Gaussian Processes on Distributions based on Regularized Optimal Transport

Students:
Louis ALLAIN
Léonard GOUSSET
Julien HEURTIN

Referents :
Brian Staber
Sébastien Da Veiga

Methodological Project

January 2024

Abstract

After giving reminders on kernel methods, optimal transport, regularized optimal transport and the Sinkhorn algorithm, we show that the kernel proposed in [1] works in a kernel ridge regression setting. We achieve similar performances as the Gaussian Processes model and explore a little beyond the toy experiment.

Glossary

- **Algorithm**:
  An algorithm is a systematic and precise set of step-by-step instructions designed to perform a specific task or solve a particular problem. It serves as a computational procedure, outlining the sequence of operations required to achieve a desired outcome.

- **Machine Learning**:
  This a field of science where we give a problem to a machine and it learns the optimal solution to this problem. Usually this is an *optimization* problem. The machine learns the best parameters for a problem.

- **Regression**:
  Regression is a specific task in machine learning. It involves the knowledge of input and output data, and makes the machine learns how to predict the output from the input data.

- **Distribution**:
  A distribution refers to a set of all possible values and their corresponding probabilities. It describes how the values of a variable are spread or distributed across different outcomes.

- **Kernel**:
  The word kernel is used in a variety of mathematics domains. In our use case it is simply a (positive definite) function that takes two arguments and outputs a real number. It can be seen as a similarity measure between those two objects.

- **Transport**:
  If you have two histograms, you want to know how to ”transform” one into the other. They are both density measures, therefore their integral is one. You slice the histograms (or density) into units and the transport correspond to the way you map each unit of the first hsitogram to units of the second. There are infinite ways to do so.

- **Optimal Transport**:
  The *optimal* transport is the way you map one histogram to another that minimizes a given cost function. This function essentially tells you how much does it costs to map every unit of the first density to the second.

- **Regularization**:
  In machine learning, the word regularization refers to the reformulation of the initial optimization problem that adds a constraint (usually on the models parameter). It is very useful to bypass computational difficulties one may encounter in machine learning.

- **Sinkhorn**:
  The Sinkhorn algorithm, is an iterative numerical method used for regularizing and solving optimal transport problems. It greatly helps with computational performances.

- **Dual formulation**:
  In a constraint optimization problem, the dual formulation introduces ”dual” variables associated with the constraint. Dual formulations are often useful in optimization theory and algorithms, providing insights into the problem structure and facilitating efficient solution techniques.

- **Complexity**:
  The complexity of an algorithm is the way the computation time scales with the data the algorithm receives. Usually, one aims for the smallest complexity, such as a linear or logarithmic complexity.

Contents

1 Introduction

In recent years, the intersection of Gaussian Processes (GPs) and Optimal Transport (OT) has emerged as a fertile ground for innovation in statistical learning. Gaussian Processes, known for their flexibility and robustness, are a cornerstone in the field of machine learning, providing a probabilistic approach to learning in kernel-based models. On the other hand, Optimal Transport offers a powerful framework for comparing probability distributions, with applications ranging from economics to image processing.

The development presented in François Bachoc et al.'s paper[1] introduces a significant advancement in statistical analysis and algorithm design through the creation of a new kernel using Gaussian Processes, enhanced by techniques from Regularized Optimal Transport. This report offers a comprehensive overview of this novel kernel, detailing its conceptual foundation and potential applications in various scientific and technological fields.

In this project, we are collaborating with Safran, a leading aerospace manufacturer, to pioneer innovative approaches in designing aircraft engine components, specifically focusing on the blades of a propeller. Our objective is to leverage advanced machine learning techniques, particularly in the realm of regression analysis, to optimize the efficiency of these critical components. The target variable in our machine learning model is the efficiency metric of the blade, a quantifiable measure of performance under various operational conditions. Intriguingly, the feature variable is the probabilistic distribution of the blade's structure, represented as a point cloud in a three-dimensional space ($\mathbb{R}^3$). This representation is not just a mere geometrical depiction but encapsulates the intricate variability and physical characteristics of the blade. Through the analysis of this relationship between the shape distribution and efficiency, our goal is to identify underlying patterns and gain insights that could inform the engineering of propeller blades with enhanced efficiency and durability. This research represents a significant step in aerospace engineering, merging advanced statistical methods with engineering challenges, and contributes to the ongoing evolution of aircraft propulsion technology.

