives and a better understanding of the mCI kernel. Figure 1 provides a diagram of the relationships among the various spaces discussed next.

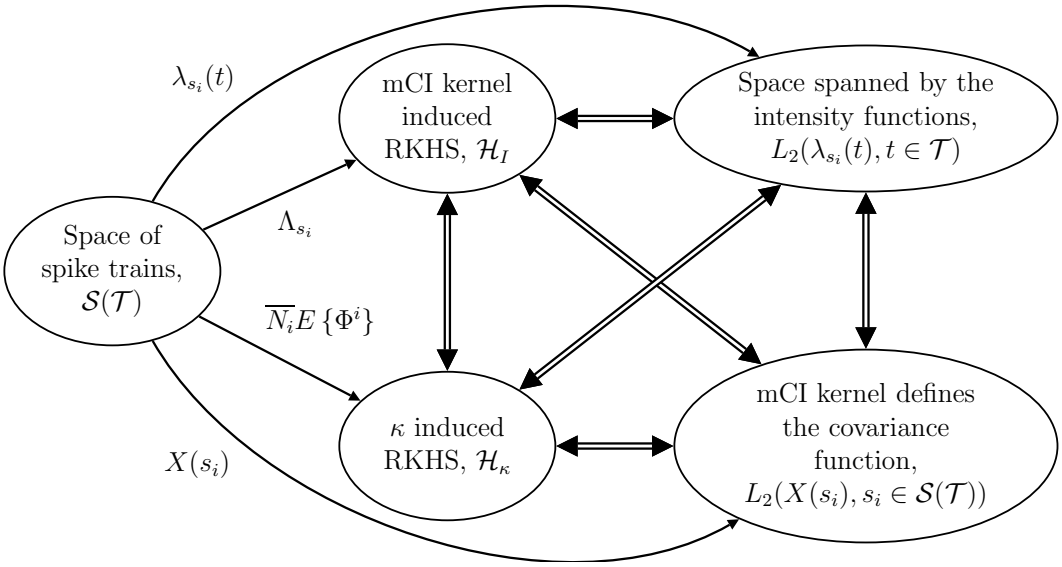


Figure 1: Relation between the original space of spike trains $\mathcal{S}(\mathcal{T})$ and the various Hilbert spaces. The double-line bi-directional connections denote congruence between spaces.

6.1 Space spanned by intensity functions

In the introduction of the mCI kernel the usual dot product in $L_2(\mathcal{T})$, the space of square integrable intensity functions defined on $\mathcal{T}$, was utilized. The definition of the inner product in this space provides an intuitive understanding to the reasoning involved. $L_2(\lambda_{s_i}(t), t \in \mathcal{T}) \subset L_2(\mathcal{T})$ is clearly an Hilbert space with inner product and norm defined in equations (12) and (13). Notice that the span of this space contains also elements (functions) that may not be valid intensity functions since, by definition, intensity functions are always non-negative. However, since our interest is mainly on the evaluation of the inner product this is of no consequence. The key limitation however is that $L_2(\lambda_{s_i}(t), t \in \mathcal{T})$ is *not* an RKHS. This should be clear because elements in this space are functions defined on $\mathcal{T}$, whereas elements in the RKHS $\mathcal{H}_I$ must be functions defined on $\mathcal{S}(\mathcal{T})$.

Despite the differences, the spaces $L_2(\lambda_{s_i}(t), t \in \mathcal{T})$ and $\mathcal{H}_I$ are closely related. In fact, $L_2(\lambda_{s_i}(t), t \in \mathcal{T})$ and $\mathcal{H}_I$ are congruent. This congruence can be verified explicitly since there is clearly a one-to-one mapping,

$$\lambda_{s_i}(t) \in L_2(\lambda_{s_i}(t), t \in \mathcal{T}) \quad \longleftrightarrow \quad \Lambda_{s_i}(s) \in \mathcal{H}_I,$$

and, by definition of the mCI kernel,

$$I(s_i, s_j) = \left\langle \lambda_{s_i}, \lambda_{s_j} \right\rangle_{L_2(\mathcal{T})} = \left\langle \Lambda_{s_i}, \Lambda_{s_j} \right\rangle_{\mathcal{H}_I}. \tag{19}$$

A direct implication of the basic congruence theorem is that the two spaces have the same dimension [11].

6.2 Induced RKHS

In section 5.1 it was shown that the mCI kernel is symmetric and positive definite (properties 1 and 3, respectively) and consequently, by the Moore-Aronszajn theorem [20], there exists an Hilbert space $\mathcal{H}_I$ in which the mCI kernel evaluates the inner product and is a reproducing kernel (property 4). This means that $I(s_i, \cdot) \in \mathcal{H}_I$ for any $s_i \in \mathcal{S}(\mathcal{T})$ and, for any $\xi \in \mathcal{H}_I$, the reproducing property holds

$$\langle \xi, I(s_i, \cdot) \rangle_{\mathcal{H}_I} = \xi(s_i). \tag{20}$$

As a result the kernel trick follows,

$$I(s_i, s_j) = \langle I(s_i, \cdot), I(s_j, \cdot) \rangle_{\mathcal{H}_I}. \tag{21}$$

Written in this form, it is easy to verify that the point in $\mathcal{H}_I$ corresponding to a spike train $s_i \in \mathcal{S}(\mathcal{T})$ is $I(s_i, \cdot)$. In other words, given any spike train $s_i \in \mathcal{S}(\mathcal{T})$, this spike train is mapped to $\Lambda_{s_i} \in \mathcal{H}_I$, given explicitly (although unknown in closed form) as $\Lambda_{s_i} = I(s_i, \cdot)$. Then equation (21) can be restated in the more usual form

$$I(s_i, s_j) = \left\langle \Lambda_{s_i}, \Lambda_{s_j} \right\rangle_{\mathcal{H}_I}. \tag{22}$$

It must be remarked that $\mathcal{H}_I$ is in fact a functional space. More specifically, that points in $\mathcal{H}_I$ are functions of spike trains defined on $\mathcal{S}(\mathcal{T})$. This is a key difference between the space of intensity functions $L_2(\mathcal{T})$ explained above and the RKHS $\mathcal{H}_I$, in that the latter allows for statistics of the transformed spike trains to be estimated *as functions of spike trains*. The usefulness of an RKHS for optimization and general computation with spike trains can be appreciated, for example, in the derivation of principal component analysis in section 7.

