Etude de noyaux de semigroupe pour objets structurés dans le cadre de l'apprentissage statistique

Marco Cuturi

Learning from Structured Objects with Semigroup Kernels

Marco Cuturi
École des Mines de Paris

ver. 18 Janvier 2006

Thèse de doctorat soutenue le 17 Novembre 2005 à l'École des Mines de Paris, 60 Blvd. Saint-Michel, en présence du jury constitué par

Olivier Bousquet , Chercheur, société Pertinence,
Stéphane Boucheron, Professeur, Université Paris VII,
Olivier Catoni, Professeur, Université Paris VI (rapporteur, président du jury),
Kenji Fukumizu, Professeur, Institute of Statistical Mathematics, Tokyo,
John Lafferty, Professeur, Carnegie-Mellon University, Pittsburgh (rapporteur),
Jean-Philippe Vert, Directeur du Centre de Bioinformatique, ENSMP.

Foreword

Silent gratitude isn't much use to anyone.
—G. Stein

Meeting and interacting with remarkable persons during these last three years was the very root of this work, making this PhD thesis a very special time for me.

First, I would like to thank Jean-Philippe for having been such an inspiring supervisor. Doing research under Jean-Philippe's guidance has brought me a lot more, as a person, than what I had ever expected from work and studies in general. In all respects, I am indebted for years to come to his generosity and his constant support.

Before even being given the chance to work with Jean-Philippe, Donald Geman, Jean-Pierre Nadal and Jérémie Jakubowicz all gave me enlightening pieces of advice to understand better academia. I would like to thank them for giving me at that time the confidence to start this PhD.

During the course of this thesis, I have been given several opportunities to work in Japan. This would not have been conceivable without the early support of Shun-ichi Amari and the opportunity he gave me to spend a summer internship in his laboratory at Riken BSI. As a graduate student just entering the world of academic research, I could not have dreamed of a more impressive and charismatic way to discover Japan than meeting Professor Amari.

I was given a few months later the chance to work in the Institute of Statistical Mathematics, Tokyo, for a year. I owe a lot in this sense to the spontaneous interest of Kunio Tanabe in my work. I am also very grateful to the Japanese Society for Promoting Science (JSPS) for financing my stay in Japan.

During this year at ISM, I had the chance to work under Kenji's enlightening guidance. Kenji's generosity and support helped me produce significant parts of the work presented here.

While in Japan, I was constantly given a very warm welcome from all researchers I met, and I would like to thank specially in that respect Tatsuya Akutsu and his staff at Kyoto University as well as Kenta Nakai and his lab at Tokyo University. Stéphane Sénécal and Momoko Ono gave me a wonderful welcome at

ISM, soon to be followed by the kindness of Tomoko Matsui, Tomoyuki Higuchi and Shiro Ikeda. I also have in mind the great times I had with fellow graduate students Hiroto Saigo, Oystein Birkenes and Dukka Bahadur

In conferences and meetings, I was also able to share thoughts with Manfred Opper, Matthias Seeger, Massimiliano Pontil, Lauren Zwald, Gilles Blanchard and Régis Vert who have all inspired and fueled part of the work presented here. I am most thankful to Francis Bach, Koji Tsuda, Klaus-Robert Müller and Alexandre d'Aspremont for giving me important insights on my work. Spending long hours discussing with Arnaud Doucet and Alvaro Cassinelli about science and more rank among the most enjoyable times I have had these last years.

I would also like to thank the whole bioinformatics team at ENSMP for the excellent atmosphere in the unit, notably my coworkers Martial Hue, Pierre Mahé and Yoshihiro Yamanishi. I am most thankful to Ecole des Mines de Paris, notably Jean-Paul Chilès, Nathalie Dietrich and Isabelle Schmitt in Fontainebleau, for their help and trust during these three years.

Sharing ideas with other friends, Jérémie Jakubowicz, Simon Cauchemez, Luc Foubert and Mustapha Boukraa, was also a great way to benefit from their experience as grad-students. All my thanks to Xavier Dupré who was always there to discuss my work and help me on code related topics.

I am also very thankful to Olivier Bousquet and Stéphane Boucheron for accepting to take part in my defence's jury, and specially grateful to Olivier Catoni and John Lafferty for taking the time to carefully review this work, as well as for being constant sources of inspiration through their work.

Finally I would like to thank my grand-mother and my uncle, my family and specially my parents, for having given me literally access to so many wonderful professional opportunities, in France and elsewhere.

Résumé

Les méthodes à noyaux désignent une famille récente d'outils d'analyse de données, pouvant être utilisés dans une grande variété de tâches classiques comme la classification ou la régression. Ces outils s'appuient principalement sur le choix a priori d'une fonction de similarité entre paires d'objets traités, communément appelée "noyau" en apprentissage statistique et analyse fonctionnelle. Ces méthodes ont récemment gagné en popularité auprès des praticiens par leur simplicité d'utilisation et leur performance.

Le choix d'un noyau adapté à la tâche traitée demeure néanmoins un problème épineux dans la pratique, et nous proposons dans cette thèse plusieurs noyaux génériques pour manipuler des objets structurés, tels que les séquences, les graphes ou les images. L'essentiel de notre contribution repose sur la proposition et l'étude de différents noyaux pour nuages de points ou histogrammes, et plus généralement de noyaux sur mesures positives. Ces approches sont principalement axées sur l'utilisation de propriétés algébriques des ensembles contenant les objets considérés, et nous faisons ainsi appel pour une large part à la théorie des fonctions harmoniques sur semigroupes. Nous utilisons également la théorie des espaces de Hilbert à noyau reproduisant dans lesquels sont plongées ces mesures, des éléments d'analyse convexe ainsi que plusieurs descripteurs de ces mesures utilisés en statistiques ou en théorie de l'information, comme leur variance ou leur entropie. En considérant tout objet structuré comme un ensemble de composants, à l'image d'une séquence transformée en un ensemble de sous-séquences ou d'images en ensembles de pixels, nous utilisons ces noyaux sur des données issues principalement de la bioinformatique et de l'analyse d'images, en les couplant notamment avec des méthodes discriminantes comme les machines à vecteurs de support.

Nous terminons ce mémoire sur une extension de ce cadre, en considérons non plus chaque objet comme un seul nuage de point, mais plutôt comme une suite de nuages emboîtés selon un ensemble d'évènements hierarchisés, et aboutissons à travers cette approche à une famille de noyaux de multirésolution sur objets structurés.

Abstract

Kernel methods refer to a new family of data analysis tools which may be used in standardized learning contexts, such as classification or regression. Such tools are grounded on an *a priori* similarity measure between the objects to be handled,

which have been named "kernels" in the statistical learning and functional analysis literature. The simplicity of kernel methods comes from the fact that, given a learning task, such methods only require the definition of a kernel to compare the objects to yield practical results.

The problem of selecting the right kernel for a task is nonetheless tricky, notably when the objects have complex structures. We propose in this work various families of generic kernels for composite objects, such as strings, graphs or images. The kernels that we obtain are tailored to compare clouds of points, histograms or more generally positive measures. Our approach is mainly motivated by algebraic considerations on the sets of interests, which is why we make frequent use of the theory of harmonic functions on semigroups in this work. The theoretical justification for such kernels is further grounded on the use of reproducing kernel Hilbert spaces, in which the measures are embedded, along with elements of convex analysis and descriptors of the measures used in statistics and information theory, such as variance and entropy. By mapping any structured object to a cloud of components, e.g., taking a string and turning it into a cloud or a histogram of substrings, we apply these kernels on composite objects coupled with discriminative methods, such as the support vector machine, to address classification problems encountered in bioinformatics or image analysis.

We extend this framework in the end of the thesis to propose a different viewpoint where objects are no longer seen as clouds of points but rather as nested clouds, where each cloud is labelled according to a set of events endowed with a hierarchy. We show how to benefit from such a description to apply a multiresolution comparison scheme between the objects.

Contents

Chapter 1

Introduction

Résumé

Les champs d'application et la portée théorique des méthodes à noyaux se sont considérablement étoffés ces dix dernières années; ce chapitre se propose de dresser un panorama général de ces outils qui sera un préalable à la lecture de cette thèse. Nous insistons en tout début de chapitre sur ce qui différencie l'approche par noyaux, basée sur le choix d'une mesure de similarité entre objets traités, des approches paramétriques qui nécéssitent la définition d'un modèle statistique vraisemblable pour les données étudiées. Nous poursuivons cet exposé en section 1.1 avec une présentation des différentes interprétations données à la notion de noyau défini positif en statistique mathématique (1.1.1) pour introduire par la suite (1.1.2) les différentes machines qui peuvent être directement utilisées avec ces noyaux pour mener à bien des tâches d'apprentissage machine. La section 1.2 précise davantage le cadre de cette thèse qui est de définir des noyaux sur objets structurés au travers d'outils empruntés à la modélisation statistique et à la géométrie de l'information, en faisant un point sur les contributions passées dans ce domaine. Les contributions de cette thése sont plus spécifiquement détaillées dans la Section 1.3.

Industries, public institutions and academia from all fields are drowning under data, stored in data warehouses that have now become an inexpensive and integral part of larger information systems. Computational means have expanded in such a way that massive parallel clusters are now an affordable commodity for most laboratories and small companies. This abundance of measurements matched with cheap computational power is ripe for statistics and machine learning to address crucial problems, from most fields of science and social sciences alike.

From a historic viewpoint in the trend of statistics, this situation confronts more than ever statisticians with real-life problems: ill-conceived information systems and slow computers are no longer an excuse, while practitioners now expect data analysis tools to be efficient with little if no prior tuning. The demand for such algorithms is but expanding, aimed at superseding human intelligence for repetitive tasks or even surpass it for large-scale studies that involve monitoring and extracting knowledge out of millions of measurements.

This situation has spurred in the last decades fertile discoveries in the field. It has also raised questions to rethink statistical sciences in the digital age. Arguably, one of the most interesting of these shifts is the increasing diversity of data structures we are now faced with. Some inherently complex data types that come from real-life applications do not fit anymore in the vectorial paradigm that was once the benchmark way of considering objects. When the task on such data types can be translated into elementary subtasks that involve either regression, binary or multi-class classification, dimensionality reduction, canonical correlation analysis or clustering, a novel class of algorithms known as kernel methods have proven to be effective, if not reach state-of-the art performance on many of these problems.

The mathematical machinery of kernel methods can be traced back to the seminal presentation of reproducing kernel Hilbert spaces by Aronszajn (1950) followed by its utilization in statistics by Parzen (1962). However, most of the practical ideas that make kernels widespread in machine learning today derive from the concurrent exposition of efficient *kernel machines* – such as gaussian processes with sparse representations (Csató and Opper, 2002) or the popular support vector machine (Cortes and Vapnik, 1995) – which are direct translations of the principles guiding statistical learning (Cucker and Smale, 2002; Vapnik, 1998), with efficient *kernel design* schemes that go beyond the simple usage of nonlinear kernels applied on vectorial data, with pioneering work by Haussler (1999); Watkins (2000). Two prominent features of kernel methods have been often pointed out to justify their success, that is, their ability to cope in a unified framework with the complexities and multimodalities of real-life data.

Versatile framework for structured data

Structured objects such as (to cite a few) strings, 3D structures, trees and networks, time-series, histograms, images, and texts have become in an increasing number of applications the *de facto* inputs for learning algorithms. Modelling their generation is no longer sufficient and practitioners expect now to infer complex rules that directly use them. The originality of kernel methods is to address this diversity through a single approach. The kernel viewpoint starts by defining or choosing a similarity measure between pairs of objects. Hence, and no matter how complex the objects might be, dealing with a learning problem through kernels involves translating a set of n data points into a symmetric $n \times n$ similarity matrix that will be the sole input used by an adequate algorithm called a kernel machine. On the contrary, most data analysis methods inherited from parametric approaches in statistics and connectionism impose a functional class beforehand (e.g. a family of statistical models or a neural architecture), that is, either tailored to fit vectorial data – which requires a feature extraction procedure to avoid very large or noisy vectorial representations –, or tailored to fit a particular data type (hidden Markov models with strings, Markov random fields with images, etc.).

Versatile framework for multimodality and semi-supervised learning

Additionally, all previously quoted objects can often be regarded as different heterogeneous representations of a single entity, seen under diverse forms. For instance, a protein can be successively treated as an amino-acid sequence, a macro-molecule with a 3D-structure, an expression level in a DNA-chip, a node in a biological pathway or in a phylogenetic tree. The interrelations between these modalities is likely to have a determinant role in our ability to predict a response variable, and hopefully in our understanding of the underlying mechanisms at work. Kernel methods provide an elegant way of integrating multimodalities through convex kernel combinations, at an earlier stage than standard techniques which usually only do so by aggregating final decision functions. A wide range of techniques have been designed to do so through convex optimization and the use of unlabelled data (Lanckriet et al., 2004; Sindhwani et al., 2005). Kernels can thus be seen as atomic elements that focus on certain types of similarities for the objects, which can be combined to correspond better to the learning task.

Following this brief presentation, we review in the next section some of the mathematical machinery behind kernel methods, before presenting in Section 1.2 the statistical approach to design kernels on structured objects, followed by a short overview in Section 1.3 of the contributions presented in the remaining chapters of this thesis.

1.1 Nuts and Bolts of Kernel Methods

Let us start this section by providing the reader with a definition for kernels, since the term "kernel" itself is used in different branches of mathematics, from linear

algebra, density estimation to integral operators theory. Some classical kernels used in estimation theory, such as the Epanechnikov kernel[1], are not, for instance, kernels in the sense of the terminology adopted in this work. We develop in this section elementary insights on kernels, inspired by sources such as (Berlinet and Thomas-Agnan, 2003; Berg et al., 1984; Schölkopf and Smola, 2002) to which the reader may refer for a more complete exposition.

1.1.1 The Multiple Facets of Kernels in Machine Learning

Quite like mathematics in general, statistical learning also admits different definitions for kernel functions. However, they are all equivalent and ultimately boil down to the same family of mathematical objects. We focus in this section on four major approaches that are the most common in the literature. Let $\mathcal{X}$ be a non-empty set sometimes referred to as the index set, and k a symmetric real-valued[2] function on $\mathcal{X} \times \mathcal{X}$.

Positive Definite Kernels

For practitioners of kernel methods, a kernel is above all a positive definite function in the following sense:

Definition 1.1 (Real-valued Positive Definite Kernels). *A symmetric function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a positive definite (p.d.) kernel on $\mathcal{X}$ if*

$$\sum_{i,j=1}^{n} c_i c_j \, k\left(x_i, x_j\right) \geq 0 \tag{1.1}$$

holds for any $n \in \mathbb{N}, x_1, \ldots, x_n \in \mathcal{X}$ and $c_1 \ldots, c_n \in \mathbb{R}$.

Functions for which the sum in Equation (1.1) is (strictly) positive when $c \neq 0$ are sometimes referred to as positive definite functions, in contrast with functions for which this sum is only non-negative, which are termed positive *semi*-definite. We will use for convenience throughout this thesis the term positive definite for kernels that simply comply with non-negativity, and will consider indifferently positive semi-definite and positive definite functions. Most theoretical results that will be presented are indifferent to this distinction, and in numerical practice definiteness and semi-definiteness will be equivalent.

One can easily deduce from Definition 1.1 that the set of p.d. kernels is a closed, convex pointed cone[3]. Furthermore, the positive definiteness of kernel functions translates in practice into p.d. matrices that correspond to a sample of points $X = \{x_i\}_{i \in I}$ in $\mathcal{X}$, that is matrices

$$K_X = \left[k(x_i, x_j)\right]_{i,j \in I} .$$

[1] for $h > 0$, $k_h(x, y) = \frac{3}{4}\left(1 - \left(\frac{x-y}{h}\right)^2\right)^+$

[2] Kernels are complex valued in the general case, but we only consider the real case here, which is the common practice in statistics and machine learning.

[3] A set C is a cone if for any $\lambda > 0, x \in C \Rightarrow \lambda x \in C$, pointed if $x \in C, -x \in C \Rightarrow x = 0$

Elementary properties of the set of kernel functions such as its closure under pointwise and tensor products are directly inherited from well known results in Kronecker and Schur (or Hadamard) algebras of matrices (Bernstein, 2005, Chapter 7). This matrix viewpoint on kernels often challenges the functional viewpoint itself, notably in the semi-supervised setting where defining a kernel matrix on a dataset is sufficient to use kernel methods. Kernel matrices for a sample X can be obtained by applying transformations r that conserve positive definiteness to a prior Gram matrix K_X, and in that case use a matrix $r(K_X)$ directly without having to define explicit formulas for the constructed kernel on the whole space $\mathcal{X} \times \mathcal{X}$. Examples of such constructions are the computation of the diffusion kernel on elements of a graph through its Laplacian matrix (Kondor and Lafferty, 2002) or direct transformations of the kernel matrix through unlabelled data (Sindhwani et al., 2005).

Let us add that Equation (1.1) appears to practitioners as the essence of a kernel. Equation (1.1) is considered as a numerical safeguard to use an arbitrary similarity measure on $\mathcal{X} \times \mathcal{X}$ with a kernel method such as Gaussian processes or SVM's, for if this function does not comply with Equation (1.1) the convergence of such machines may not be guaranteed, since they rely on convex optimizations and matrix inversions. In the case of support vector machines however, a looser constraint of conditional positive definiteness[4] for the kernel suffices to ensure convergence of the algorithm. More recent studies show that no positive definiteness at all of the considered similarity may sometimes translate with SVM like optimization into exploitable results (Haasdonk, 2005).

The fact that the whole mathematical machinery of kernels is hidden behind the positive definiteness constraint explains part of the practical success of kernel methods. Indeed, defining a p.d. similarity measure appears for most practitioners a much easier task than defining a family of statistical models or a connectionist architecture with its adequate estimation schemes. The criticism that kernel machines are black boxes is however misled, since the functional view described below and the regularization schemes exposed in Section 1.1.2 make the kernel approach conceptually transparent.

Reproducing Kernels

Kernels are also integral part of functional analysis, since with each kernel k on $\mathcal{X}$ is associated a Hilbertian space $\mathcal{H}_k$ of real valued functions on $\mathcal{X}$.

Definition 1.2 (Reproducing Kernel). *A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a reproducing kernel of a Hilbert space $\mathcal{H}$ of real-valued functions on $\mathcal{X}$ if and only if*

$$\begin{aligned} &i)\ \forall t \in \mathcal{X},\ \ k(\cdot, t) \in \mathcal{H}; \\ &ii)\ \forall t \in \mathcal{X}, \forall f \in \mathcal{H},\ \ \langle f, k(\cdot, t)\rangle = f(t). \end{aligned}$$

[4]The definition of conditional positive definiteness (c.p.d.) is the same as that of positive definiteness, except that non-negativity of the sum in Equation (1.1) only has to be ensured when $\sum_i c_i = 0$. See (Berg et al., 1984, Section 3.2) for a review

A Hilbert space that is endowed with such a kernel is called a reproducing kernel Hilbert space (RKHS) or a proper Hilbert space. Conversely, a function on $\mathcal{X} \times \mathcal{X}$ for which such a Hilbert space $\mathcal{H}$ exists is a reproducing kernel and we usually write $\mathcal{H}_k$ for this space which is unique. It turns out that both Definitions 1.1 and 1.2 are equivalent, a result known as the Moore-Aronszajn theorem (Aronszajn, 1950). First, a reproducing kernel is p.d., since it suffices to write the expansion of Equation (1.1) to obtain the squared norm of the function $\sum_{i=1}^{n} c_i k(x_i, \cdot)$, that is

$$\sum_{i,j=1}^{n} c_i c_j \, k\left(x_i, x_j\right) = \left\| \sum_{i=1}^{n} c_i k(x_i, \cdot) \right\|_{\mathcal{H}}^2, \tag{1.2}$$

which is non-negative. To prove the opposite in a general setting, that is not limited to the case where $\mathcal{X}$ is compact which is the starting hypothesis of the Mercer representation theorem (Mercer, 1909) to be found in (Schölkopf and Smola, 2002, p.37), we refer the reader to the progressive construction of the RKHS associated with a kernel k and its index set $\mathcal{X}$ presented in (Berlinet and Thomas-Agnan, 2003, Chapter 1.3). In practice, the RKHS boils down to the completed linear span of elementary functions indexed by $\mathcal{X}$, that is

$$\mathcal{H}_k \stackrel{\text{def}}{=} \overline{\text{span}}\{k(x, \cdot), x \in \mathcal{X}\}$$

The consequences of this are striking: defining a positive definite kernel k on any set $\mathcal{X}$ suffices to inherit a Hilbert space of functions $\mathcal{H}_k$. By selecting a kernel k, we hope that the space $\mathcal{H}_k$ – though made up of linear combinations of elementary functions – may contain useful functions with low norm, just as we could hope polynomials of low degree may approximate some functions of interest on a given interval. Ideally, these functions should be used to carve structures in clouds of points, or translate efficiently into decision functions with continuous or discrete outputs.

Another crucial aspect of RKHS is the simplicity of their induced norms and dot-products which are both inherited from the reproducing kernel. The fact that this norm is easy to compute for finite expansions, as seen in Equation (1.2), will be a decisive tool to quantify the complexity of a function in $\mathcal{H}_k$, paving the way for Tikhonov regularization schemes (Tikhonov and Arsenin, 1977). Additionally, the dot-product between two functions in the RKHS can be expressed through the values taken directly by the kernel on the index set, since

$$\left\langle \sum_{i \in I} a_i k(x_i, \cdot), \sum_{j \in J} b_j k(y_j, \cdot) \right\rangle = \sum_{i \in I, j \in J} a_i b_j k(x_i, y_j).$$

The fact that in $\mathcal{H}_k$ the dot-product $\langle k(x, \cdot), k(y, \cdot) \rangle_{\mathcal{H}_k}$ is equal to $k(x, y)$ illustrates the next viewpoint, which stresses further the fact that the kernel is a dot-product rather than the fact that $k(x, \cdot)$, $x \in \mathcal{X}$ are continuous evaluation functionals on $\mathcal{H}_k$.

Feature Maps

The theorem below (Berlinet and Thomas-Agnan, 2003, p.22) gives an interpretation of kernel functions, seen as dot-products between feature representations of their arguments in a space of sequences.

Theorem 1.3. *A function k on $\mathcal{X} \times \mathcal{X}$ is a positive definite kernel if and only if there exists a set T and a mapping ϕ from $\mathcal{X}$ to $l^2(T)$, the set of real sequences $\{u_t, t \in T\}$ such that $\sum_{t\in T} |u_t|^2 < \infty$, where*

$$\forall (x, y) \in \mathcal{X} \times \mathcal{X}\,,\, k(x, y) = \sum_{t\in T} \phi\,(x)_t \phi\,(y)_t = \langle \phi(x), \phi(y) \rangle_{l^2(X)}$$

The proof is derived from the fact that for any Hilbert space (notably $\mathcal{H}_k$) there exists a space $l^2(X)$ to which it is isometric. As can be glimpsed from this sketch, the feature map viewpoint and the RKHS one are somehow redundant, since

$$x \mapsto k(x, \cdot)$$

is a feature map by itself. If the RKHS is of finite dimension, functions of the RKHS turn out to be the dual space of the Euclidian space of feature projections. Although closely connected, it is rather the feature map viewpoint than the RKHS one which actually spurred most of the initial advocation for kernel methods in machine learning, notably the SVM as presented in (Cortes and Vapnik, 1995; Schölkopf and Smola, 2002). The intuition behind kernel machines was then that they would first map all the inputs into a high-dimensional feature space, that is translate all points to their feature representation,

$$\{x_1, \cdots, x_n\} \mapsto \{\phi(x_1), \cdots, \phi(x_n)\},$$

to find a linear decision surface to separate the points in two distinct classes of interest. This interpretation actually coincided with the practical choice of using polynomial kernels[5] on vectors, for which the feature space is of finite dimension. The feature map viewpoint is also illustrated in the constructive approach in (Cuturi and Vert, 2005) presented in Chapter 2.

However, the feature map approach was progressively considered to be restrictive, since it imposes to consider first the extracted features and then compute the kernel that matches them. Yet, useful kernels obtained directly from a similarity between objects do not always translate into clear feature maps, as in (Cuturi et al., 2005) with the inverse generalized variance for measures. Kernels without explicit feature maps may also be obtained through more complex kernel manipulations as in Chapter 5. The feature map formulation, particularly advocated in the early days of SVM's, also misled some observers into thinking that the kernel mapping was but a piece of the SVM machinery. Instead, SVM should be rather seen as an efficient computational approach – among many others – deployed to select a

[5] $k(x, y) = (\langle x, y \rangle + b)^d, d \in \mathbb{N}, b \in \mathbb{R}^+$

"good" function f in the RKHS $\mathcal{H}_k$ given examples, as presented in Section 1.1.2. We review next a classical connection between reproducing kernels and stochastic processes that is pertinent when considering Gaussian processes from a machine learning viewpoint.

Covariance Kernels of Stochastic Processes

Reproducing kernels on $\mathcal{X}$ are not only equivalent to positive definite ones, they are also connected to real-valued second order stochastic processes indexed on $\mathcal{X}$. Let $(\Omega, \mathcal{A}, \rho)$ be a probability space, and write $L^2(\Omega, \mathcal{A}, \rho)$ for the corresponding space of second order random variables on Ω, which is a Hilbert space with the inner product $\langle X, Y \rangle = E_\rho[XY]$. We consider a stochastic process X_s, with s ranging over $\mathcal{X}$, for which all random variables X_s are second order. In that case we write

$$R(t, s) = E_\rho[X_t X_s]$$

for the second moment function of X. We then have that

Theorem 1.4 (Loève). *R is a second moment function of a second order stochastic process indexed by $\mathcal{X}$ if and only if R is a positive definite function.*

This correspondence between positive definite kernels and stochastic processes explains why Gaussian processes and their usage in machine learning are closely linked with kernel design, as shown in (Wahba, 1990). The interested reader may refer to (Seeger, 2004) for a shorter overview of this correspondence.

1.1.2 Using Reproducing Kernel Hilbert Spaces in Practice

From the four previous interpretations of kernels, we will mainly make use of the reproducing kernel viewpoint in this section. However, we will refer frequently to the positive definite and feature map facets of kernels throughout this thesis. We review first different methods to analyze clouds of unlabelled data and extract a structural knowledge from such clouds using functions in the RKHS. We also introduce the kernel approach to supervised problems, notably binary classification, regression, dimensionality reduction and graph inference. Most of these results wouldn't be valid however without the following theorem.

The Representer Theorem

Most estimation procedures presented in the statistical literature to perform a dimensionality reduction or infer a decision function out of sampled points rely on the optimization of a criterion which is usually carried out over a class of linear functionals. Indeed, PCA, CCA, logistic regression and least-square regression and its variants (lasso or ridge regression) all look for linear transformations of the original data points to address the learning task. When these optimizations are led instead on an infinite dimensional space of functions, namely in the RKHS $\mathcal{H}_k$, the

optimization can be performed in finite subspaces of $\mathcal{H}_k$ if the criterion only depends on the values of the target function on a finite sample of points. This result is known as the representer theorem and explains why so many linear algorithms can be "kernelized" when trained on finite datasets.

Theorem 1.5 (Representer Theorem (Kimeldorf and Wahba, 1971)). *Let $\mathcal{X}$ be a set endowed with a kernel k and $\mathcal{H}_k$ its corresponding RKHS. Let $\{x_i\}_{1\leq i\leq n}$ be a finite set of points of $\mathcal{X}$ and let $\Psi : \mathbb{R}^{n+1} \to \mathbb{R}$ be any function that is strictly increasing with respect to its last argument. Then any solution to the problem*

$$\min_{f\in\mathcal{H}_k} \Psi\left(f(x_1), \cdots, f(x_n), \|f\|_{\mathcal{H}_k}\right)$$

is in the finite dimensional subspace $\operatorname{span}\{k(x_i,\cdot), 1 \leq i \leq n\}$ *of $\mathcal{H}_k$.*

The theorem in its original form was cast in a more particular setting, where the term $\|f\|_{\mathcal{H}_k}$ would be simply added to an empirical risk as shown below. This generalized version allows us to regard both supervised and unsupervised settings from the same viewpoint, and we make constant use of this generalized version in the next sections.

Studying Unlabelled Data through Eigenfunctions in $\mathcal{H}_k$

In the unsupervised setting, the structure that underlies a set $X = \{x_i\}_{1\leq i\leq n}$ of points of $\mathcal{X}$ is the focus of study. Additionally, the quantification of the correlation between the measurements observed in a set X with that of another set of points with the same indexes $Y = \{y_i\}_{1\leq i\leq n}$ from a set $\mathcal{Y}$ can be of interest, notably when each index refers to the same underlying object cast in different modalities (Vert and Kanehisa, 2003). If both $\mathcal{X}$ and $\mathcal{Y}$ are Euclidian spaces, classical linear techniques can be applied, such as:

- Principal component analysis (PCA), which aims at defining an orthonormal base $v_1, \cdots, v_{\dim(\mathcal{X})}$ of $\mathcal{X}$ such that for $1 \leq j \leq \dim(\mathcal{X})$,

  $$v_j = \underset{v\in\mathcal{X}, \|v\|_{\mathcal{X}}=1, v\perp\{v_1,\cdots,v_{j-1}\}}{\operatorname{argmax}} \operatorname{var}_X[\langle v, x\rangle_{\mathcal{X}}],$$

  where for any function $f : \mathcal{X} \to \mathbb{R}$, $\operatorname{var}_X[f]$ denotes the empirical variance with respect to the points enumerated in X, that is $E_X[(f - E_X[f])^2]$. The set is then usually represented by only considering the projections of the points in the subspace generated by an arbitrary number of r first eigenvectors $v_1, \cdots, v_r$ supposed to capture the main features of the data.

- Canonical correlation analysis (CCA), quantifies how X and Y are related, by computing

  $$\begin{aligned}\rho(X,Y) &= \max_{\xi\in\mathcal{X},\zeta\in\mathcal{Y}} \operatorname{corr}_{X,Y}[\langle\xi, x\rangle_{\mathcal{X}}, \langle\zeta, y\rangle_{\mathcal{Y}}] \\ &= \max_{\xi\in\mathcal{X},\zeta\in\mathcal{Y}} \frac{\operatorname{cov}_{X,Y}[\langle\xi, x\rangle_{\mathcal{X}}, \langle\zeta, y\rangle_{\mathcal{Y}}]}{\sqrt{\operatorname{var}_X[\langle\xi, x\rangle_{\mathcal{X}}]\operatorname{var}_Y[\langle\zeta, y\rangle_{\mathcal{Y}}]}}\end{aligned}$$

where for two real valued functions $f : \mathcal{X} \to \mathbb{R}$ and $g : \mathcal{Y} \to \mathbb{R}$ we write

$$\mathrm{cov}_{X,Y}[f, g] = E_{X,Y}[(f - E_X[f])(g - E_Y[g])].$$

We observe that both optimizations search for vectors in $\mathcal{X}$ (and $\mathcal{Y}$ for CCA) that will be representative of the data dependencies. The kernelization of such algorithms is natural when one thinks in the same terms, but in the reproducing kernel Hilbert spaces instead. We write for convenience $\mathcal{H}_\mathcal{X}$ and $\mathcal{H}_\mathcal{Y}$ for the RKHS associated with $\mathcal{X}$ and $\mathcal{Y}$ with respective kernels $k_\mathcal{X}$ and $k_\mathcal{Y}$. Using RKHS, we are now looking for vectors – that is functions – that are directions of interest in the sense that they capture most of the variance in the data now seen as a cloud of functions. The fact that we select these functions based on a criterion which satisfies the requirements of the representer theorem leads in practice to computations led on finite subspaces of $\mathcal{H}_k$ that are analogous to the standard case. The two previous optimizations become

$$f_j = \underset{f \in \mathcal{H}_\mathcal{X}, \|f\|_{\mathcal{H}_\mathcal{X}} = 1, f \perp \{f_1, \cdots, f_{j-1}\}}{\mathrm{argmax}} \mathrm{var}_X[\langle f, k_\mathcal{X}(x, \cdot)\rangle_{\mathcal{H}_\mathcal{X}}],$$

for $1 \leq j \leq n$ this time and

$$\rho(X, Y) = \max_{f \in \mathcal{H}_\mathcal{X}, g \in \mathcal{H}_\mathcal{Y}} \frac{\mathrm{cov}_{X,Y}[\langle f, k_\mathcal{X}(x, \cdot)\rangle_{\mathcal{H}_\mathcal{X}}, \langle g, k_\mathcal{Y}(y, \cdot)\rangle_{\mathcal{H}_\mathcal{X}}]}{\sqrt{\mathrm{var}_X[\langle f, k_\mathcal{X}(x, \cdot)\rangle_{\mathcal{H}_\mathcal{X}}] \, \mathrm{var}_Y[\langle g, k_\mathcal{Y}(y, \cdot)\rangle_{\mathcal{H}_\mathcal{Y}}]}}. \tag{1.3}$$

The first problem has been termed kernel-PCA in the seminal work of Schölkopf et al. (1998) and boils down to a simple singular value decomposition of the kernel matrix K_X. Note that kernelizing weighted PCA is not as straightforward and can be only carried out through a generalized eigendecomposition, as briefly formulated in (Cuturi and Vert, 2005). Similarly to PCA, practitioners are usually interested in the first eigenfunctions of a set of points, and little attention is usually paid to eigenfunctions with lower eigenvalues. The reader may look at Figures 3.2, 3.4 and 3.5 for a comparison between standard PCA and kernel-PCA for clouds of points in $[0, 1]^2$ and may refer to (Schölkopf and Smola, 2002, Section 14.2) for a more detailed review.

The second optimization, called kernel-CCA (Akaho, 2001), is ill-posed if Equation (1.3) is used directly, and requires a regularization scheme to produce satisfying results. Understanding better kernel-CCA is a topic of current research, and the reader may consult a recent overview in (Fukumizu et al., 2005). The topic of supervised dimensionality reduction, explored in (Fukumizu et al., 2004), is also linked to the kernel-CCA approach. The author look for a sparse representation of the data that will select an effective subspace for $\mathcal{X}$ and delete all the directions in $\mathcal{X}$ that are not correlated to what is observed in $\mathcal{Y}$, based on the samples X and Y. In linear terms, such a sparse representation can be described as a projection of the points of $\mathcal{X}$ into a subspace of much lower dimension while conserving the correlations observed with corresponding points in $\mathcal{Y}$. When kernelized, the approach is roughly equivalent in the RKHS, up to regularization schemes.

Kernel Regression and Classification

Suppose that we wish to infer now from what is observed in the samples X and Y a causal relation between all the points of $\mathcal{X}$ and $\mathcal{Y}$. This type of inference is usually restricted to finding a mapping f from $\mathcal{X}$ to $\mathcal{Y}$ that is consistent with the collected data and has desirable smoothness properties so that it appears as a "natural" decision function seen from a prior perspective. If $\mathcal{X}$ is Euclidian and $\mathcal{Y}$ is $\mathbb{R}$, the latter approach is a well studied field of mathematics known as approximation theory, rooted a few centuries ago in polynomial interpolation of given couples of points, and developed in statistics through spline regression (Wahba, 1990) and basis expansions (Hastie et al., 2001, Chapter 5).

Statistical learning theory starts its course when a probabilistic knowledge about the generation of the points (x, y) is assumed, and the reader may refer to (Cucker and Smale, 2002) for a valuable review. We skip its rigorous exposition, and favour intuitive arguments next. A sound guess for the learning rule f would be a function with a low empirical risk,

$$R_c^{\text{emp}}(f) \stackrel{\text{def}}{=} \frac{1}{n}\sum_{i=1}^{n} c\left(f(x_i), y_i\right),$$

quantified by a cost function $c : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+$ that penalizes wrong predictions and which is nil on the diagonal. Minimizing directly R_c^{emp} given training sets X and Y is however unlikely to give interesting functions for f. If the function class $\mathcal{F}$ from which f is selected is large, the problem becomes ill-posed in the sense that many solutions to the minimization exist, of which few will prove useful in practice. On the contrary, if the function class is too restricted, there will be no good minimizer of the empirical risk that may serve in practice. To take that tradeoff into account, and rather than constraining $\mathcal{F}$, assume that $J : \mathcal{F} \to \mathbb{R}$ is a function that quantifies the roughness of a function which is used to penalize the empirical risk,

$$R_c^{\lambda}(f) \stackrel{\text{def}}{=} \frac{1}{n}\sum_{i=1}^{n} c\left(f(x_i), y_i\right) + \lambda J(f). \tag{1.4}$$

Here $\lambda > 0$ balances the tradeoff between two desired properties for the function f, that is a good fit for the data at hand and a smoothness as measured by J. This formulation is used in most regression and classification settings to select a good function f as

$$\hat{f} = \underset{f \in \mathcal{F}}{\operatorname{argmin}} R_c^{\lambda}.$$

We recover through this viewpoint a large variety of methods, notably when the penalization is directly related to the norm of the function in a RKHS:

- When $\mathcal{X}$ is Euclidian and $\mathcal{Y} = \mathbb{R}$, $\mathcal{F} = \mathcal{X}^*$, the dual of $\mathcal{X}$ and $c(f(x), y) = (y - f(x))^2$, minimizing R_c^{λ} is known as least-square regression when $\lambda = 0$, ridge regression when $\lambda > 0$ and J is the Euclidian 2-norm, and the lasso when $\lambda > 0$ and J is the 1-norm.

- When $\mathcal{X} = [0,1]$, $\mathcal{Y} = \mathbb{R}$, $\mathcal{F}$ is the space of m-times differentiable functions on $[0,1]$ and $J = \int_{[0,1]} \left(f^{(m)}(t)\right)^2 dt$, we obtain regression by natural splines of order m. This setting actually corresponds to the usage of thin-base splines which can also be regarded as a RKHS type method (Wahba, 1990), see (Girosi et al., 1995, Table 3) for other examples.

- When $\mathcal{X}$ is an arbitrary set endowed with a kernel k and $\mathcal{Y} = \{-1, 1\}$, $\mathcal{F} = \mathcal{H}_k$, $J = \|\cdot\|_{\mathcal{H}_k}$ and the hinge loss $c(f(x), y) = (1 - yf(x))^+$ is used, we obtain the support vector hyperplane. Replacing the cost function by the loss $c(f(x), y) = \ln(1 + e^{-yf(x)})$, we obtain logistic regression, which, when used along with non-linear kernels has been coined down kernel logistic regression (Zhu and Hastie, 2002).

- When $\mathcal{X}$ is an arbitrary set endowed with a kernel k and $\mathcal{Y} = \mathbb{R}$, $\mathcal{F} = \mathcal{H}_k$, $J = \|\cdot\|_{\mathcal{H}_k}$ and $c(f(x), y) = (|y - f(x)| - \varepsilon)^+$, the ε-insensitive loss function, we obtain the support vector regression.