To approach François Bachoc et al.'s paper with a comprehensive understanding, we will first elucidate the methods of kernels in statistical learning, with a specific focus on kernel ridge regression (see sec. 2). Additionally, we will revisit the genesis of optimal transport and its more recent formulation in an entropic framework, along with its resolution using the Sinkhorn algorithm (see sec. 3). Subsequently, we will provide a detailed exposition of the kernel developed within the paper under examination (see sec. 4).

2 Kernel methods

2.1 Framework

2.1.1 Regression

Regression is a fundamental task in machine learning. Let $\mathcal{X}$ be a nonempty set, often called the *input space* and let $f\colon \mathcal{X} \to \mathcal{Y}$. Because we are focusing on a regression problem, the *target* or *input space* is going to be continuous, meaning $\mathcal{Y} \subset \mathbb{R}$. Assume, for $n \in \mathbb{N}$, that we are given the following training data $\mathcal{D}_n = ((x_i, y_i))_{1 \leq i \leq n} \in (\mathcal{X} \times \mathbb{R})^n$ such that:

$$y_i = f(x_i) + \varepsilon_i, \quad i = 1, \dots n$$

where ε_i are random variable of mean 0 and variance σ^2. They represent the "noise" of the data, the variability of y_i that is not explained by the input data. The regression task is to estimate the unknown function f from the training data $\mathcal{D}_n$. The function f is called the regression function and is the conditional mean of the output knowing an input:

$$f(x) = \mathbb{E}\left[y|x\right]$$

2.1.2 Positive definite kernel

A primary constituent of the methods discussed here is the positive definite kernel. The kernel is a way to measure similarity in the input space $\mathcal{X}$. We remind that this input space can be anything, $\mathbb{R}$ for tabular data, the set of words for natural language processing, images for computer vision or the set of distributions as in our context. Here we give a definition of a positive definite kernel and a matrix interpretation of it. Then we move on to properties and examples.

Definition 1. *Let $\mathcal{X}$ be a nonempty set. A symmetric function $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a **positive definite kernel**, if for any $n \in \mathbb{N}, (x_1, \dots x_n) \subset \mathcal{X}$ and $\forall\, (c_1, \dots c_n) \subset \mathbb{R}$,*

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

Remark 1. *We consider here only real-valued kernel, which is why k needs to be symmetric. Results exists for complex-valued kernels but they are out of scope here.*

A few example of kernels are given below.

Example 1 (Linear kernel)**.** *Let $\mathcal{X} \subset \mathbb{R}^d$. The linear kernel $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is the most simple kernel and is defined by*

$$k\left(x, x'\right) = x^\intercal x', \quad x, x' \in \mathcal{X}$$

Example 2 (Polynomial kernel)**.** *Let $\mathcal{X} \subset \mathbb{R}^d$. For $c > 0$ and $m \in \mathbb{N}$, the polynomial kernel $k_{c,m}\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is defined by*

$$k_{c,m}\left(x, x'\right) = \left(x^\intercal x' + c\right)^m, \quad x, x' \in \mathcal{X}$$

Example 3 (Gaussian kernel). *Let $\mathcal{X} \subset \mathbb{R}^d$. For $\gamma > 0$, the Gaussian kernel $k_\gamma \colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is defined by*

$$k_\gamma(x, x') = e^{-\frac{\|x - x'\|^2}{\gamma^2}}, \quad x, x' \in \mathcal{X}$$

where $\|.\|$ can be any norm on $\mathbb{R}^d$.