6.3 mCI kernel and the RKHS induced by κ

The mCI kernel estimator in equation (18) shows the evaluation written in terms of elementary kernel operations on the spike times. This fact alone provides a different perspective on how the mCI kernel uses the statistics of the spike times. To see this more clearly, if κ is chosen according to section 3 as symmetric positive definite, then it can be substituted by its inner product (equation (4)) in the mCI kernel estimator, yielding

$$\begin{aligned} \hat{I}(s_i, s_j) &= \sum_{m=1}^{N_i} \sum_{n=1}^{N_j} \left\langle \Phi_m^i, \Phi_n^j \right\rangle_{\mathcal{H}_\kappa} \\ &= \left\langle \sum_{m=1}^{N_i} \Phi_m^i, \sum_{n=1}^{N_j} \Phi_n^j \right\rangle_{\mathcal{H}_\kappa}. \end{aligned} \tag{23}$$

When the number of samples approaches infinity (so that the intensity functions and, consequently the mCI kernel, can be estimated exactly) the mean of the transformed spike times approaches the expectation. Hence, equation (23) results in

$$I(s_i, s_j) = \overline{N_i}\, \overline{N_j} \left\langle E\left\{\Phi^i\right\}, E\left\{\Phi^j\right\} \right\rangle_{\mathcal{H}_\kappa}, \tag{24}$$

where $E\left\{\Phi^i\right\}$, $E\left\{\Phi^j\right\}$ denotes the expectation of the transformed spike times and $\overline{N_i}, \overline{N_j}$ are the expected number of spikes in spike trains s_i and s_j, respectively.

Equation (23) explicitly shows that the mCI kernel can be computed as an inner product of the expectation of the transformed spike times in the RKHS $\mathcal{H}_\kappa$ induced by κ. In other words, there is a congruence $\mathscr{G}$ between $\mathcal{H}_\kappa$ and $\mathcal{H}_I$ in this case given explicitly by the expectation of the transformed spike times, $\mathscr{G}(\Lambda_{s_i}) = \overline{N_i} E\left\{\Phi^i\right\}$, such that

$$\left\langle \Lambda_{s_i}, \Lambda_{s_j} \right\rangle_{\mathcal{H}_I} = \left\langle \mathscr{G}(\Lambda_{s_i}), \mathscr{G}(\Lambda_{s_j}) \right\rangle_{\mathcal{H}_\kappa} = \left\langle \overline{N_i} E\left\{\Phi^i\right\}, \overline{N_j} E\left\{\Phi^j\right\} \right\rangle_{\mathcal{H}_\kappa}. \tag{25}$$

Recall that the transformed spike times form a manifold (the subset of an hypersphere) and, since these points have constant norm, the kernel inner product depends only on the angle between points. This is typically not true for the average of these points, however. Observe that the circular variance [32] of the transformed spike times of spike trains s_i is

$$\begin{aligned} \operatorname{var}(\Phi^i) &= E\left\{ \left\langle \Phi^i_m, \Phi^i_m \right\rangle_{\mathcal{H}_\kappa} \right\} - \left\langle E\left\{\Phi^i\right\}, E\left\{\Phi^i\right\} \right\rangle_{\mathcal{H}_\kappa} \\ &= \kappa(0) - \left\| E\left\{\Phi^i\right\} \right\|^2_{\mathcal{H}_\kappa}. \end{aligned} \tag{26}$$

So, the norm of the mean transformed spike times is inversely proportional to the variance of the elements in $\mathcal{H}_\kappa$. This means that the inner product between two spike trains depends also on the dispersion of these average points. This fact is important because data reduction techniques rely heavily on optimization with the data variance. For instance, kernel principal component analysis [33] directly maximizes the variance expressed by equation (26) [34].

6.4 mCI kernel as a covariance kernel

In section 5.1 it was shown that the mCI kernel is indeed a symmetric positive definite kernel. As mentioned in section 2, Parzen [23] showed that any symmetric and positive definite kernel is also a covariance function of a random process defined in the original space of the kernel (see also Wahba [18, chapter 1]). In the case of the mCI kernel, this means the random processes are defined on $\mathcal{S}(\mathcal{T})$.

Let X denote this random process. Then, for any $s_i \in \mathcal{S}(\mathcal{T})$, $X(s_i)$ is a random variable on a probability space $(\Omega, \mathcal{B}, P)$ with measure P. As proved by Parzen, this random process is Gaussian distributed with zero mean and covariance function

$$I(s_i, s_j) = E_\omega \left\{ X(s_i) X(s_j) \right\}. \tag{27}$$

Notice that the expectation is over $\omega \in \Omega$ since $X(s_i)$ is a random variable defined on Ω, a situation which can be written explicitly as $X(s_i, \omega)$, $s_i \in \mathcal{S}(\mathcal{T})$, $\omega \in \Omega$. This means that X is actually a doubly stochastic random process. An intriguing perspective is that, for any given ω, $X(s_i, \omega)$ is an ordered and almost surely non-uniform sampling of $X(\cdot, \omega)$. The space spanned by these random variables is $L_2(X(s_i), s_i \in \mathcal{S}(\mathcal{T}))$ since X is obviously square integrable (that is, X has finite covariance).

The RKHS $\mathcal{H}_I$ induced by the mCI kernel and the space of random functions $L_2(X(s_i), s_i \in \mathcal{S}(\mathcal{T}))$ are clearly congruent. This fact is a consequence of the basic congruence theorem [23] since the two spaces have the same dimension or, alternatively, by verifying that the

congruence mapping between the two space exist. For this reason we may consider the mCI kernel also as a covariance measure of random variables directly dependent on the spike trains with well defined statistical properties. Allied to our familiarity and intuitive knowledge of the use of covariance (which is nothing but cross-correlation between centered random variables) this concept can be of great importance in optimization and design of optimal learning algorithms that work with spike trains. This is because linear methods are known to be optimal for Gaussian distributed random variables.

7 Principal component analysis

To exemplify the importance of the developments shown here, in the following we derive the algorithm to perform principal component analysis (PCA) of spike trains. The PCA algorithm will be derived from two different perspectives to show the generality of an RKHS framework for optimization with spike trains.

First, PCA will be derived directly in the RKHS induced by the mCI kernel. This approach highlights that optimization with spike trains is possible by the definition of an inner product, and more specifically through the mathematical structure provided by the RKHS. This is also the traditional approach in the functional analysis literature [35] and has the advantage of being completely general, regardless of the spike train kernel definition. A well known example of discrete PCA done in an RKHS is kernel PCA [33].

In the second approach we will derive PCA in the space spanned by the intensity functions utilizing the inner product defined in this space. Since the RKHS is congruent to this space and therefore the inner products in the two spaces are isometric the outcome will be found to be the same. However, this approach has the advantage that the eigenfunctions are explicitly available. In general, the eigenfunctions are not available in the RKHS because the transformation to the RKHS is unknown. However, this approach is possible here due to the linearity of the space spanned by the intensity functions with the inner product we defined.