Finally, we quote another example of RKHS type regularization. In the context of supervised graph inference, Vert and Yamanishi (2005) consider a set of connected points $\{x_i\}_{1 \leq i \leq n}$ whose connections are summarized in the combinatorial Laplacian matrix L of the graph, that is for $i \neq j$, $L_{i,j} = -1$ if i and j are connected and 0 otherwise, and $L_{i,i} = -\sum_{j \neq i} L_{i,j}$. The authors look for a sequence of functions $\{f_i\}_{1 \leq i \leq d}$ of a RKHS $\mathcal{H}_k$ to map the original points in $\mathbb{R}^d$, and hope to recover the structure of the original graph through this representation. Namely, the projection is optimized such that the points, once projected in $\mathbb{R}^d$, will have graph interactions in that metric (that is by linking all nearest neighbours up to some distance threshold) that will be consistent with the original interactions. This leads to successive minimizations that may recall those performed in kernel-PCA, although very different in nature through the regularization term in λ:

$$f_j = \underset{f \in \mathcal{H}_k, f \perp \{f_1, \cdots, f_{j-1}\}}{\operatorname{argmax}} \frac{f_X^\top L f_X + \lambda \|f\|_{\mathcal{H}_k}}{f_X^\top f_X}.$$

The supervised aspect is here included in the vectors f_X where

$$f_X \stackrel{\text{def}}{=} (f(x_1), \cdots, f(x_n))^\top.$$

The term $f_X^\top L f_X$ above can be interpreted as cost functions with respect to the observable graph L, that will favour values for f that are close for two connected nodes.

All these regularization schemes and the practical ways to solve the corresponding optimizations have fostered considerable research. The subject of this thesis is not, however, related to this part of the kernel machinery. Instead, we turn more specifically to the natural issues that arise from the use of kernels, without considering the issue of the search for interesting functions in $\mathcal{H}_k$.

1.1.3 Selecting and Designing the Kernel

In supervised learning, where performance can be quantified by the error of the algorithm on a test set of points, most practitioners are very much aware of the sensitivity of kernel methods to the kernel that is selected. When a single family of parameterizable kernels is used (e.g., Gaussian kernel), this involves choosing a parameter range that will yield a good generalization; when many kernels are available for the task this involves picking the right kernel with an adequate parameter, or a good combination of such kernels.

Hence, if the kernel – that is the RKHS of candidate functions chosen for the task – is not conveniently selected, the low-performance of kernel methods such as the SVM cannot be avoided and should not be attributed to the SVM optimization itself, as has sometimes been reported (Hastie et al., 2001, Section 12.3.4), but rather to the inadequateness of the kernel. It is thus no surprise that, after the early successes of SVM as a kernel machine, followed by faster implementations (Platt, 1999), the focus progressively shifted on designing and combining efficient kernel representations for data, with important concepts laid out by Haussler (1999) and pioneering applications (Watkins, 2000; Jaakkola and Haussler, 1999; Joachims, 1997; Brown et al., 2000; Pontil and Verri, 1998).

Tuning and Combining Kernels

Other than through cross-validation, the topic of selecting the parameters through different error estimates has been addressed in a variety of papers, notably (Chapelle et al., 2002). The subject itself can be linked to the regularization penalty used in most learning schemes, and is still the subject of current research (Vert and Vert, 2005). A more general avenue to have kernels that are suited to the task is to combine a few of them. As recalled in Section 1.1.1, the set of kernels is a convex cone of functions which is further closed under pointwise product. Therefore, any polynomial on the set of kernels with positive coefficients is itself a kernel. For computational reasons, linear combinations rather than multiplicative ones have been considered so far, that is given a family $k_1, \cdots, k_d$ of d kernels, consider kernels of the form

$$k_\alpha = \sum_{i=1}^{d} \alpha_i k_i \ , \ \alpha_i > 0.$$

Although surprisingly good results can be obtained from a kernel that is just the mean of other kernels (Vert and Yamanishi, 2005), various schemes to optimize the weights α have been proposed, notably through semi-definite programming (Lanckriet et al., 2004; Bach et al., 2004). Going one step higher in abstraction, Ong et al. (2005) propose a framework where kernels themselves are selected according to a prior knowledge, which translates in practice to a supplementary regularization term for the regularized empirical risk,

$$R_c^{\lambda,\mu}(f,k) = \frac{1}{n}\sum_{i=1}^{n} c\left(f(x_i), y_i\right) + \lambda ||f||_{H_k}^2 + \mu ||k||_{\underline{H}}^2.$$

where $\|.\|_{\underline{H}}$ stands for the norm of k in a "hyper-RKHS" defined by a kernel... on kernel functions. Finally, Tsuda and Noble (2004) propose to estimate directly a kernel matrix on selected points (and not the whole kernel function itself) by choosing the matrix with maximal von Neumann entropy[6] from a set of matrices whose coefficients satisfy convex constraints derived from empirical observations. This criterion is set so that the obtained matrix has a large rank which may reveal interesting features, a work further formalized and generalized with a larger family of divergences in (Tsuda et al., 2005). Alternatively Tsuda et al. (2003) also propose a framework to fill in kernel matrices with missing values. This problem arises when objects of interest may have different modalities, but the measurements about some of the modalities may be incomplete. Proteins for instance have well known amino-acid sequences, while their 3D structure still remains difficult to measure. The authors propose to fill in these missing values by using auxiliary matrices that may come from other modalities through a series of projections in successive matrix spaces.

Designing Kernels

The other method to define kernels is to construct them explicitly, inspired by available knowledge on the objects. Kernels for vectors have been long studied in the context of spline regression (Wahba, 1990) and in spatial statistics, through the seminal works of Matheron (1965). For structured objects however the task was hardly studied in practice before (Haussler, 1999), although important theoretical foundations were laid out well before by (Berg et al., 1984). Defining such kernels is still a topic of open research, as it involves making arbitrary choices motivated by applications. Consider for instance the alphabet $\{A, B, C\}$ and the following strings

$$m_1 = AAAA,\ m_2 = BBBB,\ m_3 = AAAAAAAAAAAA,$$
$$m_4 = ABAB,\ m_5 = BCBC,\ m_6 = BBAA.$$

From m_1 to m_3, and from m_4 to m_6, one can think of many different situations where m_1 (resp. m_4) should be more similar to m_3 (resp. m_5) than to m_2 (resp. m_6), and vice-versa. This arbitrary has fostered considerable research, so that most common concepts of similarity available for objects could be accordingly translated into p.d. kernels, positive definiteness being often the stumbling block. Applications where such non-trivial kernels have significantly improved performance include bioinformatics (Schölkopf et al., 2004) and text categorization (Joachims, 2002). In string, speech and image analysis, the similarities that have inspired kernel design involve in large parts the minimization of a criterion, notably the dynamic time warping alignment (Shimodaira et al., 2002), the edit-distance, or Smith-Waterman score (Vert et al., 2004), and the tangent distance (Haasdonk and Keysers, 2002) respectively. Each of these criterions is defined through a set S

[6] The von Neumann entropy of a p.d. matrix is the entropy of its eigenvalues normalized such that they sum to 1.

of bijective transformations for the objects, where each transformation $s \in S$ is weighted by a nonnegative cost $c(s)$. Given two objects x and y, the computation of such criterions involves finding the sequence of transformations of S that will transform x into y[7] with minimum total cost, where the criterion is set to be the cost of this optimal path. The computation of this "shortest" path from x to y is usually carried out through dynamic programming algorithms. However, the exact value of the criterion, that is the total cost of the optimal path, does not translate easily into p.d. kernels – as most operations involving min's and max's – unless clever approximations are carried out as in (Vert et al., 2004). More generally and when a similarity measure inspired by common practice cannot be easily turned into a p.d. kernel, another procedure known as the empirical kernel map (Schölkopf et al., 2002) can be used at a great computational cost but with good results on experiments so far (Liao and Noble, 2002). Given a set of non-necessarily labelled points $\{x_i\}_{i\in I}$, the empirical feature map

$$\phi : x \mapsto \{k(x_i, x)\}_{i\in I}$$

is used to define a kernel on x, y through their representations $\phi(x)$ and $\phi(y)$, with a simple dot-product $\phi(x) \cdot \phi(y)$ usually applied to the vectors. We leave for the next section an important category of kernels that is grounded on statistical modelling.

1.2 Blending Discriminative and Generative Approaches with Kernels

For a wide variety of objects, notably sequences, researchers in the field of kernel methods quickly turned to existing statistical generative models, long tailored for these objects, to extract features of interest.

1.2.1 Statistical Models as Feature Extractors

Jaakkola and Haussler (1999) first thought of using generative models to build kernels that would provide in turn the inputs for discriminative machines. In the case of sequences for instance, the hidden Markov model (HMM), that was known to capture efficiently the behaviour of amino-acid sequences, quickly turned out to be an efficient feature extractor. The authors did so by defining for each sequence a vector of features that would be derived from an estimated HMM model, namely the Fisher score. Given a measurable space $(\mathcal{X}, \mathcal{B}, \nu)$ and a parametric family of absolutely continuous measures of $\mathcal{X}$ represented by their densities $\{p_\theta, \theta \in \Theta \subset \mathbb{R}^d\}$, the Fisher kernel between two elements x, y of $\mathcal{X}$ is

$$k_{\hat{\theta}}(x, y) = \left(\frac{\partial \ln p_\theta(x)}{\partial \theta}|_{\hat{\theta}}\right)^\top J_{\hat{\theta}}^{-1} \left(\frac{\partial \ln p_\theta(y)}{\partial \theta}|_{\hat{\theta}}\right),$$

where $\hat{\theta}$ is a parameter selected beforehand to match the whole training set, and $J_{\hat{\theta}}$ is the Fisher information matrix computed in $\hat{\theta}$. The statistical model not only

[7] or equivalently y into x if we assume that the cost of a transformation and its inverse are the same.

acts as a feature extractor through the score vectors, but also defines the metric to use with these vectors through $J_{\hat{\theta}}$.

The Fisher kernel was perceived by the community as a promising bridge between generative and discriminative approaches, which fostered a lot of research and extensions in this sense, notably in (Tsuda et al., 2002a; Smith and Gales, 2002). The motivation behind these contributions was to overcome the limiting assumption that the parameter $\hat{\theta}$ on which the score vectors are evaluated is unique and fits the whole set of points at hand. Rather, Tsuda et al. (2002a) and Smith and Gales (2002) proposed simultaneously to incorporate in the context of binary classification two parameters $\hat{\theta}_1$ and $\hat{\theta}_2$ for each class respectively, and consider the score vector of the likelihood ratio between the two classes evaluated in x,

$$\phi_{\hat{\theta}_1,\hat{\theta}_2} : x \mapsto \left(\frac{\partial \ln \frac{p_{\theta_1}(x)}{p_{\theta_2}(x)}}{\partial \vartheta} \Big|_{\vartheta=(\hat{\theta}_1,\hat{\theta}_2)} \right)$$

where $\vartheta = (\theta_1, \theta_2)$ is now in Θ^2, to propose the kernel

$$(x, y) \mapsto \phi_{\hat{\theta}_1,\hat{\theta}_2}(x)^\top \phi_{\hat{\theta}_1,\hat{\theta}_2}(y).$$

The Fisher kernel was the source of further theoretical work in (Tsuda et al., 2004) where its statistical consistency was studied. It was also related to a wider class of kernels coined down as mutual information kernels by Seeger (2002). Starting also from a set of distributions $\{p_\theta, \theta \in \Theta\}$ where Θ is measurable, and from a given prior $\omega \in L_2(\Theta)$, the mutual information kernel k_ω between two elements x and y is defined as

$$k_\omega(x, y) = \int_\Theta p_\theta(x) p_\theta(y)\, \omega(d\theta).$$

As noted in (Seeger, 2002), the Fisher kernel can be regarded as a maximum *a posteriori* approximation of the mutual information kernel, by setting the prior ω to the multivariate Gaussian density $\mathcal{N}(\hat{\theta}, J_{\hat{\theta}}^{-1})$, following the approximation of Laplace's method. We proposed in (Cuturi and Vert, 2005), reported in Chapter 2, a kernel on strings that is based on the mutual information approach without using such an approximation. In this work the distributions $\{p_\theta, \theta \in \Theta\}$ are Markov chain densities with finite depths, weighted by a prior ω set to convenient conjugate priors, namely a combination of branching process priors for the structure of the chain and mixtures of Dirichlet priors for the transition parameters. This setting yields closed computational formulas for the kernel, benefitting from similar computations in universal coding (Willems et al., 1995; Catoni, 2004).

In the framework of sequence analysis first (Tsuda et al., 2002b), and then in comparisons of graphs (Kashima et al., 2003), further attention was given to latent variable models to define kernels in a way that also generalized the Fisher kernel. In a latent variable model, the probability of emission of an element x is conditioned by an unobserved latent variable $s \in \mathcal{S}$, where $\mathcal{S}$ is a finite space of possible states. When a string is considered under the light of a hidden Markov model, to its chain $x = x_1 \cdots x_n$ of letters is associated a similar sequence $s = s_1 \cdots s_n$ of

states that is not usually observed. When the sequence of states s is known, the probability of x under such a model is then determined by the marginal probabilities $p(x_i|s_i)$. Building adequate transition structures for the emitting states, and their corresponding emission probabilities is one of the goals of HMM estimations. The marginalized kernel assumes that this sequence is not known for objects x and y, but it performs, given an available structure of states, an averaging

$$k(x,y) = \sum_{s\in\mathcal{S}} \sum_{s'\in\mathcal{S}} p(s|x)\, p(s'|y)\, \kappa\left((x,s),(y,s')\right)$$

of arbitrary kernel evaluations κ weighted by posterior probabilities which are estimated from data. In this setting, κ can be any arbitrary kernel on $\mathcal{X}\times\mathcal{S}$. For particular choices of κ the kernel can be computed in closed form, both on sequences and graphs (Mahé et al., 2004).

Finally, and rather than marginalizing the probabilities of the objects through a prior belief on the parameters, or the opposite by considering marginalized likelihoods, a different approach takes the direction of comparing two objects by considering directly the parameters that fits them better, that is, map first

$$\mathcal{X}^2 \ni (x,y) \mapsto (\hat{\theta}_x, \hat{\theta}_y) \in \Theta^2,$$

through maximum likelihood estimation for instance, and then compare x and y through a kernel k_Θ on Θ,

$$k(x,y) = k_\Theta(\hat{\theta}_x, \hat{\theta}_y).$$

Notable examples of this approach include the survey of Jebara et al. (2004) which presents the family of kernels

$$k_\beta(x,y) = \int_{\mathcal{X}} p_{\hat{\theta}_x}(z)^\beta p_{\hat{\theta}_y}(z)^\beta dz$$

for $\beta > 0$, the case $\beta = \frac{1}{2}$ being the well known Bhattacharrya affinity between densities. The authors review a large family of statistical models for which these kernels can be computed in closed form, ranging from graphical models, Gaussian multivariate densities and multinomials to hidden Markov models. Aiming also at computing functions k_Θ of interest, Lafferty and Lebanon (2005) propose to follow Kondor and Lafferty (2002) and use diffusion processes to define kernels. To do so they express solutions for the heat equation in the Riemannian manifold induced by the Fisher metric of the considered statistical models, under the light of information geometry as defined by Amari and Nagaoka (2001). They derive *information diffusion* kernels out of such solutions which, when specialized to multinomials, that is elements of the simplex[8], boil down to kernels of the form

$$k_{\Sigma_d}(\theta,\theta') = e^{-\frac{1}{t}\arccos^2(\sqrt{\theta\cdot\theta'})}, \tag{1.5}$$

where $t > 0$ is the diffusion parameter. Note that the squared arc-cosine in Equation (1.5) is the squared geodesic distance between θ and θ' seen as elements from

[8] writing Σ_d for the canonical simplex of dimension d, i.e., $\Sigma_d = \{\xi = (\xi_i)_{1\leq i\leq d} : \xi_i \geq 0, \sum \xi_i = 1\}$.

the unit sphere (that is when each θ_i is mapped to $\sqrt{\theta_i}$). Based on the seminal work of Schoenberg (1942), Zhang et al. (2005) rather advocate the direct use of the geodesic distance in the context of text classification which is also addressed by Lafferty and Lebanon (2005). They prove that the geodesic distance is a negative definite kernel[9] on the whole sphere, while its square used in Equation (1.5) does not seem to be in numerical practice. If the points are restricted to lie in the positive orthant, which is the case for multinomials, both approaches yield however positive definite kernels.

1.2.2 Nonparametric Kernels on Measures

When the considered objects for the task are composite, that is built by the aggregation of more atomic elements in a set $\mathcal{X}$, and when these elements can be assumed to have been generated independently according to a probability measure in $M_+^1(\mathcal{X})$, one can represent each structured object by a cloud of weighted points on a measurable space $(\mathcal{X}, \mathcal{B}, \nu)$, that is a molecular measure[10], or a histogram when $\mathcal{X}$ is finite. An early reference for this approach in a discriminative framework was brought forward by Chapelle et al. (1999), and whose advocated kernel on Σ_d,

$$k_{a,b}(\theta, \theta') = e^{-\sum_{i=1}^{d} |\theta_i^a - \theta_i'^a|^b}, \quad a > 0, b > 0,$$

has shown good results in most of our applications, as illustrated in Chapter 5. The general approach of considering objects as clouds of points was spurred by a few practical examples:

- **sequences** have been typically considered as sets of subsequences, called n-grams in text analysis (Lodhi et al., 2002) or k-mers in bioinformatics (Leslie et al., 2002, 2003; Rätsch and Sonnenburg, 2004).
- **images** can be decomposed into histograms of colors (Chapelle et al., 1999), but also sets of salient points (Eichhorn and Chapelle, 2004).
- **bags of words** representations for **texts** have long been advocated in the framework of natural language processing (Salton, 1989), and notably used for state-of-the-art kernel methods in the field (Joachims, 2002).
- **graphs and 3D structures** and their decomposition as sets of paths have been used in (Kashima et al., 2003), notably for the analysis of molecules by Ralaivola et al. (2005) and Mahé et al. (2004).

The family of kernels designed on these nonnegative counters of components encompasses thus the family of kernels on multinomials, histograms, clouds of points and densities. Such a family has been coined down more generally as "kernels on measures", and we adopt this terminology from now on. By designing general families of kernels on measures, we aim at having a generic toolbox of kernels

[9] a negative definite kernel is the negative of a c.p.d. kernel

[10] A molecular measure, also called an atomic measure in the literature.

that can be used efficiently to compare structured objects decomposed into sets components.

The setting adopted in (Chapelle et al., 1999) does not rely on a statistical model, but rather on a simple notion of proximity between the empirical measures. Arguably, this viewpoint might be better suited for kernel methods. Indeed, statistical generative models have long been designed to be interpretable and fit existing data, that is to provide an interpretation for the underlying natural phenomenon, and provide an efficient framework to estimate their parameters. Putting the focus on a good estimation of $\hat{\theta}_x$ may not be of any use at all, since $\hat{\theta}_x$ it but computed to compare it directly with $\hat{\theta}_y$, regardless of the accuracy of both estimators in representing the data. This question is raised in the experimental account in Section 2.6 where a very simple model, namely Markov chains of order 2, suffices to define an efficient kernel on protein sequences. Coupled with SVM's for classification, this kernel yields far better results than the estimation of complex HMM models for each considered family used then as a plug-in rule, which can be observed in (Liao and Noble, 2002).

These remarks spurred the definition of nonparametric kernels that would use directly the geometry of $M_+^b(\mathcal{X})$ through adequate metrics. A survey of such kernels derived from a family of metrics on $\mathbb{R}^+$ is presented in (Hein and Bousquet, 2005). The authors consider the family of distances

$$d^2_{\alpha,\beta} = \frac{2^\beta (x^\alpha + y^\alpha)^{\frac{1}{\alpha}} + 2^\alpha (x^\beta + y^\beta)^{\frac{1}{\beta}}}{2^{\frac{1}{\alpha}} - 2^{\frac{1}{\beta}}},$$

presented in (Fuglede and Topsøe, 2004), for $\alpha \in [1, \infty]$ and $\beta \in [\frac{1}{2}, \alpha]$ or $\beta \in [-\infty, -1]$. Given two absolutely continuous measures μ, μ' with densities p, p' w.r.t. ν, the authors propose to integrate pointwise the distances between the densities, defining n.d. kernels

$$d^2_{\alpha,\beta}(\mu, \mu') = \int_{\mathcal{X}} d^2_{\alpha,\beta}(p(x), p'(x)\nu(dx), \tag{1.6}$$

to characterize the following family of p.d. kernels[11],

$$k_{1,-1}(\mu, \mu') = \int_{\mathcal{X}} \frac{p(x)p'(x)}{p(x) + p'(x)} \nu(dx), \quad k_{\frac{1}{2},1}(\mu, \mu') = \int_{\mathcal{X}} \sqrt{p(x)p'(x)}\nu(dx),$$
$$k_{1,1}(\mu, \mu') = \frac{-2}{\ln 2} \int_{\mathcal{X}} p(x) \ln \frac{p(x)}{p(x) + p'(x)} + p'(x) \ln \frac{p'(x)}{p(x) + p'(x)} \nu(dx),$$
$$k_{\infty,1}(\mu, \mu') = \int_{\mathcal{X}} \min(p(x), q(x))\nu(dx),$$

which correspond respectively to the symmetric χ^2, Hellinger, Jensen-Shannon and total variation metrics. All these metrics are invariant under a change of the base measure ν.

[11] through the equivalence that if $\phi(x, y) = \psi(x, x_0) + \psi(y, x_0) - \psi(x, y) - \psi(x_0, x_0)$, ϕ is p.d. if and only if ψ is n.d. and choosing in this case $x_0 = 0$, see (Berg et al., 1984, p. 74)

In parallel to such non-parametric approaches, further research was given to the incorporation in these kernels of a prior knowledge on the component space $\mathcal{X}$ itself. When this knowledge is a kernel on the components, "kernelized" estimates of kernels on measures can be computed. An early approach which is numerically non-satisfactory because it leads to diagonally dominant kernels is presented in (Wolf and Shashua, 2003), to compare two clouds of points of equal size using a kernel on the space where the points are taken from. Beyond a few points however, the kernel has negligible values which prevents using it in most problems.

The issue when kernelizing a kernel on measures is well illustrated in the differences between two papers from the same authors, (Jebara et al., 2004) and (Kondor and Jebara, 2003). In the first paper, the authors review a large family of models for which the Bhattacharrya family of kernels can be computed directly. In the latter, which can be seen as a seminal reference to our knowledge, the authors recall the expression of the Bhattacharrya affinity for sets of points seen as empirical samples from a Gaussian law; the affinity, which only depends on the empirical means and variances of the respective clouds, is then kernelized. This modification of the original algorithm is possible thanks to analogies that can be drawn between the two first moments of a measure and the kernel matrix of the elements of its support. The kernelization which is proposed in (Kondor and Jebara, 2003) is however complex and demands extensive calculations. Hein and Bousquet (2005) also propose such a kernelization, and coin down two of these kernelized estimates *structural kernels*. They incorporate in the integration of Equation (1.6) an additional kernel term through two forms:

$$k_I(\mu,\mu') = \int_{\mathcal{X}}\int_{\mathcal{X}} k(x,y)\hat{k}(p(x),p'(y)\nu(dx)\nu(dy).$$
$$k_{II}(\mu,\mu') = \int_{\mathcal{X}}\int_{\mathcal{X}} s(x,y)\tilde{k}(p(x),p(y))\tilde{k}(p'(x),p'(y))\nu(dx)\nu(dy),$$

where k,$\hat{k}$ and $\tilde{k}$ are kernels with desirable properties and s is a non-negative valued function. The implementation of such techniques is only ensured under certain restrictions on the class of considered kernels and require usually a pre-treatment of the submitted measures.

1.3 Contribution of this Thesis

We present in the following chapters different contributions that are all aimed at understanding better kernels on structured objects, notably strings in Chapter 2, but more generally measures and histograms, in Chapters 3 and 4. Chapter 5 contains echoes from the first chapter, and we apply certain ideas of (Willems et al., 1995) to generalize the approach of (Cuturi and Vert, 2005) to turn it into a general template approach for all kinds of objects seen as multiresolution measures. A unifying trait of most of these contributions is that they make frequent use of the notion of semigroup, that is simple sets endowed with an associative operation and a neutral element, but where all elements do not necessarily have an inverse. Many objects encountered in real life applications have this property, since substraction

is not such a natural thing after all.

1.3.1 A String Kernel Inspired by Universal Coding

The main inspiration behind the context-tree kernel is the algorithmic efficiency of the context-tree weighting (CTW) algorithm presented by (Willems et al., 1995) and further studied in (Catoni, 2004). The CTW algorithm provides a framework to compute a mixture of source distributions for strings,

$$p_\omega(X) = \int_{\theta \in \Theta} p_\theta(X)\, \omega\,(d\theta\,),$$

where $\{\rho_\theta, \theta \in \Theta\}$ is a set of Markov chain distributions and ω is a prior on the parameters. Rather than considering the integration of $p_\theta(X)$, we consider instead the integration of $p_\theta(X)p_\theta(Y)$ and benefit from the same computational trick given that $\{p_\theta, \theta \in \Theta\}$ is an exponential family. The context-tree kernel translates into a kernel on strings that can be computed in linear time and which translates into a similarity measure that is seemingly more useful than more simple approaches based on the same counters (Leslie et al., 2002), while using no biological knowledge.

1.3.2 Semigroup Kernels on Measures

The following work, first proposed in (Cuturi and Vert, 2005) and extended in (Cuturi et al., 2005), was inspired by two remarks formulated while investigating the properties of the context-tree kernel. First, while many kernels used on vectors x, y are translation invariant and boil down to kernels of the type $k(x, y) = h(x - y)$, this choice is impossible with strings where no minus operation exists. Harmonic analysis on semigroups, surveyed in (Berg et al., 1984), proposes a more general theoretical framework to compare objects through their sum (concatenation for strings) rather than only through their difference. Second, observing that all the calculations performed to compute the context-tree kernel can be translated into operations on histograms of transitions, we chose to generalize this approach by considering directly kernels on measures. The natural structure of positive measures is moreover that of semigroup, since the substraction of two positive measures doesn't yield one in the general case. This remark made us investigate Bochner-type theorems to characterize kernels on measures that would only depend on their sum. Finally, the trend of defining kernels that could be kernelized led us to an approximation grounded on the second order moment of a measure, drawing interesting connections between variance matrices and kernel matrices.

1.3.3 Spectral Semigroup Kernels on Measures

We reformulate the parallel between the variance of a molecular measures and the Gram matrix of the elements of its support to define a wider family of positive definite kernels on measures than the one proposed in (Cuturi et al., 2005). We coin down this family of kernels "spectral semigroup kernels" because these kernels only depend on the spectrum of the variance of their mixture. Besides proposing

a general formula which can be related to the characteristic function of the convex cone of positive definite matrices, we propose various examples of such functions that can be computed in a close computational form, as well as a very simple case of such kernels that yields a fast kernel on clouds of points.

1.3.4 A Multiresolution Framework for Nested Measures

In parallel to the previous study, we present in this chapter a framework inspired by (Willems et al., 1995) to extend the applicability of kernels on measures. The easiest criticism that can be drawn against the idea of representing objects as histograms of components is that crucial information, about the components' localization for instance, might be lost in this translation. Rather than considering a single measure on the space of components, we use in this context families of nested sub-probability measures that sum up to the overall histogram and which can be assembled using a hierarchical knowledge on how these sub-probabilities, which may be each linked to a specific event, are extracted. The approach is at an early stage and calls for the further study of kernels that can handle both sub-probabilities and probabilities.

Chapter 2

The Context Tree Kernel for Strings

Résumé

Nous proposons dans ce chapitre un nouveau noyau pour chaînes de caractères particulièrement adapté à l'étude de séquences biologiques. Etant données deux chaînes de caractères X et Y, ce noyau est construit en considérant les probabilités de X, Y, puis de leur concaténation XY évaluées toutes les trois selon des densités extraites d'une large famille de modèles markoviens. Ces probabilités sont ensuite moyennées selon une approche bayésienne sur les divers paramètres des modèles considérés, une approche inspirée de la théorie du codage universel et de la compression. Nous sommes alors en mesure, en comparant ces trois probabilités moyennées, de proposer une quantité définie positive en les chaînes qui peut être utilisée comme un noyau. Le calcul rapide de ce noyau, d'une complexité linéaire en temps et espace mémoire, facilite son utilisation sur des données biologiques (plusieurs milliers de chaînes de plusieurs centaines de caractères, en l'occurrence d'acides aminés), notamment dans des expériences de classification où il peut être couplé avec diverses méthodes à noyaux telles que les machines à vecteurs de support. Les performances de ce noyau sur la base de données étudiée sont encourageantes, alors qui n'utilise aucune connaissance biologique *a priori* sur le type de séquences traitées.

This work is co-authored with Jean-Philippe Vert and was published in a slightly different form in *Neural Networks*, September 2005 (Cuturi and Vert, 2005).

2.1 Introduction

The need for efficient analysis and classification tools for strings remains a key issue in machine learning. This is notably the case in computational biology where the availability of an ever-increasing quantity of biological sequences calls for efficient and computationally feasible algorithms to detect, cluster, and annotate functional similarities between DNA or amino-acid sequences.

Recent years have witnessed the rapid development of a class of algorithms called *kernel methods* (Schölkopf and Smola, 2002) that may offer useful tools for these tasks. In particular, the Support Vector Machine (SVM) algorithms (Boser et al., 1992; Vapnik, 1998) provide state-of-the-art performance in many real-world problems of classifying objects into predefined classes. SVMs have already been applied with success to a number of issues in computational biology, including but not limited to protein homology detection (Jaakkola et al., 2000; Leslie et al., 2002, 2003; Noble and Liao, 2002; Ben-Hur and Brutlag, 2003; Vert et al., 2004), functional classification of genes (Liao and Noble, 2002; Vert, 2002), or prediction of gene localization (Hua and Sun, 2001). A more complete survey of the application of kernel methods in computational biology is presented in (Schölkopf et al., 2004).

The basic ingredient shared by all kernel methods is the kernel function, that measures similarities between pairs of objects to be analyzed or classified. To use kernel methods in the field of string classification requires a prior design of an efficient kernel function on strings. Indeed, while early-days SVM focused on the classification of vector-valued objects, for which kernels are well understood and easily represented, recent attempts to use SVM for the classification of more general objects have resulted in the development of several kernels for structured objects such as strings (Watkins, 2000; Haussler, 1999; Jaakkola et al., 2000; Leslie et al., 2002, 2003; Noble and Liao, 2002; Ben-Hur and Brutlag, 2003; Vert et al., 2004), graphs (Kashima et al., 2003), or even phylogenetic profiles (Vert, 2002).

A useful kernel for sequences, as the one we wish to propose in this work, should have several properties. It should represent a meaningful measure of similarity between two sequences and be general enough to be efficient on different datasets without excessive tuning. This similarity measure needs to be further positive definite to be applied in the general framework of kernel methods and rapid to compute to sustain large-scale implementations (typically, have a linear complexity

with respect to the lengths of the compared sequences). Such an ideal kernel probably does not exist, and different kernels might be useful in different situations. For large-scale studies which might involve comparing thousands of sequences, yielding to millions of kernel evaluations, or to answer simple queries which could be found in on-line applications, the computation cost becomes critical and only fast kernels, such as the spectrum (Leslie et al., 2002) and mismatch (Leslie et al., 2003) kernels can be accepted. In applications where accuracy is more important than speed, slower kernels that include more biological knowledge such as the Fisher (Jaakkola et al., 2000), pairwise (Liao and Noble, 2002) or local alignment (Vert et al., 2004) kernels might be accepted if they improve the performance of a classifier.

Our contribution in this paper is to introduce a new class of string kernels which are both fast to compute and based on the spectrum of the considered strings. The spectrum of a string as defined by (Leslie et al., 2002) is the weighted list of k-mers (or k-grams, that is a substring of k letters) contained in the string, where the weights stand for the occurrence (or relative frequency with respect to the string's length) of the considered k-mer in the string. While the work of (Leslie et al., 2002) uses a linear dot-product on that representation, we propose in this work an alternative class of kernels on those counters.

The motivation behind these kernels is grounded on information theory, in a similar way to the work proposed recently in (Li et al., 2004). By applying an information theoretic viewpoint on the information carried out by strings, we present a way to compare strings through kernel methods using little prior knowledge on the structure of the alphabet, just as universal coding (Cover and Thomas, 1991) aims at giving a sound compression of sequences with no prior assumptions on the nature of those sequences. This information theoretic viewpoint takes the form of a string compression algorithm, which is first applied on two strings X and Y to be compared taken separately, and then on their concatenation XY. Intuitively if the compression behaves in a similar way (in terms of gain for instance) for X, Y and XY, one can expect the strings to share similar properties. On the opposite, one might conclude that the strings are dissimilar if their concatenation cannot be efficiently compressed. This intuition can be translated mathematically in terms of differences in coding redundancy between X and Y with respect to XY, in the light of noiseless coding theory for instance (Cover and Thomas, 1991).

The compression method we choose in this work is the popular context-tree weighting (CTW) algorithm (Willems et al., 1995), and we show how to derive a kernel out of it. The compression performed by the CTW algorithm involves a Bayesian averaging of the probability of a string under a large collection of weighted source distributions. These source distributions are chosen among variable-length Markov chains, which are also known as context-tree (CT) models. Using the CTW algorithm to derive a kernel brings a sound answer to the criterions expressed previously, since it guarantees positive definiteness, computational speed, and an additional interpretation (other than the one considered by compression) to our kernel.

Indeed, the integral representation of the CTW compression, not shared with ad-hoc heuristics such as the Lempel-Ziv algorithm, first enables us to cast easily the proposed kernels in the framework of *mutual information kernels* (Seeger, 2002),

which ensures their positive definiteness. Second, the Bayesian integration over Markovian (and hence exponential) models performed by such kernels provides us with an alternative probabilistic interpretation of their computation. Following that alternative perspective, the kernels project each sequence to be compared to the set of their probabilities under all distributions contained in the class of CT models, and compare different sequences in the light of their respective projections. These projections can be intuitively considered as feature extractions, where each considered context-tree distribution acts as a feature extractor, providing a feature which is the likelihood of the distribution for the considered sequence. Because we find that perspective to be clearer, we will favor this interpretation and present the family of context-tree kernels in a constructive manner and as a special case of mutual information kernels. However the reader should keep in mind that most choices in models and priors taken to devise such kernels are chosen to match the CTW algorithm's ones, so as to benefit from its properties including notably computational tricks presented by the authors of (Willems et al., 1995) to ensure linear (in time and space) computational costs.

The paper is organized as follows. In Section 2.2 we present the general strategy of devising mutual information kernels from families of probabilistic models. In Section 2.3 we define a kernel for sequences based on context-tree models. Its efficient implementation, derived from the CTW algorithm, is presented in Section 2.4. We present further interpretations of the context-tree kernel's computation as well as links with universal coding in Section 2.5. Experimental results on a benchmark problem of remote protein homology detection are then presented in Section 2.6.

2.2 Probabilistic Models and Mutual Information Kernels

A parametric probabilistic model on a measurable space $\mathcal{X}$ is a family of distributions $\{P_\theta, \theta \in \Theta\}$ on $\mathcal{X}$, where θ is the parameter of the distribution P_θ. Typically, the set of parameters Θ is a subset of $\mathbb{R}^n$, in which case n is called the dimension of the model. As an example, a hidden Markov model (HMM) for sequences is a parametric model, the parameters being the transition and emission probabilities (Durbin et al., 1998). A family of probabilistic models is a family $\{P_{f,\theta_f}, f \in \mathcal{F}, \theta_f \in \Theta_f\}$, where $\mathcal{F}$ is a finite or countable set, and $\Theta_f \subset \mathbb{R}^{\dim(f)}$ for each $f \in \mathcal{F}$, where $\dim(f)$ denotes the dimension of f. An example of such a family would be a set of HMMs with different architectures and numbers of states. Probabilistic models are typically used to model sets of elements $X_1, \ldots, X_n \in \mathcal{X}$, by selecting a model $\hat{f}$ and a choosing a parameter $\hat{\theta}_{\hat{f}}$ that best "fits" the dataset, using criteria such as penalized maximum likelihood or maximum *a posteriori* probability Durbin et al. (1998).

Alternatively, probabilistic models can also be used to characterize each single element $X \in \mathcal{X}$ by the feature representation

$$\phi(X) = \left(P_{f,\theta_f}(X)\right)_{f \in \mathcal{F}, \theta_f \in \Theta_f} \quad , \tag{2.1}$$

spanning all possible probabilities of X within the considered families. If the

probabilistic models are designed in such a way that each distribution is roughly characteristic of a class of objects of interest, then the representation $\phi(X)$ quantifies how X fits each class. In this representation, each distribution can be seen as a filter that extracts from X an information, namely the probability of X under this distribution, or equivalently how much X fits the class modelled by this distribution.

Kernels are real-valued function $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ that can be represented in the form of a dot-product $\kappa(X, Y) = \langle \psi(X), \psi(Y) \rangle_F$ for some mapping ψ from $\mathcal{X}$ to a Hilbert space F(Schölkopf and Smola, 2002). Given the preceding mapping ϕ of Equation (2.1), a natural way to derive a kernel from a family of probabilistic models is to endow the set of representations $\phi(X)$ with a dot-product, and set $\kappa(X, Y) = \langle \phi(X), \phi(Y) \rangle$. This can be done for example if a prior probability $\pi(f, d\theta_f)$ can be defined on the set of distributions in the models, by considering the following dot-product:

$$\kappa(X, Y) = \langle \phi(X), \phi(Y) \rangle \stackrel{\text{def}}{=} \sum_{f \in \mathcal{F}} \pi(f) \int_{\Theta_f} P_{f,\theta_f}(X)\ P_{f,\theta_f}(Y)\ \pi(d\theta_f). \qquad (2.2)$$

By construction, the kernel in Equation (2.2) is a valid kernel, that belongs to the class of mutual information kernels (Seeger, 2002). Observe that contrary to the Fisher kernel that also uses probabilistic models, no model or parameter estimation is required in Equation (2.2). Intuitively, for any two elements X and Y the kernel of Equation (2.2) automatically detects the models and parameters that explain both X and Y, with consequent weights if the models and parameters are likely to appear under the prior π. On the other hand, models and parameters for which X and Y present no simultaneous fit bring a marginal contribution to the value of the kernel and are thus ignored.

There is of course some arbitrariness in the previous definition, both in the definition of the models and in the choice of the prior distribution π. This arbitrary can be used to include prior knowledge in the kernel. For example, if one wants to detect similarity with respect to families of sequences known to be adequately modelled by HMMs, then using HMM models constrains the kernel to detect such similarities. However, these choices need to be decided having computational limitations in mind. The calculations involved in Equation (2.2), namely the computation of the likelihood of a distribution for two given sequences and the integration of those likelihoods over a set of parameters, should not only be tractable under a closed form but also fast to compute. This is not likely to be the case for most families of models and most choices of priors. We consider those limitations under the light of the solution proposed by the CTW algorithm in the framework of universal coding, to define below a suitable set of models and prior distributions.