Remark 2. *For simplicity, examples here are only given for kernels on $\mathcal{X} \subset \mathbb{R}^d$. But one strength of kernel methods is that the kernel can be defined over any nonempty set $\mathcal{X}$, for instance the space of distribution as it used here. From there on, any model can be used with the defined kernel. The kernel's role is to measure the similarity in the input space $\mathcal{X}$.*

This definite positive property is essential as it transforms complex machine learning problems into convex optimisation problems, for which we have tools to solve. Consequently they are widely used in numerous machine learning methods. Quite surprisingly they arise in different machine learning framework. In the Bayesian paradigm they appear as covariance matrices in Gaussian Processes. But they also emerge in the more traditional frequentist paradigm as a way to take into account non linearity without adding too much complexity, subject of the next section 2.2.

2.2 Reproducing Kernel Hilbert Space

To estimate the regression function, numerous method have been developed. Most of them works well for the linear case. However, in the real world arise dependencies and relations that are non linear. Reproducing Kernel Hilbert Spaces (RKHS) methods combines the two worlds, allowing us to analyse non linear relationships in a linear setting. The kernel (a positive definite kernel as mentioned in paragraph 2.1.2) corresponds to a dot product in a high dimensional space, often called the *feature space*. In this space, the algorithms and methods used are linear. As long as we can express the calculations using the kernel, none of the computations has to be performed in the complex feature space. We assume from now on that we are given a nonempty input space $\mathcal{X}$ and a positive definite kernel k. We start be defining the *feature space* and *feature map* notions.

Definition 2. *Let $\mathcal{X}$ be a nonempty set. We call $\mathcal{H} \coloneqq \{f \colon \mathcal{X} \to \mathbb{R}\}$ the **feature space**.*

Definition 3. *A **feature map** is an application mapping input space objects into the feature space:*

$$\Phi \colon \mathcal{X} \to \mathcal{H}$$

To execute such a computation, we need the kernel to satisfy, for a given feature map Φ:

$$\forall x, x' \in \mathcal{X}^2, \quad k(x, x') = \langle \Phi(x), \Phi(x') \rangle_{\mathcal{H}} \tag{1}$$

It is shown in Schölkopf et al.[2] that the set of kernels satisfying relation 1, meaning kernels that admit a representation as a dot product in a feature space, is equal to the set of positive definite kernels studied in 2.1.2. This shows that the right class of kernel to study is the definite positive ones. The construction of kernels from feature maps is developped in Schölkopf et al.[2]. We now present spaces related to such a kernel and expose strong theorem about those spaces and kernels.

Definition 4. *Let $\mathcal{X}$ a nonempty set and k a positive definite kernel on $\mathcal{X}$. A Hilbert space $\mathcal{H}_k$ of function over $\mathcal{X}$ endowed with an inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}_k}$ is a **Reproducing Kernel Hilbert Space** of reproducing kernel k if:*

1. $\forall x \in \mathcal{X},\ k(\cdot, x) \in \mathcal{H}_k$

2. $\forall f \in \mathcal{H}_k,\ \forall x \in \mathcal{X},\ \ f(x) = \langle f, k(\cdot, x)\rangle_{\mathcal{H}_k}$ (Reproducing property)

Remark 3. *For all $x \in \mathcal{X},\ k(\cdot, x) : \mathcal{X} \to \mathbb{R}$ is the canonical feature map of x, because inside $\mathcal{H}_k$, k writes as a scalar product: $k(x, y) = \langle k(\cdot, x), k(\cdot, y)\rangle_{\mathcal{H}_k}$*

Next comes a powerful theorem that tight things up between a RKHS and a kernel:

Theorem 1 (Moore-Aronszajn)**.** *The association between a kernel and a RKHS is unique:*

- *For every positive definite kernel k there exist a uniquely associated RKHS $\mathcal{H}_k$.*
- *In the other way, for every RKHS $\mathcal{H}$, there exist a unique kernel k with the reproducing property. And this kernel is positive definite.*

The application of this theorem in a machine learning context is extremely powerful, it applies in optimization problem that arises in machine learning:

Theorem 2 (Representer Theorem)**.** *We consider the following machine learning optimization problem:*