7.1 Optimization in the RKHS

Suppose we are given a set of spike trains, $\{s_i \in \mathcal{S}(\mathcal{T}), i = 1, \ldots, N\}$, for which we wish to determine the principal components. Computing the principal components of the spike trains directly is not feasible because we would not know how to define a principal component (PC), however, this is a trivial task in an RKHS.

Let $\{\Lambda_{s_i} \in \mathcal{H}_I, i = 1, \ldots, N\}$ be the set of elements in the RKHS $\mathcal{H}_I$ corresponding to the given spike trains. Denote the mean of the transformed spike trains as

$$\bar{\Lambda} = \frac{1}{N} \sum_{i=1}^{N} \Lambda_{s_i}, \tag{28}$$

and the centered transformed spike trains (i.e., with the mean removed) can be obtained as

$$\tilde{\Lambda}_{s_i} = \Lambda_{s_i} - \bar{\Lambda}. \tag{29}$$

PCA finds an orthonormal transformation providing a compact description of the data. Determining the principal components of spike trains in the RKHS can be formulated as the problem of finding the set of orthonormal vectors in the RKHS such that the projection of the centered transformed spike trains $\{\tilde{\Lambda}_{s_i}\}$ has the *maximum variance*. This means that the principal components can be obtained by solving an optimization problem in the RKHS. A function $\xi \in \mathcal{H}_I$ (i.e., $\xi : \mathcal{S}(\mathcal{T}) \longrightarrow \mathbb{R}$) is a principal component if it maximizes the cost function

$$J(\xi) = \sum_{i=1}^{N} \left[\text{Proj}_{\xi}(\tilde{\Lambda}_{s_i})\right]^2 - \rho\left(\|\xi\|^2 - 1\right) \tag{30}$$

where $\text{Proj}_{\xi}(\tilde{\Lambda}_{s_i})$ denotes the projection of the ith centered transformed spike train onto ξ, and ρ is the Lagrange multiplier to the constraint $\left(\|\xi\|^2 - 1\right)$ imposing that the principal components have unit norm. To evaluate this cost function one needs to be able to compute the projection and the norm of the principal components. However, in an RKHS, an inner product is the projection operator and the norm is naturally defined (see equation (13)). Thus, the above cost function can be expressed as

$$J(\xi) = \sum_{i=1}^{N} \left\langle \tilde{\Lambda}_{s_i}, \xi \right\rangle_{\mathcal{H}_I}^2 - \rho\left(\langle \xi, \xi \rangle_{\mathcal{H}_I} - 1\right), \tag{31}$$

Because in practice we always have a finite number of spike trains, ξ is restricted to the subspace spanned by the centered transformed spike trains $\{\tilde{\Lambda}_{s_i}\}$. Consequently, there exist coefficients $b_1, \ldots, b_N \in \mathbb{R}$ such that

$$\xi = \sum_{j=1}^{N} b_j \tilde{\Lambda}_{s_j} = \boldsymbol{b}^T \tilde{\boldsymbol{\Lambda}} \tag{32}$$

where $\boldsymbol{b}^T = [b_1, \ldots, b_N]$ and $\tilde{\boldsymbol{\Lambda}}(t) = \left[\tilde{\Lambda}_{s_1}(t), \ldots, \tilde{\Lambda}_{s_N}(t)\right]^T$. Substituting in equation (31) yields

$$\begin{aligned} J(\xi) &= \sum_{i=1}^{N} \left(\sum_{j=1}^{N} b_j \left\langle \tilde{\Lambda}_{s_i}, \tilde{\Lambda}_{s_j} \right\rangle \right) \left(\sum_{k=1}^{N} b_k \left\langle \tilde{\Lambda}_{s_i}, \tilde{\Lambda}_{s_k} \right\rangle \right) \\ &\quad + \rho \left(1 - \sum_{j=1}^{N} \sum_{k=1}^{N} b_j b_k \left\langle \tilde{\Lambda}_{s_i}, \tilde{\Lambda}_{s_k} \right\rangle \right) \\ &= \boldsymbol{b}^T \tilde{\boldsymbol{I}}^2 \boldsymbol{b} + \rho\left(1 - \boldsymbol{b}^T \tilde{\boldsymbol{I}} \boldsymbol{b}\right). \end{aligned} \tag{33}$$

where $\tilde{\boldsymbol{I}}$ is the Gram matrix of the centered spike trains; that is, the $N \times N$ matrix with elements

$$\begin{aligned} \tilde{\boldsymbol{I}}_{ij} &= \left\langle \tilde{\Lambda}_{s_i}, \tilde{\Lambda}_{s_j} \right\rangle \\ &= \left\langle \Lambda_{s_i} - \bar{\Lambda}, \Lambda_{s_j} - \bar{\Lambda} \right\rangle \\ &= \left\langle \Lambda_{s_i}, \Lambda_{s_j} \right\rangle - \frac{1}{N} \sum_{l=1}^{N} \left\langle \Lambda_{s_i}, \Lambda_{s_l} \right\rangle - \frac{1}{N} \sum_{l=1}^{N} \left\langle \Lambda_{s_l}, \Lambda_{s_j} \right\rangle + \frac{1}{N^2} \sum_{l=1}^{N} \sum_{n=1}^{N} \left\langle \Lambda_{s_l}, \Lambda_{s_n} \right\rangle. \end{aligned} \tag{34}$$

In matrix notation,

$$\tilde{\boldsymbol{I}} = \boldsymbol{I} - \frac{1}{N}(\mathbf{1}_N \boldsymbol{I} + \boldsymbol{I}\mathbf{1}_N) + \frac{1}{N^2}\mathbf{1}_N \boldsymbol{I}\mathbf{1}_N, \tag{35}$$

where $\boldsymbol{I}$ is the Gram matrix of the inner product of spike trains $\boldsymbol{I}_{ij} = \left\langle \Lambda_{s_i}, \Lambda_{s_j} \right\rangle$, and $\mathbf{1}_N$ is the $N \times N$ matrix with all ones. This means that $\tilde{\boldsymbol{I}}$ can be computed directly in terms of $\boldsymbol{I}$ without the need to explicitly remove the mean of the transformed spike trains.