Prior to this definition, we note that some biases might appear when attempting to compare sequences of different lengths, which is likely to be the case for most applications. Indeed, as the probability of a sequence under most models defined on strings (including Markovian models) decreases roughly exponentially with its length, the value of the kernel (2.2) can not only be strongly biased if we directly consider the probabilities of two strings of very different lengths, but will also quickly tend to negligible values when comparing long strings. This is a classical

issue with many string kernels that leads to bad performance in classification with SVM (Schölkopf et al., 2002; Vert et al., 2004). This undesirable effect can easily be controlled in our case by normalizing the likelihoods as follows:

$$\kappa_\sigma(X,Y) = \sum_{f \in \mathcal{F}} \pi(f) \int_{\Theta_f} P_{f,\theta_f}(X)^{\frac{\sigma}{N_X}} \; P_{f,\theta_f}(Y)^{\frac{\sigma}{N_Y}} \; \pi(d\theta_f). \tag{2.3}$$

where σ is a width parameter and N_X and N_Y stand for the lengths of both sequences. Equation (2.3) is clearly a valid kernel (only the feature extractor ϕ is modified), and the parameter σ controls the range of values it takes independently of the lengths of the sequences used.

2.3 A Mutual Information Kernel Based on Context-Tree Models

In this section we derive explicitly a mutual information kernel for strings based on context-tree models with mixtures of Dirichlet priors. Context-tree models, also known as probabilistic suffix trees, are Markovian models which are actually equivalent to Markov chains up to a different parametrization as we will see below. They have been shown useful to model several families of sequences, including biological ones as illustrated by their use in (Bejerano and Yona, 1999; Eskin et al., 2000) where different techniques to estimate such models on protein sequences were proposed. Note however that the use of context-trees in the present work should not be related excessively to their previous success in representing sequences, notably protein families. Arguably, we both believe and observe in our experiments that the overall performance of the kernels proposed in this paper does not rely so much on the individual ability of such distributions to model specific families of sequences, but rather on their overall efficiency to extract features out of strings.

2.3.1 Framework and notations

Starting with basic notations and definitions, let E be a finite set of size d called the alphabet. In our experiments E will be the 20 letters alphabet of amino-acids. For a given depth $D \in \mathbb{N}$ corresponding to the maximal memory of the Markovian models, we write E_D^* for the set of strings of E of length smaller or equal to D, i.e., $E_D^* = \cup_{i=0}^{D} E^i$, which includes $\varnothing$, the empty word. We introduce $\mathcal{X} = \cup_{n=0}^{\infty} (E^D \times E)^n$, the set on which we choose to define our kernel. Observe that we do not define directly the kernel on the set of finite-length sequences, but rather in a slightly different framework which stresses the fact that we are chiefly interested in the local behaviour of the sequence. Indeed, we see sequences as finite sets of (*context,letter*) couples, where the *context* is a D-letters long subsequence of the initial sequence and the *letter* is the element next to it. This transformation is justified by the fact that we consider Markovian models with a memory limited to D letters, and is equivalent to the information contained by the spectrum of order $D+1$ of a string. An element $X \in \mathcal{X}$ can therefore be written as $X = \{(x_c^i, x_l^i)\}_{i=1..N_X}$ where N_X is the cardinality of X, $x_c^i \in E^D$ and $x_l^i \in E$ for all $1 \leq i \leq N_X$.

By considering strings as collections of transitions (or equivalently substrings of length $D+1$) we do not only follow previous approaches such as (Leslie et al., 2003, 2002; Ben-Hur and Brutlag, 2003) but also refer to a recent framework in kernel design (Kondor and Jebara, 2003; Cuturi et al., 2005) which aims at computing kernels on compound objects (such as long strings) as kernels for collections of smaller components ($D+1$-mers in this case).

2.3.2 Context-Tree Models

Context-tree distributions require the definition of a complete suffix dictionary (c.s.d) $\mathcal{D}$. A c.s.d is a finite set of words of E_D^* such that any left-infinite sequence has a unique suffix in $\mathcal{D}$, but no word in $\mathcal{D}$ has a suffix in $\mathcal{D}$. We write $L(\mathcal{D})$ for the length of the longest word contained in $\mathcal{D}$ and $\mathcal{F}_D$ for the set of c.s.d $\mathcal{D}$ that satisfy $L(\mathcal{D}) \leq D$. We note that c.s.d are in correspondence with suffix trees based on E as illustrated in Figure 2.1. Once this dictionary $\mathcal{D}$ or the equivalent suffix tree structure is set, a distribution on $\mathcal{X}$ can be defined by attaching a multinomial distribution[12] $\theta_s \in \Sigma_d$ to *each* word s of $\mathcal{D}$. Indeed, through the family of parameters $\theta = (\theta_s)_{s\in\mathcal{D}}$ we define a conditional distribution on $\mathcal{X}$ by the following equation:

$$P_{\mathcal{D},\theta}(X) = \prod_{i=1}^{N_X} \theta_{\mathcal{D}(x_c^i)}(x_l^i), \tag{2.4}$$

where for any word m in E^D, $\mathcal{D}(m)$ is the unique suffix of m in $\mathcal{D}$. Note that Markov chains are a simple case of context-tree distributions when the c.s.d. is set to E^D. Conversely a context-tree distribution $\mathcal{D}$ can be easily expressed as a Markov chain by assigning the transition parameter θ_s to all the contexts in E^D which admit s as their unique suffix in $\mathcal{D}$. Context-trees can thus be seen as an alternative parametrization and a handier representation of Markov chains, where the importance of some suffixes is highlighted by developing further or stopping the tree expansion in branches which have more or less significance in the generation of our string. We present in Figure 2.1 an example where the alphabet has been set to $E = \{A, B, C\}$ and the maximal depth D to 3. We write $\mathcal{P}_D$ for $\{P_{\mathcal{D},\theta} : \mathcal{D} \in \mathcal{F}_D, \theta \in \Theta_{\mathcal{D}}\}$, the set of context-tree distributions of depth D.

2.3.3 Prior Distributions on Context-Tree Models

We define in this section priors on the family of distributions $\mathcal{P}_D$ introduced in the previous section, following the framework set in Equation (2.3). Namely, we propose a prior probability $\pi(\mathcal{D}, d\theta)$ on $\mathcal{P}_D$ to finalize the definition of the family of kernels presented in this paper, which we name *context-tree kernels*. Note that we use and adapt the priors proposed by (Willems et al., 1995) to our computation to ensure the computation feasibility of the proposed kernels. The prior probability $\pi(\mathcal{D}, d\theta)$ on $\mathcal{P}_D$ factorizes as $\pi(\mathcal{D}, d\theta) = \pi(\mathcal{D})\,\pi(d\theta|\,\mathcal{D})$, two terms which are defined as follows.

[12] writing Σ_d for the canonical simplex of dimension d, i.e., $\Sigma_d = \{\xi = (\xi_i)_{1\leq i\leq d} : \xi_i \geq 0, \sum \xi_i = 1\}$.

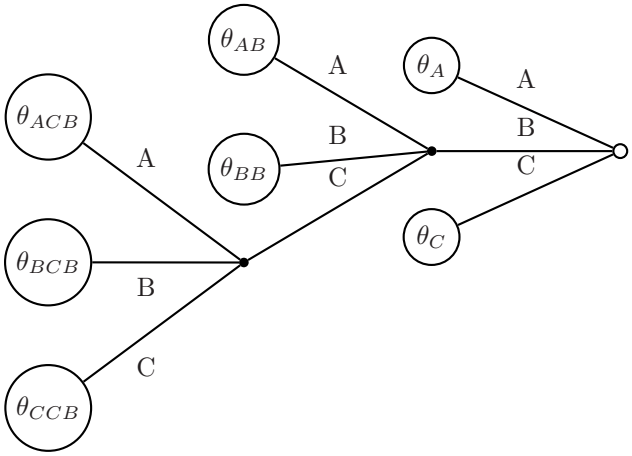


Figure 2.1. *Tree representation of a context-tree distribution. The alphabet E is set to $\{A, B, C\}$, the maximal depth D to 3 and the complete suffix dictionary $\mathcal{D}$ is the set of strings $\{A, AB, BB, ACB, BCB, CCB, C\}$. Each parameter θ_s for $s \in \mathcal{D}$ is in that case a vector of the 3-dimensional simplex Σ_3.*

Prior on the Tree Structure

The set $\mathcal{F}_D$ of complete suffix dictionaries is equivalent to the set of complete d-ary trees of depth smaller than D, namely the set of trees where each node has either d sons or none, up to nodes of depth D which can only be leaves. Following (Willems et al., 1995) we define a simple probability π_D on the set $\mathcal{F}_D$ of trees that is the direct translation of an intuitive random generation of trees stopped at depth D. Starting from the root, the tree generation process follows recursively the following rule: up to depth $D-1$, each node has probability ε of giving birth to d children, and probability $1-\varepsilon$ of having no children, that is probability $1-\varepsilon$ of becoming a leaf; if the node is however located at depth D of the tree, it becomes automatically a leaf with no successors. In mathematical terms, this defines a branching process on d-ary trees, truncated at depth D. The typical outcome of this generation is completely parameterized by ε, since a low value will favour short-depth trees while values closer to 1 will yield fully grown trees of depth D up to the case where $\varepsilon = 1$ and only the full tree of depth D is considered. If we denote by $\mathring{\mathcal{D}}$ the set of all strict suffixes (corresponding to inner nodes of the tree) of elements of $\mathcal{D}$, the probability of a tree is given by:

$$\pi_D(\mathcal{D}) = \prod_{s \in \mathring{\mathcal{D}}} \varepsilon \prod_{\substack{s \in \mathcal{D} \\ l(s) < D}} (1-\varepsilon) = \varepsilon^{\frac{|\mathcal{D}|-1}{d-1}} (1-\varepsilon)^{\mathrm{card}\{s \in \mathcal{D} \,|\, l(s) < D\}}. \tag{2.5}$$

This probability is illustrated with the case of the tree shown in Figure 2.2, with a prior value for that example of $\varepsilon^3(1-\varepsilon)^4$.

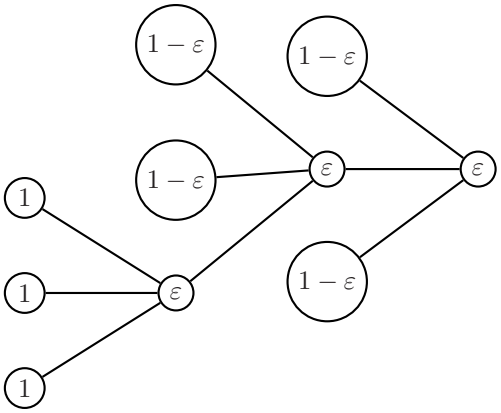


Figure 2.2. *Branching-process generation of the example shown in Figure 2.1 with a depth $D = 3$. The prior value for that tree is $\varepsilon^3(1-\varepsilon)^4$.*

Priors on Multinomial Parameters

For a given tree $\mathcal{D}$ we now define a prior on the family of multinomial parameters $\Theta_{\mathcal{D}} = (\Sigma_d)^{\mathcal{D}}$ which fully characterizes a context-tree distribution based on a dictionary of suffixes $\mathcal{D}$. We assume an independent prior among multinomials attached to each of those suffixes as

$$\pi(d\theta|\mathcal{D}) = \prod_{s\in\mathcal{D}} \omega(d\theta_s),$$

where ω is a prior distribution on the simplex Σ_d. Following Willems et al. (1995) a simple choice is to make use of Dirichlet priors:

$$\omega_\beta(d\theta) = \frac{1}{\sqrt{d}} \frac{\Gamma(\sum_{i=1}^d \beta_i)}{\prod_{i=1}^d \Gamma(\beta_i)} \prod_{i=1}^d \theta_i^{\beta_i - 1} \lambda(d\theta),$$

where λ is Lebesgue's measure and $\beta = (\beta_i)_{i=1..d}$ is the parameter of the Dirichlet distribution. The parameter β incorporates all the prior belief we have on the distribution of the alphabet. It can be either tuned based on empirical data or chosen having theoretical considerations in mind. A natural choice in the latter case is to use Jeffrey's prior (Amari and Nagaoka, 2001, p.44) also known as the Krichevski-Trofimov prior (Willems et al., 1995) and set $\beta_i = \frac{1}{2}$ for $1 \leq i \leq d$. Alternative choices, such as Laplace's successor rule ($\beta_i = 1$) or the Schurmann-Grassberger estimate ($\beta_i = \frac{1}{d}$) have been advocated in the literature and will also be explored in the experimental section of this work, taking into account discussions presented in (Nemenman et al., 2002) for instance. Furthermore, the use of a simple Dirichlet prior can be extended to additive mixtures of Dirichlet priors since the latter have been shown to incorporate more efficiently information on the distributions of amino-acids (Brown et al., 1993). We propose to include such priors in the construction of our kernel and extend the computational framework of the

CTW by doing so. An additive mixture of n Dirichlet distributions is defined by a family of n Dirichlet parameters $\beta^{(1)}, \ldots, \beta^{(n)}$ and n weights $\gamma^{(1)}, \ldots, \gamma^{(n)}$ (with $\sum_{k=1}^{n} \gamma^{(k)} = 1$) to yield the prior:

$$\omega_{\gamma,\,\beta}(d\theta_s) = \sum_{k=1}^{n} \gamma^{(k)}\, \omega_{\beta^{(k)}}\,(d\theta_s). \tag{2.6}$$

2.3.4 Triple Mixture Context-Tree Kernel

Combining the definition of the kernel of Equation (2.3) with the definition of the context-tree model distributions in Equation (2.4) and of the priors on the set of distributions of Equations (2.5), (2.6), we obtain the following expression for the context-tree kernel:

$$\kappa_\sigma(X,Y) = \sum_{\mathcal{D}\in\,\mathcal{F}_D} \pi_D(\mathcal{D}) \int_{\Theta_\mathcal{D}} P_{\mathcal{D},\,\theta}(X)^{\frac{\sigma}{N_X}}\; P_{\mathcal{D},\,\theta}(Y)^{\frac{\sigma}{N_Y}} \prod_{s\in\,\mathcal{D}} \left(\sum_{k=1}^{n} \gamma^{(k)} \omega_{\beta^{(k)}}(d\theta_s) \right). \tag{2.7}$$

We observe that Equation (2.7) involves three summations respectively over the trees, the Dirichlet components used in our additive mixtures, and the multinomial parameters over which a Bayesian averaging is performed. This generalizes the double mixture performed in (Willems et al., 1995) in the context of sequence compression by adding a mixture of Dirichlet priors.

2.4 Kernel Implementation

As pointed out in the introduction, the models and priors selected to define the mutual information kernel of Equation (2.7) may not fit in the best way the natural process which generates the considered sequences. Some distributions favoured by these priors may not even correspond to the ones that are frequently observed in sequences generated by the natural phenomenon. While this may already seem arguably not so important in the context of this paper (which highlights feature extraction as opposed to parameter estimation), we also advocate such choices having in mind they yield an efficient computation of the value of Equation (2.7).

For $r \in \mathbb{N}$, and $\beta = (\beta_i)_{1\leq\, i\leq\, r} \in (\mathbb{R}^{+*})^r$ and $\alpha = (\alpha_i)_{1\leq\, i\leq\, r} \in (\mathbb{R}^{+})^r$ we write $\mathbf{G}_\beta(\alpha)$ for

$$\mathbf{G}_\beta(\alpha) \stackrel{\text{def}}{=} \int_{\Sigma_r} \prod_{i=1}^{r} \theta_i^{\alpha_i} \omega_\beta(d\theta) = \frac{\Gamma(\beta_\bullet)}{\prod_{i=1}^{r}\Gamma(\beta_i)} \frac{\prod_{i=1}^{r}\Gamma(\alpha_i+\beta_i)}{\Gamma(\alpha_\bullet+\beta_\bullet)},$$

where Γ is the Gamma function, Σ_r the r-dimensional simplex, $\beta_\bullet = \sum_{i=1}^{r}\beta_i$, and $\alpha_\bullet = \sum_{i=1}^{r}\alpha_i$. The quantity $\mathbf{G}_\beta(\alpha)$ corresponds to the averaging of the multinomial likelihood $P_\theta(\alpha)$ under a Dirichlet prior of parameter β when θ spans Σ_r. As a reference to Chapter 3, we note that for a family $\beta \in (\mathbb{R}^{+*})^r$, $\mathbf{G}_\beta$ is a semigroup positive definite function on $(\mathbb{R}^{+})^r$ endowed with the usual addition, or on Σ_r when restricted on multinomials in the sense of Definition 4.1.

The computation of the context-tree kernel on two strings can be divided into two phases for more clarity, which can be implemented side by side. A look at Figure 2.3 may give a better intuition on the computations actually performed by the CTW algorithm.

2.4.1 Defining Counters

The first step of the algorithm is to compute for $m \in E^D$ the counter

$$\rho_m(X) \stackrel{\text{def}}{=} \sum_{i=1}^{N_X} \mathbb{1}(x_c^i = m),$$

which simply counts the occurrences of m within contexts enumerated in X. For contexts present in the string X, that is words m such that $\rho_m(X) > 0$, the empirical behaviour of transitions can be estimated as

$$\hat{\theta}_{m,\,e}(X) \stackrel{\text{def}}{=} \frac{\sum_{i=1}^{N_X} \mathbb{1}(x_c^i = m, x_l^i = e)}{\rho_m(X)}.$$

$\hat{\theta}_{m,\,e}$ summarizes the empirical probability of the appearance of letter e after m has been observed. We finally define a last counter:

$$a_{m,\,e}(X,Y) \stackrel{\text{def}}{=} \frac{\rho_m(X)}{N_X}\,\hat{\theta}_{m,\,e}(X) + \frac{\rho_m(Y)}{N_Y}\,\hat{\theta}_{m,\,e}(Y).$$

$a_{m,\,e}(X,Y)$ is a weighted average of the transitions encountered in X and Y. Once those counters are computed on visited contexts, which are up to $N_X + N_Y$, the following downward recursion on the length of the string m (when m spans all strict suffixes of visited contexts) computes equivalent counters for shorter suffixes:

$$\begin{aligned}
\rho_m(X) &= \sum_{f \in E} \rho_{f.\,m}(X), \\
\hat{\theta}_{m,\,e}(X) &= \frac{\sum_{f \in E}\ \rho_{f.\,m}(X)\ \hat{\theta}_{f.\,m,\,e}(X)}{\rho_m(X)}, \\
a_{m,\,e}(X,Y) &= \sum_{f \in E} a_{f.\,m,\,e}(X,Y).
\end{aligned}$$

So far, the memory needed to store the information on which the kernel will be computed (essentially counters a which can be stored in the leaves of a suffix tree generated while scanning only visited contexts) is linear with respect to the size of our strings and is loosely upper-bounded by $D(N_X + N_Y)$.

2.4.2 Recursive Computation of the Triple Mixture

We can now attach to each m for which we have calculated the previous counters the value:

$$K_m(X,Y) = \sum_{k=1}^{n} \gamma^{(k)} \mathbf{G}_{\beta^{(k)}} \left(\sigma \cdot a_{m,\,e}\,(X,Y)_{e \in E}\right),$$

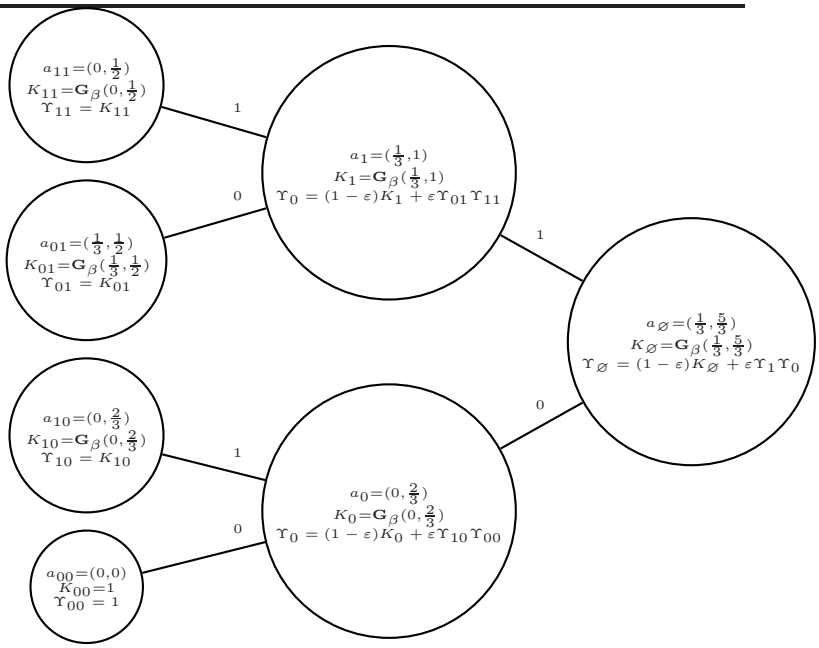


Figure 2.3. *CTW calculation tree for two binary strings* $X = 0111$ *and* $Y = 10101$*, with a depth* $D = 2$*,* $\sigma = 1$ *and an arbitrary Dirichlet parameter* β*. The two string are considered as sets of weighted transitions* $X = \{(01, 1), (11, 1)\}$ *and* $Y = \{(10, 1), (10, 1), (01, 0)\}$*, and the resulting kernel value* $K(X, Y)$ *is* $\Upsilon_\varnothing$*.*

which computes two mixtures, the first being a continuous Bayesian averaging on the possible values of θ weighted by a given Dirichlet prior and the second being a discrete weighted summation using the weighted Dirichlet distributions provided by the mixture $(\gamma^{(k)}, \beta^{(k)})_{k=1..n}$. A numerical approximation of $\mathbf{G}_{\beta^{(k)}}$ can be used in practice, through Lanczos' approximation of the $\ln\Gamma$ function for instance. By defining the quantity $\Upsilon_m(X, Y)$, which is also attached to each visited word m and computed recursively through

$$\Upsilon_m(X,Y) = \begin{cases} K_m(X,Y) & \text{if } l(m) = D, \\ (1-\varepsilon)K_m(X,Y) + \varepsilon \prod_{e\in E} \Upsilon_{e.\,m}(X,Y) & \text{if } l(m) < D. \end{cases}$$

we actually perform the third mixture over all possible tree structures by taking into account the branching probability ε. Indeed, we finally have, recalling $\varnothing$ is the empty word, that:

$$\kappa_\sigma(X,Y) = \Upsilon_\varnothing(X,Y). \tag{2.8}$$

Proof. For a c.s.d model $(\mathcal{D}, \theta)$ and two sets of transitions $X = (x_c^i, x_l^i)_{i=1\le N_X}$ and $Y = (y_c^i, y_l^i)_{1\le i\le N_Y}$ we have that

$$P_{\mathcal{D},\theta}(X)^{\frac{\sigma}{N_X}} P_{\mathcal{D},\theta}(Y)^{\frac{\sigma}{N_Y}} = \prod_{s\in\mathcal{D}} \prod_{e\in E} \theta_s(e)^{\sigma a_{s,\,e}(X,Y)}.$$

The latter product of likelihoods can thus be calculated using only counter a, and we further have that

$$\begin{aligned}
&\int_{\Theta_{\mathcal{D}}} P_{\mathcal{D},\theta}(X)^{\frac{\sigma}{N_X}} P_{\mathcal{D},\theta}(Y)^{\frac{\sigma}{N_Y}} \prod_{s\in\mathcal{D}} \left(\sum_{k=1}^{n} \gamma^{(k)} \omega_{\beta^{(k)}}(d\theta_s) \right) \\
&\quad = \int_{\Theta_{\mathcal{D}}} \prod_{s\in\mathcal{D}} \left[\prod_{e\in E} \theta_s(e)^{\sigma a_{s,e}(X,Y)} \left(\sum_{k=1}^{n} \gamma^{(k)} \omega_{\beta^{(k)}}(d\theta_s) \right) \right] \\
&\quad\quad = \prod_{s\in\mathcal{D}} \sum_{k=1}^{n} \gamma^{(k)} \int_{\Sigma_d} \prod_{e\in E} \theta_s(e)^{\sigma a_{s,e}(X,Y)} \omega_{\beta^{(k)}}(d\theta_s) \\
&\quad\quad\quad = \prod_{s\in\mathcal{D}} \sum_{k=1}^{n} \gamma^{(k)} \mathbf{G}_{\beta^k} \left(\sigma \left(a_{s,e}(X,Y) \right)_{e\in E} \right) = \prod_{s\in\mathcal{D}} K_s(X,Y),
\end{aligned}$$

where we have used Fubini's theorem to factorize the integral in the second line. Having in mind Equation (2.7), we have thus proved that

$$\kappa_\sigma(X,Y) = \sum_{\mathcal{D}\in\mathcal{F}_D} \pi_D(\mathcal{D}) \prod_{s\in\mathcal{D}} K_s(X,Y).$$

The second part of the proof is identical to the one given in (Willems et al., 1995), and developed in (Catoni, 2004) whose recursive treatment we adopt. Let us prove by induction, with respect to successively decreasing lengths of m (i.e., over words m such that $l(m) = D, ..., 0$), that

$$\Upsilon_m(X,Y) = \sum_{\mathcal{D}\in\mathcal{F}_{D-l(m)}} \pi_{D-l(m)}(\mathcal{D}) \prod_{s\in\mathcal{D}} K_{s.m}(X,Y), \tag{2.9}$$

where $\pi_{D-l(m)}$ is the distribution of a tree according to the branching process prior previously presented stopped at level $D - l(m)$. We notice that the set $\mathcal{F}_{D-l(m)}$ of c.s.d's of depth $D - l(m)$ can be further divided into:

$$\mathcal{F}_{D-l(m)} = \left\{ \{(s,y) : y \in E, s \in \mathcal{D}_y\}, \mathcal{D}_y \in \mathcal{F}_{D-l(m)-1} \right\} \cup \{\{\varnothing\}\},$$

where we have that:

$$\begin{aligned}
\pi_{D-l(m)}(\{(s,y) : y \in E, s \in \mathcal{D}_y\}) &= \varepsilon \prod_{y\in E} \pi_{D-l(m)-1}(\mathcal{D}_y), \\
\pi_{D-l(m)}(\{\varnothing\}) &= 1 - \varepsilon.
\end{aligned}$$

Starting our recursion with words of length $d = D$, where Equation (2.9) is valid by the recursive definition of Υ, we assume Equation (2.9) to be valid with words of length d and prove that it holds for words of length $d - 1$. Given m such that

$l(m) = d - 1$, we can write:

$$\begin{aligned}
\Upsilon_m(X,Y) &= (1-\varepsilon)K_m(X,Y) + \varepsilon \prod_{y\in E} \Upsilon_{y.m}(X,Y) \\
&= (1-\varepsilon)K_m(X,Y) + \varepsilon \prod_{y\in E} \sum_{\mathcal{D}\in\mathcal{F}_{D-d}} \pi_{D-d}(\mathcal{D}) \prod_{s\in\mathcal{D}} K_{s.y.m}(X,Y) \\
&= (1-\varepsilon)K_m(X,Y) + \varepsilon \sum_{(\mathcal{D}_y)\in(\mathcal{F}_{D-d})^E} \prod_{y\in E} \pi_{D-d}(\mathcal{D}_y) \prod_{s\in\mathcal{D}_y} K_{s.y.m}(X,Y) \\
&= \pi_{D-l(m)}(\{\varnothing\})K_m(X,Y) \\
&\quad + \sum_{(\mathcal{D}_y)\in(\mathcal{F}_{D-d})^E} \pi_{D-l(m)}(\{(s,y) : y\in E, s\in\mathcal{D}_y\}) \prod_{(s,y)\in\mathcal{D}_y\times E} K_{s.y.m}(X,Y) \\
&= \sum_{\mathcal{D}\in\mathcal{F}_{D-d}} \pi_{D-d}(\mathcal{D}) \prod_{s\in\mathcal{D}} K_{s.m}(X,Y)
\end{aligned}$$

Applying Equation (2.9) to the case where $m = \varnothing$ we finally prove Equation (2.8).■

As previously recalled, the computation of the counters has a linear cost in time and memory with respect to $D(N_X + N_Y)$. As only counters that correspond to visited suffixes of X and Y are created, recursive computation of Υ_m is also linear in time and space (the values Υ_m for suffixes m not encountered, such that $\rho_m(X) = \rho_m(Y) = 0$, being equal to 1). As a final result, the computation of the kernel is linear in time and space with respect to $D(N_X + N_Y)$.

2.5 Source Coding and Compression Interpretation

There is a very classical duality between source distributions (a random model to generate infinite sequences) and sequence compression (Cover and Thomas, 1991). Roughly speaking, if a finite sequence X has a probability $P(X)$ under a source distribution P, then one can design a binary code to represent X by $r(X) = -\log_2 P(X)$ bits, up to 2 bits, using for example arithmetic coding. In this section, we provide an interpretation of the context-tree kernel in terms of information theory and compression, and highlight its differences with the spectrum kernel.

When sequences are generated by an unknown source P, it is classical to form a coding source distribution by averaging several *a priori* sources. Under reasonable assumptions, one can design this way universal codes, in the sense that the average length of the codes be almost as short as if P was known and the best source was used. As an example, the context-tree weighting (CTW) algorithm (Willems et al., 1995) defines a coding probability P_π for sequences by averaging source distributions defined by context-trees as follows:

$$P_\pi(X) \stackrel{\text{def}}{=} \sum_{\mathcal{D}\in\mathcal{F}_D} \pi(\mathcal{D}) \int_{\Theta_\mathcal{D}} P_{\mathcal{D},\theta}(X) \prod_{s\in\mathcal{D}} \omega_\beta(d\theta_s), \tag{2.10}$$

where ω_β is the Krichevski-Trofimov prior. Up to the mixture of Dirichlet and the exponents (used to renormalize the probabilities with respect to the sequences'

lengths), we therefore see, by comparing (2.10) with (2.7), that the context-tree kernel between two sequences can be roughly interpreted as the probability under P_π of the concatenation of the two sequences. Our kernel actually considers a sequence as a list of weighted empirical distributions $\{(\rho_m, \hat{\theta}_m)\}_{m \in E^D} \in (\mathbb{R}^+ \times \Sigma_d)^{E^D}$ which summarizes the local behaviour of its letter transitions. These coordinates, whose information is equivalent to the one contained in the spectrum of the sequence, can be used to compute the likelihood of a specific context-tree distribution $(\mathcal{D}, \theta)$ on such a set by deriving $\{(\rho_s, \hat{\theta}_s), s \in \mathcal{D}\}$ recursively, as in the previous computation.

We write $\mathrm{kl}(\theta||\theta')$ for the Kullback-Leibler divergence between θ and θ', two multinomial parameters of size d, i.e $\mathrm{kl}(\theta||\theta') = \sum_{i=1}^{d} \theta_i \ln \frac{\theta_i}{\theta'_i}$. We also note $h(\theta)$ the entropy of θ, i.e., $h(\theta) = -\sum_{i=1}^{d} \theta_i \ln \theta_i$. We use the following identity on θ and θ':

$$\begin{aligned}\prod_{i=1}^{d} \theta_i^{\theta'_i} &= e^{\sum_{i=1}^{d} \theta'_i \ln \theta_i} \\ &= e^{\sum_{i=1}^{d} \theta'_i \ln \frac{\theta_i}{\theta'_i} + \sum_{i=1}^{d} \theta'_i \ln \theta'_i} \\ &= e^{-h(\theta') - kl(\theta'||\theta)},\end{aligned}$$

to reformulate the mixture coding probability P_π on $\mathcal{X}$ in the context of the context-tree kernel computation. Indeed, following the priors previously defined on $\mathcal{P}_D$, the following formula expresses the value of the coding probability of a given string through its counters ρ and $\hat{\theta}$:

$$P_\pi(\rho, \hat{\theta}) = \sum_{\mathcal{D} \in \mathcal{F}_D} \pi(\mathcal{D}) \prod_{s \in \mathcal{D}} e^{-\sigma \rho_s h(\hat{\theta}_s)} \int_{\Sigma_d} e^{-\sigma \rho_s kl(\hat{\theta}_s||\theta)} \, \omega_{\gamma,\beta}(d\theta).$$

We write r_π for $-\ln P_\pi$, $\hat{\rho}(X)$ for the normalized counters $\frac{1}{N_X}\rho(X)$ and introduce the following function t_π of two strings,

$$\begin{aligned}t_\pi(X, Y) =& \frac{1}{2}\left[r_\pi\left(\hat{\rho}(X), \hat{\theta}(X)\right) + r_\pi\left(\hat{\rho}(Y), \hat{\theta}(Y)\right)\right] \\ &- r_\pi\left(\frac{\hat{\rho}(X) + \hat{\rho}(Y)}{2}, \frac{\hat{\theta}(X) + \hat{\theta}(Y)}{2}\right).\end{aligned} \tag{2.11}$$

Finally we have, by defining the renormalized kernel $\tilde{\kappa}_\sigma$ as

$$\tilde{\kappa}_\sigma(X, Y) = \kappa_\sigma(X, Y) / \sqrt{\kappa_\sigma(X, X)\kappa_\sigma(Y, Y)},$$

that

$$\tilde{\kappa}_\sigma(X, Y) = e^{-t_\pi(X,Y)}.$$

We note here that the function t_π can be interpreted in the light of semigroup kernels on sets of components or measures, as proposed in (Cuturi and Vert, 2005; **?**). A semigroup is roughly a set with an associative composition law, which in our case is just the addition of counters and estimated transitions as in Equation (2.11). What the structure of t_π highlights is that the similarity computed by context-tree

kernels between two strings, and more precisely the sequences of counters indexed on E^D that describe them, is just a function of their sum. This is opposed to the computations led by the spectrum kernel, which considers products on those counters (namely a linear-dot-product on those vectors of counters). The whole family of context-tree kernels are hence defined through a prior belief on the behaviour of sequences of counters (tuned through a selection of specific priors), which is first applied to the sequences individually, $\left(\hat{\rho}(X), \hat{\theta}(X)\right)$ and $\left(\hat{\rho}(Y), \hat{\theta}(Y)\right)$, before evaluating it on their mean $\left(\frac{\hat{\rho}(X)+\hat{\rho}(Y)}{2}, \frac{\hat{\theta}(X)+\hat{\theta}(Y)}{2}\right)$. This formulation makes the link with compression more precise, where instead of concatenating strings we rather perform counter averaging. This viewpoint can also bring forward a geometrical perspective on the actual computation which is performed. The choice of a compression algorithm (namely a selection of priors) defines the shape of the function r_π on the whole space of counters, and the similarity between two sequences is measured through the difference between three evaluations of r_π, first taken on the two points taken apart and then on their average, which is directly related to the convexity of r_π.

2.6 Experiments

2.6.1 Protein Domain Homology Detection Benchmark

We report results concerning the performance of the context-tree family of kernels on a benchmark experiment that tests the capacity of SVMs to detect remote homologies between protein domains. This is simulated by recognizing domains that are in the same SCOP (Structural Classification of Proteins (Hubbard et al., 1997), ver. 1.53) superfamily, but not in the same family, using the procedure described in (Jaakkola et al., 2000). We used the files compiled by the authors of (Noble and Liao, 2002), which consist in 4352 sequences extracted from the Astral database of protein domains. For each of the 54 tested families, the protein domains within the family where considered positive *test* examples while protein domains within the superfamily but outside the family were considered as positive *training* examples. This results in 54 classification experiments with at least 10 positive training examples and 5 positive test examples. Negative examples were selected outside of the positive sequences' fold with a similar ratio. Following previous studies of this benchmark, we computed the ROC (Receiving Operator Characteristic, (Gribskov and Robinson, 1996)), ROC50 and RFP (Rate of False Positives) of each of the classification performed by a SVM based on various parameter settings of the context-tree kernel. The ROC score (or AUC, Area Under the ROC Curve) is the normalized area under the curve which plots the number of true positives as a function of false positives; the ROC50 is the area under the ROC curve up to 50 false positives while the median RFP is the number of false positives scoring as high or better than the median scoring true positives. We average those criterions on the 54 experiments to provide an overall measure of the performance of the considered kernels on this task.

2.6.2 Parameter Tuning and Comparison with Alternative String Kernels

Let us now recall, along with the formula of the context-tree kernel, the different parameters which need to be set to control the output of the family of context-tree kernels;

$$\kappa_\sigma(X,Y) = \sum_{\mathcal{D}\in\mathcal{F}_D} \pi_D(\mathcal{D}) \int_{\Theta_\mathcal{D}} P_{\mathcal{D},\theta}(X)^{\frac{\sigma}{N_X}} P_{\mathcal{D},\theta}(Y)^{\frac{\sigma}{N_Y}} \prod_{s\in\mathcal{D}} \left(\sum_{k=1}^{n} \gamma^{(k)} \omega_{\beta^{(k)}}(d\theta_s) \right).$$

- σ represents the width taken by the probabilities used to compute the kernel, allowing us to control the range of values appearing in Gram matrices. Large values of σ will favor diagonal-dominant matrices while lower values will tend to create Gram matrices of similar elements. We thus tuned these values empirically, so that none of the two previous problematic cases appears. Using a σ value between 1 and 5 typically ensures this and we usually set $\sigma = 2$.

- The branching-process probability π_D is parameterized by ε, which controls the typical amount of suffixes numbered in dictionaries in relation with D, their maximal depth. A sound choice for ε, as well as being validated by experiments and used in the original paper (Willems et al., 1995) is to set $\varepsilon = 1/d$, as this keeps a good balance between small trees which might capture simple interactions and larger trees which might detect longer range interactions.

- The depth parameter D controls the maximal memory of our Markovian models. This parameter influences the complexity of our features extractors and adds computational time to most calculations. The submitted sequences have typical lengths of roughly two hundred amino-acids. Hence lengths set between 2 and 4 for the substrings (that is contexts of length 1 to 3) should suffice to capture most of the available information, following the empirical observation of (Li et al., 2004) that the base d logarithm of the average length of the sequences suffices as a context length to capture most of the letter-to-letter transition information. Those lengths were also shown to give the best performance on the datasets.

- Finally, different Dirichlet priors but also families of Dirichlet mixtures

  $$(\gamma^{(k)}, \beta^{(k)})_{1\leq k\leq n},$$

  can be considered to compute mixtures at the level of each node.

  We tested three popular uniform priors, namely the Jeffrey prior ($\beta_i = 1/2$) as used in (Willems et al., 1995), the Laplace successor rule ($\beta_i = 1$) and the Schurmann-Grassberger estimate ($\beta_i = 1/d$). The first two choices yielded equivalently good results in practice and better than the third one. We also tested mixtures of Dirichlet priors, hoping they would prove more accurate in comparing biological strings. We considered 3, 9 and 20 components additive mixtures (respectively hydro-cons.3comp, byst-4.5-0-3.9comp, recode3.20comp,

fournier20.comp and dist20.comp) which can be downloaded from a Dirichlet mixture repository[13]. These mixtures gave disappointing results when averaged over the 54 families (considering ROC average this means a performance of roughly 87% to 88%) but produced somehow different results for some families which seemed hard to classify through other methods. However, we interpret the fact that those families of Dirichlet mixtures did not improve overall accuracy as a form of overfitting. Again, while this biological knowledge might improve the selection of a specific model to fit sequences (notably Hidden Markov Models), it does not seem to work in our framework where we only use statistical models as feature extraction tools.