- *Let $\mathcal{X}$ a nonempty set and k a positive definite kernel on $\mathcal{X}$. We denote by $\mathcal{H}_k$ the (unique) associated Reproducing Kernel Hilbert Space.*
- *We keep the same notation for the data: $\mathcal{D}_n = ((x_i, y_i))_{1 \leq i \leq n} \in (\mathcal{X} \times \mathbb{R})^n$*
- *Let $\Omega\colon [0, +\infty[\to \mathbb{R}$ a strictly monotonic increasing function.*
- *Let l be a loss function.*
- *We also consider $\lambda > 0$ a real positive constant, which plays the role of weight of the regularization.*

The Representer Theorem *states that for any solution f^* of the optimization problem that is minimizing the empirical risk associated to the loss function, regularized with the function Ω, over the space of function $\mathcal{H}_k$:*

$$\min_{f \in \mathcal{H}_k} \sum_{i=1}^{n} l(y_i, f(x_i)) + \lambda \Omega(\|f\|_{\mathcal{H}_k})$$

there exists $\alpha_i \in \mathbb{R}$ for $i = 1, \ldots, n$ such that for any $x \in \mathcal{X}$:

$$f^*(x) = \sum_{i=1}^{n} \alpha_i k(x, x_i)$$

Remark 4. *Kernel methods and RKHS were made very popular because of the **kernel trick**. This* trick *consists in replacing in all computations the use of the dot product and the feature map with the kernel.*

Therefore, kernel methods enable to compute linear algorithm on non linear data without having to compute anything in the Hilbert space itself, thanks to the kernel trick. It makes such methods very powerful and useful in modern machine learning. Another huge benefit of such methods is given by the Riesz Representation theorem.

Theorem 3 (Riesz Representation Theorem)**.** *Let $\mathcal{H}$ be a Hilbert space and f a continuous linear functional defined over $\mathcal{H}$. Then, there exist a unique element y of $\mathcal{H}$ such that for all $x \in \mathcal{H}$:*

$$f(x) = \langle x, y \rangle_{\mathcal{H}}$$

The consequences of this theorem are of great magnitude. By specifying the three terms, being the loss function, the regularization function and the positive definite kernel, you can perform a lot of known models such as logistic regression, ridge regression and even kernel-PCA and kernel-Nearest Neighbour. To a certain extend Gaussian Process and Neural Networks such as CNN, RNN, Transformers and GPTs can be linked to RKHS and more broadly kernel methods. In the following section we will develop on the Kernel Ridge Regression.

2.3 Kernel Ridge Regression

Ridge Regression is a usual linear regression performed with a regularization on the coefficient of the regression. Therefore the input space is a subset of $\mathbb{R}^d$. The **Kernel** Ridge Regression is an extension of the Ridge Regression. Indeed, the Ridge Regression is using the linear kernel seen here 1. Generalizing this with any kernel gives us the general formulation of the Kernel Ridge Regression. This generalization allows for this method to be used in a broader range of settings such as the one that interests us here, a regression over the set of distribution.

Example 4. *We present the results of the **Kernel Ridge Regression**. We keep the same notions as above and consider the quadratic loss function. The optimization problem thus writes:*

$$\min_{f \in \mathcal{H}_k} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \|f\|^2_{\mathcal{H}_k}$$

We know with the Representer Theorem that the optimization problem writes in a matrix formulation:

$$\min_{\alpha \in \mathbb{R}^n} \|Y - K\alpha\|^2 + \lambda \alpha^\intercal K \alpha$$

To solve this optimization problem, one must derive this expression with regard to α. Writing $\hat{\alpha} = (\hat{\alpha}_i, \dots, \hat{\alpha}_n) \in \mathbb{R}^n$ we have the Kernel Ridge estimator for all x in $\mathcal{X}$:

$$\hat{f} = \sum_{i=1}^{n} \hat{\alpha}_i k(\cdot, x_i)$$

This estimator is extremely powerful. Indeed, the regression can be performed on any set of data $\mathcal{X}$ and all the benefits of using the Ridge Regression applies. A few remark must be made on this model.