From equation (33), finding the principal components simplifies to the problem of estimating the coefficients $\{b_i\}$ that maximize $J(\xi)$. Since $J(\xi)$ is a quadratic function its extrema can be found by equating the gradient to zero. Taking the derivative with regards to $\boldsymbol{b}$ (which characterizes ξ) and setting it to zero results in

$$\frac{\partial J(\xi)}{\partial \boldsymbol{b}} = 2\tilde{\boldsymbol{I}}^2\boldsymbol{b} - 2\rho\tilde{\boldsymbol{I}}\boldsymbol{b} = 0, \tag{36}$$

and thus corresponds to the eigendecomposition problem[3]

$$\tilde{\boldsymbol{I}}\boldsymbol{b} = \rho\boldsymbol{b}. \tag{37}$$

This means that any eigenvector of the centered Gram matrix is a solution of equation (36). Thus, the eigenvectors determine the coefficients of equation (32) and characterize the principal components. It is easy to verify that, as expected, the variance of the projections onto each principal component equals the corresponding eigenvalue. So, the ordering of ρ specifies the relevance of the principal components.

To compute the projection of a given input spike train s onto the kth principal component (corresponding to the eigenvector with the kth largest eigenvalue) we need only to compute in the RKHS the inner product of Λ_s with ξ_k. That is,

$$\begin{aligned} \mathrm{Proj}_{\xi_k}(\Lambda_s) &= \langle \Lambda_s, \xi_k \rangle_{\mathcal{H}_I} \\ &= \sum_{i=1}^{N} b_{ki} \left\langle \Lambda_s, \tilde{\Lambda}_{s_i} \right\rangle \\ &= \sum_{i=1}^{N} b_{ki} \left(I(s, s_i) - \frac{1}{N}\sum_{j=1}^{N} I(s, s_j) \right). \end{aligned} \tag{38}$$

7.2 Optimization in the space spanned by the intensity functions

As before, let $\{s_i \in \mathcal{S}(\mathcal{T}), i = 1, \ldots, N\}$ denote the set of spike trains for which we wish to determine the principal components, and $\{\lambda_{s_i}(t), t \in \mathcal{T}, i = 1, \ldots, N\}$ the corresponding intensity functions. The mean intensity function is

$$\bar{\lambda}(t) = \frac{1}{N}\sum_{i=1}^{N} \lambda_{s_i}(t), \tag{39}$$

and therefore the centered intensity functions are

$$\tilde{\lambda}_{s_i}(t) = \lambda_{s_i}(t) - \bar{\lambda}(t). \tag{40}$$

[3]Note that the simplification in the eigendecomposition problem is valid regardless if the Gram matrix is invertible or not, since $\tilde{\boldsymbol{I}}^2$ and $\tilde{\boldsymbol{I}}$ have the same eigenvectors and the eigenvalues of $\tilde{\boldsymbol{I}}^2$ are the eigenvalues of $\tilde{\boldsymbol{I}}$ squared.

Again, the problem of finding the principal components of a set of data can be stated as the problem of finding the eigenfunctions of unit norm such that the projections have maximum variance. This can be formulated in terms of the following optimization problem. A function $\zeta(t) \in L_2(\lambda_{s_i}(t), t \in \mathcal{T})$ is a principal component if it maximizes the cost function

$$\begin{aligned} J(\zeta) &= \sum_{i=1}^{N} \left[\text{Proj}_{\zeta}(\tilde{\lambda}_{s_i})\right]^2 - \gamma\left(\|\zeta\|^2 - 1\right) \\ &= \sum_{i=1}^{N} \left\langle \tilde{\lambda}_{s_i}, \zeta \right\rangle_{L_2}^2 - \gamma\left(\|\zeta\|^2 - 1\right), \end{aligned} \tag{41}$$

where γ is the Lagrange multiplier constraining ζ to have unit norm. It can be shown that $\zeta(t)$ lies in the subspace spanned by the intensity functions $\{\tilde{\lambda}_{s_i}(t), i = 1, \ldots, N\}$. Therefore, there exist coefficients $b_1, \ldots, b_N \in \mathbb{R}$ such that

$$\zeta(t) = \sum_{j=1}^{N} b_j \tilde{\lambda}_{s_j}(t) = \boldsymbol{b}^T \tilde{\boldsymbol{r}}(t). \tag{42}$$

with $\boldsymbol{b}^T = [b_1, \ldots, b_N]$ and $\tilde{\boldsymbol{r}}(t) = \left[\tilde{\lambda}_{s_1}(t), \ldots, \tilde{\lambda}_{s_N}(t)\right]^T$. Substituting in equation (31) yields

$$\begin{aligned} J(\zeta) = &\sum_{i=1}^{N} \left(\sum_{j=1}^{N} b_j \left\langle \tilde{\lambda}_{s_i}, \tilde{\lambda}_{s_j} \right\rangle \right) \left(\sum_{k=1}^{N} b_k \left\langle \tilde{\lambda}_{s_i}, \tilde{\lambda}_{s_k} \right\rangle \right) \\ &+ \gamma \left(1 - \sum_{j=1}^{N} \sum_{k=1}^{N} b_j b_k \left\langle \tilde{\lambda}_{s_i}, \tilde{\lambda}_{s_k} \right\rangle \right) \\ = &\boldsymbol{b}^T \tilde{\boldsymbol{I}}^2 \boldsymbol{b} + \gamma\left(1 - \boldsymbol{b}^T \tilde{\boldsymbol{I}} \boldsymbol{b}\right). \end{aligned} \tag{43}$$

where $\tilde{\boldsymbol{I}}$ is the gram matrix of the centered intensity functions (i.e., $\tilde{\text{I}}_{ij} = \left\langle \tilde{\lambda}_{s_i}, \tilde{\lambda}_{s_j} \right\rangle_{L_2}$).

As expected, since the inner product is the same and the two spaces are congruent, this cost function yields the same solution. However, unlike the previous, this presentation has the advantage that it shows the role of the eigenvectors of the gram matrix and, most importantly, how to obtain the principal component functions in the space of intensity functions. From equation (42), the coefficients of the eigenvectors of the gram matrix provide a weighting for the intensity functions of each spike trains and therefore expresses how important a spike train is to represent others. In a different perspective, this suggests that the principal component functions should reveal general trends in the intensity functions.