Except for the poor performances of context-tree kernels defined with Dirichlet mixtures, the few experiments we led on different parameters yielded no surprises and favoured ranges of parameters which were theoretically motivated, namely short depths, a branching process prior of roughly $1/d$ and uniform Dirichlet priors (either the Laplace of the Krichevski-Trofimov rule). Note further that the variety of all 54 protein families used in the experiment prevents overfitting since an increase in performance over certain families usually implies a decrease in other ones. We compare the performance of context-tree kernels with other string kernels, where the performances we report were computed according to the parameters known to perform in a good way on that dataset and proposed by the respective authors of those kernels. We present here the best mismatch kernel (5,1) reported in (Leslie et al., 2003), which can also be computed in linear time and space, but also more greedy algorithms such as the pairwise kernel (Liao and Noble, 2002) and the two local alignment kernels (LA-Eig, LA-Ekm) presented in (Vert et al., 2004), which, as opposed to the context-tree Kernel, take into account relevant information known to be of capital importance for biological sequences (such as gaps, deletions or mutations of amino-acids). We also report the results of the spectrum kernel (Leslie et al., 2002) with depth 3 and 4 and show that based on the same information (D-grams) the context-tree kernel clearly outperforms the latter. The classification was led using the Gist (version 2.1.1) implementation of SVM[14], where all parameters specific to SVM optimization were set to default values (elementary attempts to tune the latter parameters did not yield significative improvements in accuracy).

2.6.3 Mean Performances and Curves

We present in Figure 2.4 the performance of all previously quoted kernels, along with an implementation of the context-tree kernel where $\sigma = 2$, $D = 4$, $\varepsilon = 1/20$ and where a uniform Jeffrey prior was used. The results show that the CTK performs roughly better than the mismatch kernel and overall similarly to the pairwise kernel, notably in regions where classification becomes more difficult and ROC scores become lower for all techniques. Except in those regions, it is outperformed by both versions of the local-alignment kernels. The CTK is computed in linear time and without any biological knowledge, a property exclusively shared with the spectrum

[13] `http://www.cse.ucsc.edu/research/compbio/dirichlets/`
[14] `http://microarray.cpmc.columbia.edu/gist/download.html`

Method	ROC	ROC50	RFP
CTK	0.894	0.371	0.0869
Spectrum 3	0.781	0.277	-
Spectrum 4	0.716	0.208	-
Mismatch 5,1	0.872	0.400	0.0837
Pairwise	0.894	0.461	0.0846
LA-ekm	0.934	0.663	0.0525
LA-eig	0.923	0.646	0.0552

Table 2.1. *Mean results for ROC, ROC50 and RFP as produced over the 54 families by all compared kernels, where CTK denotes the context-tree kernel set with* $\sigma = 2$, $\varepsilon = 1/20$, *Jeffrey's prior and depth* $D = 4$.

kernel whose curve in the figure is significantly below that of the CTK (only results obtained for the spectrum with a depth 3 have been represented in the plot).

Table 2.1 summarizes the three main statistics used to compare performances over the studied benchmark between context-tree kernels and all other kernels. In this table, context-tree kernels perform (relatively to other kernels) better in terms of ROC score than in terms of ROC50 and RFP, and we have no explanation for this. As can be easily deduced from the previous figure, the context-tree kernel clearly outperforms the spectrum kernel while using exactly the same information. In the general case where only the spectrum information of a string is available, the context-tree kernel may hence prove more useful than the simple spectrum kernel.

Additionally, we report that using the 20-components mixture fournier20 with usual parameters ($D = 4$ and $\varepsilon = 0.05$) produced the means $(0.887, 0.366, 0.096)$ for ROC, ROC50 and RFP scores respectively. Simpler mixtures with less components did not yield a substantial increase in performance either and we hence did not use them further, notably because of their computational cost. However, we observed important variations on the performance for each family with respect to other context-tree kernels which only use uniform priors, while their overall performance was similar or slightly worse. This might be interpreted as some complementary between the two kinds of kernels and may be a subject of future research, through a linear combination of kernels for instance (Lanckriet et al., 2004). Finally we present in Table 2.2 a few results for meaningful settings of the context-tree kernels using Jeffrey's prior. These results show that an increase in the complexity of the models used to perform the Bayesian mixture does not yield better results in practice. Surprisingly, a context-tree kernel of depth 1 suffices to provide good results, while more complex models which require far more computational cost give relatively poor results. These observations show once more that in the context of mutual information kernels, the relevance of distributions to model the data does not seem to be an important criterion.

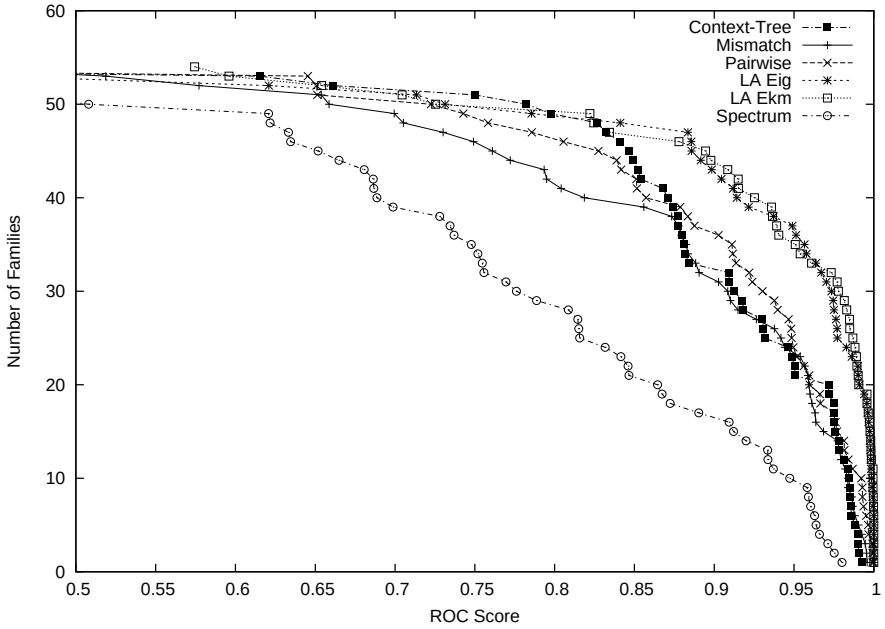


Figure 2.4. *Performance of all considered kernels on the problem of recognizing domain's superfamily. The curve shows the total number of families for which a given methods exceeds a ROC score threshold. CTK denotes the context-tree kernel set with* $\sigma = 2$, $\varepsilon = 1/20$, *Jeffrey's prior and depth* $D = 4$.

2.7 Closing remarks

We introduced a novel class of kernels for sequences that are fast to compute while only using the spectrum of the submitted strings. The kernel is a mutual information kernel based on a family of context-tree models, and makes a link between the comparison of two string and the ability of universal coding algorithms to compress them when taken together. On a benchmark experiment of remote homology detection it performs at a level close to state-of-the-art levels reached by kernels which involve heavier computational cost and make use of biological knowledge. The context-tree kernels clearly outperform the spectrum kernel on the same benchmark while using exactly the same information. The context-tree kernel, whose computation is inspired by universal coding theory, may thus share one of the qualities of the latter algorithms, which is to appear as a sound prior choice to explore similarities between sequences for whom little knowledge is available and at a reasonable computational cost.

Parameters (with Jeffrey's prior and $\sigma = 2$)	ROC	ROC50	RFP
$D = 1,\ \varepsilon = 1/20$	0.886	0.373	0.0796
$D = 2,\ \varepsilon = 1/20$	0.892	0.391	0.0857
$D = 3,\ \varepsilon = 1/20$	0.895	0.385	0.0865
$D = 4,\ \varepsilon = 1/20$	0.894	0.371	0.0869
$D = 4,\ \varepsilon = 1/4$	0.893	0.378	0.0857
$D = 4,\ \varepsilon = 1/2$	0.889	0.367	0.0877
$D = 4,\ \varepsilon = 1$	0.872	0.326	0.101
$D = 6,\ \varepsilon = 1/20$	0.889	0.362	0.0923
$D = 8,\ \varepsilon = 1/20$	0.885	0.355	0.0986

Table 2.2. *From short trees to long and dense trees: mean results of ROC, ROC50 and RFP scores for different settings of the branching process prior and of the length of the models selected. Note that when only the complete tree is selected ($\varepsilon = 1$) the performance decreases significantly. In that case, namely when no mixture is performed on the class of models, the context-tree computation resembles the simpler computation performed by the spectrum kernel. Note also that a good performance is reached when the context-tree only uses contexts of length* 1 *(namely Markov chains of depth* 1*), which shows that models should be selected to extract features and not to model sequences, a hint which is further confirmed by the fact that long trees do not perform very well despite their better ability to absorb more knowledge about the strings' transitions.*

Chapter 3

Semigroup Kernels on Measures

Résumé

Nous étudions dans ce chapitre une nouvelle famille de noyaux sur listes d'éléments, histogrammes ou plus généralement sur des mesures positives bornées sur un espace mesurable $\mathcal{X}$. Cette famille de noyaux, inspirée des noyaux présentés dans le chapitre 2, se distingue de précédentes contributions dans le domaine des noyaux sur mesures par le fait que nous comparons ici deux mesures μ, μ' de $M_+^b(\mathcal{X})$ en ne considérant qu'une fonction à valeurs réelles $\varphi : M_+^b(\mathcal{X}) \to \mathbb{R}$, évaluée en leur somme, i.e., $\varphi(\mu + \mu')$. Nous étudions la famille des fonctions φ, dites de semigroupe, telles que l'application $(\mu, \mu') \mapsto \varphi(\mu + \mu')$ est définie positive sur $M_+^b(\mathcal{X}) \times M_+^b(\mathcal{X})$. Nous proposons une représentation intégrale de ces fonctions, inspirée des travaux de Berg et al. (1984). Après avoir considéré la divergence de Jensen entre deux mesures, qui utilise le cas particulier de l'entropie d'une mesure comme fonction de semigroupe, nous nous intéressons plus spécifiquement à l'inverse de la variance généralisée de la moyenne de deux mesures positives. Nous montrons que cette quantité, en plus d'être définie positive sur $M_+^b(\mathcal{X})$, peut être facilement transformée pour être applicable dans les cas, fréquents dans la pratique des méthodes à noyaux, où l'espace de départ $\mathcal{X}$ n'est pas Euclidien. Nous terminons ce chapitre sur des expériences menées dans le domaine de la reconnaissance automatique de chiffres manuscrits qui montrent l'intérêt pratique de notre approche.

This work is co-authored with Kenji Fukumizu Jean-Philippe Vert and was published in a slightly different form in the *Journal of Machine Learning Research*, July 2005 (Cuturi et al., 2005).

3.1 Introduction

The challenge of performing classification or regression tasks over complex and non vectorial objects is an increasingly important problem in machine learning, motivated by diverse applications such as bioinformatics or multimedia document processing. The kernel method approach to such problems (Schölkopf and Smola, 2002) is grounded on the choice of a proper similarity measure, namely a positive definite (p.d.) kernel defined between pairs of objects of interest, to be used alongside with kernel methods such as support vector machines (Boser et al., 1992). While natural similarities defined through dot-products and related distances are available when the objects lie in a Hilbert space, there is no standard dot-product to compare strings, texts, videos, graphs or other structured objects. This situation motivates the proposal of various kernels, either tuned and trained to be efficient on specific applications or useful in more general cases.

One possible approach to kernel design for such complex objects consists in representing them by sets of basic components which are easier to manipulate, and designing kernels on such sets. Such basic components can be typically subparts of the original complex objects, where the sets can be obtained by exhaustive enumeration or random sampling. For instance, a very common way to represent a text for applications such as text classification and information retrieval is to break it into words and consider it as a bag-of-words, that is, a finite set of weighted components. Another frequent use of this scheme in sequence analysis is to extract all fixed-length blocks of consecutive letters of a string and represent the string by the vector of counts of all blocks (Leslie et al., 2002), or even to add to this representation additional blocks obtained by slight modifications of the blocks present in the text with different weighting schemes (Leslie et al., 2003). Similarly, a grey-level digitized image can be considered as a finite set of points of $\mathbb{R}^3$ where each point (x, y, I) stands for the intensity I displayed on the pixel (x, y) in that image (Kondor and Jebara, 2003).

Once such a representation is obtained, different strategies have been adopted to design kernels on these descriptions of complex objects. When the set of basic components is finite, this representation amounts to encoding a complex object as a finite-dimensional vector of counters, and any kernel for vectors can be then trans-

lated to a kernel for complex object through this feature representation (Joachims, 2002; Leslie et al., 2002, 2003). For more general situations, several authors have proposed to handle such weighted lists of points by first fitting a probability distribution to each list, and defining a kernel between the resulting distributions (Lafferty and Lebanon, 2002; Jebara et al., 2004; Kondor and Jebara, 2003; Hein and Bousquet, 2005). Alternatively, Cuturi and Vert (2005) use a parametric family of densities and a Bayesian framework to define a kernel for strings based on the mutual information between their sets of variable-length blocks, using the concept of mutual information kernels (Seeger, 2002). Finally, Wolf and Shashua (2003) recently proposed a formulation rooted in kernel canonical correlation analysis (Bach and Jordan, 2002; Melzer et al., 2001; Akaho, 2001) which makes use of the principal angles between the subspaces generated by the two sets of points to be compared when considered in a feature space.

We explore in this contribution a different direction to kernel design for weighted lists of basic components. Observing that such a list can be conveniently represented by a molecular measure on the set of basic components, that is a weighted sum of Dirac measures, or that the distribution of points might be fit by a statistical model and result in a density on the same set, we formally focus our attention on the problem of defining a kernel between finite measures on the space of basic components. More precisely, we explore the set of kernels between measures that can be expressed as a function of their sum, that is:

$$k(\mu, \mu') = \varphi(\mu + \mu'). \tag{3.1}$$

The rationale behind this formulation is that if two measures or sets of points μ and μ' overlap, then it is expected that the sum $\mu + \mu'$ is more concentrated and less scattered than if they do not. As a result, we typically expect φ to quantify the dispersion of its argument, increasing when it is more concentrated. This setting is therefore a broad generalization of the observation by Cuturi and Vert (2005) that a valid kernel for strings, seen as bags of variable-length blocks, is obtained from the compression rate of the *concatenation* of the two strings by a particular compression algorithm.

The set of measures endowed with the addition is an Abelian semigroup, and the kernel (3.1) is exactly what Berg et al. (1984) call a *semigroup kernel*. The main contribution of this paper is to present several valid positive definite (p.d.) semigroup kernels for molecular measures or densities. As expected, we prove that several functions φ that quantify the dispersion of measures through their entropy or through their variance matrix result in valid p.d. kernels. Using entropy to compare two measures is not a new idea (Rao, 1987) but it was recently restated within different frameworks (Hein and Bousquet, 2005; Endres and Schindelin, 2003; Fuglede and Topsøe, 2004). We introduce entropy in this paper slightly differently, noting that it is a semigroup negative definite function defined on measures. On the other hand, the use of generalized variance to derive a positive definite kernel between measures as proposed here is new to our knowledge. We further show how such kernels can be applied to molecular measures through regularization operations. In the case of the kernel based on the spectrum of the variance matrix, we show how

it can be applied implicitly for molecular measures mapped to a reproducing kernel Hilbert space when a p.d. kernel on the space of basic components is provided, thanks to an application of the "kernel trick".

Besides these examples of practical relevance, we also consider the question of characterizing *all* functions φ that lead to a p.d. kernel through (3.1). Using the general theory of semigroup kernels we state an integral representation of such kernels and study the semicharacters involved in this representation. This new result provides a constructive characterization of such kernels, which we briefly explore by showing that Bayesian mixtures over exponential models can be seen as natural functions φ that lead to p.d. kernels, thus making the link with the particular case treated by Cuturi and Vert (2005).

This paper is organized as follows. We first introduce elements of measure representations of weighted lists and define the semigroup formalism and the notion of semigroup p.d. kernel in Section 3.2. Section 3.3 contains two examples of semigroup p.d. kernels, which are however usually not defined for molecular measures: the entropy kernel and the inverse generalized variance (IGV) kernel. Through regularization procedures, practical applications of such kernels on molecular measures are proposed in Section 3.4, and the approach is further extended by kernelizing the IGV through an *a priori* kernel defined itself on the space of components in Section 3.5. Section 3.6 contains the general integral representation of semigroup kernels and Section 3.7 makes the link between p.d. kernels and Bayesian posterior mixture probabilities. Finally, Section 5.4 contains an empirical evaluation of the proposed kernels on a benchmark experiment of handwritten digits classification.

3.2 Notations and Framework

In this section we set up the framework and notations of this paper, in particular the idea of semigroup kernel on the semigroup of measures.

3.2.1 Measures on Basic Components

We model the space of basic components by a Hausdorff space $(\mathcal{X}, \mathcal{B}, \nu)$ endowed with its Borel σ-algebra and a Borel dominant measure ν. A positive Radon measure μ is a positive Borel measure which satisfies $(i)\, \mu(C) < +\infty$ for any compact subset $C \subseteq \mathcal{X}$ and $(ii)\, \mu(B) = \sup\{\mu(C) | C \subseteq B, C \text{ compact } \}$ for any $B \in \mathcal{B}$ (see for example Berg et al. (1984) for the construction of Radon measures on Hausdorff spaces). The set of positive bounded (i.e., $\mu(\mathcal{X}) < +\infty$) Radon measures on $\mathcal{X}$ is denoted by $M_+^b(\mathcal{X})$. We introduce the subset of $M_+^b(\mathcal{X})$ of molecular (or atomic) measures $\mathrm{Mol}_+(\mathcal{X})$, namely measures such that

$$\mathrm{supp}(\mu) \stackrel{\text{def}}{=} \{x \in \mathcal{X} |\, \mu(U) > 0, \text{ for all open subset } U \text{ s.t. } x \in U\}$$

is finite, and we denote by $\delta_x \in \mathrm{Mol}_+(\mathcal{X})$ the molecular (Dirac) measure of weight 1 on x. For a molecular measure μ, an *admissible base* of μ is a finite list γ of weighted points of $\mathcal{X}$, namely $\gamma = (x_i, a_i)_{i=1}^d$, where $x_i \in \mathcal{X}$ and $a_i > 0$ for $1 \leq i \leq d$, such that $\mu = \sum_{i=1}^d a_i \delta_{x_i}$. We write in that case $|\gamma| = \sum_{i=1}^d a_i$ and $l(\gamma) = d$. Reciprocally,

a measure μ is said to be the image measure of a list of weighted elements γ if the previous equality holds. Finally, for a Borel measurable function $f \in \mathbb{R}^{\mathcal{X}}$ and a Borel measure μ, we write $\mu[f] = \int_{\mathcal{X}} f d\mu$.

3.2.2 Semigroups and Sets of Points

We follow in this paper the definitions found in Berg et al. (1984), which we now recall. An *Abelian semigroup* $(\mathcal{S}, +)$ is a nonempty set S endowed with an *associative* and *commutative composition* $+$ and a neutral element 0. Referring further to the notations used in Berg et al. (1984), note that we will only use auto-involutive semigroups in this paper, and will hence not discuss other semigroups which admit different involutions.

A function $\varphi : \mathcal{S} \to \mathbb{R}$ is called a *positive definite* (resp. *negative definite*, n.d.) function on the semigroup $(S, +)$ if $(s, t) \mapsto \varphi(s+t)$ is a p.d. (resp. n.d.) kernel on $\mathcal{S} \times \mathcal{S}$. The symmetry of the kernel being ensured by the commutativity of $+$, the positive definiteness is equivalent to the fact that the inequality

$$\sum_{i,j=1}^{N} c_i c_j \, \varphi(x_i + x_j) \geq 0$$

holds for any $N \in \mathbb{N}, (x_1, \ldots, x_N) \in \mathcal{S}^N$ and $(c_1 \ldots, c_n) \in \mathbb{R}^N$. Using the same notations, and adding the additional condition that $\sum_{i=1}^{n} c_i = 0$ yields the definition of negative definiteness as φ satisfying now

$$\sum_{i,j=1}^{N} c_i c_j \, \varphi(x_i + x_j) \leq 0.$$

Hence semigroup kernels are real-valued functions φ defined on the set of interest $\mathcal{S}$, the similarity between two elements s, t of $\mathcal{S}$ being just the value taken by that function on their composition, namely $\varphi(s + t)$.

Recalling our initial goal to quantify the similarity between two complex objects through finite weighted lists of elements in $\mathcal{X}$, we note that $(\mathcal{P}(\mathcal{X}), \cup)$ the set of subsets of $\mathcal{X}$ equipped with the usual union operator $\cup$ is a semigroup. Such a semigroup might be used as a feature representation for complex objects by mapping an object to the set of its components, forgetting about the weights. The resulting representation would therefore be an element of $\mathcal{P}(\mathcal{X})$. A semigroup kernel k on $\mathcal{P}(\mathcal{X})$ measuring the similarity of two sets of points $A, B \in \mathcal{P}(\mathcal{X})$ would use the value taken by a given p.d. function φ on their union, namely $k(A, B) = \varphi(A \cup B)$. However we put aside this framework for two reasons. First, the union composition is idempotent (i.e., for all A in $\mathcal{P}(\mathcal{X})$, we have $A \cup A = A$) which as noted in Berg et al. (1984, Proposition 4.4.18) drastically restricts the class of possible p.d. functions. Second, such a framework defined by sets would ignore the frequency (or weights) of the components described in lists, which can be misleading when dealing with finite sets of components. Other problematic features would include the fact that $k(A, B)$ would be constant when $B \subset A$ regardless of its characteristics, and that comparing sets of very different sizes should be difficult.

In order to overcome these limitations we propose to represent a list of weighted points $z = (x_i, a_i)_{i=1}^d$, where for $1 \leq i \leq d$ we have $x_i \in \mathcal{X}$ and $a_i > 0$, by its image measure $\delta_z = \sum_{i=1}^d a_i \delta_{x_i}$, and focus now on the Abelian semigroup $(M_+^b(\mathcal{X}), +)$ to define kernels between lists of weighted points. This representation is richer than the one suggested in the previous paragraph in the semigroup $(\mathcal{P}(\mathcal{X}), \cup)$ to consider the merger of two lists. First it performs the union of the supports; second the sum of such molecular measures also adds the weights of the points common to both measures, with a possible renormalization on those weights. Two important features of the original list are however lost in this mapping: the order of its elements and the original frequency of each element within the list as a weighted singleton. We assume for the rest of this paper that this information is secondary compared to the one contained in the image measure, namely its unordered support and the *overall* frequency of each point in that support. As a result, we study in the following sections p.d. functions on the semigroup $(M_+^b(\mathcal{X}), +)$, in particular on molecular measures, in order to define kernels on weighted lists of simple components.

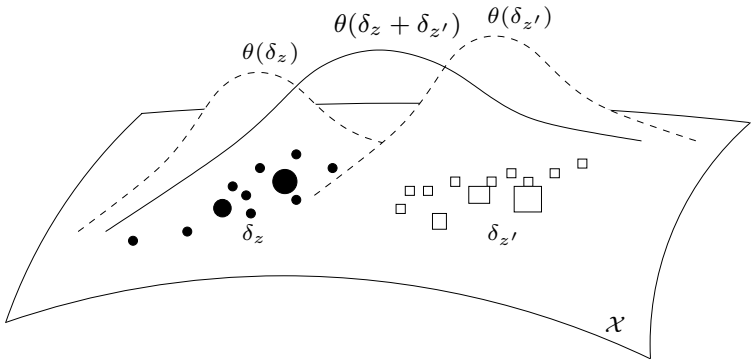


Figure 3.1. *Measure representations of two lists z and z'. Each element of z (resp. z') list is represented by a black circle (resp. a white square), the size of which represents the associated weight. Five measures of interest are represented: the image measures δ_z and $\delta_{z'}$ of those weighted finite lists, the smoothed density estimates $\theta(\delta_z)$ and $\theta(\delta_{z'})$ of the two lists of points, and the smoothed density estimate $\theta(\delta_z + \delta_{z'})$ of the union of both lists.*

Before starting the analysis of such p.d. functions, it should however be pointed out that several interesting semigroup p.d. kernels on measures are not directly applicable to molecular measures. For example, the first function we study below is only defined on the set of absolutely continuous measures with finite entropy. In order to overcome this limitation and be able to process complex objects in such situations, it is possible to think about alternative strategies to represent such objects by measures, as illustrated in Figure 5.3:

- The molecular measures δ_z and $\delta_{z'}$ as the image measures corresponding to the two weighted sets of points of z and z', where dots and squares represent

the different weights applied on each points;

- Alternatively, smoothed estimates of these distributions obtained for example by non-parametric or parametric statistical density estimation procedures, and represented by $\theta(\delta_z)$ and $\theta(\delta_{z'})$ in Figure 5.3. Such estimates can be considered if a p.d. kernel is only defined for absolutely continuous measures. When this mapping takes the form of estimation among a given family of densities (through maximum likelihood for instance) this can also be seen as a prior belief assumed on the distribution of the objects;
- Finally, a smoothed estimate of the sum $\delta_z + \delta_{z'}$ corresponding to the merging of both lists, represented by $\theta(\delta_z + \delta_{z'})$, can be considered. Note that $\theta(\delta_z + \delta_{z'})$ might differ from $\theta(\delta_z) + \theta(\delta_{z'})$.

A kernel between two lists of points can therefore be derived from a p.d. function on $(M_+^b(\mathcal{X}), +)$ in at least three ways:

$$k(z, z') = \begin{cases} \varphi(\delta_z + \delta_{z'}), & \text{using } \varphi \text{ directly on molecular measures,} \\ \varphi\left(\theta(\delta_z) + \theta(\delta_{z'})\right), & \text{using } \varphi \text{ on smoothed molecular measures,} \\ \varphi\left(\theta(\delta_z + \delta_{z'})\right), & \text{evaluating } \varphi \text{ on a smoothed version of the sum.} \end{cases}$$

The positive definiteness of φ on $M_+^b(\mathcal{X})$ ensures positive definiteness of k only in the first two cases. The third expression can be seen as a special case of the first one, where we highlight the usage of a preliminary mapping on the sum of two measures; in that case $\varphi \circ \theta$ should in fact be p.d. on $(M_+^b(\mathcal{X}), +)$, or at least $(\mathrm{Mol}_+(\mathcal{X}), +)$. Having defined the set of representations on which we will focus in this paper, namely measures on a set of components, we propose in the following section two particular cases of positive definite functions that can be computed through an addition between the considered measures. We then show how those quantities can be computed in the case of molecular measures in Section 3.4.

3.3 The Entropy and Inverse Generalized Variance Kernels

In this section we present two basic p.d. semigroup kernels for measures, motivated by a common intuition: the kernel between two measures should increase when the sum of the measures gets more "concentrated". The two kernels differ in the way they quantify the concentration of a measure, using either its entropy or its variance. They are therefore limited to a subset of measures, namely the subset of measures with finite entropy and the subset of sub-probability measures with non-degenerate variance, but are extended to a broader class of measures, including molecular measures, in Section 3.4.

3.3.1 Entropy Kernel

We consider the subset of $M_+^b(\mathcal{X})$ of absolutely continuous measures with respect to the dominant measure ν, and identify in this section a measure with its correspond-

ing density with respect to ν. We further limit the subset to the set of non-negative valued ν-measurable functions on $\mathcal{X}$ with finite sum, such that

$$M_+^h(\mathcal{X}) \stackrel{\text{def}}{=} \{f : \mathcal{X} \to \mathbb{R}^+ | \, f \text{ is } \nu\text{-measurable}\,, |h(f)| < \infty, |f| < \infty\}$$

where we write for any measurable non-negative valued function g,

$$h(g) \stackrel{\text{def}}{=} -\int_{\mathcal{X}} g \ln g \, d\nu,$$

(with $0 \ln 0 = 0$ by convention) and $|g| \stackrel{\text{def}}{=} \int_{\mathcal{X}} g \, d\nu$, consistently with the notation used for measures. If g is such that $|g| = 1$, $h(g)$ is its differential entropy. Using the following inequalities,

$$(a+b)\ln(a+b) \leq a \ln a + b \ln b + (a+b) \ln 2, \text{ by convexity of } x \mapsto x \ln x,$$
$$(a+b)\ln(a+b) \geq a \ln a + b \ln b,$$

we have that $(M_+^h(\mathcal{X}), +)$ is an Abelian semigroup since for f, f' in $M_+^h(\mathcal{X})$ we have that $h(f+f')$ is bounded by integrating pointwise the inequalities above, the boundedness of $|f+f'|$ being also ensured. Following Rao (1987) we consider the quantity

$$J(f, f') \stackrel{\text{def}}{=} h(\frac{f+f'}{2}) - \frac{h(f)+h(f')}{2}, \tag{3.2}$$

known as the *Jensen divergence* (or Jensen-Shannon divergence) between f and f', which as noted by Fuglede and Topsøe (2004) can be seen as a symmetrized version of the Kullback-Leibler (KL) divergence D, since

$$J(f, f') = \frac{1}{2} D(f || \frac{f+f'}{2}) + \frac{1}{2} D(f' || \frac{f+f'}{2}).$$

The expression of Equation (3.2) fits our framework of devising semigroup kernels, unlike the direct use of the KL divergence (Moreno et al., 2004) which is neither symmetric nor negative definite. As recently shown in Endres and Schindelin (2003) and Österreicher and Vajda (2003), $\sqrt{J}$ is a metric on $M_+^h(\mathcal{X})$ which is a direct consequence of J's negative definiteness proven below, through Berg et al. (1984, Proposition 3.3.2) for instance. The Jensen-Divergence was also recently reinterpreted as a special case of a wider family of metrics on $M_+^b(\mathcal{X})$ derived from a particular family of Hilbertian metrics on $\mathbb{R}_+$ as presented in Hein and Bousquet (2005). The comparison between two densities f, f' is in that case performed by integrating pointwise the squared distance between both densities $d^2(f(x), f'(x))$ over $\mathcal{X}$, using for d a distance chosen among a suitable family of metrics in $\mathbb{R}_+$ to ensure that the final value is independent of the dominant measure ν. The considered family for d is described in Fuglede and Topsøe (2004) through two parameters, a family of which the Jensen Divergence is just a special case as detailed in Hein and Bousquet (2005). The latter work shares with this paper another similarity, which lies in the "kernelization" of such quantities defined on measures through a prior kernel on the space of components, as will be reviewed in Section 3.5. However,

of all the Hilbertian metrics introduced in Hein and Bousquet (2005), the Jensen-Divergence is the only one that can be related to the semigroup framework used throughout this paper.

Note finally that a positive definite kernel k is said to be infinitely divisible if $-\ln k$ is a negative definite kernel. As a consequence, any positive exponentiation $k^\beta, \beta > 0$ of an infinitely divisible kernel is a positive definite kernel.

Proposition 3.1. *h is a negative definite function on the semigroup $M_+^h(\mathcal{X})$. As a consequence e^{-h} is a positive definite function on $M_+^h(\mathcal{X})$ and its normalized counterpart, $k_h \stackrel{\text{def}}{=} e^{-J}$ is an infinitely divisible positive definite kernel on $M_+^h(\mathcal{X}) \times M_+^h(\mathcal{X})$.*

Proof. It is known that the real-valued function $r : y \mapsto -y \ln y$ is n.d. on $\mathbb{R}_+$ as a semigroup endowed with addition (Berg et al., 1984, Example 6.5.16). As a consequence the function $f \mapsto r \circ f$ is n.d. on $M_+^h(\mathcal{X})$ as a pointwise application of r since $r \circ f$ is integrable w.r.t ν. For any real-valued n.d. kernel k and any real-valued function g, we have trivially that $(y, y') \mapsto k(y, y') + g(y) + g(y')$ remains negative definite. This allows first to prove that $h(\frac{f+f'}{2})$ is also n.d. through the identity $h(\frac{f+f'}{2}) = \frac{1}{2}h(f+f') + \frac{\ln 2}{2}(|f|+|f'|)$. Subtracting the normalization factor $\frac{1}{2}(h(f)+h(f'))$ gives the negative definiteness of J. This finally yields the positive definiteness of k_h as the exponential of the negative of a n.d. function through Schoenberg's theorem (Berg et al., 1984, Theorem 3.2.2). □

Note that only e^{-h} is a semigroup kernel strictly speaking, since e^{-J} involves a normalized sum (through the division by 2) which is not associative. While both e^{-h} and e^{-J} can be used in practice on non-normalized measures, we name more explicitly $k_h = e^{-J}$ the *entropy kernel*, because what it indeed quantifies when f and f' are normalized (i.e., such that $|f| = |f'| = 1$) is the difference of the average of the entropy of f and f' from the entropy of their average. The subset of absolutely continuous *probability* measures on $(\mathcal{X}, \nu)$ with finite entropies, namely $\left\{f \in M_+^h(\mathcal{X}),\ \text{s.t.}\, |f| = 1\right\}$ is not a semigroup since it is not closed by addition, but we can nonetheless define the restriction of J and hence k_h on it to obtain a p.d. kernel on probability measures inspired by semigroup formalism.

3.3.2 Inverse Generalized Variance Kernel

We assume in this subsection that $\mathcal{X}$ is an Euclidian space of dimension n endowed with Lebesgue's measure ν. Following the results obtained in the previous section, we propose under these restrictions a second semigroup p.d. kernel between measures which uses generalized variance. The generalized variance of a measure, namely the determinant of its variance matrix, is a quantity homogeneous to a volume in $\mathcal{X}$. This volume can be interpreted as a typical volume occupied by a measure when considering only its second order moments, making it hence a useful quantification of its dispersion. Besides being easy to compute in the case of molecular measures, this quantity is also linked to entropy if we consider that for normal

laws $\mathcal{N}(m, \Sigma)$ the following relation holds:

$$\frac{1}{\sqrt{\det \Sigma}} \propto e^{-h(\mathcal{N}(m,\Sigma))}.$$

Through this observation, we note that considering the Inverse of the Generalized Variance (IGV) of a measure is equivalent to considering the value taken by e^{-2h} on its maximum likelihood normal law. We will put aside this interpretation in this section, before reviewing it with more care in Section 3.7.

Let us define the variance operator on measures μ with finite first and second moment of $M_+^b(\mathcal{X})$ as

$$\Sigma(\mu) \stackrel{\text{def}}{=} \mu[xx^\top] - \mu[x]\mu[x]^\top.$$

Note that $\Sigma(\mu)$ is always a positive semi-definite matrix when μ is a sub-probability measure, that is when $|\mu| \leq 1$, since

$$\Sigma(\mu) = \mu[(x - \mu[x])\,(x - \mu[x])^\top] + (1 - |\mu|)\,\mu[x]\mu[x]^\top.$$

We call $\det \Sigma(\mu)$ the generalized variance of a measure μ, and say a measure μ is *non-degenerate* if $\det \Sigma(\mu)$ is non-zero, meaning that $\Sigma(\mu)$ is of full rank. The subset of $M_+^b(\mathcal{X})$ of such measures with total weight less than or equal to 1 is denoted by $M_+^v(\mathcal{X})$; $M_+^v(\mathcal{X})$ is convex through the following proposition:

Proposition 3.2. *$M_+^v(\mathcal{X}) \stackrel{\text{def}}{=} \left\{\mu \in M_+^b(\mathcal{X}) : |\mu| = 1, \det \Sigma(\mu) > 0\right\}$ is a convex set, and more generally for $\lambda \in [0, 1)$, $\mu' \in M_+^b(\mathcal{X})$ such that $|\mu'| = 1$ and $\mu \in M_+^v(\mathcal{X})$, $(1 - \lambda)\mu + \lambda\mu' \in M_+^v(\mathcal{X})$.*

Proof. We use the following identity,

$$\Sigma\left((1 - \lambda)\mu + \lambda\mu'\right) = (1 - \lambda)\Sigma(\mu) + \lambda\Sigma(\mu') + \lambda(1 - \lambda)\left(\mu[x] - \mu'[x]\right)\left(\mu[x] - \mu'[x]\right)^\top,$$

to derive that $\Sigma((1 - \lambda)\mu + \lambda\mu')$ is a (strictly) positive-definite matrix as the sum of two positive semi-definite matrices and a strictly positive definite matrix $\Sigma(\mu)$. □

$M_+^v(\mathcal{X})$ is not a semigroup, since it is not closed under addition. However we will work in this case on the mean of two measures in the same way we used their standard addition in the semigroup framework of $M_+^b(\mathcal{X})$.

Proposition 3.3. *The real-valued kernel k_v defined on elements μ, μ' of $M_+^v(\mathcal{X})$ as*

$$k_v(\mu, \mu') = \frac{1}{\det \Sigma(\frac{\mu+\mu'}{2})}$$

is positive definite.

Proof. Let y be an element of $\mathcal{X}$. For any $N \in \mathbb{N}$, any $c_1, ..., c_N \in \mathbb{R}$ such that $\sum_i c_i = 0$ and any $\mu_1, ..., \mu_N \in M_+^v(\mathcal{X})$ we have

$$\sum_{i,j} c_i c_j y^\top \Sigma(\frac{\mu_i + \mu_j}{2}) y = \sum_{i,j} c_i c_j y^\top \left(\frac{1}{2}\mu_i[xx^\top] + \frac{1}{2}\mu_j[xx^\top] - \right.$$
$$\left. \frac{1}{4}\left(\mu_i[x]\mu_i[x]^\top + \mu_j[x]\mu_j[x]^\top + \mu_j[x]\mu_i[x]^\top + \mu_i[x]\mu_j[x]^\top\right) \right) y$$
$$= -\frac{1}{4}\sum_{i,j} c_i c_j y^\top \left(\mu_j[x]\mu_i[x]^\top + \mu_i[x]\mu_j[x]^\top\right) y$$
$$= -\frac{1}{2}\left(\sum_i c_i y^\top \mu_i[x]\right)^2 \leq 0,$$

making thus the function $\mu, \mu' \mapsto y^\top \Sigma(\frac{\mu+\mu'}{2}) y$ negative-definite for any $y \in \mathcal{X}$. Using again Schoenberg's theorem (Berg et al., 1984, Theorem 3.2.2) we have that $\mu, \mu' \mapsto e^{-y^\top \Sigma(\frac{\mu+\mu'}{2})y}$ is positive definite and so is the sum $\frac{1}{(2\pi)^{\frac{n}{2}}} \int_{\mathcal{X}} e^{-y^\top \Sigma(\frac{\mu+\mu'}{2})y} \nu(dy)$ which is equal to $1/\sqrt{\det \Sigma(\frac{\mu+\mu}{2})}$ ensuring thus the positive-definiteness of k_v as its square. □

Both entropy and IGV kernels are defined on subsets of $M_+^b(\mathcal{X})$. Since we are most likely to use them on molecular measures or smooth measures (as discussed in Section 3.2.2), we present in the following section practical ways to apply them in that framework.