Remark 5. *The first remark there is to make is that this Kernel Ridge Regression is an extension of the regression with ridge regularization model. If we use the linear kernel we immediately have $K = X^\intercal X$. We found the exact same formulation as the regression with ridge regularization.*

Remark 6. *When λ tends to 0 (with the linear kernel) we found the usual linear regression. When λ tends to ∞, we found $\hat{\alpha} = 0_{\mathbb{R}^d}$. The regularization allows for a solution that is unique and that always exists.*

The Kernel Ridge Regression is both a versatile and easy model to use in a regression setting. Nevertheless, it is sometime necessary to turn towards more complex and high performance models. Specifically, using Bayesian model allows one to use prior knowledge to enhance the model. Gaussian Processes [3, 4] are such models. They also rely on a positive definite kernel as covariance matrix. Therefore the benefit of kernel methods, that is, be usable in any context if one can find the right kernel, still applies for Gaussian Processes.

3 Optimal Transport

3.1 Introduction to Optimal Transport : The Monge formulation

The Monge Transport Problem, named after 18th-century mathematician Gaspard Monge, represents a seminal development in Optimal Transport. Originally conceptualized to address practical challenges in mass transportation, Monge's formulation has evolved into a comprehensive theoretical framework. It involves finding the most cost-effective strategy to transport mass from one distribution to another.

Consider a scenario with n distinct sources and n distinct destinations, each source-destination pair incurs a specific transportation cost, represented in a cost matrix $(C_{i,j})_{i\in[\![n]\!],j\in[\![n]\!]}$, where i and j index the sources and destinations, respectively. If we denote σ a bijection from the set of sources to the set of destinations, and $\text{Perm}(n)$ the set of all possible permutations of n elements, the Monge Transport Problem seeks to minimize the total transportation cost, formalized as:

$$\min_{\sigma\in\text{Perm}(n)} \sum_{i=1}^{n} C_{i,\sigma(i)} \tag{2}$$

Optimal Transport (OT) can be conceptualized across various scenarios, including continuous-continuous, discrete-continuous, and discrete-discrete distributions. Each scenario presents unique challenges and methodologies within the framework of OT. However, for the purpose of our study, we will concentrate on the discrete-discrete case. This focus aligns with the scenarios where both the source and destination distributions are discrete, as is often encountered in practical applications relevant to our field of study.

Monge problem between discrete measures : Considering two discretes measures

$$\alpha = \sum_{i=1}^{n} a_i \delta_{x_i} \quad \text{and} \quad \beta = \sum_{j=1}^{m} b_j \delta_{y_j},$$

the Monge problem seeks a map $T : \{x_1, \dots, x_n\} \to \{y_1, \dots, y_m\}$ that associates to each point x_i a single point y_j, and which must push the mass of α toward the mass of β. The map must satisfy the following condition for every $j \in m$:

$$\forall j \in m, \quad b_j = \sum_{i:T(x_i)=y_j} a_i \tag{3}$$

which can be written in compact form as $T\#\alpha = \beta$. This map must minimize a transportation cost parameterized by a function $c(x, y)$ defined for points $(x, y) \in X \times Y$, expressed as:

$$\min_T \sum_i c(x_i, T(x_i)) \quad : \quad T\#\alpha = \beta. \tag{4}$$

Remark 7. *This problem, however, does not always have a unique solution. Particularly, when $n = m$ and weights are uniform ($a_i = b_j = 1/n$), it simplifies to an optimal matching problem 2, representable with a cost matrix C_{ij}. On the other hand, if $n \neq m$ or weights vary, a Monge map might not exist, necessitating more complex or generalized approaches.*

Such variations underscore the challenges in addressing the Monge problem for diverse distribution scenarios. It is in this context that the Kantorovich formulation becomes particularly relevant.