7.3 Results

To illustrate the algorithm just derived we performed a simple experiment. We generated two template spike trains comprising of 10 spikes uniformly random distributed over an interval of 0.25s. In a specific application these template spike trains could correspond, for example, to the average response of a culture of neurons to two distinct but fixed input stimuli. For the computation of the coefficients of the eigendecomposition ("training set"),

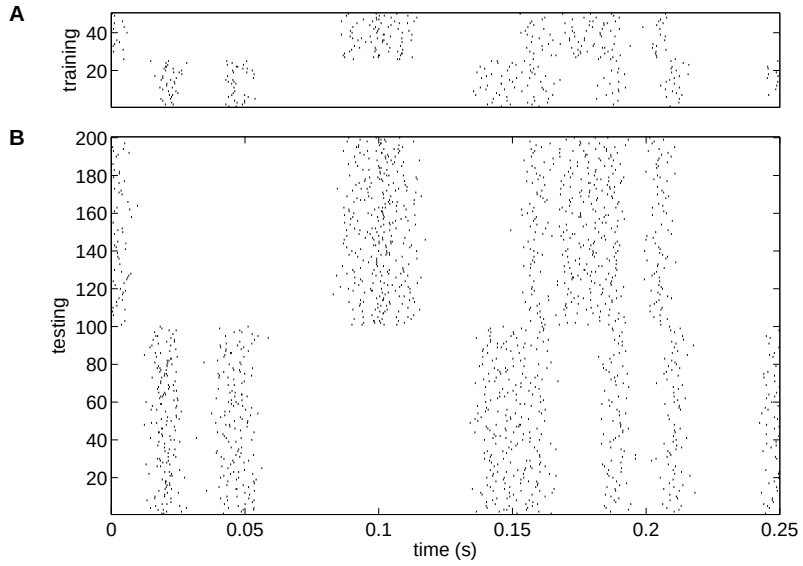
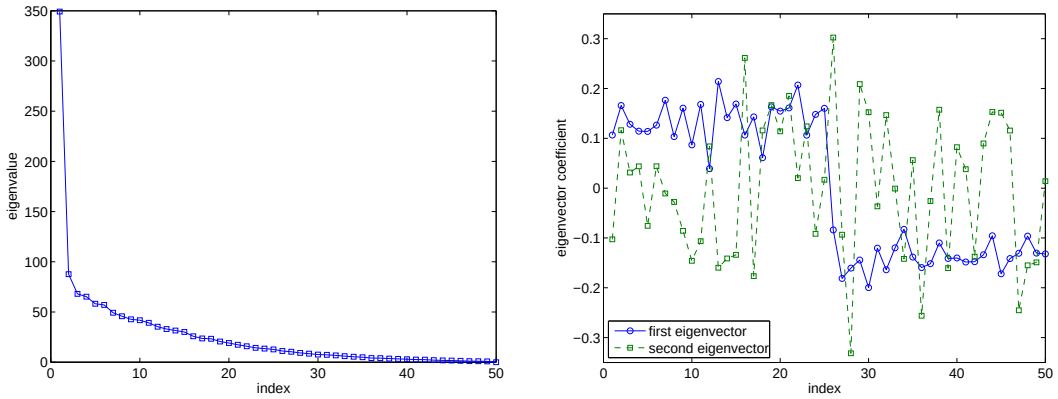


Figure 2: Spike trains used for evaluation of the eigendecomposition coefficients of PCA algorithm (A), and for testing of the result (B). In either case, the first half of spike trains corresponds to the first template and the remaining to the second template.

we generated a total of 50 spike trains, half for each template, by randomly copying each spike from the template with probability 0.8 and adding zero mean Gaussian distributed jitter with standard deviation 3ms. For testing of the obtained coefficients, 200 spike trains were generated following the same procedure. The simulated spike trains are shown in figure 2.

According to the PCA algorithm derived previously, we computed the eigendecomposition of the matrix $\tilde{\boldsymbol{I}}$ as given by equation (35) so that it solves equation (37). The evaluation of the mCI kernel was estimated from the spike trains according to equation (12), and computed with a Gaussian kernel with size 2ms. The eigenvalues $\{\rho_l, l = 1, \ldots, 100\}$ and first two eigenvectors are shown in figure 3. The first eigenvalue alone accounts for more than 26% of the variance of the dataset in the RKHS space. Although this value is not impressive, its importance is clear since it is nearly 4 times higher than the second eigenvalue (6.6%). Furthermore, notice that the first eigenvector clearly shows the separation between spike trains generated from different templates (Fig. 3(b)). This again can be seen in the first principal component function, shown in figure 4, which reveals the location of the spike times used to generate the templates while discriminating between them with opposite signs. Around periods of time where the spike from both templates overlap the first principal component is zero. As can be seen from the second principal component function, the role of the second eigenvector is to account for the dispersion in the data capable of differentiate spike trains generate from different templates.

Both datasets, for evaluation and testing, where projected onto the first two principal components. Figure 5 shows the projected spike trains. As noted from the difference between



(a) Eigenvalues $\{\rho_l, l = 1, \ldots, 100\}$ in decreasing order.

(b) First two eigenvectors.

Figure 3: Eigendecomposition of the spike trains Gram matrix $\tilde{\boldsymbol{I}}$.

the first and second eigenvalues, the first principal component is the main responsible for the dispersion between classes of the projected spike trains. This happens because the direction of maximum variance is the one that passes through both clusters of points in the RKHS due to the small dispersion within class. The second principal component seems to be responsible for dispersion due to the jitter noise introduced in the spike trains, and suggests that other principal components play a similar role.

A more specific understanding can be obtained from the considerations done in section 6.3. There, the congruence between the RKHS induced by the mCI kernel, $\mathcal{H}_I$, and the RKHS induced by κ, $\mathcal{H}_\kappa$, was utilized to show that the mCI kernel is inversely related to the variance of the transformed spike times in $\mathcal{H}_\kappa$. In this dataset and for the kernel size utilized, this guaranties that the value of the mCI kernel within class is always smaller than inter class. This is a reason why in this scenario the first principal component always suffices to project the data in a way that distinguishes between spike trains generated each of the templates.

Conventional PCA was also applied to this dataset by binning the spike trains. Although cross-correlation is an inner product for spike trains and therefore the above algorithm could have been used, for comparison, the conventional approach was followed [36, 37]. That is, to compute the covariance matrix with each binned spike train taken as a data vector. This means that the dimensionality of the covariance matrix is determined by the number of bins per spike train, which may be problematic if long spike trains are used or small bin sizes are needed for high temporal resolution.