3.4 Semigroup Kernels on Molecular Measures

The two positive definite functions defined in Sections 3.3.1 and 3.3.2 cannot be applied in the general case to $\mathrm{Mol}_+(\mathcal{X})$ which as exposed in Section 3.2 is our initial goal. In the case of the entropy kernel, molecular measures are generally not absolutely continuous with respect to ν (except on finite spaces), and they have therefore no entropy; we solve this problem by mapping them into $M_+^h(\mathcal{X})$ through a smoothing kernel. In the case of the IGV, the estimates of variances might be poor if the number of points in the lists is not large enough compared to the dimension of the Euclidean space; we perform in that case a regularization by adding a unit-variance correlation matrix to the original variance. This regularization is particularly important to pave the way to the kernelized version of the IGV kernel presented in the next section, when $\mathcal{X}$ is not Euclidian but simply endowed with a prior kernel κ.

The application of both the entropy kernel and the IGV kernel to molecular measures requires a previous renormalization to set the total mass of the measures to 1. This technical renormalization is also beneficial, since it allows a consistent comparison of two weighted lists even when their size and total mass is very different.

All molecular measures in this section, and equivalently all admissible bases, will hence be supposed to be normalized such that their total weight is 1, and $\mathrm{Mol}_+^1(\mathcal{X})$ denotes the subset of $\mathrm{Mol}_+(\mathcal{X})$ of such measures.

3.4.1 Entropy Kernel on Smoothed Estimates

We first define the Parzen smoothing procedure which allows to map molecular measures onto measures with finite entropy:

Definition 3.4. *Let κ be a probability kernel on $\mathcal{X}$ with finite entropy, i.e., a real-valued function defined on $\mathcal{X}^2$ such that for any $x \in \mathcal{X}$, $\kappa(x,\cdot) : y \mapsto \kappa(x,y)$ satisfies $\kappa(x,\cdot) \in M_+^h(\mathcal{X})$ and $|\kappa(x,\cdot)| = 1$. The κ-Parzen smoothed measure of μ is the probability measure whose density with respect to ν is $\theta_\kappa(\mu)$, where*

$$\begin{aligned} \theta_\kappa : \mathrm{Mol}_+^1(\mathcal{X}) &\longrightarrow M_+^h(\mathcal{X}) \\ \mu &\mapsto \sum_{x \in \operatorname{supp}\mu} \mu(x)\kappa(x,\cdot). \end{aligned}$$

Note that for any admissible base $(x_i, a_i)_{k=1}^d$ of μ we have that $\theta_\kappa(\mu) = \sum_{i=1}^d a_i \kappa(x_i,\cdot)$. Once this mapping is defined, we use the entropy kernel to propose the following kernel on two molecular measures μ and μ',

$$k_h^\kappa(\mu,\mu') = e^{-J\left(\theta_\kappa(\mu),\, \theta_\kappa(\mu')\right)}.$$

As an example, let $\mathcal{X}$ be an Euclidian space of dimension n endowed with Lebesgue's measure, and κ the isotropic Gaussian RBF kernel on that space, namely

$$\kappa(x,y) = \frac{1}{(2\pi\sigma)^{\frac{n}{2}}} e^{-\frac{\|x-y\|^2}{2\sigma^2}}.$$

Given two weighted lists z and z' of components in $\mathcal{X}$, $\theta_\kappa(\delta_z)$ and $\theta_\kappa(\delta_{z'})$ are thus mixtures of Gaussian distributions on $\mathcal{X}$. The resulting kernel computes the entropy of $\theta_\kappa(\delta_z)$ and $\theta_\kappa(\delta_{z'})$ taken separately and compares it with that of their mean, providing a positive definite quantification of their overlap.

3.4.2 Regularized Inverse Generalized Variance of Molecular Measures

In the case of a molecular measure μ defined on an Euclidian space $\mathcal{X}$ of dimension n, the variance $\Sigma(\mu)$ is simply the usual empirical estimate of the variance matrix expressed in an orthonormal basis of $\mathcal{X}$:

$$\Sigma(\mu) = \mu[xx^\top] - \mu[x]\mu[x]^\top \quad = \sum_{i=1}^d a_i x_i x_i^\top - \left(\sum_{i=1}^d a_i x_i\right)\left(\sum_{i=1}^d a_i x_i\right)^\top,$$

where we use an admissible base $\gamma = (x_i, a_i)_{i=1}^d$ of μ to give a matrix expression of $\Sigma(\mu)$, with all points x_i expressed as column vectors. Note that this matrix

expression, as would be expected from a function defined on measures, does not depend on the chosen admissible base. Given such an admissible base, let $X_\gamma = [x_i]_{i=1..d}$ be the $n \times d$ matrix made of all column vectors x_i and Δ_γ the diagonal matrix of weights of γ taken in the same order $(a_i)_{1\leq 1\leq d}$. If we write I_d for the identity matrix of rank d and $\mathbb{1}_{d,d}$ for the $d \times d$ matrix composed of ones, we have for any base γ of μ that:

$$\Sigma(\mu) = X_\gamma(\Delta_\gamma - \Delta_\gamma \mathbb{1}_{d,d}\Delta_\gamma)X_\gamma^\top,$$

which can be rewritten as

$$\Sigma(\mu) = X_\gamma(I_d - \Delta_\gamma \mathbb{1}_{d,d})\Delta_\gamma(I_d - \mathbb{1}_{d,d}\Delta_\gamma)X_\gamma^\top,$$

noting that $(\Delta_\gamma \mathbb{1}_{d,d})^2 = \Delta_\gamma \mathbb{1}_{d,d}$ since $\operatorname{tr}\Delta_\gamma = 1$.

The determinant of $\Sigma(\mu)$ can be equal to zero when the size of the support of μ is smaller than n, the dimension of $\mathcal{X}$, or more generally when the linear span of the points in the support of μ does not cover the whole space $\mathcal{X}$. This problematic case is encountered in Section 3.5 when we consider kernelized versions of the IGV, using an embedding of $\mathcal{X}$ into a functional Hilbert space of potentially infinite dimension. Mapping an element of $\mathrm{Mol}_+^1(\mathcal{X})$ into $M_+^v(\mathcal{X})$ by adding to it any element of $M_+^v(\mathcal{X})$ through Proposition 3.2 would work as a regularization technique; for an arbitrary $\rho \in M_+^v(\mathcal{X})$ and a weight $\lambda \in [0, 1)$ we could use the kernel defined as

$$\mu, \mu' \mapsto \frac{1}{\det \Sigma\left(\lambda \frac{\mu+\mu'}{2} + (1-\lambda)\rho\right)}.$$

We use in this section a different strategy inspired by previous works (Fukumizu et al., 2004; Bach and Jordan, 2002) further motivated in the case of covariance operators on infinite dimensional spaces as shown by Cuturi and Vert (2005). The considered regularization consists in modifying directly the matrix $\Sigma(\mu)$ by adding a small diagonal component ηI_n where $\eta > 0$ so that its spectrum never vanishes. When considering the determinant of such a regularized matrix $\Sigma(\mu) + \eta I_n$ this is equivalent to considering the determinant of $\frac{1}{\eta}\Sigma(\mu) + I_n$ up to a factor η^n, which will be a more suitable expression in practice. We thus introduce the regularized kernel k_v^η defined on measures $(\mu, \mu') \in M_+^b(\mathcal{X})$ with finite second moment as

$$k_v^\eta(\mu, \mu') \stackrel{\text{def}}{=} \frac{1}{\det\left(\frac{1}{\eta}\Sigma\left(\frac{\mu+\mu'}{2}\right) + I_n\right)}.$$

It is straightforward to prove that the regularized function k_v^η is a positive definite kernel on the measures of $M_+^b(\mathcal{X})$ with finite second-order moments using the same proof used in Proposition 3.3. If we now introduce

$$K_\gamma \stackrel{\text{def}}{=} \left[x_i^\top x_j\right]_{1\leq i,j\leq d},$$

for the $d \times d$ matrix of dot-products associated with the elements of a base γ, and

$$\tilde{K}_\gamma \stackrel{\text{def}}{=} \left[(x_i - \sum_{k=1}^d a_k x_k)^\top (x_j - \sum_{k=1}^d a_k x_k)\right]_{1\leq i,j\leq d} = (I_d - \mathbb{1}_{d,d}\Delta_\gamma)K_\gamma(I_d - \Delta_\gamma \mathbb{1}_{d,d}),$$

for its centered expression with respect to the mean of μ, we have the following result:

Proposition 3.5. *Let $\mathcal{X}$ be an Euclidian space of dimension n. For any $\mu \in \mathrm{Mol}^1_+(\mathcal{X})$ and* any *admissible base γ of μ we have*

$$\det\left(\frac{1}{\eta}\tilde{K}_\gamma\Delta_\gamma + I_{l(\gamma)}\right) = \det\left(\frac{1}{\eta}\Sigma(\mu) + I_n\right).$$

Proof. We omit the references to μ and γ in this proof to simplify matrix notations, and write $d = l(\gamma)$. Let $\tilde{X}$ be the $n \times d$ matrix $[x_i - \sum_{j=1}^d a_j x_j]_{i=1..d}$ of centered column vectors enumerated in γ, namely $\tilde{X} = X(I_d - \Delta\mathbb{1}_{d,d})$. We have

$$\Sigma = \tilde{X}\Delta\tilde{X}^\top,$$
$$\tilde{K}\Delta = \tilde{X}^\top\tilde{X}\Delta.$$

Through the singular value decomposition of $\tilde{X}\Delta^{\frac{1}{2}}$, it is straightforward to see that the non-zero elements of the spectrums of matrices $\tilde{K}\Delta, \Delta^{\frac{1}{2}}\tilde{X}^\top\tilde{X}\Delta^{\frac{1}{2}}$ and Σ are identical. Thus, regardless of the difference between n and d, we have

$$\det\left(\frac{1}{\eta}\tilde{K}\Delta + I_d\right) = \det\left(\frac{1}{\eta}\Delta^{\frac{1}{2}}\tilde{X}^\top\tilde{X}\Delta^{\frac{1}{2}} + I_d\right)$$
$$= \det\left(\frac{1}{\eta}\tilde{X}\Delta\tilde{X}^\top + I_n\right) = \det\left(\frac{1}{\eta}\Sigma + I_n\right),$$

where the addition of identity matrices only introduces an offset of 1 for all eigenvalues. □

Given two measures $\mu, \mu' \in \mathrm{Mol}^1_+(\mathcal{X})$, the following theorem can be seen as a regularized equivalent of Proposition 3.3 through an application of Proposition 4.7 to $\mu'' = \frac{\mu+\mu'}{2}$.

Theorem 3.6. *Let $\mathcal{X}$ be an Euclidian space. The kernel k_v^η defined on two measures μ, μ' of $\mathrm{Mol}^1_+(\mathcal{X})$ as*

$$k_v^\eta(\mu, \mu') = \frac{1}{\det\left(\frac{1}{\eta}\tilde{K}_\gamma\Delta_\gamma + I_{l(\gamma)}\right)},$$

where γ is any admissible base of $\frac{\mu+\mu'}{2}$, is p.d. and independent of the choice of γ.

Given two objects z, z' and their respective molecular measures δ_z and $\delta_{z'}$, the computation of the IGV for two such objects requires in practice an admissible base of $\frac{\delta_z+\delta_{z'}}{2}$ as seen in Theorem 3.6. This admissible base can be chosen to be of the cardinality of the support of the mixture of δ_z and $\delta_{z'}$, or alternatively be

the simple merger of two admissible bases of z and z' with their weights divided by 2, without searching for overlapped points between both lists. This choice has no impact on the final value taken by the regularized IGV-kernel and can be arbitrated by computational considerations.

If we now take a practical look at the IGV's definition, we note that it can be applied but to cases where the component space $\mathcal{X}$ is Euclidian, and only if the studied measures can be summarized efficiently by their second order moments. These limitations do not seem very realistic in practice, since $\mathcal{X}$ may not have a vectorial structure, and the distribution of the components may not even be well represented by Gaussians in the Euclidian case. We propose to bypass this issue and introduce the usage of the IGV in a more flexible framework by using the kernel trick on the previous quantities, since the IGV of a measure can be expressed only through the dot-products between the elements of the support of the considered measure.

3.5 Inverse Generalized Variance on the RKHS associated with a Kernel κ

As with many quantities defined by dot-products, one is tempted to replace the usual dot-product matrix $\tilde{K}$ of Theorem 3.6 by an alternative Gram-matrix obtained through a p.d. kernel κ defined on $\mathcal{X}$. The advantage of such a substitution, which follows the well known "kernel trick" principle (Schölkopf and Smola, 2002), is multiple as it first enables us to use the IGV kernel on any non-vectorial space endowed with a kernel, thus in practice on any component space endowed with a kernel; second, it is also useful when $\mathcal{X}$ is a dot-product space where a non-linear kernel can however be used (e.g., using Gaussian kernel) to incorporate into the IGV's computation higher-order moment comparisons. We prove in this section that the inverse of the regularized generalized variance, computed in Proposition 4.7 through the centered dot-product matrix $\tilde{K}_\gamma$ of elements of any admissible base γ of μ, is still a positive definite quantity if we replace $\tilde{K}_\gamma$ by a centered Gram-matrix $\tilde{\mathcal{K}}_\gamma$, computed through an *a priori* kernel κ on $\mathcal{X}$, namely

$$\mathcal{K}_\gamma = [\kappa(x_i, x_j)]_{1 \leq i,j \leq d}$$
$$\tilde{\mathcal{K}}_\gamma = (I_d - \mathbb{1}_{d,d}\Delta_\gamma)\mathcal{K}_\gamma(I_d - \Delta_\gamma \mathbb{1}_{d,d}).$$

This substitution follows also a general principle when considering kernels on measures. The "kernelization" of a given kernel defined on measures to take into account a prior similarity on the components, when computationally feasible, is likely to improve its overall performance in classification tasks, as observed by Kondor and Jebara (2003) but also by Hein and Bousquet (2005) under the "Structural Kernel" appellation. The following theorem proves that this substitution is valid in the case of the IGV.

Theorem 3.7. *Let $\mathcal{X}$ be a set endowed with a p.d. kernel κ. The kernel*

$$k_\kappa^\eta(\mu,\mu') = \frac{1}{\det\left(\frac{1}{\eta}\tilde{\mathcal{K}}_\gamma \Delta_\gamma + I_{l(\gamma)}\right)}, \tag{3.3}$$

defined on two elements μ, μ' in $\mathrm{Mol}_+^1(\mathcal{X})$ is positive definite, where γ is any *admissible base of $\frac{\mu+\mu'}{2}$.*

Proof. Let $N \in \mathbb{N}$, $\mu_1, .., \mu_N \in \mathrm{Mol}_+^1(\mathcal{X})$ and $(c_i)_{i=1}^N \in \mathbb{R}^N$. Let us now study the quantity $\sum_{i=1}^N c_i c_j\, k_\kappa^\eta(\mu_i,\mu_j)$. To do so we introduce by the Moore-Aronszajn theorem (Berlinet and Thomas-Agnan, 2003, p.19) the reproducing kernel Hilbert space Ξ with reproducing kernel κ indexed on $\mathcal{X}$. The usual mapping from $\mathcal{X}$ to Ξ is denoted by ϕ, that is $\phi : \mathcal{X} \ni x \mapsto \kappa(x,\cdot)$. We define

$$\mathcal{Y} \stackrel{\text{def}}{=} \mathrm{supp}\left(\sum_{i=1}^N \mu_i\right) \subset \mathcal{X},$$

the finite set which numbers all elements in the support of the N considered measures, and

$$\Upsilon \stackrel{\text{def}}{=} \mathrm{span}\,\phi(\mathcal{Y}) \subset \Xi,$$

the linear span of the elements in the image of $\mathcal{Y}$ through ϕ. Υ is a vector space whose finite dimension is upper-bounded by the cardinality of $\mathcal{Y}$. Endowed with the dot-product inherited from Ξ, we further have that Υ is Euclidian. Given a molecular measure $\mu \in \mathrm{Mol}_+^1(\mathcal{Y})$, let $\phi(\mu)$ denote the image measure of μ in Υ, namely $\phi(\mu) = \sum_{x\in\mathcal{Y}} \mu(x)\delta_{\phi(x)}$. One can easily check that any admissible base $\gamma = (x_i, a_i)_{i=1}^d$ of μ can be used to provide an admissible base $\phi(\gamma) \stackrel{\text{def}}{=} (\phi(x_i), a_i)_{i=1}^d$ of $\phi(\mu)$. The weight matrices Δ_γ and $\Delta_{\phi(\gamma)}$ are identical and we further have $\tilde{\mathcal{K}}_\gamma = \tilde{K}_{\phi(\gamma)}$ by the reproducing property, where $\tilde{K}$ is defined by the dot-product of the Euclidian space Υ induced by κ. As a result, we have that $k_\kappa^\eta(\mu_i,\mu_j) = k_v^\eta(\phi(\mu_i),\phi(\mu_j))$ where k_v^η is defined on $\mathrm{Mol}_+^1(\Upsilon)$, ensuring the non-negativity

$$\sum_{i=1}^N c_i c_j\, k_\kappa^\eta(\mu_i,\mu_j) = \sum_{i=1}^N c_i c_j\, k_v^\eta(\phi(\mu_i),\phi(\mu_j)) \geq 0$$

and hence positive-definiteness of k_κ^η. □

As observed in the experimental section, the kernelized version of the IGV is more likely to be successful to solve practical tasks since it incorporates meaningful information on the components. Before observing these practical improvements, we provide a general study of the family of semigroup kernels on $M_+^b(\mathcal{X})$ by casting the theory of integral representations of positive definite functions on a semigroup (Berg et al., 1984) in the framework of measures, providing new results and possible interpretations of this class of kernels.

3.6 Integral Representation of p.d. Functions on a Set of Measures

In this section we study a general characterization of *all* p.d. functions on the whole semigroup $(M_+^b(\mathcal{X}),+)$, including thus measures which are not normalized. This characterization is based on a general integral representation theorem valid for any semigroup kernel, and is similar in spirit to the representation of p.d. functions obtained on Abelian groups through Bochner's theorem (Rudin, 1962). Before stating the main results in this section we need to recall basic definitions of semicharacters and exponentially bounded function (Berg et al., 1984, chap. 4).

Definition 3.8. *A real-valued function ρ on an Abelian semigroup $(S,+)$ is called a* semicharacter *if it satisfies the following properties:*

(i) $\rho(0)=1$

(ii) $\forall s,t\in S, \rho(s+t)=\rho(s)\rho(t)$.

It follows from the previous definition and the fact that $M_+^b(\mathcal{X})$ is *2-divisible* (i.e., $\forall\mu\in M_+^b(\mathcal{X}), \exists\mu'\in M_+^b(\mathcal{X})$ s.t. $\mu=2\mu'$) that semicharacters are nonnegative valued since it suffices to write that $\rho(\mu)=\rho(\frac{\mu}{2})^2$. Note also that semicharacters are trivially positive definite functions on S. We denote by S^* the set of semicharacters on $M_+^b(\mathcal{X})$, and by $\hat{S}\subset S^*$ the set of bounded semicharacters. S^* is a Hausdorff space when endowed with the topology inherited from $\mathbb{R}^S$ having the topology of pointwise convergence. Therefore we can consider the set of Radon measures on S^*, namely $M_+^b(S^*)$.

Definition 3.9. *A function $f: M_+^b(\mathcal{X})\to\mathbb{R}$ is called* exponentially bounded *if there exists a function $\alpha: M_+^b(\mathcal{X})\to\mathbb{R}_+$ (called an* absolute value*) satisfying $\alpha(0)=1$ and $\alpha(\mu+\mu')\leq\alpha(\mu)\alpha(\mu')$ for $\mu,\mu'\in M_+^b(\mathcal{X})$, and a constant $C>0$ such that:*

$$\forall\mu\in M_+^b(\mathcal{X}),\quad f(\mu)\leq C\alpha(\mu).$$

We can now state two general integral representation theorems for p.d. functions on semigroups (Berg et al., 1984, Theorems 4.2.5 and 4.2.8). These theorems being valid on any semigroup, they hold in particular on the particular semigroup $(M_+^b(\mathcal{X}),+)$.

Theorem 3.10.

- *A function $\varphi: M_+^b(\mathcal{X})\to\mathbb{R}$ is p.d. and exponentially bounded if and only if it has an integral representation:*

$$\varphi(s)=\int_{S^*}\rho(s)d\omega(\rho),$$

with $\omega\in M_+^c(S^*)$ (the set of Radon measures on S^* with compact support).

- *A function $\varphi : M_+^b(\mathcal{X}) \to \mathbb{R}$ is p.d. and bounded if and only if it has an integral representation of the form:*

$$\varphi(s) = \int_{\hat{S}} \rho(s) d\omega(\rho),$$

with $\omega \in M_+(\hat{S})$.

In both cases, if the integral representation exists, then there is uniqueness of the measure ω in $M_+(S^*)$.

In order to make these representations more constructive, we need to study the class of (bounded) semicharacters on $(M_+^b(\mathcal{X}), +)$. Even though we are not able to provide a complete characterization, even of bounded semicharacters, the following proposition introduces a large class of semicharacters, and completely characterizes the *continuous* semicharacters. For matters related to continuity of functions defined on $M_+^b(\mathcal{X})$, we will consider the weak topology of $M_+^b(\mathcal{X})$ which is defined in simple terms through the *portmanteau* theorem (Berg et al., 1984, Theorem 2.3.1). Note simply that if μ_n converges to μ in the weak topology then for any *bounded* measurable and continuous function f we have that $\mu_n[f] \to \mu[f]$. We further denote by $C(\mathcal{X})$ the set of continuous real-valued functions on $\mathcal{X}$ and by $C^b(\mathcal{X})$ its subset of bounded functions. Both sets are endowed with the topology of pointwise convergence. For a function $f \in \mathbb{R}^{\mathcal{X}}$ we write ρ_f for the function $\mu \mapsto e^{\mu[f]}$ when the integral is well defined.

Proposition 3.11. *A semicharacter $\rho : M_+^b(\mathcal{X}) \to \mathbb{R}$ is continuous on $(M_+^b(\mathcal{X}), +)$ endowed with the weak topology if and only if there exists $f \in C^b(\mathcal{X})$ such that $\rho = \rho_f$. In that case, ρ is a bounded semicharacter on $M_+^b(\mathcal{X})$ if and only if $f \leq 0$.*

Proof. For a continuous and bounded function f, the semicharacter ρ_f is well-defined. If a sequence μ_n in $M_+^b(\mathcal{X})$ converges to μ weakly, we have $\mu_n[f] \to \mu[f]$, which implies the continuity of ρ_f. Conversely, suppose ρ is weakly continuous. Define $f : \mathcal{X} \to [-\infty, \infty)$ by $f(x) = \log \rho(\delta_x)$. If a sequence x_n converges to x in $\mathcal{X}$, obviously we have $\delta_{x_n} \to \delta_x$ in the weak topology, and

$$\rho(\delta_{x_n}) \quad \to \quad \rho(\delta_x),$$

which means the continuity of f. To see the boundedness of f, assume the contrary. Then, we can find $x_n \in \mathcal{X}$ such that either of $0 < f(x_n) \to \infty$ or $0 > f(x_n) \to -\infty$ holds. Let $\beta_n = |f(x_n)|$. Because the measure $\frac{1}{\beta_n}\delta_{x_n}$ converges weakly to zero, the continuity of ρ means

$$\rho\big(\tfrac{1}{\beta_n}\delta_{x_n}\big) \to 1,$$

which contradicts with the fact $\rho(\frac{1}{\beta_n}\delta_{x_n}) = e^{\frac{1}{\beta_n} f(x_n)} = e^{\pm 1}$. Thus, ρ_f is well-defined, weakly continuous on $M_+^b(\mathcal{X})$ and equal to ρ on the set of molecular measures. It is further equal to ρ on $M_+^b(\mathcal{X})$ through the denseness of molecular measures in $M_+^b(\mathcal{X})$, both in the weak and the pointwise topology (Berg et al.,

1984, Proposition 3.3.5). Finally suppose now that ρ_f is bounded and that there exists x in $\mathcal{X}$ such that $f(x) > 0$. By $\rho_f(n\,\delta_x) = e^{nf(x)}$ which diverges with n we see a contradiction. The converse is straightforward. □

Let ω be a bounded nonnegative Radon measure on the Hausdorff space of continuous real-valued functions on $\mathcal{X}$, namely $\omega \in M_+^b(C(\mathcal{X}))$. Given such a measure, we first define the subset M_ω of $M_+^b(\mathcal{X})$ as

$$M_\omega = \{\mu \in M_+^b(\mathcal{X}) \mid \sup_{f \in \operatorname{supp}\omega} \mu[f] < +\infty\}.$$

M_ω contains the null measure and is a semigroup.

Corollary 3.12. *For any bounded Radon measure $\omega \in M_+^b(C(\mathcal{X}))$, the following function φ is a p.d. function on the semigroup $(M_\omega, +)$:*

$$\varphi(\mu) = \int_{C(\mathcal{X})} \rho_f(\mu)\, d\omega(f). \tag{3.4}$$

If $\operatorname{supp}\omega \subset C^b(\mathcal{X})$ then φ is continuous on M_ω endowed with the topology of weak convergence.

Proof. For $f \in \operatorname{supp}\omega$, ρ_f is a well defined semicharacter on M_ω and hence positive definite. Since

$$\varphi(\mu) \leq |\omega| \sup_{f \in \operatorname{supp}\omega} \mu[f]$$

is bounded, φ is well defined and hence positive definite. Suppose now that $\operatorname{supp}\omega \subset C^b(\mathcal{X})$ and let μ_n be a sequence of M_ω converging weakly to μ. By the bounded convergence theorem and continuity of all considered semicharacters (since all considered functions f are bounded) we have that:

$$\lim_{n\to\infty} \varphi(\mu_n) = \int_{C(\mathcal{X})} \lim_{n\to\infty} \rho_f(\mu_n)\, d\omega(f) = \varphi(\mu).$$

and hence φ is continuous w.r.t the weak topology. □

When the measure ω is chosen in such a way that the integral (3.4) is tractable or can be approximated, then a valid p.d. kernel for measures is obtained; an example involving mixtures over exponential families is provided in Section 3.7.

Before exploiting this constructive representation, a few remarks should be pointed out. When using non-bounded functions (as is the case when using expectation or second-order moments of measures) the continuity of the integral φ is left undetermined to our knowledge, even when its existence is ensured. However, when $\mathcal{X}$ is compact we have that $C(\mathcal{X}) = C^b(\mathcal{X})$ and hence continuity on M_ω of any function φ constructed through corollary 3.12. Conversely, there exist continuous p.d. functions on $(M_+^b(\mathcal{X}), +)$ that can not be represented in the form (3.4). Although any continuous p.d. function can necessarily be represented as an integral

of semicharacters by Theorem 3.10, the semicharacters involved in the representation are not necessarily continuous as in (3.4). An example of such a continuous p.d. function written as an integral of non-continuous semicharacters is exposed in Appendix A. It is an open problem to our knowledge to fully characterize continuous p.d. functions on $(M_+^b(\mathcal{X}), +)$.

3.7 Projection on Exponential Families through Laplace's Approximation

The constructive approach presented in corollary 3.12 can be used in practice to define kernels by restricting the space $C(\mathcal{X})$ to subspaces where computations are tractable. A natural way to do so is to consider a vector space of finite dimension s of $C(\mathcal{X})$, namely the span of a free family of s non-constant functions $f_1, ..., f_s$ of $C(\mathcal{X})$, and define a measure on that subspace by applying a measure on the weights associated with each function. The previous integral representation (3.4) would then take the form:

$$\varphi(\mu) = \int_\Theta e^{\mu[\sum_{i=1}^s \theta_i f_i]}\, \omega(d\theta),$$

where ω is now a bounded measure on a compact subset $\Theta \subseteq \mathbb{R}^s$ and μ is such that $\mu[f_i] < +\infty$ for $1 \leq i \leq s$. The subspace of $C(\mathcal{X})$ considered in this section is however slightly different, in order to take advantage of the natural benefits of exponential densities generated by all functions $f_1, ..., f_s$. Following Amari and Nagaoka (2001, p.69), this requires the definition of the cumulant generating function of ν with respect to $f_1, ..., f_s$ as

$$\psi(\theta) \stackrel{\text{def}}{=} \log \nu[e^{\sum_{i=1}^s \theta_i f_i}],$$

such that for each $\theta \in \Theta$,

$$p_\theta \stackrel{\text{def}}{=} \exp\left(\sum_{i=1}^s \theta_i f_i - \psi(\theta)\right) \nu,$$

is a probability density, which defines an exponential family of densities on $\mathcal{X}$ as θ varies in Θ. Rather than the direct span of functions $f_1, ..., f_s$ on Θ, this is equivalent to considering the hypersurface $\{\sum_{i=1}^s \theta_i f_i - \psi(\theta)\}$ in $\text{span}\{f_1, .., f_s, -1\}$. This yields the following expression:

$$\varphi(\mu) = \int_\Theta e^{\mu[\sum_{i=1}^s \theta_i f_i - \psi(\theta)]}\, \omega(d\theta).$$

Following the notations of Amari and Nagaoka (2001) the η-parameters (or expectation parameters) of μ are defined as

$$\hat{\eta}_i \stackrel{\text{def}}{=} \frac{1}{|\mu|}\mu[f_i],\ 1 \leq i \leq s,$$

and $\hat{\theta}$ stands for the θ-parameters of $\hat{\eta}$. We assume in the following approximations that $\hat{\theta} \in \Theta$ and recall two identities (Amari and Nagaoka, 2001, Chapters 3.5 & 3.6):

$$\chi(\theta) \stackrel{\text{def}}{=} \sum_{i=1}^{s} \theta_i \eta_i - \psi(\theta) = -h(\theta), \text{ the dual potential,}$$

$$D(\theta||\theta') = \psi(\theta) + \chi(\theta') - \sum_{i=1}^{s} \theta_i \eta'_i, \text{ the KL divergence,}$$

where we used the abbreviations $h(\theta) = h(p_\theta)$ and $D(\theta||\theta') = D(p_\theta||p_{\theta'})$. We can then write

$$\begin{aligned}
\mu[\sum_{i=1}^{s} \theta_i f_i - \psi(\theta)] &= |\mu| \left(\sum_{i=1}^{s} \theta_i \hat{\eta}_i - \psi(\theta) \right) \\
&= |\mu| \left(\sum_{i=1}^{s} \hat{\theta}_i \hat{\eta}_i - \psi(\hat{\theta}) + \sum_{i=1}^{s} (\theta_i - \hat{\theta}_i) \hat{\eta}_i + \psi(\hat{\theta}) - \psi(\theta) \right) \\
&= -|\mu| \left(h(\hat{\theta}) + D(\hat{\theta}||\theta) \right),
\end{aligned}$$

to obtain the following factorized expression,

$$\varphi(\mu) = e^{-|\mu| h(\hat{\theta})} \int_{\Theta} e^{-|\mu| D(\hat{\theta}||\,\theta)} \omega(d\theta). \tag{3.5}$$

The quantity $e^{-|\mu| h(\hat{\theta})}$ was already evoked in Section 3.3.2 when multivariate normal distributions were used to express the IGV kernel. When $\mathcal{X}$ is an Euclidian space of dimension n, this is indeed equivalent to defining $s = n + n(n+1)/2$ base functions, more precisely $f_i = x_i$ and $f_{ij} = x_i x_j$, and dropping the integral of Equation (3.5). Note that such functions are not bounded and that M_ω corresponds here to the set of measures with finite first and second order moments.

The integral of Equation (3.5) cannot be computed in a general case. The use of conjugate priors can however yield exact calculations, such as in the setting proposed by Cuturi and Vert (2005). In their work $\mathcal{X}$ is a finite set of short sequences formed over an alphabet, functions f_i are all possible indicator functions of $\mathcal{X}$ and ω is an additive mixture of Dirichlet priors. The kernel value is computed through a factorization inspired by the context-tree weighting algorithm (Willems et al., 1995). In the general case a numerical approximation can also be derived using Laplace's method (Dieudonné, 1968) under the assumption that $|\mu|$ is large enough. To do so, first notice that

$$\frac{\partial D(\hat{\theta}||\theta)}{\partial \theta_i} \Big|_{\theta = \hat{\theta}} = \frac{\partial \psi}{\partial \theta_i} \Big|_{\theta = \hat{\theta}} - \hat{\eta}_i = 0,$$

$$\frac{\partial D(\hat{\theta}||\theta)}{\partial \theta_i \partial \theta_j} = \frac{\partial \psi}{\partial \theta_i \partial \theta_j} = g_{ij}(\theta),$$

where $G_\theta = [g_{ij}(\theta)]$ is the Fisher information matrix computed in θ and hence a p.d. matrix. The following approximation then holds:

$$\varphi(\mu) \underset{|\mu|\to\infty}{\sim} e^{-|\mu|h(\hat{\theta})} \int_{\mathbb{R}^s} \omega(\hat{\theta}) e^{-\frac{|\mu|}{2}(\theta-\hat{\theta})^\top G_{\hat{\theta}}(\theta-\hat{\theta})} d\theta = e^{-|\mu|h(\hat{\theta})} \left(\frac{2\pi}{|\mu|}\right)^{\frac{s}{2}} \frac{\omega(\hat{\theta})}{\sqrt{\det G_{\hat{\theta}}}}$$

which can be simplified by choosing ω to be Jeffrey's prior (Amari and Nagaoka, 2001, p.44), namely

$$\omega(d\theta) = \frac{1}{V}\sqrt{\det G_\theta}\, d\theta, \qquad \text{where } V = \int_\Theta \sqrt{\det G_\theta}\, d\theta.$$

Up to a multiplication by V this provides an approximation of φ by $\tilde{\varphi}$ as

$$\varphi(\mu) \underset{|\mu|\to\infty}{\sim} \tilde{\varphi}(\mu) \stackrel{\text{def}}{=} e^{-|\mu|h(\hat{\theta})} \left(\frac{2\pi}{|\mu|}\right)^{\frac{s}{2}}.$$

The η-coordinates of μ are independent of the total weight $|\mu|$, hence $\tilde{\varphi}(2\mu) = \tilde{\varphi}(\mu)^2(\frac{|\mu|}{4\pi})^{\frac{s}{2}}$. This identity can be used to propose a renormalized kernel for two measures as

$$k(\mu,\mu') \stackrel{\text{def}}{=} \frac{\tilde{\varphi}(\mu+\mu')}{\sqrt{\tilde{\varphi}(2\mu)\tilde{\varphi}(2\mu')}} = \frac{e^{-(|\mu+\mu'|)h(p_{\mu+\mu'})}}{e^{-|\mu|h(p_\mu)-|\mu'|h(p_{\mu'})}} \left(\frac{2\sqrt{|\mu||\mu'|}}{|\mu|+|\mu'|}\right)^{\frac{s}{2}}.$$

where p_μ stands for $p_{\hat{\theta}_\mu}$. When μ and μ' are normalized such that their total weight coincides and is equal to β, we have that

$$k(\mu,\mu') = e^{-2\beta\left(h(p_{\mu''})-\frac{h(p_\mu)+h(p_{\mu'})}{2}\right)}, \tag{3.6}$$

where $\mu'' = \mu + \mu'$. From Equation (3.6), we see that β can be tuned in practice and thought of as a width parameter. It should be large enough to ensure the consistency of Laplace's approximation and thus positive definiteness, while not too large at the same time to avoid diagonal dominance issues. In the case of the IGV kernel this tradeoff can however be put aside since the inverse of the IGV is directly p.d. as was proved in Proposition 3.3. However and to our knowledge this assertion does not stand in a more general case when the functions $f_1, ..., f_s$ are freely chosen.

3.8 Experiments on images of the MNIST database

We present in this section experimental results and discussions on practical implementations of the IGV kernels on a benchmark experiment of handwritten digits classification. We focus more specifically on the kernelized version of the IGV and discuss its performance with respect to other kernels. The entropy kernel performed very poorly in the series of experiments presented here, besides requiring a time consuming Monte Carlo computation, which is why we do not consider it in this section. We believe however that in more favourable cases, notably when the considered measures are multinomials, the entropy kernel and its structural variants (Hein and Bousquet, 2005) may provide good results.

3.8.1 Linear IGV Kernel

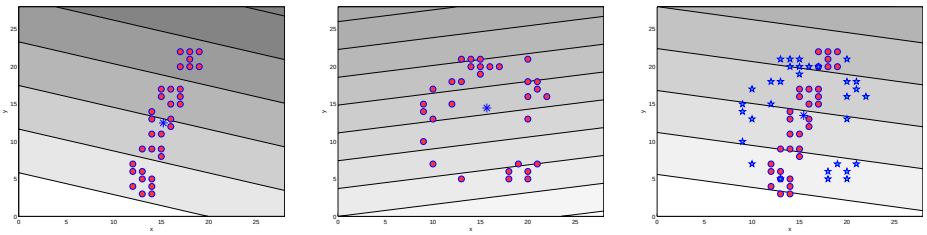
Following the previous work of Kondor and Jebara (2003), we have conducted experiments on 500 and 1000 images (28×28 pixels) taken from the MNIST database of handwritten digits (black shapes on a white background), with 50 (resp. 100) images for each digit. To each image z we randomly associate a set of d distinct points which are black (intensity superior to 190) in the image. In this case the set of components is $\{1, .., 28\} \times \{1, .., 28\}$ which we map onto points with coordinates between 0 and 1, thus defining $\mathcal{X} = [0, 1]^2$. The linear IGV kernel as described in Section 3.3.2 is equivalent to using the linear kernel $\kappa((x_1, y_1), (x_2, y_2)) = x_1 x_2 + y_1 y_2$ on a non-regularized version of the kernelized-IGV. It also boils down to fitting Gaussian bivariate-laws on the points and measuring the similarity of two measures by performing variance estimation on the samples taken first separately and then together. The resulting variances can be diagonalized to obtain three diagonal variance matrices, which can be seen as performing PCA on the sample,

$$\Sigma(\mu) = \begin{pmatrix} \Sigma_{1,1} & 0 \\ 0 & \Sigma_{2,2} \end{pmatrix}, \quad \Sigma(\mu') = \begin{pmatrix} \Sigma'_{1,1} & 0 \\ 0 & \Sigma'_{2,2} \end{pmatrix}, \quad \Sigma(\mu'') = \begin{pmatrix} \Sigma''_{1,1} & 0 \\ 0 & \Sigma''_{2,2} \end{pmatrix}.$$

and the value of the kernel is computed through

$$k_v(\mu, \mu') = \frac{\sqrt{\Sigma_{1,1}\Sigma_{2,2}\,\Sigma'_{1,1}\Sigma'_{2,2}}}{\Sigma''_{1,1}\Sigma''_{2,2}}.$$

This ratio is for instance equal to 0.3820 for two handwritten digits in the case shown in Figure 3.2. The linear IGV manages a good discrimination between ones and



$\Sigma_{1,1} = 0.0552$ $\Sigma_{2,2} = 0.0013$ | $\Sigma'_{1,1} = 0.0441$ $\Sigma'_{2,2} = 0.0237$ | $\Sigma''_{1,1} = 0.0497$ $\Sigma''_{2,2} = 0.0139$

Figure 3.2. *Weighted PCA of two different measures and their mean, with their first principal component shown. Below are the variances captured by the first and second principal components, the generalized variance being the product of the two values.*

zeros. Indeed, ones are shaped as sticks, and hence usually have a strong variance carried by their first component, followed by a weak second component. On the other hand, the variance of zeros is more equally distributed between the first and second axes. When both weighted sets of points are united, the variance of the mean

of both measures has an intermediary behaviour in that respect, and this suffices to discriminate numerically both images. However this strategy fails when using numbers which are not so clearly distinct in shape, or more precisely whose surface cannot be efficiently expressed in terms of Gaussian ellipsoids. To illustrate this we show in Figure 3.3 the Gram matrix of the linear IGV on 60 images, namely 20 zeros, 20 ones and 20 twos. Though images of ones can be efficiently discriminated from the two other digits, we clearly see that this is not the case between zeros and twos, whose support may seem similar if we try to capture them through Gaussian laws. In practice, the results obtained with the linear IGV on this particular task where so unadapted to the learning goal that the SVM's trained based on that methodology did not converge in most cases, which is why we discarded it.