3.2 Kantorovich's Relaxation

Leonid Kantorovich, a mathematician in the 20-th century has developed a new formulation for the problem of optimal transport. His formulation was conceived to address critical constraints in the Monge problem, notably the challenge of finding feasible solutions that adhere to mass conservation. Kantorovich's method effectively manages the combinatorial complexity of the assignment problem and overcomes the nonconvex nature of feasible sets in the Monge formulation. It accomplishes this by introduces 'mass splitting', allowing the distribution of source mass to multiple destinations, enhancing both flexibility and computational feasibility.

This flexibility is encoded using a coupling matrix $\mathbf{P} \in \mathbb{R}_+^{n \times m}$, where $\mathbf{P}_{i,j}$ describes the amount of mass flowing from bin i toward bin j, or from mass at x_i toward y_j in discrete measures. Admissible couplings are characterized as:

$$\mathbf{U}(\mathbf{a}, \mathbf{b}) \coloneqq \left\{ \mathbf{P} \in \mathbb{R}_+^{n \times m} : \mathbf{P}\mathbb{1}_m = \mathbf{a} \text{ and } P^T\mathbb{1}_n = \mathbf{b} \right\}$$

with matrix-vector notation:

$$\mathbf{a} = \left(\sum_i \mathbf{P}_{i,j} \right)_i \in \mathbb{R}^n \text{ and } \mathbf{b} = \left(\sum_i \mathbf{P}_{i,j} \right)_j \in \mathbb{R}^m.$$

The set $U(a, b)$ is a convex polytope, defined by $n + m$ equality constraints.

In contrast to the Monge formulation's asymmetric nature, the Kantorovich formulation exhibits inherent symmetry. This is demonstrated by the relationship between the coupling matrix P and its transpose P^T: P is a member of $U(a, b)$ if and only if P^T is included in $U(b, a)$. This lead to the Kantorovich optimal transport problem defined as :

$$L_{\mathbf{C}(\mathbf{a},\mathbf{b})} \coloneqq \min_{\mathbf{P} \in \mathbf{U}(\mathbf{a},\mathbf{b})} \langle \mathbf{C}, \mathbf{P} \rangle \coloneqq \sum_{i,j} \mathbf{C}_{i,j} \mathbf{P}_{i,j}. \tag{5}$$

Remark 8. *In the context of Optimal Transport, the linear program in Equation 5 is solved to derive a transportation plan $\mathbf{P}^*$, quantifying the goods $\mathbf{P}_{i,j}$ to be transported from the sources i to the destinations j. For permutation matrices as couplings, $\mathbf{P}_\sigma$ represents the permutation matrix for $\sigma \in Perm(n)$, with $(\mathbf{P}_\sigma)_{i,j} = \frac{1}{n}$ if $j = \sigma(i)$, else 0. This formulation allows recasting the assignment problem as a Kantorovich problem, where couplings are restricted to permutation matrices. The set of permutation matrices is a subset of the Birkhoff polytope $U(\frac{1}{n}, \frac{1}{n})$, illustrating that the Kantorovich problem's minimum is achieved with permutation matrices when dealing with uniform measures.*

Remark 9 (Kantorovich problem between discrete measures)**.** *For discrete measures α, β of the form 4, we store in the matrix C all pairwise costs between points in the supports of α, β, namely $C_{i,j} \coloneqq c(x_i, y_j)$, to define*

$$L_c(\alpha, \beta) \coloneqq L_C(\mathbf{a}, \mathbf{b}). \tag{6}$$

Therefore, the Kantorovich formulation of optimal transport between discrete measures is the same as the problem between their associated probability weight vectors $\mathbf{a}, \mathbf{b}$ except that the cost matrix C depends on the support of α and β.

Building upon our focus on the discrete measure in the Kantorovich formulation, our project further evolves to embrace Regularized Optimal Transport. This advanced variation of Optimal Transport is particularly suited to the practical constraints and computational demands we face in analyzing high-dimensional data, such as point clouds in $\mathbb{R}^d$. Regularized Optimal Transport offers improved computational efficiency and robustness, crucial for handling the complexity of probability distributions in our study. The integration of regularization techniques into Optimal Transport thus aligns seamlessly with our objective of utilizing the kernel for efficient and accurate comparison of these distributions.