The results of PCA using bin size of 5ms are shown in figure 6 and figure 7. The bin size was chosen to provide a good compromise between temporal resolution and smoothness of the eigenfunctions (important for interpretability). Comparing these results the ones using the mCI kernel, the distribution of the eigenvalues is quite similar and the first eigenfunction does reveals somewhat of the same trend as in figure 4. The same is not true for the second

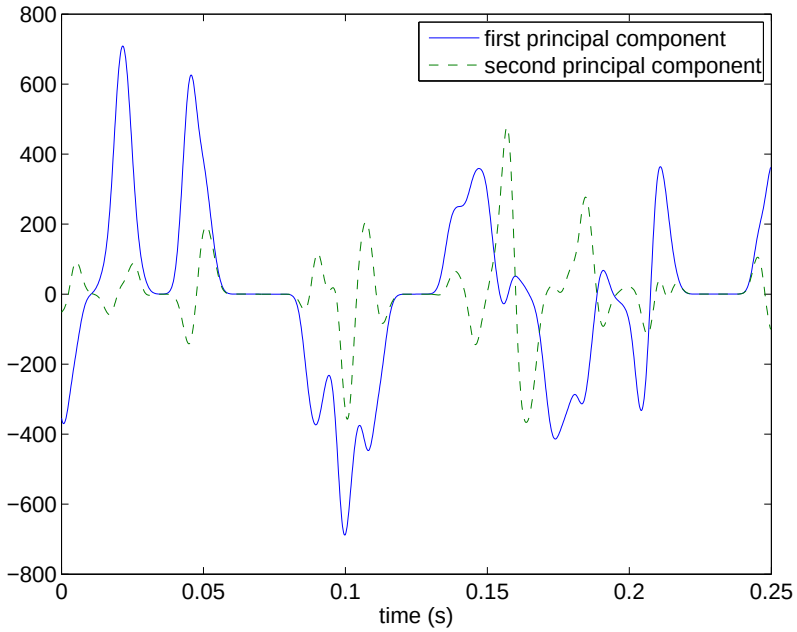
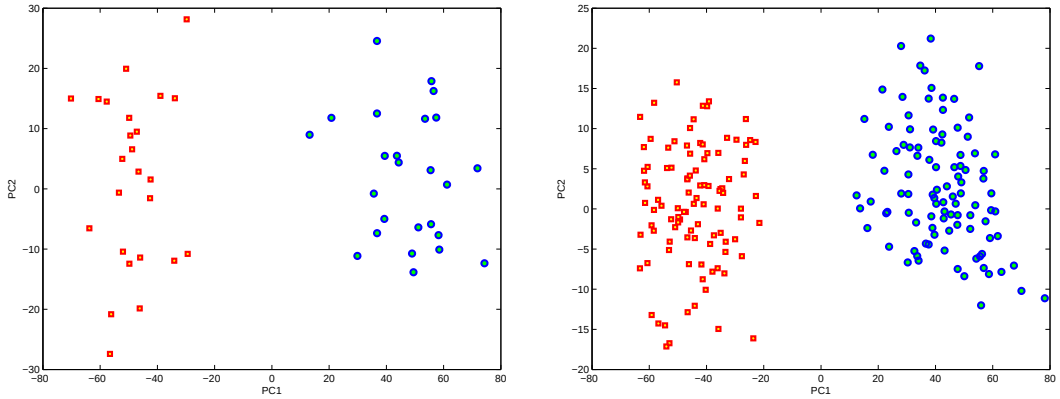


Figure 4: First two principal component functions (i.e., eigenfunctions) in the space of intensity functions. They are computed by substituting the coefficients of the first two eigenvectors of the Gram matrix in equation (42).

eigenfunction, however, which looks much more "jaggy." In fact, as figure 7 shows, in this case the projections along the first two principal directions are not orthogonal. This means that the covariance matrix does not fully express the structure of the spike trains. It is noteworthy that this is not only because the covariance matrix is being estimated with a small number of data vectors. In fact, even if the binned cross-correlation was utilized directly in the above algorithm as the inner product the same effect was observed, meaning that the *binned cross-correlation does not characterize the spike train structure in sufficient detail*. Since the binned cross-correlation and the mCI kernel are conceptually equivalent apart from the discretization introduced by binning, this shows the ill effects of this preprocessing step for analysis and computation with spike train, and point process realizations in general.

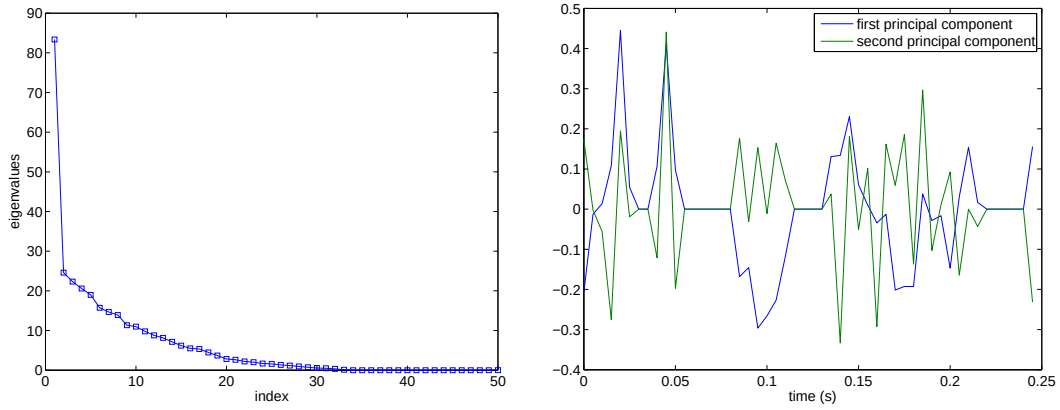
8 Conclusion

A reproducing kernel Hilbert space (RKHS) framework for optimization with spike trains is introduced. Although the application of kernel methods to spike trains without binning is not entirely novel [8, 9], a more general view of the problem is presented. Instead of a top-down approach often taken in kernel methods, the mCI kernel was built bottom-up from the concept of intensity functions which are basic statistical descriptors of spike trains. Indeed, intensity functions are the core concept of the statistical analysis of spike trains



(a) Projection of the spike trains in the training set. (b) Projection of the spike trains in the testing set.

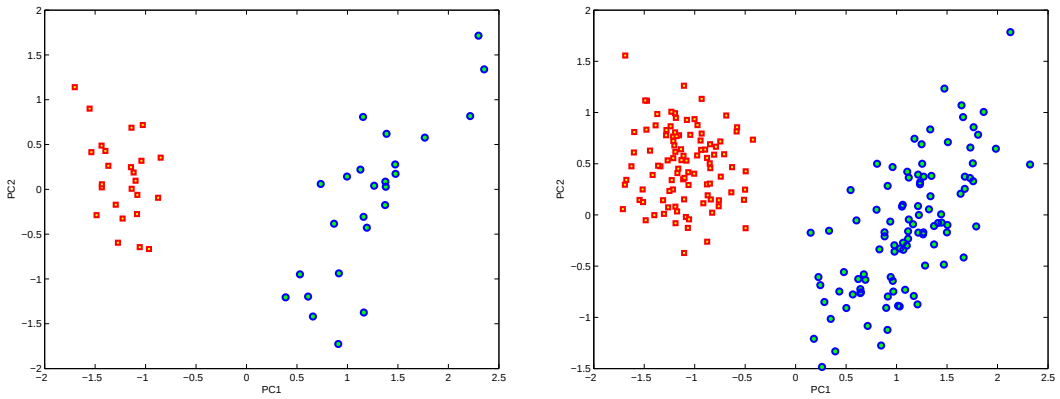
Figure 5: Projection of spike trains onto the first two principal components. The different point marks differentiate between spike trains corresponding to each one of the classes.