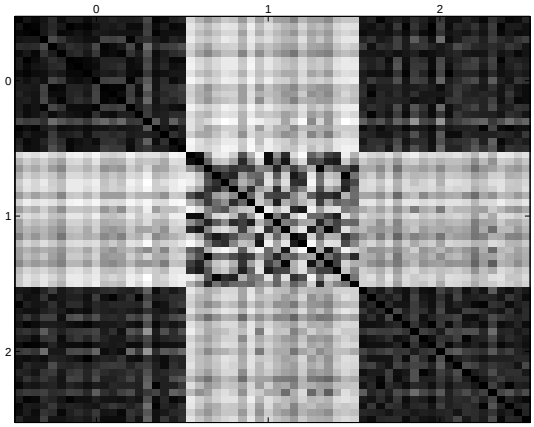


Figure 3.3. *Normalized Gram matrix computed with the linear IGV kernel of twenty images of "0", "1" and "2" displayed in that order. Darker spots mean values closer to 1, showing that the restriction to "0" and "1" yields good separation results, while "0" and "2" can hardly be discriminated using variance analysis.*

3.8.2 Kernelized IGV

Following previous works (Kondor and Jebara, 2003; Wolf and Shashua, 2003) and as suggested in the initial discussion of Section 3.5, we use in this section a Gaussian kernel of width σ to incorporate a prior knowledge on the pixels, and equivalently to define the reproducing kernel Hilbert space Ξ by using

$$\kappa\left((x_1,y_1),(x_2,y_2)\right) = e^{-\frac{(x_1-x_2)^2+(y_1-y_2)^2}{2\sigma^2}}.$$

As pointed out by Kondor and Jebara (2003), the pixels are no longer seen as points but rather as functions (Gaussian bells) defined on the components space $[0,1]^2$. To illustrate this approach we show in Figure 3.8.2 the first four eigenfunctions of three measures μ_1, μ_0 and $\frac{\mu_1+\mu_0}{2}$ built from the image of a handwritten "1" and

"0" with their corresponding eigenvalues, as well as for images of "2" and "0" in Figure 3.8.2.

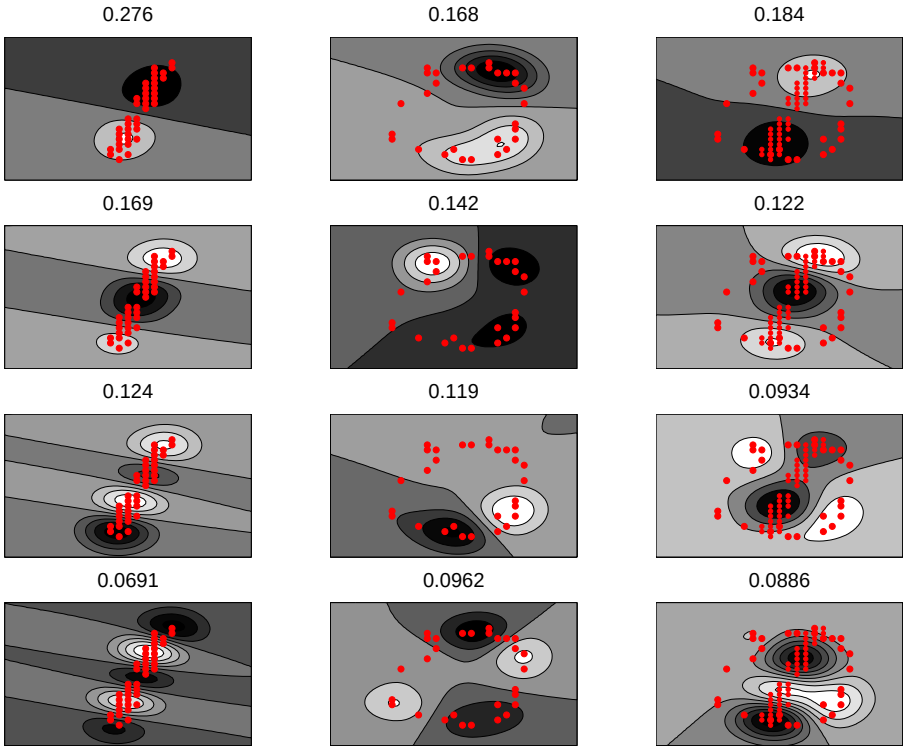


Figure 3.4. *The four first eigenfunctions of respectively three empirical measures μ_1 (first column), μ_0 (second column) and $\frac{\mu_1+\mu_0}{2}$ (third column), displayed with their corresponding eigenvalues, using $\eta = 0.01$ and $\sigma = 0.1$.*

Setting σ, the width of κ, to define the functions contained in the RKHS Ξ is not enough to fully characterize the values taken by the kernelized IGV. We further need to define η, the regularization parameter, to control the weight assigned to smaller eigenvalues in the spectrum of Gram matrices. Both parameters are strongly related, since the value of σ controls the range of the typical eigenvalues found in the spectrum of Gram matrices of admissible bases, whereas η acts as a scaling parameter for those eigenvalues as can be seen in Equation (3.3). Indeed, using a very small σ value, which means Ξ is only defined by peaked Gaussian bells around each pixels, yields diagonally dominant Gram matrices very close to the identity matrix. The resulting eigenvalues for $\tilde{\mathcal{K}}\Delta$ are then all very close to $\frac{1}{d}$, the inverse of the amount of considered points. On the contrary, a large value for σ yields higher values for the kernel, since all points would be similar to each other and Gram matrices would turn close to the matrix $\mathbb{1}_{d,d}$ with a single significant eigenvalue and

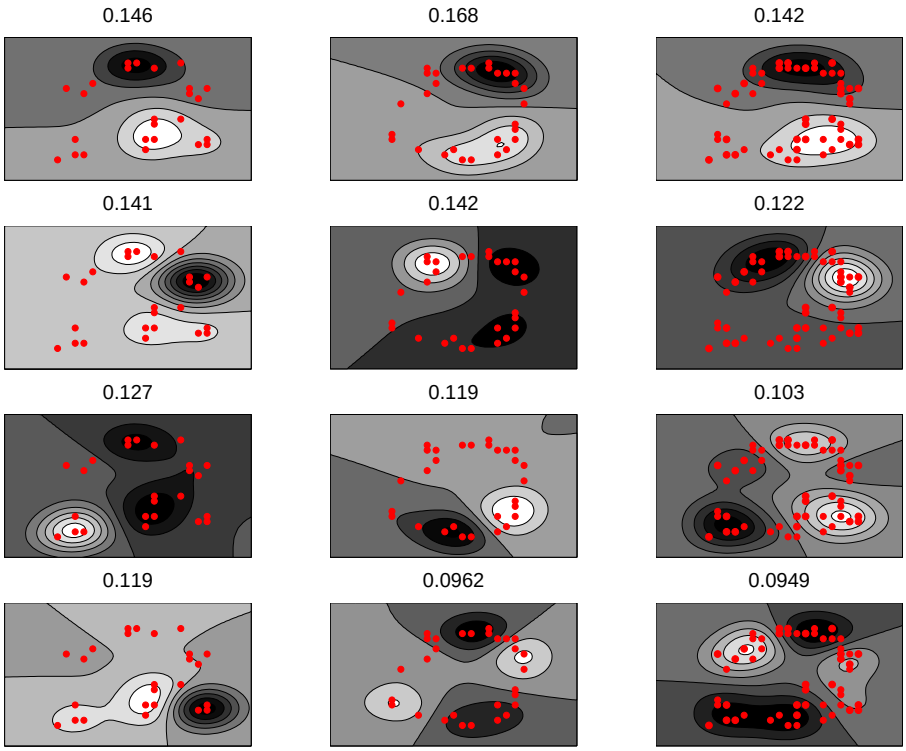


Figure 3.5. *Same representation as in Figure 3.8.2, with* μ_2, μ_0 *and* $\frac{\mu_2+\mu_0}{2}$.

all others close to zero. We address these issues and study the robustness of the final output of the k-IGV kernel in terms of classification error by doing preliminary experiments where both η and σ vary freely.

3.8.3 Experiments on the SVM Generalization Error

To study the behaviour and the robustness of the IGV kernel under different parameter settings, we used two ranges of values for η and σ:

$$\eta \in 10^{-2} \times \{0.1, 0.3, 0.5, 0.8, 1, 1.5, 2, 3, 5, 8, 10, 20\}$$
$$\sigma \in \{0.05, 0.1, 0.12, 0.15, 0.18, 0.20, 0.25, 0.3\}.$$

For each kernel k_κ^η defined by a (σ, η) couple, we trained 10 binary SVM classifiers (each one trained to recognize each digit versus all other digits) on a training fold of our 500 images dataset such that the proportion of each class was kept to be one tenth of the total size of the training set. Using then the test fold, our decision for each submitted image was determined by the highest SVM score proposed by the 10 trained binary SVM's. To determine train and test points, we

led a 3-fold cross validation, namely randomly splitting our total dataset into 3 balanced subsets, using successively 2 subsets for training and the remaining one for testing (that is roughly 332 images for training and 168 for testing). The test error was not only averaged on those cross-validations folds but also on 5 different fold divisions. All the SVM experiments in this experimental section were run using the spider[15] toolbox. Most results shown here did not improve by choosing different soft margin C parameters, we hence just set $C = \infty$ as suggested by default by the authors of the toolbox.

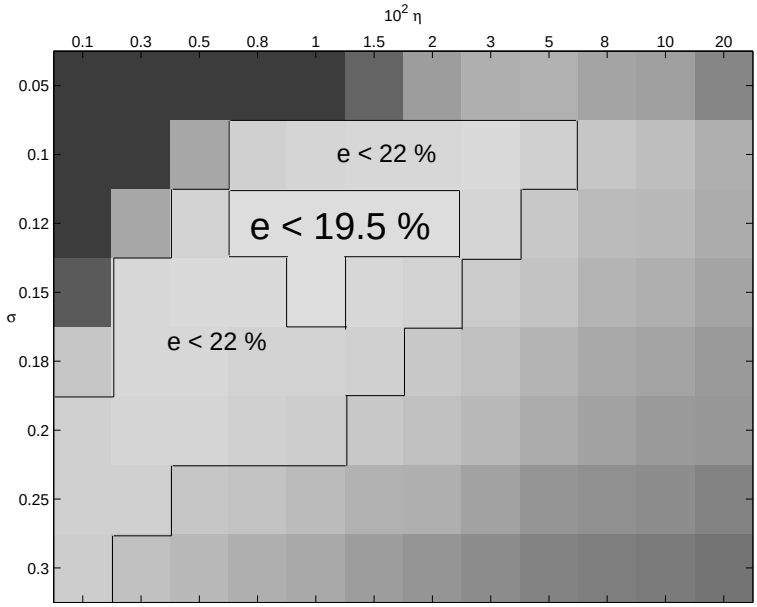


Figure 3.6. *Average test error (displayed as a grey level) of different SVM handwritten character recognition experiments using 500 images from the MNIST database (each seen as a set of 25 to 30 randomly selected black pixels), carried out with 3-fold (2 for training, 1 for test) cross validations with 5 repeats, where parameters η (regularization) and σ (width of the Gaussian kernel) have been tuned to different values.*

The error rates are graphically displayed in Figure 3.6 using a grey-scale plot. Note that for this benchmark the best testing errors were reached using a σ value of 0.12 with an η parameter within 0.008 and 0.02, this error being roughly 19.5%. All values below and on the right side of this zone are below 32.5%, which is the value reached on the lower right corner. All standard deviations with respect to multiple cross-validations of those results were inferior to 2.3%, the whole region under 22% being under a standard deviation of 1%. Those preliminary tests show that the

[15] see http://www.kyb.tuebingen.mpg.de/bs/people/spider/

IGV kernel has an overall robust performance within what could be considered as a sound range of values for both η and σ. Note that the optimal range of parameter found in this experiment only applies to the specific sampling procedure that was used in this case (25 to 30 points), and may not be optimal for larger matrices. However the stability observed here led us to discarding further tuning of σ and η when the amount of sampled points is different. We simply applied $\sigma = 0.1$ and $\eta = 0.01$ for the remaining of the experimental section.

As in Kondor and Jebara (2003), we also compared the results obtained with the k-IGV to the standard RBF kernel performed on the images seen as binary vectors of $\{0,1\}^{28\times 28}$ further normalized so that their components sums up to 1. Using the same range for σ that was previously tested, we applied the formula $k(z,z') = e^{-\frac{\|z-z'\|}{2\sigma^2}}$. Since the RBF kernel is grounded on the exact overlapping between two images we expect it to perform poorly with few pixels and significantly better when d grows, while we expect the k-IGV to capture more quickly the structure of the images with fewer pixels through the kernel κ. This is illustrated in Figure 3.7 where the k-IGV outperforms significantly the RBF kernel, reaching with a sample of less than 30 points a performance the RBF kernel only reaches above 100 points. Taking roughly all black points in the images, by setting $d = 200$ for instance, the RBF kernel error is still 17.5%, an error the IGV kernel reaches with roughly 35 points.

Finally, we compared the kernelized-version of the Bhattacharrya kernel (k-B) proposed in Kondor and Jebara (2003), the k-IGV, the polynomial kernel and the RBF kernel by using a larger database of the first 1,000 images in MNIST (100 images for each of the 10 digits), selecting randomly $d = 40, 50, 60, 70$ and 80 points and performing the cross-validation methodology previously detailed. The polynomial kernel was performed seeing the images as binary vectors of $\{0,1\}^{28\times 28}$ and applying the formula $k_{b,d}(z,z') = (z \cdot z' + b)^d$. We followed the observations of Kondor and Jebara (2003) concerning parameter tuning for the k-B kernel but found out that it performed better using the same set of parameters used for the k-IGV. The results presented in Table 5.1 of the k-IGV kernel show a consistent improvement over all other kernels for this benchmark of 1000 images, under all sampling schemes.

We did not use the kernel described by Wolf and Shashua (2003) in our experiments because of its poor scaling properties for a large amount of considered points. Indeed, the kernel proposed by Wolf and Shashua (2003) takes the form of the product of d cosines values where d is the cardinality of the considered sets of points, thus yielding negligible values in practice when d is large as in our case. Their SVM experiments were limited to 6 or 7 points while we mostly consider lists of more than 40 points here. This problem of poor scaling which in practice produces a diagonal-dominant kernel led us to discarding this method in our comparison. All semigroup kernels presented in this paper are grounded on statistical estimation, which makes their values stable under variable sizes of samples through renormalization, a property shared with the work of Kondor and Jebara (2003). Beyond a minimal amount of points needed to perform sound estimation, the size of submitted samples influences positively the accuracy of the k-IGV kernel. A

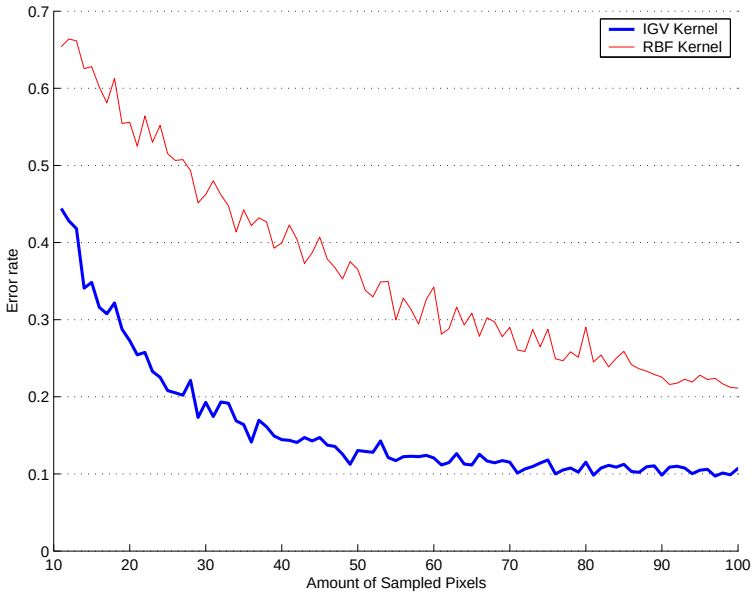


Figure 3.7. *Average test error with RBF ($\sigma = 0.2$) and k-IGV ($\sigma = 0.1$ and $\eta = 0.01$) kernels led on 90 different samplings of 500 images. The curves show an overall trend that both kernels perform better when they are given more points to compute the similarity between two images. If we consider $d = 200$, the RBF kernel error is 0.175, that is 17.5%, a threshold the IGV kernel reaches with slightly more than 35 points. Each sampling corresponds to a different amount of sampled points d, those samplings being ordered increasingly with d. Each sampling has been performed independently which explains the bumpiness of those curves.*

large sample size can lead however to computational problems since the value of the k-IGV-kernel requires not only the computation of the centered Gram-matrix $\mathcal{K}$ and a few matrix multiplications, but also the computation of a determinant, an operation which can quickly become prohibitive since it has a complexity of $O(d^{2.3})$ where d is the size of the considered Gram matrix. Although we did not optimize the computations of both k-B and k-IGV kernels (by storing precomputed values for instance or using numerical approximations in the computation of the determinant), this computational cost in the case of a naive implementation, illustrated by the running times displayed in Table 5.1, remains an issue that needs to be addressed in practical applications.

3.9 Closing remarks

We presented in this work a new family of kernels between measures. Such kernels are defined through prior functions which should ideally quantify the concentration

Sample Size	Gaussian $\sigma = 0.1$	Polynomial $b = 10$; $d = 4$	k-B $\eta = 0.01$; $\sigma = 0.1$	k-IGV $\eta = 0.01$; $\sigma = 0.1$
40 pixels	32.2 (1)	31.3 (1.5)	19.1 (1500)	16.2 (1000)
50 "	28.5 (1)	26.3 (1.5)	17.1 (2500)	14.7 (1400)
60 "	24.5 (1)	22.0 (1.5)	15.8 (3600)	14.6 (2400)
70 "	22.2 (1)	19.5 (1.5)	15.1 (4100)	13.1 (2500)
80 "	20.3 (1)	17.4 (1.5)	14.5 (5500)	12.8 (3200)

Table 3.1. *SVM Error rate in percents of different kernels used on a benchmark test of recognizing digits images, where only 40 to 80 black points where sampled from the original images. The 1,000 images where randomly split into 3 balanced sets to perform cross validation (2 for training and 1 for testing), the error being first averaged over 5 such splits, the whole process being repeated again over 3 different random samples of points. Running times are indicated in minutes.*

of a measure. Once such a function is properly defined, the kernel computation goes through the evaluation of the function on the two measures to be compared and on their mixture. As expected when dealing with concentration of measures, two intuitive tools grounded on information theory and probability, namely entropy and variance, prove to be useful to define such functions. Their expression is however still complex in terms of computational complexity, notably for the k-IGV kernel. Computational improvements or numerical simplifications should be brought forward to ensure a feasible implementation for large-scale tasks involving tens of thousands of objects.

An attempt to define and understand the general structure of p.d. functions on measures was also presented, through a representation as integrals of elementary functions known as semicharacters. We are investigating further theoretical properties and characterizations of both semicharacters and positive definite functions on measures. The choice of alternative priors on semicharacters to propose other meaningful kernels, with convenient properties on molecular measures for instance, is also a subject of future research. As for practical applications, these kernels can be naturally applied on complex objects seen as molecular measures. We also expect to perform further experiments to measure the performance of semigroup kernels on a diversified sample of challenging tasks, including cases where the space of components is not a vector space, notably when the considered measures are multinomials on a finite component space endowed with a kernel.

Appendix A : an Example of Continuous Positive Definite Function Given by Noncontinuous Semicharacters

Let $\mathcal{X}$ be the unit interval $[0,1]$ hereafter. For any t in $\mathcal{X}$, a semicharacter on $M_+^b(\mathcal{X})$ is defined by

$$\rho_{h_t}(\mu) = e^{\mu([0,t])},$$

where $h_t(x) = I_{[0,t]}(x)$ is the index function of the interval $[0,t]$. Note that ρ_{h_t} is *not* continuous for $t \in [0,1)$ by Proposition 3.11.

For $\mu \in M_+^b(\mathcal{X})$, the function $t \mapsto \mu([0,t))$ is bounded and non-decreasing, thus, Borel-measurable, since the discontinuous points are countable at most. A positive definite function on $M_+^b(\mathcal{X})$ is defined by

$$\varphi(\mu) = \int_0^1 \rho_{h_t}(\mu) dt.$$

This function is continuous, while it is given by the integral of noncontinuous semicharacters.

Proposition *The positive definite function φ is continuous and exponentially bounded.*

Proof. Suppose μ_n converges to μ weakly in $M_+^b(\mathcal{X})$. We write $F_n(t) = \mu_n([0,t])$ and $F(t) = \mu([0,t])$. Because μ_n and μ are finite measures, the weak convergence means

$$F_n(t) \quad \to \quad F(t)$$

for any continuous point of F. Since the set of discontinuous points of F is at most countable, the above convergence holds almost everywhere on X with Lebesgue measure. From the weak convergence, we have $F_n(1) \to F(1)$, which means there exists $M > 0$ such that $\sup_{t\in\mathcal{X}, n\in\mathbb{N}} F_n(t) < M$. By the bounded convergence theorem, we obtain

$$\lim_{n\to\infty} \varphi(\mu_n) = \lim_{n\to\infty} \int_0^1 e^{F_n(t)} dt = \int_0^1 e^{F(t)} dt = \varphi(\mu).$$

For the exponential boundedness, by taking an absolute value $\alpha(\mu) = e^{\mu(X)}$, we have

$$|\varphi(\mu)| \leq \int_0^1 \alpha(\mu) dt = \alpha(\mu).$$

□

Chapter 4

Semigroup Spectral Functions on Measures

Résumé

Nous poursuivons dans ce chapitre la caractérisation de noyaux de semigroupe pour mesures présentés dans le chapitre précédent. Nous nous intéressons plus spécifiquement aux noyaux définis positifs de deux mesures qui sont calculés à partir du spectre de la matrice de variance du mélange de ces deux mesures. Nous montrons que les fonctions caractéristiques du cône des matrices définies positives $\mathbf{\Sigma}_n^+$ munies de différentes mesures de probabilité peuvent être directement utilisées pour définir des noyaux de ce type, en les appliquant directement à la matrice de variance du mélange considéré. Nous proposons de nouvelles formules pour certains cas particuliers de ces fonctions caractéristiques, que nous particularisons dans les cas simples où les mesures sont des mesures moléculaires ou des histogrammes lorsque $\mathcal{X}$ est fini.

4.1 Introduction

Defining positive definite kernels on measures has attracted a lot of attention recently (Cuturi et al., 2005; Jebara et al., 2004; Hein and Bousquet, 2005; Lafferty and Lebanon, 2005), motivated by potential applications of kernel methods on composite objects seen as measures on spaces of components. The key idea behind this approach is to represent an object as a measure, possibly a distribution in a statistical model, and use kernels (or combinations of kernels as proposed in Chapter 5) on these representations. The advantage of this approach is two fold:

- First, and once the space of components is set, the representation of an object as a measure can be efficiently obtained through statistical estimation, to represent and regularize properly these measures, e.g., the use of uniform Dirichlet priors in Cuturi and Vert (2005) to regularize letters' counters or tfidf type frequencies for words (Lafferty and Lebanon, 2005; Joachims, 2002).
- Second, defining the kernel on such measures is a matter of choosing the right metric from a large collection of available metrics (Amari and Nagaoka, 2001; Hein and Bousquet, 2005), using for instance cross validation.

However, two major and independent issues arise when following this approach: the arbitrariness in the definition of the space of components, and the fact that a single measure for the whole object (that is a single histogram if the space of components is finite) might be a poor representation. The latter issue has been recently investigated in (Cuturi and Fukumizu, 2005) whose content is exposed in Chapter 5. We review in this Chapter a few ideas to cope with the first problem, which may translate into novel directions to design kernels on measures.

We discuss with more details the issues that arise when mapping objects onto measures in Section 4.2. A class of functions that may be useful to cope with such issues is presented in Section 4.3, labelled as semigroup spectral functions. Finally, we review in Section 4.4 a simple particular case of the previous family that translates into a fast kernel on clouds of points.

4.2 Comparing Measures on Arbitrary Spaces through Kernelized Estimates

Let us start this section with a few notations. We write $\mathbf{\Sigma}_n^+$ for the set of positive semidefinite matrices of order n, and for two elements A, B of $\mathbf{\Sigma}_n^+$, $\langle A, B \rangle$ denotes

the Frobenius dot-product $\mathrm{tr}(AB)$. We write I_d for the identity matrix of rank d and $\mathbb{1}_{d,d}$ for the $d \times d$ matrix composed of ones. For a vector $u \in \mathbb{R}^d$ and a $d \times d$ real positive semidefinite matrix K, the seminorm $\|\cdot\|_K$ is defined as

$$\|u\|_K = \sqrt{u^\top K u}.$$

Hölder norms for $\mathbb{R}^d$ are defined as the family of norms,

$$\|u\|_p = \left(\sum_{i=1}^{d} |u_i|^p \right)^{1/p},$$

with $p \geq 1$.

4.2.1 Measure Representations of Objects and Component Spaces

In some applications, the choice of a component space $\mathcal{X}$ to decompose a complex object is unambiguous and motivated by an *a priori* knowledge on the objects. Proteins for instance can be seen as macromolecules and are usually sliced into smaller chains of amino-acids known as domains (Hubbard et al., 1997). In learning tasks when there is not such a natural option – as guided by intuition and knowledge on the task – choosing a good size and complexity for the space of components is a major issue.

For example, the k in the size of k-mers used to slice amino-acids sequences into shorter subsequences usually has to be selected through cross-validation techniques. The corpus of words used in a bag-of-words representation of texts is also chosen following empirical considerations, that yield in practice to the erasure of frequent words with poor discriminative impact. Finally, the far more general issue of turning continuous values into finite number of bins, e.g., selecting a color depth for digitalized images, also falls into this type of problem if one wishes to represent the objects as multinomials of such bins.

When the space of components is too small, crucial details on the objects is definitely lost. When the component space is large, more information is captured but the obtained measures may be too sparse to be of practical use. One of the ways to cope with such a sparsity is, as recalled in the introduction, to regularize the empirical measures using prior knowledge. In bag of words representations of texts, Lafferty and Lebanon (2005) and Joachims (2002) use tf or tfidf estimators; for histograms of letter transitions, Cuturi and Vert (2005) use Dirichlet priors, while for histograms of colors Chapelle et al. (1999) apply nonlinear transformation to the counters. Finally, a common practice in bioinformatics is to regularize the k-mers counters through BLOSUM-type similarity matrices for amino-acids (Leslie et al., 2003; Rätsch and Sonnenburg, 2004). All these approaches share the following approach: first regularize the empirical measures and then compare them through kernels on multinomials which assume that the bins are interchangeable (Lafferty and Lebanon, 2005; Hein and Bousquet, 2005).

Rather than regularizing the empirical measures directly, a different direction incorporates this smoothing step into the expression of the kernel on measures itself, using in that sense a kernel κ on the component space. This operation is usually carried out theoretically by mapping a molecular measure on $\mathcal{X}$ to a molecular measure in a finite subspace of the RKHS $\mathcal{H}_\kappa$ associated with κ, and performing then the original calculations in $\mathcal{H}_\kappa$. In practice, most calculations only involve a finite number of kernel evaluations between the components that are directly used to construct the final kernel value between the two measures. When the space of components is finite, the bins that are translated into the RKHS $\mathcal{H}_\kappa$ are not necessarily orthogonal, and hence not interchangeable anymore, but rather depend on κ directly. This approach has been investigated to our knowledge in two papers; Kondor and Jebara (2003) and Cuturi and Vert (2005) have recently proposed kernelized versions of p.d. kernels designed for measures. These approaches can be related to (Wolf and Shashua, 2003), although the latter reference does not yield a kernel that can be used in practice as such. A slightly different approach, presented in (Hein and Bousquet, 2005), performs such a kernelization too but can only be computed in closed form with certain families of kernels. Through such kernelizations, all previous authors aim at including prior knowledge on the components at the level of the kernel itself, and reduce in parallel the arbitrariness of choosing the right parameterization for the component space.

We extend in this chapter results from (Cuturi et al., 2005) and characterize a few semigroup kernels on measures defined through variance, which from our viewpoint is a possible first step to perform such a kernelization.

4.2.2 Computing Kernels on Measures through Variance

While variance can be regarded as an efficient descriptor of a measure, it is also well suited for a further kernelization that takes into account similarities between elements of the original space. Indeed, if $\mathcal{X}$ is Euclidian, the variance Σ of a molecular measure, that is its centered second-order moment, has natural links with the dot-product matrix of the centered elements of the support of the measure. The variance of a measure μ can be roughly expressed in a matrix form as

$$\Sigma = \tilde{X}\tilde{X}^\top,$$

with $\tilde{X}$ being the matrix of all centered points contained in the support of the measure in a column form, while its centered dot-product matrix $\tilde{K}$ would take the form

$$\tilde{K} = \tilde{X}^\top\tilde{X}.$$

This remark suffices to note that all non-zero eigenvalues of both matrices are the same with the same multiplicity. Real valued p.d. kernels that only rely on the spectrum of variance matrices, that is spectral functions (Davis, 1957; Lewis, 2003) of the variance matrices, may hence be kernelized by computing the spectrum of the Gram matrices of the support of the measures rather than that of the variance matrix itself (which may not even exist when $\mathcal{X}$ is not Euclidian) and apply such formulas to the spectrum of the dot-product matrix. To ensure a stable formulation,

the spectral function used should further be invariant if we add an arbitrary number of zeros to the spectrum. We focus first on the sole issue of defining spectral functions on variances that may translate into p.d. kernels on measures.

In Euclidian spaces, variance matrices can be seen as an elementary characterization of the dispersion of a measure. When quantified by its determinant, the volume of a variance matrix has close connections with the entropy of the maximum likelihood multivariate Gaussian density that fits best the measure (Cuturi et al., 2005). However, comparing straightforwardly the centered variances of two measures ignores the discrepancy between the respective means. This has not been explored so far to our knowledge, but might translate into a useful translation invariance. In this framework, using spectral p.d. kernels on $\mathbf{\Sigma}_n^+$ should be sufficient to ensure the resulting kernelization.

When such an invariance is not pertinent, a better grounded alternative is to map the measures onto multivariate Gaussian distributions, that is consider both first and second order moments of the empirical measures, and use classical affinities between these distributions. The Bhattacharrya affinity was recently exposed in this context (Jebara et al., 2004), along with its kernelized counterpart (Kondor and Jebara, 2003). The drawback of such an approach is that it requires, when kernelized, an implicit construction of an orthogonal base for the space spanned by the support of the two measures, on which the means of the respective measures have to be projected. When performed in the RKHS this construction can be very costly.

4.2.3 The Semigroup Framework for Measures

Instead, the direction exploited in (Cuturi et al., 2005) is to use directly the second order moment of the mean of two measures rather than only consider the measures separately. Considering the mixture of two measures rather than the separate variances is a non-linear mapping that takes into account the discrepancies between the means since

$$\Sigma(\mu + \mu') = \Sigma(\mu) + \Sigma(\mu') - \left(\bar{\mu}\bar{\mu}'^{\top} + \bar{\mu}'\bar{\mu}^{\top}\right), \tag{4.1}$$

where we write as a shorthand $\bar{\mu}$ for $\mu[x]$. Taking the variance of the mixture of two measures is however grounded on geometric considerations: positive measures lie in a convex cone, and it is natural to add or average two measures. In probabilistic terms this amounts to merging the outcome of two experiments. On the other hand, there is no proper substraction between measures that respects their non-negativeness, unlike usual vector spaces. Semigroup kernels can be seen as a way of constructing a distance between points when no substraction operation on the set is available, through a semigroup negative definite function. Intuitively, one aims at defining a distance through addition, as one would reconstruct the first hand of the following equation by using only its second-hand,

$$\|x - y\|^2 = -\|x + y\|^2 + \frac{1}{2}\|x + x\|^2 + \frac{1}{2}\|y + y\|^2.$$

Other types of objects often encountered in machine learning, such as sequences, motivate the use of the semigroup framework. Indeed, while it is natural to merge

or concatenate strings, no natural substraction can be proposed on sequences. Furthermore, while the square of the Hilbert norm of a difference is always negative definite (Berg et al., 1984, Section 3.3) in the two arguments, the converse is not true in general. One reaches thus more generality by considering all possible negative definite semigroup kernels rather than focusing only on norms, which is why for n.d. functions ψ the expression

$$\psi(\mu+\mu') - \frac{1}{2}\psi(\mu+\mu) - \frac{1}{2}\psi(\mu'+\mu')$$

may reach more generality to design kernels. If we connect this approach to our previous discussion on functions on measures that take as an argument variance matrices, we obtain the formulation

$$\Psi(\Sigma(\mu+\mu')) - \frac{1}{2}\Psi(\Sigma(2\mu)) - \frac{1}{2}\Psi(\Sigma(2\mu')).$$

where $\Psi \circ \Sigma$ is both negative definite and Ψ is a spectral function defined on $\mathbf{\Sigma}_n^+$. We name such a composed function $\psi = \Psi \circ \Sigma$ a *semigroup spectral negative definite* (s.s.n.d.) function on measures. Note that this is *not equivalent* to defining directly negative definite functions Ψ for matrices in $\mathbf{\Sigma}_n^+$, since the underlying semigroup operation is the addition of measures and not that of the variance matrices of the measures.

Following this motivation, we propose below a general framework to devise semigroup spectral functions.

4.3 Semigroup Spectral Functions of Measures

We assume in the beginning of this section that $\mathcal{X}$ is an Euclidian space of dimension n endowed with Lebesgue's measure ν.

The variance of a measure μ of $M_+^V(\mathcal{X})$, the semigroup of measures with finite first and second moment of $M_+^b(\mathcal{X})$, is the matrix

$$\Sigma(\mu) = \mu[xx^\top] - \mu[x]\mu[x]^\top.$$

Note that $\Sigma(\mu)$ belongs to $\mathbf{\Sigma}_n^+$ when μ is a sub-probability measure, that is when $|\mu| \leq 1$, since

$$\Sigma(\mu) = \mu[(x-\mu[x])\,(x-\mu[x])^\top] + (1-|\mu|)\,\mu[x]\mu[x]^\top.$$

4.3.1 An Extended Framework for Semigroup Kernels

In practice, the assumption that $\Sigma(\mu) \in \mathbf{\Sigma}_n^+$ will be crucial in the following sections. The set on which this is ensured for any measure is the subset of $M_+^V(\mathcal{X})$ of sub-probability measures. This subset is not, however, a semigroup, since it is not closed under addition. To cope with this contradiction, that is to use semigroup like functions of the type $(\mu,\mu') \to \psi(\mu+\mu')$ where ψ is only defined on a subset

of the original semigroup, and where this subset may not be itself a semigroup, we define the following extension to the original definition of semigroup functions.

Definition 4.1 (Semigroup kernels on subsets). *Let $(S,+)$ be an auto-involutive semigroup and $U \subset S$ a nonempty subset of S. A function $\psi : U \to \mathbb{R}$ is a p.d. (resp. n.d.) semigroup function on U if*

$$\sum_{i,j} c_i c_j \psi(s_i + s_j) \geq 0 \;\; (resp \leq 0)$$

holds for any $n \in \mathbb{N}$; any $s_1, \ldots, s_n \in S$ such that $s_i + s_j \in U$ for $1 \leq i \leq j \leq n$; and any $c_1 \ldots, c_n \in \mathbb{R}$ (resp. with the additional condition that $\sum_i c_i = 0$)

The subset $M_+^v(\mathcal{X})$ of sub-probability measures μ of $M_+^V(\mathcal{X})$ with non degenerated variances, that is such that $\det \Sigma(\mu) \neq 0$, will be used throughout this exposition, and all semigroup functions on $M_+^v(\mathcal{X})$ should be considered following this definition. In practice and for such a semigroup kernel ψ defined on the subset $M_+^v(\mathcal{X})$, we will consider kernels of the form

$$M_+^v(\mathcal{X})^2 \ni (\mu, \mu') \mapsto \psi\left(\frac{\mu}{2} + \frac{\mu'}{2}\right)$$

to ensure that the mean $\frac{\mu+\mu'}{2}$ remains in $M_+^v(\mathcal{X})$. In other words, we will consider the product space $\frac{1}{2}M_+^v(\mathcal{X}) \times \frac{1}{2}M_+^v(\mathcal{X})$ as the index set of all kernels derived from a s.s.p.d. function Ψ.

4.3.2 Characteristic Functions and s.s.p.d. Kernels

We establish a link in this section between a classical tool of convex analysis, used notably in convex optimization through barrier functions (Boyd and Vandenberghe, 2004), and spectral semigroup kernels. We prove first a simple lemma which characterizes a large family of s.s.n.d. on measures.

Lemma 4.2. *For any $S \in \mathbf{\Sigma}_n^+$, the real-valued function defined on $M_+^v(\mathcal{X})$, a subset of the semigroup $M_+^V(\mathcal{X})$, as*

$$\mu \mapsto \langle \Sigma(\mu), S \rangle$$

is a negative definite semigroup function.

Proof. For any $k \in \mathbb{N}$, any $c_1, ..., c_k \in \mathbb{R}$ such that $\sum_i c_i = 0$ and any $\mu_1, ..., \mu_k \in$

$M_+^V(\mathcal{X})$ such that $\mu_i + \mu_j \in M_+^v(\mathcal{X})$, we have using Equation (4.1) that

$$\begin{aligned}\sum_{i,j} c_i c_j \left\langle \Sigma\left(\mu_i + \mu_j\right), S \right\rangle &= \left\langle \sum_{i,j} c_i c_j \Big(\Sigma(\mu_i) + \Sigma(\mu_j) - \left(\bar{\mu}_i \bar{\mu}_j^\top + \bar{\mu}_j \bar{\mu}_i^\top\right) \Big), S \right\rangle \\ &= -\left\langle \sum_{i,j} c_i c_j \left(\bar{\mu}_i \bar{\mu}_j^\top + \bar{\mu}_j \bar{\mu}_i^\top\right), S \right\rangle \\ &= -2 \sum_{i,j} c_i c_j \bar{\mu}_i^\top S \bar{\mu}_j \leq 0.\end{aligned}$$

□

Note that $\mu \mapsto \langle \Sigma(\mu), S \rangle$ is actually a semigroup function on the whole of $M_+^V(\mathcal{X})$, but we will mainly use its restriction on $M_+^v(\mathcal{X})$. The simple case $S = I_n$, that is considering $\mu \mapsto \operatorname{tr} \Sigma(\mu)$, provides interesting results in practice and boils down to a fast kernel on clouds of points. This case is considered separately with more depth in Section 4.4. We characterize in the next proposition a large family of semigroup spectral functions derived from Lemma 4.2.