(a) Eigenvalues in decreasing order. (b) First two eigenvectors/eigenfunctions.

Figure 6: Eigendecomposition of the binned spike trains covariance matrix.

and is perhaps one of reasons why binning is such a well established technique, at any time-scale of interest [27, 1]. Kernel methods applied before to spike trains seemed to have no connection to intensity estimation. This paper, however, bridges these two perspectives seamlessly. In one perspective, the mCI kernel approximates our intuitive understanding regarding intensity functions as functional descriptors of point processes. On the other hand, the evaluation (or estimation) of the mCI kernel for given spike trains easily links to other methodologies in the literature. Most importantly, the approach taken lends itself to generalization to other point process models and spike train kernels nonlinear in the space of intensity functions taking advantage of the RKHS mathematical structure and without sacrifice in rigor.



(a) Projection of the spike trains in the training set. (b) Projection of the spike trains in the testing set.

Figure 7: Projection of spike trains onto the first two principal components of the covariance matrix of binned spike trains. The different point marks differentiate between spike trains corresponding to each one of the classes.

In addition to this enlightening connection of point of view, the rigorous yet general mathematical approach towards the problem of optimization for manipulating spike trains clarifies exactly from basic principles which kernels can be used and what are the general properties of the mCI kernel defined. Even though it may be argued that kernel methods can applied directly for spike trains data given a kernel, the true meaning of using such a kernel cannot be well determined. This is one of the strengths of the explicit construction followed. In this way, the general structure of the RKHS space induced is well understood allowing for methods to be derived from their basic ideas. Additionally, we were able to establish a close mathematical relationship to several congruent spaces where the derived methods can be thoroughly comprehended. Still, it must be remarked that the mCI kernel presented here will likely not be the most appropriate for a number of problems. This was not the goal of this paper. Instead one of our aims was to show how other kernels that operate with spike trains may be easily formulated. Depending on a specific application other kernels may be defined which lead to simpler solutions and/or are computationally simpler.

It is noteworthy that the mCI kernel is not restricted to applications with spike trains but rather can be applied to processing with any Poisson point processes. In fact, the mCI kernel can be applied for even more general point processes. Naturally, it might not be the optimum inner product for point processes other than Poisson processes since the intensity function does not fully characterizes the process but, in a sense, this is similar to the use of cross-correlation in continuous random processes, which is only sensitive to second-order statistics.

Acknowledgements

A. R. C. Paiva was supported by Fundação para a Ciência e a Tecnologia (FCT), Portugal, under grant SRFH/BD/18217/2004. This work was partially supported by NSF grants ECS-0422718 and CISE-0541241.

A Proofs

This section presents the proofs for properties 2, 5 and 6 in section 5.1.

Proof (Property 2): The symmetry of the matrix results immediately from property 1.

By definition, a matrix is non-negative definite if and only if $\boldsymbol{a}^T\boldsymbol{V}\boldsymbol{a} \geq 0$, for any $\boldsymbol{a}^T = [a_1, \ldots, a_n]$ with $a_i \in \mathbb{R}$. So, we have that

$$\boldsymbol{a}^T\boldsymbol{V}\boldsymbol{a} = \sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j I(s_i, s_j), \tag{44}$$

which, making use of the mCI kernel definition (equation (12)), yields

$$\begin{aligned} \boldsymbol{a}^T\boldsymbol{V}\boldsymbol{a} &= \int_{\mathcal{T}} \left(\sum_{i=1}^{n} a_i \lambda_{s_i}(t)\right)\left(\sum_{j=1}^{n} a_j \lambda_{s_j}(t)\right) dt \\ &= \left\langle \sum_{i=1}^{n} a_i \lambda_{s_i}, \sum_{j=1}^{n} a_j \lambda_{s_j} \right\rangle_{L_2(\mathcal{T})} \\ &= \left\| \sum_{i=1}^{n} a_i \lambda_{s_i} \right\|_{L_2(\mathcal{T})} \geq 0, \end{aligned} \tag{45}$$

since the norm is non-negative. □

Proof (Property 5): Consider the 2×2 CI kernel matrix,

$$\boldsymbol{V} = \begin{bmatrix} I(s_i, s_i) & I(s_i, s_j) \\ I(s_j, s_i) & I(s_j, s_j) \end{bmatrix}.$$

From property 2, this matrix is symmetric and non-negative definite. Hence, its determinant is non-negative [38, pg. 245]. Mathematically,

$$\det(\boldsymbol{V}) = I(s_i, s_i)I(s_j, s_j) - I^2(s_i, s_j) \geq 0,$$

which proves the result of equation (16). □

Proof (Property 6): Consider two spike trains, $s_i, s_j \in \mathcal{S}(\mathcal{T})$. The norm of the sum of two spike trains is

$$\begin{aligned} \left\|\lambda_{s_i} + \lambda_{s_j}\right\|^2 &= \left\langle \lambda_{s_i} + \lambda_{s_j}, \lambda_{s_i} + \lambda_{s_j} \right\rangle && (46a) \\ &= \left\langle \lambda_{s_i}, \lambda_{s_i} \right\rangle + 2\left\langle \lambda_{s_i}, \lambda_{s_j} \right\rangle + \left\langle \lambda_{s_j}, \lambda_{s_j} \right\rangle && (46b) \\ &\leq \left\|\lambda_{s_i}\right\|^2 + 2\left\|\lambda_{s_i}\right\|\left\|\lambda_{s_j}\right\| + \left\|\lambda_{s_j}\right\|^2 && (46c) \\ &= \left(\left\|\lambda_{s_i}\right\| + \left\|\lambda_{s_j}\right\|\right)^2, && (46d) \end{aligned}$$

with the upper bound in step 46c established by the Cauchy-Schwarz inequality (Property 5). □

References

[1] Peter Dayan and L. F. Abbott. *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. MIT Press, Cambridge, MA, USA, 2001.