To do so, we introduce first the notion of the characteristic function of a cone. Given a cone T endowed with a dot-product $\langle \cdot, \cdot \rangle_T$, its dual is defined as

$$T^* = \bigcap_{t \in T} \{ s \in T, \langle s, t \rangle_T \geq 0 \},$$

and we write T° for its interior. We further assume that T is a measurable space with a base measure ν.

Definition 4.3 (Characteristic Function of a Cone, Koecher (1957)). *The characteristic function $\varphi_{T,\nu} : T^\circ \to \mathbb{R}$ of a cone T endowed with a measure ν is defined as*

$$\varphi_{T,\nu}(t) = \int_{T^*} e^{-\langle t, s \rangle_T} \nu(ds).$$

Let us particularize this result to the cone $\mathbf{\Sigma}_n^+$ of positive semidefinite matrices which is in addition self-dual[16], that is $\mathbf{\Sigma}_n^{+*} = \mathbf{\Sigma}_n^+$. Indeed, this well known result follows first from the fact that for $S, T \in \mathbf{\Sigma}_n^+$,

$$\langle S, T \rangle = \operatorname{tr}(ST) = \operatorname{tr}(ST^{\frac{1}{2}} T^{\frac{1}{2}}) = \operatorname{tr}(T^{\frac{1}{2}} S T^{\frac{1}{2}}) \geq 0.$$

Then, for $S \in \mathbf{\Sigma}_n^{+*}$, we have that for any vector $y \in \mathbb{R}^n$,

$$y^\top S y = \operatorname{tr}(S y y^\top) \geq 0,$$

and hence $S \in \mathbf{\Sigma}_n^+$. Note further that $\mathbf{\Sigma}_n^{+\circ}$ is the set of positive definite matrices. We endow $\mathbf{\Sigma}_n^+$ with its Borel σ-algebra $\mathcal{B}$ and consider the Lebesgue measure on

[16] $\mathbf{\Sigma}_n^+$ is actually one of five canonical homogeneous and irreducible cones, characterized by P.Jordan, J. von Neumann and E. Wigner in a seminal paper of 1934.

$\mathbf{\Sigma}_n^+$ seen as a subset of $\mathbb{R}^{n^2}$. Following the terminology for functions, an orthogonally invariant measure ρ on $(\mathbf{\Sigma}_n^+, \mathcal{B})$ is such that for every $B \in \mathcal{B}$ and any $n \times n$ orthogonal matrix P,

$$\rho(B) = \rho(\{PSP^*, S \in B\}).$$

The next proposition characterizes a large family of s.s.p.d. kernels on measures.

Proposition 4.4. *The function $\psi_\rho : M_+^v(\mathcal{X}) \to \mathbb{R}$ defined by an orthogonally invariant measure ρ through*

$$\psi_\rho(\mu) = \varphi_{\mathbf{\Sigma}_n^+, \rho}(\Sigma(\mu)),$$

is a semigroup spectral p.d. function.

Proof. When $\mu \in M_+^v(\mathcal{X})$, $\Sigma(\mu) \in \mathbf{\Sigma}_n^{+\circ}$. The integral when it exists is a sum of p.d. semigroup functions through Schoenberg's theorem (Berg et al., 1984, Theorem 3.2.2), and is hence positive definite. The fact that the function is a spectral function is ensured by the fact that ρ is orthogonally invariant. □

4.3.3 A Few Examples of Semigroup Spectral Functions

We have already evoked the case $\rho = \delta_{I_n}$, that is $\mu \mapsto e^{-\operatorname{tr}\Sigma(\mu)}$, and we study its properties in Section 4.4. Let us review another example with Wishart densities on $\mathbf{\Sigma}_n^+$, that is densities w.r.t the Lebesgue measure ν of the type

$$p_{\Sigma,d}(S) \propto \frac{1}{\det(S)^{(n+1)/2}} \det(\Sigma^{-1}S)^{d/2} e^{-\langle \Sigma^{-1}, S\rangle},$$

for $d \geq n$. Using this density which integrates to 1 on $\mathbf{\Sigma}_n^+$, we can set $\rho = f \cdot \nu$ where $f : S \mapsto \det(S)^{\frac{d-n-1}{2}}$ to obtain that

$$\mu \mapsto \frac{1}{\det \Sigma(\mu)^{\frac{d}{2}}}$$

is a s.s.p.d. function on $M_+^v(\mathcal{X})$. In the next example, we restrict the integration domain to only consider the subspace of $\mathbf{\Sigma}_n^+$ of matrices of rank 1, that is matrices of the form $yy^\top$ where $y \in \mathbb{R}^n$. For a $n \times n$ strictly p.d. matrix A such that its multispectrum $\operatorname{mspec} A = \{\lambda_1, ..., \lambda_n\}$, we write for $1 \leq i \leq n$,

$$\gamma_i(A) \stackrel{\text{def}}{=} \sum_{|j|=i} \frac{\prod_{k=1}^n \Gamma(j_k + \frac{1}{2})}{\lambda_1^{j_1} \cdots \lambda_n^{j_n}}$$

where the summation is taken over all families $j \in \mathbb{N}^n$ such that the sum of their elements $|j|$ is equal to i. Additionnally we set $\gamma_0(A) = 1$ and remark that

$$\gamma_n(A) = \left(\frac{\sqrt{\pi}}{2}\right)^n \frac{1}{\det(A)},$$

and

$$\gamma_1(A) = \frac{\pi^{\frac{n}{2}}}{2}\operatorname{tr}(A^{-1}).$$

Corollary 4.5. *The functions*

$$\chi_i : \mu \mapsto \left(\frac{2}{\sqrt{\pi}}\right)^{\frac{n}{2}} \sqrt{\gamma_n}\cdot\gamma_i\left(\Sigma(\mu)\right),$$

are s.s.p.d. functions on $M_+^v(\mathcal{X})$ for $1 \le i \le n-1$.

Proof. The Euclidian norm $\|y\|_2^2$ of y is the only positive eigenvalue of $yy^\top$ when $y \neq 0$, hence any real-valued function of $\|y\|_2^2$ is spectral. As a consequence of Proposition 4.4, where we have restricted the integration domain on matrices of rank 1, and for any function $g : \mathbb{R}^+ \to \mathbb{R}^+$,

$$\mu \mapsto \int_{\mathbb{R}^n} e^{-y^t\Sigma(\mu)y} g(\|y\|_2^2)dy \tag{4.2}$$

is a s.s.p.d. function on $M_+^v(\mathcal{X})$. Let $\mu \in M_+^v(\mathcal{X})$, and write $\operatorname{mspec}\Sigma(\mu) = \{\lambda_1, ..., \lambda_n\}$. Then by an appropriate base change we have for $g : x \mapsto x^i$, $i \in \mathbb{N}$,

$$\begin{aligned}
\int_{\mathbb{R}^n} e^{-y^t\Sigma(\mu)y} g(\|y\|_2^2)dy &= \int_{\mathbb{R}^n} e^{-\sum_{k=1}^n \lambda_k y_k^2}\, g(\,\|y\|_2^2)dy \\
&= \int_{\mathbb{R}^n} e^{-\sum_{k=1}^n \lambda_k y_k^2} \sum_{|j|=i}\prod_{k=1}^n y_k^{2j_k} dy \\
&= \sum_{|j|=i}\prod_{k=1}^n \int_{\mathbb{R}} e^{-\lambda_k y_k^2} y_k^{2j_k} dy_k \;=\; \sum_{|j|=i}\prod_{k=1}^n \Gamma(j_k+\frac{1}{2})\lambda_k^{-j_k-\frac{1}{2}} \\
&= \left(\frac{2}{\sqrt{\pi}}\right)^{\frac{n}{2}} \sqrt{\gamma_n\left(\Sigma\left(\mu\right)\right)} \sum_{|j|=i} \frac{\prod_{k=1}^n \Gamma(j_k+\frac{1}{2})}{\lambda_1^{j_1}\cdots\lambda_n^{j_n}} \\
&= \left(\frac{2}{\sqrt{\pi}}\right)^{\frac{n}{2}} \sqrt{\gamma_n}\cdot\gamma_i\left(\Sigma\left(\mu\right)\right)
\end{aligned}$$

□

The inverse generalized variance presented in (Cuturi et al., 2005) is recovered as χ_0. We refer now to a series of identities, known as *Lancaster's formulas* (Bernstein, 2005, p.320) to express more explicitly the cases $i = 1, 2, 3$.

Proposition 4.6 (Lancaster formulas). *Let $A, B \in \mathbf{\Sigma}_n^+$. Then, for $i = 0, 1, 2, 3$ define*

$$\mathcal{I}_i \stackrel{\text{def}}{=} \frac{1}{\pi^{\frac{n}{2}}\sqrt{\det A}} \int_{\mathbb{R}^n} (y^\top By)^i e^{-y^\top Ay} dy.$$

Then

$$\begin{aligned}\mathcal{I}_0 &= 1,\\ \mathcal{I}_1 &= \operatorname{tr} AB,\\ \mathcal{I}_2 &= (\operatorname{tr} AB)^2 + 2\operatorname{tr}(AB)^2,\\ \mathcal{I}_3 &= (\operatorname{tr} AB)^3 + 6(\operatorname{tr} AB)(\operatorname{tr}(AB)^2) + 8\operatorname{tr}(AB)^3\end{aligned}$$

The proof may be found in (Miller, 1987, p.80). Semigroup positive definite functions can be derived through these formulas by using for g in the proof of Corollary 4.5 a non-spectral density function of the type $g : y \mapsto (y^\top By)^k$. When $B = I_n$, the Lancaster formulas give explicit expressions for χ_i in the cases $i = 1, 2, 3$, where we write Σ for $\Sigma(\mu)$ and C_n for $\left(\frac{2}{\sqrt{\pi}}\right)^{\frac{n}{2}}$:

$$\begin{aligned}\chi_0(\mu) =& \frac{C_n}{\sqrt{\det \Sigma}},\\ \chi_1(\mu) =& \frac{C_n}{\sqrt{\det \Sigma}} \left[\operatorname{tr} \Sigma^{-1}\right],\\ \chi_2(\mu) =& \frac{C_n}{\sqrt{\det \Sigma}} \left[(\operatorname{tr} \Sigma^{-1})^2 + 2\operatorname{tr} \Sigma^{-2}\right],\\ \chi_3(\mu) =& \frac{C_n}{\sqrt{\det \Sigma}} \left[(\operatorname{tr} \Sigma^{-1})^3 + 6(\operatorname{tr} \Sigma^{-1})(\operatorname{tr} \Sigma^{-2}) + 8\operatorname{tr} \Sigma^{-3}\right].\end{aligned}$$

Finally, let us note that the representation proposed in Proposition 4.4 is not exhaustive to our knowledge, although it shares some structural similarity with the integral representation of semigroup p.d. functions studied in Section 3.6. First, the functions

$$\mu \mapsto e^{-\langle \Sigma(\mu), S\rangle},$$

are not semicharacters, since

$$e^{-\langle \Sigma(\mu+\mu'), S\rangle} \neq e^{-\langle \Sigma(\mu)+\Sigma(\mu'), S\rangle}$$

in the general case. Second, the class of functions considered through Lemma 4.2 is far from characterizing all semigroup negative definite functions on $M_+^v(\mathcal{X})$ since, through (Berg et al., 1984, Corrolary 3.2.10, p.78), we have that for any $S \in \mathbf{\Sigma}_n^+$ and $0 < \alpha < 1$ both

$$\mu \mapsto \langle \Sigma(\mu), S\rangle^\alpha,$$

and

$$\mu \mapsto \ln(1 + \langle \Sigma(\mu), S\rangle),$$

are s.s.n.d. functions. Note that if we use for $y \in \mathbb{R}^n$ and $m \geq 1$ a n.d. function of the type

$$\mu \mapsto \frac{m+n}{2} \ln\left(1 + \frac{1}{m} y^\top \Sigma(\mu) y\right),$$

and exponentiate it in the spirit of Equation (4.2), that is define

$$\mu \mapsto \int_{\mathbb{R}^n} \frac{1}{\left(1 + \frac{1}{m} y^\top \Sigma(\mu) y\right)^{\frac{m+n}{2}}} dy,$$

we recover the integration of the Student centered multivariate distribution for vectors of $\mathbb{R}^n$,

$$S(y|\Sigma, m) = \frac{\Gamma(\frac{m+n}{2})\sqrt{\det \Sigma}}{\Gamma(\frac{m}{2})(m\pi)^{n/2}} \left(1 + \frac{1}{m} y^\top \Sigma(\mu) y\right)^{-\frac{m+n}{2}}$$

which boils down again to a kernel that is proportional to χ_0.

Before reviewing in the next section a simple kernel, we note that we have not completely fulfilled the initial objective of proposing a general family of kernels on measures that would not only be spectral, but also easily "kernelizable". Computational issues aside, the family of s.s.p.d. functions on $M_+^v(\mathcal{X})$ obtained from Proposition 4.4 particularized in Corollary 4.5 may not be used with the same ease than the regularized IGV proposed in Chapter 3. Indeed, for a measure μ with admissible base γ, only the non-zero elements of the spectrums of $\Sigma(\mu)$ and the centered dot-product matrix $\tilde{K}_\gamma$ are equal. To be kernelized, the function should not only be spectral, but also invariant under the addition of an arbitrary number of null-eigenvalues. This property is only fulfilled so far by the regularized inverse generalized variance, and by the trace kernel proposed below. Regularizing functions obtained through Proposition 4.4 so that they can satisfy this restriction is the topic of current research. An intermediate step that would be satisfactory from an empirical viewpoint would be to project the considered measures onto finite subspaces of the RKHS $\mathcal{H}_k$ obtained in a semi-supervised way, as in (Bach and Jordan, 2005), and use directly the family χ_i on such finite subspaces.

4.4 The Trace as a Semigroup Spectral Function

We focus back on the s.s.p.d. function

$$\varphi_{tr} : \mu \mapsto e^{-\frac{1}{\beta}\langle \Sigma(\mu), I_n \rangle} = e^{-\frac{1}{\beta} \operatorname{tr} \Sigma(\mu)},$$

defined for $\beta > 0$, and where φ_{tr} stands for $\varphi_{\delta_{I_n}}$ as a shorthand to the notation used in Proposition 4.4. We coin down this function the trace s.s.p.d. function, or the trace kernel between two measures μ and μ' when it is evaluated on their mean.

Presented in (Cuturi et al., 2005), the regularized inverse generalized variance (IGV) kernel quantifies how similar two probability measures μ and μ' are by considering the generalized variance of their mixture, that is the determinant of the centered variance matrix $\Sigma(\frac{\mu+\mu'}{2})$. The IGV kernel k_{igv}^η (where $\eta > 0$), defined[17] as

$$k_{igv}^\eta(\mu, \mu') \stackrel{\text{def}}{=} \frac{1}{\sqrt{\det\left(\frac{1}{\eta}\Sigma\left(\frac{\mu+\mu'}{2}\right) + I_n\right)}},$$

[17] In (Cuturi et al., 2005), the IGV kernel is presented without the square root sign. The original proof incorporates this case however, and we adopt from now on this convention to be consistent with Proposition 4.4.

can actually be written as a s.s.p.d function. It suffices to note, using the same proof used for Corrolary 4.5, that the s.s.p.d. φ_{gv} defined through

$$g : y \mapsto e^{-\|y\|_2^2},$$

and applied to a scaling $1/\eta$ of $\Sigma(\mu)$,

$$\varphi_{\mathrm{gv}} : \mu \mapsto \int_{\mathbb{R}^n} e^{-\frac{1}{\eta}\langle \Sigma(\mu), yy^\top \rangle} e^{-\|y\|_2^2} dy = \int_{\mathbb{R}^n} e^{-\langle \frac{1}{\eta}\Sigma(\mu) + I_n, yy^\top \rangle} dy$$

boils down to the identity

$$k^{\eta}_{igv}(\mu, \mu') = \varphi_{\mathrm{gv}}\left(\frac{\mu + \mu'}{2}\right).$$

$\varphi_{\mathrm{gv}}(\mu)$ and $\varphi_{\mathrm{tr}}(\mu)$ only depend on the spectrum of $\Sigma(\mu)$, and can be interpreted as two different quantifications of the size of $\Sigma(\mu)$, described by its volume and by its perimeter respectively. If we write $(\lambda_i)_{i=1..n}$ for the eigenvalues of $\Sigma(\mu)$, we have that:

$$\begin{aligned} \varphi_{\mathrm{gv}}(\mu) &= \prod_{i=1}^{n} \left(1 + \frac{\lambda_i}{\eta}\right)^{-1}, \quad \text{or} \quad \varphi_{\mathrm{gv}}(\mu) = e^{-\sum_{i=1}^{n} \ln(1+\frac{\lambda_i}{\eta})}; \\ \varphi_{\mathrm{tr}}(\mu) &= \prod_{i=1}^{n} e^{-\frac{\lambda_i}{\beta}}, \quad \text{or} \quad \varphi_{\mathrm{tr}}(\mu) = e^{-\sum_{i=1}^{n} \frac{\lambda_i}{\beta}}, \end{aligned} \tag{4.3}$$

which shows that both IGV and trace kernels apply a different regularization scheme to these eigenvalues: the trace kernel tends to give more importance to large eigenvalues when seen from the IGV kernel viewpoint expressed in the left-hand side of Equation (4.3), while the IGV kernel tends to focus more on small eigenvalues when seen from the trace kernel viewpoint expressed in the right hand side.

4.4.1 The Trace Kernel on Molecular Measures

In the case of a molecular measure μ defined on an Euclidian space $\mathcal{X}$ of dimension n, the variance $\Sigma(\mu)$ is simply the usual empirical estimate of the variance matrix expressed in an orthonormal basis of $\mathcal{X}$:

$$\Sigma(\mu) = \sum_{i=1}^{d} a_i x_i x_i^\top - \left(\sum_{i=1}^{d} a_i x_i\right)\left(\sum_{i=1}^{d} a_i x_i\right)^\top,$$

where we use an admissible base $\gamma = (x_i, a_i)_{i=1}^{d}$ of μ to give a matrix expression of $\Sigma(\mu)$, with all points x_i expressed as column vectors. Given such an admissible base, let $X_\gamma = [x_i]_{i=1..d}$ be the $n \times d$ matrix made of all column vectors x_i, and let δ_γ and Δ_γ be respectively the column vector and the diagonal matrix of weights of γ taken in the same order $(a_i)_{1 \leq 1 \leq d}$,

$$\delta_\gamma = \begin{pmatrix} a_1 \\ \vdots \\ a_i \\ \vdots \\ a_d \end{pmatrix}, \quad \Delta_\gamma = \begin{pmatrix} a_1 & 0 & \cdots & 0 \\ 0 & a_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & a_d \end{pmatrix}.$$

We then have for any base γ of μ that:

$$\Sigma(\mu) = X_\gamma(\Delta_\gamma - \Delta_\gamma \mathbb{1}_{d,d}\Delta_\gamma)X_\gamma^\top.$$

If μ is a *probability* measure,

$$\Sigma(\mu) = X_\gamma(I_d - \Delta_\gamma \mathbb{1}_{d,d})\Delta_\gamma(I_d - \mathbb{1}_{d,d}\Delta_\gamma)X_\gamma^\top, \tag{4.4}$$

where we used $(\Delta_\gamma \mathbb{1}_{d,d})^2 = \Delta_\gamma \mathbb{1}_{d,d}$ since $\operatorname{tr}\Delta_\gamma = |\mu| = 1$. This expression is however invalid when μ is a not a probability measure, and we use below a more suitable expression of $\Sigma(\mu)$ than the one in Equation (4.4) in that case. Assume further that

$$K_\gamma \stackrel{\text{def}}{=} \left[x_i^\top x_j\right]_{1\le i,j\le d},$$

namely the $d \times d$ matrix of dot-products associated with the elements of a base γ, and subsequently define

$$\tilde{K}_\gamma \stackrel{\text{def}}{=} \left[(x_i - \sum_{k=1}^d a_k x_k)^\top (x_j - \sum_{k=1}^d a_k x_k)\right]_{1\le i,j\le d} = (I_d - \mathbb{1}_{d,d}\Delta_\gamma)K_\gamma(I_d - \Delta_\gamma \mathbb{1}_{d,d}),$$

for its centered expression with respect to the mean of μ. The reformulation of Equation (4.4) paves then a way for a natural kernelization of $\varphi_{\mathrm{tr}}(\mu)$, since by writing $\tilde{X} = X(I_d - \Delta \mathbb{1}_{d,d})$ we have that the matrices $\tilde{K}_\gamma\Delta_\gamma = \tilde{X}^\top\tilde{X}\Delta_\gamma$ and $\Sigma(\mu) = \tilde{X}\Delta_\gamma\tilde{X}^\top$ share the same non-zero eigenvalues and hence the same trace. When μ is not a probability measure however, the expression of $\Sigma(\mu)$ becomes slightly more complicated, although it is still possible to "kernelize" it (that is express it through K_γ) as seen below.

Proposition 4.7. *For a measure $\mu \in \mathrm{Mol}_+(\mathcal{X})$ there exists $r > 0$ and $\theta \in \mathrm{Mol}_+^1(\mathcal{X})$ such that $\mu = r\,\theta$. If γ is an admissible base of θ, then*

$$\begin{aligned} \varphi_{tr}(\mu) &= r \operatorname{tr}\Sigma(\theta) + (r - r^2)\|\theta[x]\|_2^2, \\ &= r \operatorname{tr}(K_\gamma\Delta_\gamma) - r^2\|\delta_\gamma\|_{K_\gamma}^2. \end{aligned} \tag{4.5}$$

Proof. The first identity is obtained by subtracting below the second line to the first line:

$$\begin{aligned} \Sigma(r\theta) &= r\theta[xx^\top] - r^2\theta[x]\theta[x]^\top \\ r\Sigma(\theta) &= r\theta[xx^\top] - r\theta[x]\theta[x]^\top, \end{aligned}$$

and using the fact that $\mathrm{tr}(\theta[x]\theta[x]^\top) = \|\theta[x]\|_2^2$. For a probability measure θ, we refer to the proof of (Cuturi et al., 2005, Proposition 5) for the identification of non-zero eigenvalues of $\Sigma(\theta)$ and $\tilde{K}_\gamma\Delta_\gamma$. Hence

$$\begin{aligned}\varphi_{\mathrm{tr}}(\theta) &= \mathrm{tr}(\tilde{K}_\gamma\Delta_\gamma)\\ &= \mathrm{tr}(I_d - \mathbb{1}_{d,d}\Delta_\gamma)K_\gamma(I_d - \Delta_\gamma\mathbb{1}_{d,d})\Delta_\gamma\\ &= \mathrm{tr}(K\Delta_\gamma - \mathbb{1}_{d,d}\Delta_\gamma K\Delta_\gamma - K\Delta_\gamma\mathbb{1}_{d,d}\Delta_\gamma + \mathbb{1}_{d,d}\Delta_\gamma K\Delta_\gamma\mathbb{1}_{d,d}\Delta_\gamma)\\ &= \mathrm{tr}(K\Delta_\gamma) - \|\delta_\gamma\|^2_{K_\gamma}\end{aligned}$$

the final identity can be then obtained by noting that $\|\theta[x]\|_2^2 = \|\delta_\gamma\|^2_{K_\gamma}$ □

The reformulation in (Cuturi et al., 2005, Proposition 5) of a quantity defined on the variance matrix in terms of matrices of dot-product is used to pave the way for a kernelization of the IGV kernel in (Cuturi et al., 2005, Theorem 7). Indeed, the latter theorem proves that changing the dot-product matrix K_γ of a base γ into any Gram matrix $\mathcal{K}_\gamma$ associated with the same base γ and a kernel κ, that is

$$\mathcal{K}_\gamma = [\kappa(x_i, x_j)]_{1\leq i,j\leq d},$$

retains the positive definiteness of the IGV kernel, while giving it a wider applicability. This substitution is valid here as well, since both IGV and TV kernels can be kernelized considering molecular measures in $\mathcal{X}$ as molecular measures on a finite subspace of the RKHS associated with a kernel κ and performing exactly the same computations. Instead of considering dot-product matrices in a strict sense, we will hence consider Gram matrices of any type induced by any *a priori* kernel κ on $\mathcal{X}$ without assuming further that $\mathcal{X}$ is Euclidian.

4.4.2 Practical Formulations for the Trace Kernel

Following our discussion in the introduction, negative definite kernels on a set $\mathcal{X}$, that is real-valued functions k defined on $\mathcal{X}\times\mathcal{X}$, and Hilbert norms, that is real valued functions on $\mathcal{X}\times\mathcal{X}$ of the form $D(x,y) = \|\Phi(x) - \Phi(y)\|_{\mathcal{H}}$ where $\mathcal{H}$ is a Hilbert space and $\Phi : X \to \mathcal{H}$, are deeply related one to the other. Remarkably, D is a Hilbert norm (that is an adequate space $\mathcal{H}$ and a mapping Φ match its expression) if and only if D^2 is negative definite. On the other hand, the scope of n.d. kernels spans a larger space of functions, see notably (Berg et al., 1984, Chapter 3.3) and (Hein and Bousquet, 2005) for a discussion in the framework of measures. Expressing a n.d. kernel k as the square of Hilbert norm can be obtained in practice by normalizing the kernel, that is writing

$$\tilde{k}(x,y) = k(x,y) - \frac{k(x,x) + k(y,y)}{2}$$

so that $\tilde{k}$ vanishes on the diagonal. Note that $\tilde{k}$ is still negative definite, as the addition of two n.d. kernels (Berg et al., 1984, Corollary 3.2.11). We use below such normalized quantities, and see that they translate naturally into families of Hilbert norms on $M_+^b(\mathcal{X})$.

Clouds of points of $\mathcal{X}$

Given two measures $\mu = r\theta$ and $\mu' = r'\theta' \in \mathrm{Mol}_+^1(\mathcal{X})$ with admissible bases γ, γ' for θ and θ', we have that

$$\tilde{\varphi_{\mathrm{tr}}}(\mu, \mu') = \varphi_{\mathrm{tr}}\left(\frac{\mu + \mu'}{2}\right) - \frac{\varphi_{\mathrm{tr}}(\mu) + \varphi_{\mathrm{tr}}(\mu')}{2} = \frac{1}{4}(r^2\|\delta\|^2_{\mathcal{K}_\gamma} + r'^2\|\delta'\|^2_{\mathcal{K}_{\gamma'}} - 2rr'\delta^\top \mathcal{K}_* \delta')$$

where $\mathcal{K}''$, the Gram matrix of the concatenation γ, γ' has been decomposed as

$$\mathcal{K}'' = \begin{pmatrix} \mathcal{K}_\gamma & \mathcal{K}_* \\ \mathcal{K}_*^\top & \mathcal{K}_{\gamma'} \end{pmatrix},$$

and $\mathcal{K}_*$ is the dot-product matrix between the elements of the support of μ and μ'. Suppose that we address the simple case $\mu = \frac{1}{d}\sum_{i=1}^{d} \delta_{x_i}$ and $\mu' = \frac{1}{d'}\sum_{i=1}^{d'} \delta_{y_i}$, then we have that

$$\tilde{\varphi_{\mathrm{tr}}}(\mu, \mu') = \frac{1}{4}\left(2\frac{|\mathcal{K}_*|}{dd'} - \frac{|\mathcal{K}_\gamma|}{d^2} - \frac{|\mathcal{K}_{\gamma'}|}{d'^2}\right)$$

where for any matrix A, $|A| \stackrel{\mathrm{def}}{=} \sum_{i,j} A_{i,j}$.

Multinomials on finite sets of components $\mathcal{X}$

Various kernels on measures on finite spaces have been proposed recently to be applied to multinomial probability distributions (Lafferty and Lebanon, 2005; Hein and Bousquet, 2005; Jebara et al., 2004). This attention has been mainly fueled by practical considerations when implementing kernel methods on real-life datasets, since in practice structured objects such as texts or digitalized images can be formulated as multinomials over finite spaces of components, typically sets of words, letters or bins of discreticized continuous values. When $\mathcal{X}$ is finite and $n = \mathrm{card}(\mathcal{X})$, a measure μ in $M_+^b(\mathcal{X})$ is described by its weights $\mu(x)$ where x spans $\mathcal{X}$. If we define an arbitrary index for the elements of $\mathcal{X}$, this is equivalent to considering the measure as a vector δ of the simplex Σ_n. The previously considered Gram matrices of the support of a measure can then be replaced by a single *a priori* Gram matrix $\mathcal{K}$ defined on the whole set of components ordered in the same way. These simplifications yield the following expression:

Proposition 4.8 (Trace Kernel for Multinomials). *Let $\mu = r\delta, \mu' = r\delta'$ be two measures on a finite set $\mathcal{X}$ endowed with a kernel κ, where δ and δ' belong to Σ_d.*

$$\tilde{\varphi_{tr}}(\mu, \mu') = \|r\delta - r'\delta'\|_{\mathcal{K}},$$

where $\mathcal{K}$ is the κ Gram matrix of elements of $\mathcal{X}$.

Proof.

$$\varphi_{\mathrm{tr}}\left(\frac{\mu + \mu'}{2}\right) - \frac{\varphi_{\mathrm{tr}}(\mu) + \varphi_{\mathrm{tr}}(\mu')}{2} = \frac{1}{4}\left(r^2\|\delta\|^2_{\mathcal{K}_\gamma} + r'^2\|\delta'\|^2_{\mathcal{K}_{\gamma'}} - 2rr'\delta^\top \mathcal{K}_* \delta'\right)$$

□

We thus obtain through a different path RBF type kernels for multinomials (and sub-multinomials alike), that is for $\sigma > 0$ the kernel

$$(\mu, \mu') \mapsto e^{-\frac{1}{\sigma}\|\mu-\mu'\|_{\mathcal{K}}}.$$

Such kernels are directly parameterized by a kernel function on the bins of interest. We believe this simple kernel may prove useful in practice to compare objects, at an additional cost however since the use the seminorm $\|\cdot\|_{\mathcal{K}}$ demands d^2 operations to compute the kernel, rather than using directly the Euclidian norm which can be computed in d steps.

Chapter 5

Multiresolution Kernels

Résumé

Nous proposons dans ce chapitre une méthodologie pour définir des noyaux pour objets structurés inspirée de l'algorithme dit du "context-tree weighting" proposé par Willems et al. (1995) et utilisé pour calculer un mélange de probabilités dans le deuxième chapitre de cette thèse. Nous faisons le constat que l'approche visant à représenter un objet complexe comme un histogramme de ces composants peut s'avérer limitante pour plusieurs applications, alors même que certains noyaux pour histogrammes peuvent à l'inverse être performants quand ils sont couplés avec des machines à vecteur de support. Nous tâchons donc d'intégrer ce problème en proposant une approche générique qui puisse tirer profit d'une représentation de chaque objet considéré non plus comme un simple histogramme mais comme une famille d'histogrammes emboîtés selon une hiérarchie d'évènements pouvant conditionner l'apparition de composants. Via l'algorithme de Willems et al. (1995) nous montrons que le calcul de ce noyau, constitué d'un vaste mélange de noyaux plus élémentaires, peut s'effectuer en un temps linéaire en le nombre d'histogrammes élémentaires considérés pour représenter les objets. Dans des applications pratiques, son implémentation peut améliorer la performance de noyaux sur mesures couplés avec des SVM, comme en témoignent les expériences exposées en fin de chapitre visant à catégoriser automatiquement des images vues comme des histogrammes de couleur.

5.1 Introduction

There is strong evidence that kernel methods (Schölkopf and Smola, 2002) can deliver state-of-the-art performance in many classification tasks when the input data lies in a vector space. Arguably, two factors contribute to this success. First, the good ability of kernel algorithms, such as the SVM, to generalize and provide a sparse formulation for the underlying learning problem; Second, the capacity of nonlinear kernels, such as the polynomial and RBF kernels, to quantify meaningful similarities between vectors, notably non-linear correlations between their components. Using kernel machines with non-vectorial data (e.g., in bioinformatics, pattern recognition or signal processing tasks) requires more arbitrary choices, both to represent the objects and to chose suitable kernels on those representations. The challenge of using kernel methods on real-world data has thus recently fostered many proposals for kernels on complex objects, notably for strings, trees, images or graphs to cite a few.

For such objects, multiple representations that lead to various kernels have been proposed in the literature. Some practitioners expect to use directly some well-known similarities in their respective fields, and the issue of their positive-definiteness has to be addressed, either through some clever adaptations (Vert et al., 2004) or more massive implementations using the so-called "empirical kernel map" (Schölkopf et al., 2002). Another trend when dealing with such objects is to consider them as bags of components. In practice this often yields kernels that should be applied on the histograms of smaller components sampled in the objects, where the kernels take into account the geometry of the underlying histograms Jebara et al. (2004); Lafferty and Lebanon (2005); Cuturi and Vert (2005); Hein and Bousquet (2005); Cuturi et al. (2005). The previous approaches coupled with SVM's combine both the advantages of using discriminative methods with generative ones, and produced convincing results on many tasks.

One of the drawbacks of such representations is however that they implicitly assume that each component has been generated independently and in a stationary way, where the empirical histogram of components is seen as a sample from an underlying stationary measure. While this viewpoint may translate into adequate properties for some learning tasks (such as translation or rotation invariance when using histograms of colors to manipulate images (Chapelle et al., 1999)), it might prove too restrictive and hence inadequate for other types of problems. Namely, tasks which involve a more subtle mix of detecting *both* conditional (with respect

to the location of the components for instance) and global similarities between the objects. Such problems are likely to arise for instance in speech, language, time series or image processing. In the first three tasks, this consideration is notably treated by most state-of-the-art methods through dynamic programming algorithms capable of detecting and penalizing accordingly local matches between the objects. Using dynamic programming to produce a kernel yielded fruitful results in different applications (Vert et al., 2004; Shimodaira et al., 2002), with the limitation that the kernels obtained in practice are not always positive definite, as reviewed in (Vert et al., 2004).

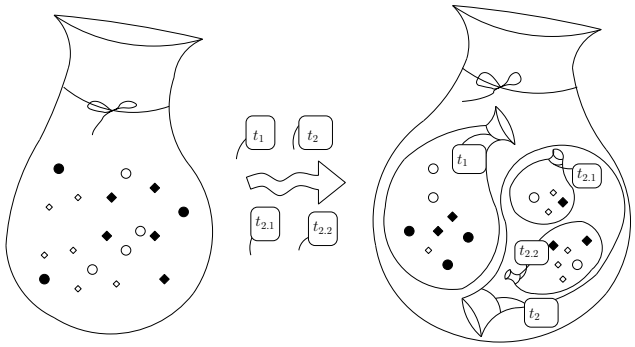


Figure 5.1. *From the bag of components representation to a set of nested bags, using a set of conditioning events.*

We propose in this work a different approach to detect such similarities, an approach that is grounded on the decomposition of structured objects into components that is also able to detect both conditional and global similarities. The motivation behind the kernels presented in this work is both intuitive and computational: intuitively, the global histogram of components, that is the simple bag of components representation of Figure 5.1, may seem inadequate if the components' generation can be particularized for certain event. This phenomenon can be taken into account by considering collections (indexed on the same set of events, to be defined) of nested bags or histograms to describe the object. Kernels that would only rely on these detailed resolutions might however miss the bigger picture that is provided by the global histogram. We propose a trade-off between both viewpoints through a combination that aims at giving a balanced account of both fine and coarse perspectives, hence the name of multiresolution kernels, which we introduce formally in Section 5.2. On the computational side, we show how such a theoretical framework can translate into an efficient factorization detailed in Section 5.3. We then provide experimental results in Section 5.4 on an image retrieval task which show that the methodology improves the performance of kernel based state-of-the art techniques in this field.

5.2 Multiresolution Kernels

In most applications, complex objects can be represented as histograms of components, such as texts as bags of words or images and sequences as histograms of colors and letters. Through this representation, objects are cast as probability laws or measures on the space $\mathcal{X}$ of components, typically multinomials if $\mathcal{X}$ is finite (Lafferty and Lebanon, 2005; Hein and Bousquet, 2005; Chapelle et al., 1999; Joachims, 2002), and compared as such through kernels on measures. An obvious drawback of this representation is that all contextual information on how the components have been sampled is lost, notably any general sense of position in the objects, but also more complex conditional information that may be induced from neighboring components, such as transitions or long range interactions.

In the case of images for instance, one may be tempted to consider not only the overall histogram of colors, but also more specialized histograms which may be relevant for the task. If some local color-overlapping in the images is an interesting or decisive feature of the learning problem, these specialized histograms may be generated arbitrarily following a grid, dividing for instance the image into 4 equal parts, and computing histograms for each corner before comparing them pairwise between two images (see Figure 5.2 for an illustration). If sequences are at stake, these may also be sliced into predefined regions to yield local histograms of letters. If the strings are on the contrary assumed to follow some Markovian behaviour (namely that the appearance of letters in the string is independent of their exact location but only depends on the few letters that precede them), an interesting index would translate into a set of contexts, typically a complete suffix dictionary as detailed in (Cuturi and Vert, 2005). While the two previous examples may seem opposed in the way the histograms are generated, both methodologies stress a particular class of events (location or transitions) that give an additional knowledge on how the components were sampled in the objects. Since both these two approaches, and possibly other ones, can be applied within the framework of this paper using a unified formalism, we present our methodology using a general notation for the index of events. Namely, we note $\mathcal{T}$ for an arbitrary set of conditioning events, assuming these events can be directly observed on the object itself, by contrast with the latent variables approach of (Tsuda et al., 2002b). Considering still, following the generative approach, that an object can be mapped onto a probability measure μ on $\mathcal{X}$, we have that the realization of an event $t \in \mathcal{T}$ can be interpreted in terms of a joint probability $\mu(x,t)$, with $x \in \mathcal{X}$, factorized through Bayes' law as $\mu(x|t)\mu(t)$ to yield the following decomposition of μ as

$$\mu = \sum_{t \in \mathcal{T}} \mu_t,$$

where each $\mu_t \stackrel{\text{def}}{=} \mu(\cdot|t)\mu(t)$ is an element of the set of sub-probability measures $M_+^s(\mathcal{X})$, that is the set of positive measures ρ on $\mathcal{X}$ such that their total mass $\rho(\mathcal{X})$ denoted as $|\rho|$ is *less than* or equal to 1. To take into account the information brought by the events in $\mathcal{T}$, objects can hence be represented as families of measures of $M_+^s(\mathcal{X})$ indexed by $\mathcal{T}$, namely elements μ contained in $M_{\mathcal{T}}(\mathcal{X}) \stackrel{\text{def}}{=} M_+^s(\mathcal{X})^{\mathcal{T}}$.