[2] Wolfgang Maass, Thomas Natschläger, and Henry Markram. Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Comp.*, 14(11):2531–2560, 2002. doi: 10.1162/089976602760407955.

[3] Wolfgang Maass and Christopher M. Bishop, editors. *Pulsed Neural Networks*. MIT Press, 1998.

[4] Sander M. Bohte, Joost N. Kok, and Han La Poutré. Error-backpropagation in temporally encoded networks of spiking neurons. *Neurocomp.*, 48(1–4):17–37, October 2002. doi: 10.1016/S0925-2312(01)00658-0.

[5] Jonathan D. Victor and Keith P. Purpura. Nature and precision of temporal coding in visual cortex: A metric-space analysis. *J. Neurophysiol.*, 76(2):1310–1326, August 1996.

[6] Jonathan D. Victor and Keith P. Purpura. Metric-space analysis of spike trains: theory, algorithms, and application. *Network: Comp. Neural Sys.*, 8:127–164, October 1997.

[7] M. C. W. van Rossum. A novel spike distance. *Neural Comp.*, 13(4):751–764, 2001.

[8] Andrew Carnell and Daniel Richardson. Linear algebra for time series of spikes. In *Proc. European Symp. on Artificial Neural Networks*, pages 363–368, Bruges, Belgium, April 2005.

[9] Benjamin Schrauwen and Jan Van Campenhout. Linking non-binned spike train kernels to several existing spike train distances. *Neurocomp.*, 70(7–8):1247–1253, March 2007. doi: 10.1016/j.neucom.2006.11.017.

[10] António R. C. Paiva, Il Park, and José C. Príncipe. Reproducing kernel Hilbert spaces for spike train analysis. In *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, ICASSP-2008*, Las Vegas, NV, USA, April 2008.

[11] Emanuel Parzen. Statistical inference on time series by Hilbert space methods. Technical Report 23, Applied Mathematics and Statistics Laboratory, Stanford University, Stanford, California, January 1959.

[12] Emanuel Parzen. Statistical inference on time series by RKHS methods. In Ronald Pyke, editor, *Proc. 12th Biennal Int. Seminar of the Canadian Mathematical Congress*, pages 1–37, 1970.

[13] Thomas Kailath. RKHS approach to detection and estimation problems–part I: Deterministic signals in gaussian noise. *IEEE Trans. Inform. Theory*, 17(5):530–549, September 1971.

[14] Thomas Kailath and Howard L. Weinert. An RKHS approach to detection and estimation problems–part II: Gaussian signal detection. *IEEE Trans. Inform. Theory*, 21 (1):15–23, January 1975.

[15] Thomas Kailath and Donald L. Duttweiler. An RKHS approach to detection and estimation problems–part III: Generalized innovations representations and a likelihood-ratio formula. *IEEE Trans. Inform. Theory*, 18(6):730–745, November 1972.

[16] Bernhard Schölkopf, Christopher J. C. Burges, and Alexander J. Smola, editors. *Advances in Kernel Methods: Support Vector Learning*. MIT Press, 1999.

[17] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.

[18] Grace Wahba. *Spline Models for Observational Data*, volume 59 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. SIAM, 1990.

[19] Simon Haykin. *Adaptive Filter Processing*. Prentice-Hall, 4nd edition, 2002.

[20] N. Aronszajn. Theory of reproducing kernels. *Trans. Am. Math. Soc.*, 68(3):337–404, May 1950.

[21] E. H. Moore. On properly positive Hermitian matrices. *Bull. Am. Math. Soc.*, 23:59, 1916.

[22] J. Mercer. Functions of positive and negative type, and their connection with the theory of integral equations. *Phil. Trans. R. Soc. London – A*, 209:415–446, 1909.

[23] Emanuel Parzen. *Time Series Analysis Papers*. Holden-Day, San Francisco, CA, 1967.

[24] José C. Príncipe, Dongxin Xu, and John W. Fisher. Information theoretic learning. In Simon Haykin, editor, *Unsupervised Adaptive Filtering*, volume 2, pages 265–319. John Wiley & Sons, 2000.

[25] Christian Berg, Jens Peter Reus Christensen, and Paul Ressel. *Harmonic Analysis on Semigroups: Theory of Positive Definite and Related Functions*. Springer-Verlag, New York, NY, 1984.

[26] D. L. Snyder. *Random Point Process in Time and Space*. John Viley & Sons, New York, 1975.

[27] Fred Rieke, David Warland, Rob de Ruyter van Steveninck, and William Bialek. *Spikes: exploring the neural code*. MIT Press, Cambridge, MA, USA, 1999. ISBN 0-262-18174-6.

[28] S. Schreiber, J. M. Fellous, D. Whitmer, P. Tiesinga, and T. J. Sejnowski. A new correlation-based measure of spike timing reliability. *Neurocomp.*, 52–54:925–931, June 2003. doi: 10.1016/S0925-2312(02)00838-X.

[29] Rolf-Dieter Reiss. *A Course on Point Processes*. Springer-Verlag, New York, NY, 1993.

[30] Emanuel Parzen. On the estimation of a probability density function and the mode. *Annals Math. Stat.*, 33(2):1065–1076, September 1962.

[31] Peter Diggle and J. S. Marron. Equivalence of smoothing parameter selectors in density and intensity estimation. *J. Acoust. Soc. Am.*, 83(403):793–800, September 1988.

[32] Kanti V. Mardia and Peter E. Jupp. *Directional Statistics*. John Wiley & Sons, West Sussex, England, 2000. ISBN 0-471-95333-4.

[33] B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comp.*, 10(5):1299–1319, 1998.

[34] António R. C. Paiva, Jian-Wu Xu, and José C. Príncipe. Kernel principal components are maximum entropy projections. In *Proc. Int. Conf. on Independent Component Analysis and Blind Source Separation, ICA-2006*, pages 846–853, Charleston, SC, March 2006. doi: 10.1007/11679363_105.

[35] J. O. Ramsay and B. W. Silverman. *Functional Data Analysis*. Springer-Verlag, 1997. ISBN 0-387-94956-9.

[36] Barry J. Richmond and Lance M. Optican. Temporal encoding of two-dimensional patterns by single units in primate inferior temporal cortex. II. Quantification of response waveform. *J. Neurophysiol.*, 51(1):147–161, January 1987.

[37] John W. McClurkin, Timothy J. Gawne, Lance M. Optican, and Barry J. Richmond. Lateral geniculate neurons in behaving primates. II. Encoding of visual information in the temporal shape of the response. *J. Neurophysiol.*, 66(3):794–808, September 1991.

[38] David A. Harville. *Matrix algebra from a statistician's perspective*. Springer, 1997.