5.2.1 Local Similarities Between Measures Conditioned by Sets of Events

To compare two objects under the light of their respective decompositions as sub-probability measures μ_t and μ'_t, we make use of an arbitrary positive definite kernel k on $M^s_+(\mathcal{X})$ to which we will refer to as the base kernel throughout the paper. For interpretation purposes only, we may sometimes assume in the following sections that k is an infinitely divisible kernel which can be written as $e^{-\psi}$ where ψ is a negative definite kernel on $M^s_+(\mathcal{X})$. Note also that the kernel is defined not only on probability measures, but also on sub-probabilities. For two elements μ, μ' of $M_{\mathcal{T}}(\mathcal{X})$ and a given element $t \in \mathcal{T}$, the kernel

$$k_t(\mu, \mu') \stackrel{\text{def}}{=} k(\mu_t, \mu'_t)$$

measures the similarity of μ and μ' by quantifying how similarly their components were generated conditionally to event t. For two different events s and t of $\mathcal{T}$, k_s and k_t can be associated through polynomial combinations with positive factors to result in new kernels, notably their sum $k_s + k_t$ or their product $k_s k_t$. This is particularly adequate if some complementarity is assumed between s and t, so that their combination can provide new insights for a given learning task. If on the contrary the events are assumed to be similar, then they can be regarded as a unique event $\{s\} \cup \{t\}$ and result in the kernel

$$k_{\{s\}\cup\{t\}}(\mu, \mu') \stackrel{\text{def}}{=} k(\mu_s + \mu_t, \mu'_s + \mu'_t),$$

which will measure the similarity of m and m' when *either* s or t occurs. The previous formula can be extended to model kernels indexed on a set $T \subset \mathcal{T}$ of similar events, through

$$k_T(m, m') \stackrel{\text{def}}{=} k\left(\mu_T, \mu'_T\right), \text{ where } \mu_T \stackrel{\text{def}}{=} \sum_{t \in T} \mu_t \text{ and } \mu'_T \stackrel{\text{def}}{=} \sum_{t \in T} \mu'_t.$$

Note that this equivalent to defining a negative kernel between elements μ and μ' conditionned by T as $\psi_T(\mu, \mu') \stackrel{\text{def}}{=} \psi(\mu_T, \mu'_T)$.

5.2.2 Resolution Specific Kernels

Let P be a finite partition of $\mathcal{T}$, that is a finite family $P = (T_1, ..., T_n)$ of sets of $\mathcal{T}$, such that $T_i \cap T_j = \varnothing$ if $1 \leq i < j \leq n$ and $\bigcup_{i=1}^n T_i = \mathcal{T}$. We write $\mathcal{P}(\mathcal{T})$ for the set of all partitions of $\mathcal{T}$. Consider now the kernel defined by a partition P as

$$k_P(\mu, \mu') \stackrel{\text{def}}{=} \prod_{i=1}^{n} k_{T_i}(\mu, \mu'). \tag{5.1}$$

The kernel k_P quantifies the similarity between two objects by detecting their joint similarity under all possible events of $\mathcal{T}$, given an *a priori* similarity assumed on the events which is expressed as a partition of $\mathcal{T}$. Note that there is some arbitrary in

this definition since, following the convolution kernels (Haussler, 1999) approach for instance, a simple multiplication of base kernels k_{T_i} to define k_P is used, rather than any other polynomial combination. More precisely, the multiplicative structure of Equation (5.1) quantifies how two objects are similar given a partition P in a way that imposes for the objects to be similar according to all subsets T_i. If k can be expressed as a function of a n.d. kernel ψ, k_P can be expressed as the exponential of

$$\psi_P(\mu, \mu') \stackrel{\text{def}}{=} \sum_{i=1}^{n} \psi_{T_i}(\mu, \mu'),$$

a quantity which penalizes local differences between the decompositions of μ and μ' over $\mathcal{T}$, as opposed to the coarsest approach where $P = \{\mathcal{T}\}$ and only $\psi(\mu, \mu')$ is considered.

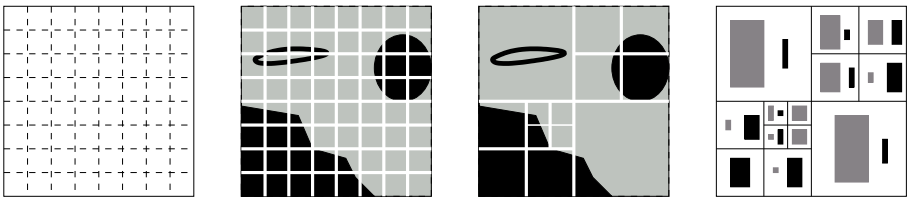


Figure 5.2. *A useful set of events $\mathcal{T}$ for images which would focus on pixel localization can be represented by a grid, such as the 8×8 one represented above. In this case P_3 corresponds to the 4^3 windows presented in the left image, P_2 to the 16 larger square obtained when grouping 4 small windows, P_1 to the image divided into 4 equal parts and P_0 is simply the whole image. Any partition of the image obtained from sets in P_0^3, such as the one represented above, can in turn be used to represent an image as a family of sub-probability measures, which reduces in the case of two-color images to binary histograms as illustrated in the right-most image.*

As illustrated in Figure 5.2 in the case of images expressed as histograms indexed over locations, a partition of $\mathcal{T}$ reflects a given belief on how events should be associated to belong to the same set or dissociated to highlight interesting dissimilarities. Hence, all partitions contained in the set $\mathcal{P}(\mathcal{T})$ of all possible partitions[18] are not likely to be equally meaningful given that some events may look more similar than others. If the index is based on location, one would naturally favor mergers between neighboring indexes. For contexts, a useful topology might also be derived by grouping contexts with similar suffixes.

Such meaningful partitions can be obtained in a general case if we assume the existence of a prior hierarchical information on the elements of $\mathcal{T}$, translated into a series

$$P_0 = \{\mathcal{T}\}, .., P_D = \{\{t\}, t \in \mathcal{T}\}$$

[18] which is quite a big space, since if $\mathcal{T}$ is a finite set of cardinal r, the cardinal of the set of partitions is known as the Bell Number of order r with $B_r = \frac{1}{e}\sum_{u=1}^{\infty} \frac{u^r}{u!} \underset{r\to\infty}{\sim} e^{r \ln r}$.

of partitions of $\mathcal{T}$, namely a hierarchy on $\mathcal{T}$. To provide a hierarchical content, the family $(P_d)_{d=1}^D$ is such that any subset present in a partition P_d is included in a (unique by definition of a partition) subset included in the coarser partition P_{d-1}, and further assume this inclusion to be strict. This is equivalent to stating that each set T of a partition P_d is divided in P_{d+1} through a partition of T which is not T itself. We note this partition $s(T)$ and name its elements the siblings of T. Consider now the subset $\mathcal{P}_D \subset \mathcal{P}(\mathcal{T})$ of all partitions of $\mathcal{T}$ obtained by using only sets in

$$P_0^D \stackrel{\text{def}}{=} \bigcup_{d=1}^{D} P_d,$$

namely $\mathcal{P}_D \stackrel{\text{def}}{=} \{P \in \mathcal{P}(\mathcal{T}) \text{ s.t. } \forall T \in P, T \in P_0^D\}$. The set $\mathcal{P}_D$ contains both the coarsest and the finest resolutions, respectively P_0 and P_D, but also all variable resolutions for sets enumerated in P_0^D, as can be seen for instance in the third image of Figure 5.2.

5.2.3 Averaging Resolution Specific Kernels

Each partition P contained in $\mathcal{P}_D$ provides a resolution to compare two objects, and generates consequently a very large family of kernels k_P when P spans $\mathcal{P}_D$. Some partitions are probably better suited for certain tasks than others, which may call for an efficient estimation of an optimal partition given a task. We take in this section a different direction by considering an averaging of such kernels based on a prior on the set of partitions. In practice, this averaging favours objects which share similarities under a large collection of resolutions.

Definition 5.1. *Let $\mathcal{T}$ be an index set endowed with a hierarchy $(P_d)_{d=0}^D$, π be a prior measure on the corresponding set of partitions $\mathcal{P}_D$ and k a base kernel on $M_+^s(\mathcal{X}) \times M_+^s(\mathcal{X})$. The multiresolution kernel k_π on $M_{\mathcal{T}}(\mathcal{X}) \times M_{\mathcal{T}}(\mathcal{X})$ is defined as*

$$k_\pi(\mu, \mu') = \sum_{P \in \mathcal{P}_D} \pi(P)\, k_P(\mu, \mu'). \tag{5.2}$$

Note that in Equation (5.2), each resolution specific kernel contributes to the final kernel value and may be regarded as a weighted feature extractor.

5.3 Kernel Computation

This section aims at characterizing hierarchies $(P_d)_{d=0}^D$ and priors π for which the computation of k_π is both tractable and meaningful. We first propose a type of hierarchy generated by trees, which is then coupled with a branching process prior to fully specify π. These settings yield a computational time for expressing k_π which is loosely upperbounded by $D \times \operatorname{card} \mathcal{T} \times c(k)$ where $c(k)$ is the time required to compute the base kernel.

5.3.1 Partitions Generated by Branching Processes

All partitions P of $\mathcal{P}_D$ can be generated iteratively through the following rule, starting from the initial root partition $P := P_0 = \{\mathcal{T}\}$. For each set T of P:

1. either leave the set as it is in P,
2. either replace it by its siblings enumerated in $s(T)$, and reapply this rule to each sibling unless they belong to the finest partition P_D.

By giving a probabilistic content to the previous rule through a binomial parameter (i.e. for each treated set assign probability $1-\varepsilon$ of applying rule 1 and probability ε of applying rule 2) a candidate prior for $\mathcal{P}_D$ can be derived, depending on the overall coarseness of the considered partition. For all elements T of P_D this binomial parameter is equal to 0, whereas it can be individually defined for any element T of the $D-1$ coarsest partitions as $\varepsilon_T \in [0,1]$, yielding for a partition $P \in \mathcal{P}_D$ the weight

$$\pi(P) = \prod_{T \in P} (1-\varepsilon_T) \prod_{T \in \overset{\circ}{P}} (\varepsilon_T),$$

where the set $\overset{\circ}{P} = \{T \in P_0^D \text{ s.t. } \exists V \in P, V \subsetneq T\}$ gathers all coarser sets belonging to coarser resolutions than P, and can be regarded as all ancestors in P_0^D of sets enumerated in P.

5.3.2 Factorization

The prior proposed in Section 5.3.1 can be used to factorize the formula in (5.2), which is summarized in this theorem, using notations used in Definition 5.1

Theorem 5.2. *For two elements m, m' of $M_{\mathcal{T}}(\mathcal{X})$, define for T spanning recursively $P_D, P_{D-1}, ..., P_0$ the quantity*

$$K_T = (1-\varepsilon_T) k_T(\mu, \mu') + \varepsilon_T \prod_{U \in s(T)} K_U.$$

Then $k_\pi(\mu, \mu') = K_{\mathcal{T}}$.

Proof. The proof follows from the prior structure used for the tree generation, and can be found in either (Catoni, 2004) or Cuturi and Vert (2005). Figure 5.3 underlines the importance of incorporating to each node K_T a weighted product of the kernels K_U computed by its siblings. □

If the hierarchy of $\mathcal{T}$ is such that the cardinality of $s(T)$ is fixed to a constant α for any set T, typically $\alpha = 4$ for images as seen in Figure 5.2, then the computation of k_π is upperbounded by $(\alpha^{D+1} - 1)c(k)$. This computational complexity may even become lower in cases where the histograms become sparse at fine resolutions, yielding complexities in linear time with respect to the size of the compared objects, quantified by the length of the sequences in Cuturi and Vert (2005) for instance.

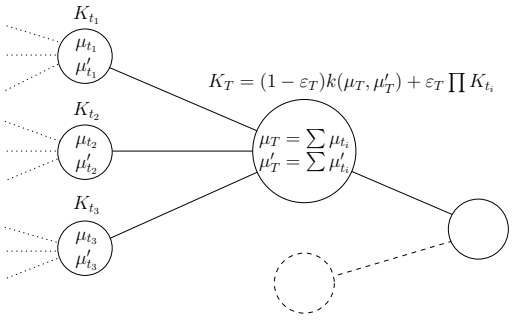


Figure 5.3. *The update rule for the computation of $k\pi$ takes into account the branching process prior by updating each node corresponding to a set T of any intermediary partitions with the values obtained for higher resolutions in $s(T)$.*

5.3.3 Numerical Consistency of the Base Kernel

Before reviewing experimental results, we discuss a numerical issue that may arise from the choice of the basic seed of the multiresolution approach, namely the base kernel k on sub-probabilities.

In our Definition 5.1, any kernel on $M_+^s(\mathcal{X})$ can be used to apply a multiresolution comparison scheme on families of measures. If we even look for more generality in our formulation, it is easy to note that a different kernel k_t might be used for each event t of $\mathcal{T}$, without altering the overall applicability of the factorization above. However, we only consider in this discussion a unique choice k for all events and partitions.

From a numerical perspective, and for a partition $P \in \mathcal{P}_D$, the kernel k_P – which is a product of many base kernels $k_T, T \in P$ – can become quickly negligible for fine partitions, notably if the base kernel is slightly diagonally dominant. Furthermore, if P is fine, the weight $\pi(P)$ that is used for k_P in Equation (5.2) does already penalize it with respect to coarser partitions. To maintain a consistent weighting framework, one would expect that for any two partitions P_1 and P_2 of $\mathcal{P}_D$, both k_{P_1} and k_{P_2} share similar ranges of values, so that their respective importance in the mixture only relies on $\pi(P_1)$ and $\pi(P_2)$. This notion is difficult to quantify, and we only formulate here a basic assumption for k. As can be observed in Figure 5.4, comparing two families of measures μ, μ' through multiresolution can be roughly seen as comparing two broken lines that reach the surface of the simplex (when $\mathcal{X}$ is finite) after a finite number of steps. The condition we propose here to ensure a basic numerical consistency is that the base kernel k is geometrically homogeneous, that is for $t > 0$ and any measures $\nu, \nu' \in M_+^s(\mathcal{X})$,

$$k(\nu, \nu') = \left[k\left(\frac{\nu}{t}, \frac{\nu'}{t}\right) \right]^t .$$

Seen from the viewpoint of Figure 5.4, this would mean that in the special case that μ and μ' are families of parallel segments of equal size, comparing them would not be

affected by a partition change, that is, for all partitions P_1 and P_2 of $\mathcal{P}_D$, $k_{P_1}(\mu, \mu') = k_{P_2}(\mu, \mu')$. If we translate this condition for infinitely divisible base kernels, we obtain a more usual notion of additive homogeneity for the n.d. kernel ψ, that is for $t > 0$ and two measures $\nu, \nu' \in M_+^s(\mathcal{X})$,

$$\psi(\nu, \nu') = t\psi\left(\frac{\nu}{t}, \frac{\nu'}{t}\right).$$

This elementary criterion is consistent with the experimental results we obtained in practice, which show better results for such kernels. We even observed that sub-homogeneous n.d. kernels, that is such that

$$t\psi\left(\frac{\nu}{t}, \frac{\nu'}{t}\right) \leq \psi(\nu, \nu'),$$

also seem to work better in practice. This issue is itself related to the branching process prior that is used, and the relations between the branching-process prior and the homogeneity behaviour of the kernel ψ that is used is a topic of current research.

5.4 Experiments

We present in this section experiments inspired by the image retrieval task first considered in (Chapelle et al., 1999) and also used in (Hein and Bousquet, 2005), although the images used here are not exactly the same. The dataset was also extracted from the Corel Stock database and includes 12 families of labelled images, each class containing 100 color images, each image being coded as 256×384 pixels with colors coded in 24 bits (16M colors). The families depict *bears, African specialty animals, monkeys, cougars, fireworks, mountains, office interiors, bonsais, sunsets, clouds, apes* and *rocks and gems*. The database is randomly split into balanced sets of 900 training images and 300 test images. The task consists in classifying the test images with the rule learned by training 12 one-vs-all SVM's on the learning fold. The object are then classified according to the SVM performing the highest score, namely with a "winner-takes-all" strategy. The results presented in this section are averaged over 4 different random splits. We used the CImg package to generate histograms and the Spider toolbox for the SVM experiments[19].

We adopted a coarser representation of 9 bits for the color of each pixel from the $98,304$ ones stored in an image, rather than the 24 bits originally available, to reduce the size of the RGB color space to $8^3 = 512$ from the original set of $256^3 = 16,777,216$ colors. In this image retrieval experiment, we used localization as the conditioning index set, dividing the images into $1, 4, 4^2 = 16, 9$ and $9^2 = 81$ local histograms (in Figure 5.2 the image was for instance divided into $4^3 = 64$ windows). To define the branching process prior, we simply set an uniform value over all the grid of ε of $1/\alpha$, an usage motivated by previous experiments led in a similar context (Cuturi and Vert, 2005). Finally, we used kernels described in

[19] http://cimg.sourceforge.net/ and http://www.kyb.tuebingen.mpg.de/bs/people/spider/

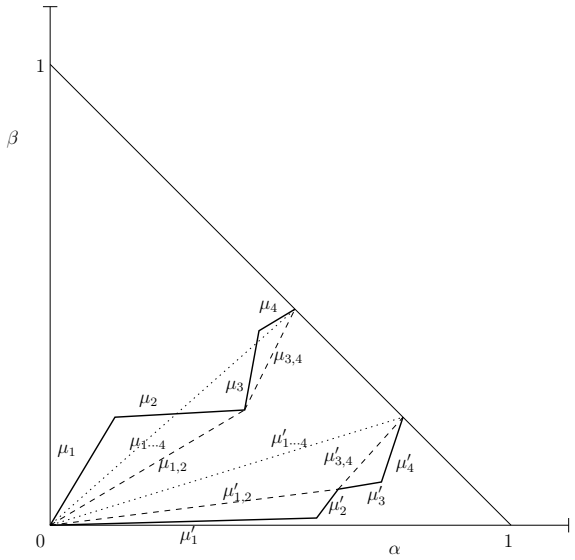


Figure 5.4. *Two families of measures μ and μ', decomposed as 4 subprobabilities seen as 4 segments in the interior of the simplex Σ_2, μ_i and μ'_i, $i = 1..4$. We write α, β for the coordinates of a multinomial, where $\alpha + \beta = 1$ on the border of the simplex. The coarsest approach only compares the averages $\mu_{1...4}$ and $\mu'_{1...4}$ represented by dotted lines. The finest one compares pairwise μ_i and μ'_i, for $i = 1, 2, 3, 4$. Other decompositions of the respective broken lines are used by the multiresolution kernel which may used local averages of short segments, such as $\mu_{1,2}$ and $\mu'_{1,2}$. As an analogy, for two points starting from zero and reaching the simplex in $\mu_{1...4}$ and $\mu'_{1...4}$, one may have a more detailed perspective than just comparing their arrival point by considering the local dissimilarities of their trajectory in-between stop-overs. Such similarities may then be averaged according to some weighting scheme on the importance of such stop-overs.*

both (Chapelle et al., 1999) and (Hein and Bousquet, 2005) to define the base kernel k. These kernels can be directly applied on sub-probability measures, which is not the case for all kernels on multinomials, notably the Information Diffusion Kernel (Lafferty and Lebanon, 2005). We report results for two families of kernels, namely the Radial Basis Function expressed for multinomials and the entropy kernel based on the Jensen divergence (Hein and Bousquet, 2005; Cuturi et al., 2005):

$$k_{a,b,\rho}(\theta, \theta') = e^{-\rho \sum |\theta_i^a - \theta_i'^a|^b} \ , \ k_h(\theta, \theta') = e^{-h\left(\frac{\theta+\theta'}{2}\right) + \frac{1}{2}\left(h(\theta)+h(\theta')\right)}.$$

For most kernels not presented here, the multiresolution approach usually improved the performance in a similar way than the results presented in Table 5.1. Finally, we also report that using only the finest resolution available in each (α, D) setting, that is a branching process prior uniformly set to 1, yielded better results than

the use of the coarsest histogram without achieving however the same performance of the multiresolution averaging framework, which highlights the interest of taking both coarse and fine perspectives into account. When $a = .25$ for instance, this setting produced 16.5% and 16.2% error rates for $\alpha = 4$ and $D = 1, 2$, and 15.8% for $\alpha = 9$ and $D = 1$.

Kernel	RBF, $b = 1$, $\rho = .01$			JD
	$a = .25$	$a = .5$	$a = 1$	
global histogram	18.5	18.3	20.3	21.4
$D = 1, \alpha = 4$	15.4	16.4	18.8	17
$D = 2, \alpha = 4$	*13.9*	*13.5*	15.8	15.2
$D = 1, \alpha = 9$	14.7	14.7	16.6	15
$D = 2, \alpha = 9$	15.1	15.1	30.5	15.35

Table 5.1. *Results for the Corel image database experiment in terms of error rate, with 4 fold cross-validation and 2 different types of tested kernels, the RBF and the Jensen Divergence.*

Bibliography

Akaho, S. (2001). A kernel method for canonical correlation analysis. In *Proceedings of International Meeting on Psychometric Society (IMPS2001)*.

Amari, S.-I. and Nagaoka, H. (2001). *Methods of Information Geometry*. AMS vol. 191.

Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68:337 – 404.

Bach, F. and Jordan, M. (2002). Kernel independent component analysis. *Journal of Machine Learning Research*, 3:1–48.

Bach, F. R. and Jordan, M. I. (2005). Predictive low-rank decomposition for kernel methods. In *Proceedings of ICML '05: Twenty-second international conference on Machine learning*. ACM Press.

Bach, F. R., Lanckriet, G. R. G., and Jordan, M. I. (2004). Multiple kernel learning, conic duality, and the smo algorithm. In *ICML '04: Twenty-first international conference on Machine learning*. ACM Press.

Bejerano, G. and Yona, G. (1999). Modeling protein families using probabilistic suffix trees. In Istrail, S., Pevzner, P., and Waterman, M., editors, *Proceedings of the third Annual International Conference on Computational Molecular Biology (RECOMB)*, pages 15–24, Lyon, France. ACM Press.

Ben-Hur, A. and Brutlag, D. L. (2003). Remote homology detection: a motif based approach. In *ISMB (Supplement of Bioinformatics)*, pages 26–33.

Berg, C., Christensen, J. P. R., and Ressel, P. (1984). *Harmonic Analysis on Semigroups*. Number 100 in Graduate Texts in Mathematics. Springer-Verlag.

Berlinet, A. and Thomas-Agnan, C. (2003). *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Kluwer Academic Publishers.

Bernstein, D. S. (2005). *Matrix Mathematics: Theory, Facts, and Formulas with Application to Linear Systems Theory*. Princeton University Press.

Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the 5th annual ACM workshop on Computational Learning Theory*, pages 144–152. ACM Press.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.

Brown, M. P., Hughey, R., Krogh, A., Mian, I. S., Sjölander, K., and Haussler, D. (1993). Using Dirichlet mixture priors to derive hidden Markov models for protein families. In *Proc. of First Int. Conf. on Intelligent Systems for Molecular Biology*, pages 47–55. AAAI/MIT Press.

Brown, M. P. S., Grundy, W. N., Lin, D., Cristianini, N., Sugnet, C. W., Furey, T. S., Ares, M. J., and Haussler, D. (2000). Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proc. Natl. Acad. Sci. USA*, 97:262–267.

Catoni, O. (2004). *Statistical learning theory and stochastic optimization, Ecole d'été de probabilités de Saint-Flour XXXI -2001*. Number 1851 in Lecture Notes in Mathematics. Springer Verlag.

Chapelle, O., Haffner, P., and Vapnik, V. (1999). SVMs for histogram based image classification. *IEEE Transactions on Neural Networks*, 10(5):1055.

Chapelle, O., Vapnik, V., Bousquet, O., and Mukherjee, S. (2002). Choosing multiple parameters for support vector machines. *Machine Learning*, 46(1/3):131.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20:273.

Cover, T. M. and Thomas, J. A. (1991). *Elements of Information Theory*. Wiley, New York.

Csató, L. and Opper, M. (2002). Sparse on-line gaussian processes. *Neural Computation*, 14(3):641–668.

Cucker, F. and Smale, S. (2002). On the mathematical foundations of learning. *BAMS: Bulletin of the American Mathematical Society*, 39.

Cuturi, M. and Fukumizu, K. (2005). Multiresolution kernels, arxiv cs.lg/0507033.

Cuturi, M., Fukumizu, K., and Vert, J.-P. (2005). Semigroup kernels on measures. *Journal of Machine Learning Research*, 6:1169–1198.

Cuturi, M. and Vert, J.-P. (2005). The context-tree kernel for strings. *Neural Networks*, 18(8).

Cuturi, M. and Vert, J.-P. (2005). Semigroup kernels on finite sets. In Saul, L. K., Weiss, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 17*, pages 329–336. MIT Press, Cambridge, MA.

Davis, C. (1957). All convex invariant functions of Hermitian matrices. *Archiv der Mathematik*, 8:276–278.

Dieudonné, J. (1968). *Calcul Infinitésimal*. Hermann, Paris.

Durbin, R., Eddy, S., Krogh, A., and Mitchison, G. (1998). *Biological sequence analysis - Probabilistic models of proteins and nucleic acids*. Cambridge University Press.

Eichhorn, J. and Chapelle, O. (2004). Object categorization with svm: Kernels for local features. Technical Report 137, MPI for Biological Cybernetics.

Endres, D. M. and Schindelin, J. E. (2003). A new metric for probability distributions. *IEEE Transactions on Information Theory*, 49(7):1858–1860.

Eskin, E., Noble, W., and Singer, Y. (2000). Protein family classification using sparse Markov transducers. *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*.

Fuglede, B. and Topsøe, F. (2004). Jensen-Shannon divergence and Hilbert space embedding. In *Proc. of the Internat. Symposium on Information Theory*, page 31.

Fukumizu, K., Bach, F., and Gretton, A. (2005). Consistency of kernel canonical correlation analysis. Technical report, Research Memorandum No.942, Institute of Statistical Mathematics.

Fukumizu, K., Bach, F., and Jordan, M. (2004). Dimensionality reduction for supervised learning with reproducing kernel hilbert spaces. *Journal of Machine Learning Research*, 5:73–99.

Girosi, F., Jones, M., and Poggio, T. (1995). Regularization theory and neural networks architectures. *Neural Computation*, 7(2):219–269.

Gribskov, M. and Robinson, N. L. (1996). Use of receiver operating characteristic (ROC) analysis to evaluate sequence matching. *Computers & Chemistry*, 20(1):25–33.

Haasdonk, B. (2005). Feature space interpretation of SVMs with indefinite kernels. *IEEE Trans. Pattern Anal. Mach. Intell*, 27(4):482–492.

Haasdonk, B. and Keysers, D. (2002). Tangent distance kernels for support vector machines. In *Proceedings of the International Conference on Pattern Recognition (2)*, pages 864–868.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). *Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Verlag.

Haussler, D. (1999). Convolution kernels on discrete structures. Technical report, UC Santa Cruz. USCS-CRL-99-10.

Hein, M. and Bousquet, O. (2005). Hilbertian metrics and positive definite kernels on probability measures. In Ghahramani, Z. and Cowell, R., editors, *Proceedings of AISTATS 2005*.

Hua, S. and Sun, Z. (2001). Support vector machine approach for protein subcellular localization prediction. *Bioinformatics*, 17(8):721–728.

Hubbard, T., Murzin, A., Brenner, S., and Chothia, C. (1997). Scop: a structural classification of proteins database. *Nucleic Acids Research*, pages 236– 239.

Jaakkola, T., Diekhans, M., and Haussler, D. (2000). A discriminative framework for detecting remote protein homologies. *Journal of Computational Biology*, 7(1,2):95–114.

Jaakkola, T. S. and Haussler, D. (1999). Exploiting Generative Models in Discriminative Classifiers. In Kearns, M. S., Solla, S. A., and Cohn, D. A., editors, *Advances in Neural Information Processing Systems 11*. MIT Press.

Jebara, T., Kondor, R., and Howard, A. (2004). Probability product kernels. *Journal of Machine Learning Research*, 5:819–844.

Joachims, T. (1997). Text categorization with support vector machines: Learning with many relevant features. Technical Report LS VIII-Report, Universität Dortmund, Dortmund, Germany.

Joachims, T. (2002). *Learning to Classify Text Using Support Vector Machines: Methods, Theory, and Algorithms*. Kluwer Academic Publishers.

Kashima, H., Tsuda, K., and Inokuchi, A. (2003). Marginalized kernels between labeled graphs. In Faucett, T. and Mishra, N., editors, *Proceedings of the Twentieth International Conference on Machine Learning*, pages 321–328. AAAI Press.

Kimeldorf, G. S. and Wahba, G. (1971). Some results on tchebycheffian spline functions. *Journal of Mathematical Analysis and Applications*, 33:82–95.

Koecher, M. (1957). Positivitätbereiche im $\mathbb{R}^m$. *Amer. Jour. Math. 79, 575–596*.

Kondor, R. and Jebara, T. (2003). A kernel between sets of vectors. In Faucett, T. and Mishra, N., editors, *Proceedings of the Twentieth International Conference on Machine Learning*, pages 361–368.

Kondor, R. and Lafferty, J. (2002). Diffusion kernels on graphs and other discrete input spaces. In *Proceedings of the Nineteenth International Conference on Machine Learning*, pages 315–322.

Lafferty, J. and Lebanon, G. (2002). Information diffusion kernels. In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, Cambridge, MA. MIT Press.

Lafferty, J. and Lebanon, G. (2005). Diffusion kernels on statistical manifolds. *Journal of Machine Learning Research*, 6:129–163.

Lanckriet, G. R. G., Bie, T. D., Cristianini, N., Jordan, M. I., and Noble, W. S. (2004). A statistical framework for genomic data fusion. *Bioinformatics*, 20(16):2626–2635.

Leslie, C., Eskin, E., and Noble, W. S. (2002). The spectrum kernel: a string kernel for svm protein classific ation. In *Proceedings of the Pacific Symposium on Biocomputing 2002*, pages 564–575. World Scientific.

Leslie, C., Eskin, E., Weston, J., and Noble, W. S. (2003). Mismatch string kernels for svm protein classification. In Becker, S., Thrun, S., and Obermayer, K., editors, *Advances in Neural Information Processing Systems 15*. MIT Press.

Lewis, A. S. (2003). The mathematics of eigenvalue optimization. *Mathematical Programming*, 97(1–2):155–176.

Li, M., Chen, X., Li, X., Ma, B., and Vitanyi, V. (2004). The similarity metric. *IEEE Transactions on Information Theory*, 50(12):3250–3264.

Liao, L. and Noble, W. S. (2002). Combining pairwise sequence similarity and support vector machines for remote protein homology detection. In *Proceedings of the Sixth Annual International Conference on Computational Molecular Biology*, pages 225–232.

Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., and Watkins, C. (2002). Text classification using string kernels. *Journal of Machine Learning Research*, 2:419–444.

Mahé, P., Ueda, N., Akutsu, T., Perret, J.-L., and Vert, J.-P. (2004). Extensions of marginalized graph kernels. In Greiner, R. and Schuurmans, D., editors, *Proceedings of the Twenty-First International Conference on Machine Learning (ICML 2004)*, pages 552–559. ACM Press.

Matheron, G. (1965). *Les variables régionalisées et leur estimation*. Masson, Paris.

Melzer, T., Reiter, M., and Bischof, H. (2001). Nonlinear feature extraction using generalized canonical correlation analysis. In *Proceedings of International Conference on Artificial Neural Networks (ICANN)*, pages 353–360.

Mercer, T. (1909). Functions of positive and negative type and their connection with the theory of integral equations. *Philosophical Transactions of the Royal Society of London, Series A*, 209:415–446.

Miller, K. S. (1987). *Some Eclectic Matrix Theory*. Robert E. Krieger Publishing Co., Krieger Drive, Malabar, FL 32950.

Moreno, P. J., Ho, P. P., and Vasconcelos, N. (2004). A kullback-leibler divergence based kernel for svm classification in multimedia applications. In Thrun, S., Saul, L., and Schölkopf, B., editors, *Advances in Neural Information Processing Systems 16*. MIT Press.

Nemenman, I., Shafee, F., and Bialek, W. (2002). Entropy and inference, revisited. In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, Cambridge, MA. MIT Press.

Noble, W. S. and Liao, L. (2002). Combining pairwise sequence similarity and support vector machines for remote protein homology detection. *Proceedings of the Sixth Annual International Conference on Research in Computational Molecular Biology*, pages 225–232.

Ong, C. S., Smola, A. J., and Williamson, R. C. (2005). Learning the kernel with hyperkernels. *Journal of Machine Learning Research*, 6:1043–1071.

Österreicher, F. and Vajda, I. (2003). A new class of metric divergences on probability spaces and its applicability in statistics. *Annals of the Institute of Statistical Mathematics*, 55:639–653.

Parzen, E. (1962). Extraction and detection problems and reproducing kernel Hilbert spaces. *Journal of the Society for Industrial and Applied Mathematics. Series A, On control*, 1:35–62.

Platt, J. (1999). Fast training of support vector machines using sequential minimal optimization. In Scholkopf, B., Burges, C., and Smola, A., editors, *Advances in Kernel Methods—Support Vector Learning*. MIT Press.

Pontil, M. and Verri, A. (1998). Support vector machines for 3D object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(6):637–646.

Ralaivola, L., Swamidass, J. S., Saigo, H., and Baldi, P. (2005). Graph kernels for chemical informatics. *Neural Networks*, 18(8).

Rao, C. (1987). Differential metrics in probability spaces. In Amari, S.-I., Barndorff-Nielsen, O., Kass, R., Lauritzen, S., and Rao, C., editors, *Differential Geometry in Statistical Inference*, Hayward,CA. Institute of Mathematical Statistics.

Rätsch, G. and Sonnenburg, S. (2004). Accurate splice site prediction for caenorhabditis elegans. In Schölkopf, B., Tsuda, K., and Vert, J.-P., editors, *Kernel Methods in Computational Biology*. MIT Press.

Rudin, W. (1962). *Fourier Analysis on Groups*. John Wiley & sons.

Salton, G. (1989). *Automatic Text Processing*. AddisonWesley.

Schoenberg, I. (1942). Positive definite functions on spheres. *Duke Math. J.*, 9:96–108.

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

Schölkopf, B. and Smola, A. J. (2002). *Learning with Kernels: Support Vector Machines, Regularization , Optimization, and Beyond*. MIT Press.

Schölkopf, B., Tsuda, K., and Vert, J.-P. (2004). *Kernel Methods in Computational Biology*. MIT Press.

Schölkopf, B., Weston, J., Eskin, E., Leslie, C., and Noble, W. S. (2002). A kernel approach for learning from almost orthogonal patterns. In Elomaa, T., Mannila, H., and Toivonen, H., editors, *Proceedings of ECML 2002, 13th European Conference on Machine Learning, Helsinki, Finland, August 19-23, 2002*, volume 2430 of *Lecture Notes in Computer Science*, pages 511–528. Springer.

Seeger, M. (2002). Covariance kernels from bayesian generative models. In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, pages 905–912. MIT Press.

Seeger, M. (2004). Gaussian processes for machine learning. *International Journal of Neural Systems*, 14(2):69–106.

Shimodaira, H., Noma, K.-I., Nakai, M., and Sagayama, S. (2002). Dynamic time-alignment kernel in support vector machine. In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, Cambridge, MA. MIT Press.

Sindhwani, V., Niyogi, P., and Belkin, M. (2005). Beyond the point cloud: from transductive to semi-supervised learning. In *Proceedings of ICML '05: Twenty-second international conference on Machine learning*. ACM Press.

Smith, N. and Gales, M. (2002). Speech recognition using svms. In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, Cambridge, MA. MIT Press.

Tikhonov, A. N. and Arsenin, V. Y. (1977). *Solution of Ill-Posed Problems*. Winston, Washington D. C.

Tsuda, K., Akaho, S., and Asai, K. (2003). The em algorithm for kernel matrix completion with auxiliary data. *Journal of Machine Learning Research*, 4:67–81.

Tsuda, K., Akaho, S., Kawanabe, M., and Müller, K.-R. (2004). Asymptotic properties of the fisher kernel. *Neural Computation*, 16(1):115–137.

Tsuda, K., Kawanabe, M., Rätsch, G., Sonnenburg, S., and Müller, K.-R. (2002a). A new discriminative kernel from probabilistic models. *Neural Computation*, 14(10):2397–2414.

Tsuda, K., Kin, T., and Asai, K. (2002b). Marginalized kernels for biological sequences. *Bioinformatics*, 18(Suppl 1):268–275.

Tsuda, K. and Noble, W. (2004). Learning kernels from biological networks by maximizing entropy. *Bioinformatics (ISMB/ECCB 2004)*, 20(Suppl. 1):i326–i333.

Tsuda, K., Rätsch, G., and Warmuth, M. K. (2005). Matrix exponentiated gradient updates for on-line learning and bregman projection. *Journal of Machine Learning Research*, 6:995–1018.

Vapnik, V. N. (1998). *Statistical Learning Theory*. Wiley.

Vert, J.-P. (2002). A tree kernel to analyze phylogenetic profiles. *Bioinformatics*, 18:S276–S284.

Vert, J.-P. and Kanehisa, M. (2003). Graph-driven features extraction from microarray data using diffusion kernels and kernel cca. In Becker, S., Thrun, S., and Obermayer, K., editors, *Advances in Neural Information Processing Systems 15*. MIT Press.

Vert, J.-P., Saigo, H., and Akutsu, T. (2004). Local alignment kernels for protein sequences. In Schölkopf, B., Tsuda, K., and Vert, J.-P., editors, *Kernel Methods in Computational Biology*. MIT Press.

Vert, J.-P. and Yamanishi, Y. (2005). Supervised graph inference. In Saul, L. K., Weiss, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 17*. MIT Press.

Vert, R. and Vert, J.-P. (2005). Consistency and convergence rates of one-class svm and related algorithms. Technical Report 1414, LRI, Université Paris Sud.

Wahba, G. (1990). *Splines Models for Observational Data*. Series in Applied Mathematics, Vol. 59, SIAM.

Watkins, C. (2000). Dynamic alignment kernels. In Smola, A., Bartlett, P., Schölkopf, B., and rmans, D. S., editors, *Advances in Large Margin Classifiers*, pages 39–50. MIT Press.

Willems, F. M. J., Shtarkov, Y. M., and Tjalkens, T. J. (1995). The context-tree weighting method: basic properties. *IEEE Transactions on Information Theory*, pages 653–664.

Wolf, L. and Shashua, A. (2003). Learning over sets using kernel principal angles. *Journal of Machine Learning Research*, 4:913–931.

Zhang, D., Chen, X., and Lee, W. S. (2005). Text classification with kernels on the multinomial manifold. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 266–273. ACM Press.

Zhu, J. and Hastie, T. (2002). Kernel logistic regression and the import vector machine. In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, pages 1081–1088, Cambridge, MA. MIT Press.