3.3 Regularized Optimal Transport

To resolve this optimal transport problem between two discrete measures, several algorithms are available, including the Hungarian algorithm and the simplex algorithm. However, these algorithms can become very time-consuming, with a worst-case complexity of $O(n^3 log(n))$ when $n = m$. To overcome those computational issue, the regularized OT introduces an entropic regularization penalty (see sec. 3.3.1). To solve this minimization problem, we're going to use the Sinkhorn algorithm (see sec. 3.3.2).

3.3.1 Entropic Regularization and formulation

Let $\mathbf{P}$ be a coupling matrix, its discrete entropy is defined as follows :

$$H(\mathbf{P}) \stackrel{\text{def}}{=} -\sum_{i,j} \mathbf{P}_{i,j}(\log(\mathbf{P}_{i,j}) - 1)$$

The term $-H(\mathbf{P})$ is thus utilized here as a regularization term to obtain approximate solutions for the optimal transport problem. We then obtain the following minimization problem:

$$L_C^{\varepsilon}(\mathbf{a}, \mathbf{b}) \stackrel{\text{def}}{=} \min_{\mathbf{P} \in U(a,b)} \langle \mathbf{P}, \mathbf{C} \rangle - \varepsilon H(\mathbf{P})$$

Since this minimizing transport is an ε-strongly convex function, it has a unique optimal solution. We can write it solution $\mathbf{P}_{\varepsilon}$. One can verify the following convergence:

$$L_C^{\varepsilon}(\mathbf{a}, \mathbf{b}) \xrightarrow[\varepsilon \to 0]{} L_C(\mathbf{a}, \mathbf{b})$$

As ε approaches 0, we approach Kantorovich's formulation. The goal is to determine an optimal value for ε. When ε is small, the solution tends to converge to the maximum entropy optimal transport coupling. However, with increasing ε, the optimal transport coupling becomes denser, signifying a greater density of non-zero entries. This densification leads to accelerated computational algorithms and improved statistical convergence[5]. We can observe this densification as ε increases in the following chart:

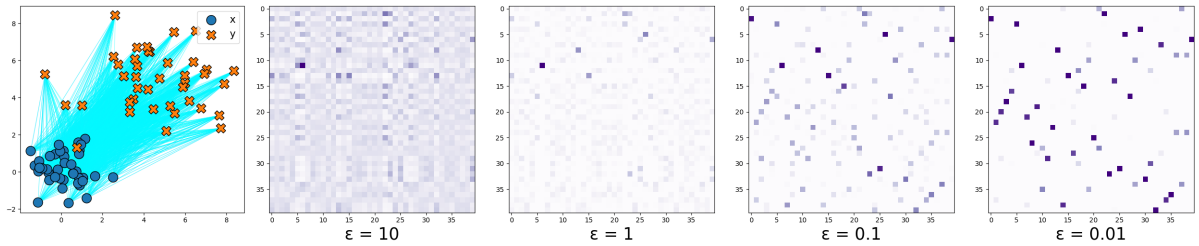


Figure 2: The influence of ε on the transport plan is illustrated. On the left are the two distributions targeted for transport optimization, while the right displays four distinct plans corresponding to different ε values.

The Kullback–Leibler divergence permit to defines the unique solution $\mathbf{P}_\varepsilon$ as a projection onto $\mathbf{U}(\mathbf{a}, \mathbf{b})$ using the Gibbs kernel associated with the cost matrix $\mathbf{C}$. The Kullback-Leibler divergence between the coupling matrix $\mathbf{P}$ and a kernel $\mathbf{K}$ is defined as:

$$\mathrm{KL}(\mathbf{P}|\mathbf{K}) \stackrel{\text{def}}{=} \sum_{i,j} \mathbf{P}_{i,j} \log\left(\frac{\mathbf{P}_{i,j}}{\mathbf{K}_{i,j}}\right) - \mathbf{P}_{i,j} + \mathbf{K}_{i,j}$$

where

$$\mathbf{K}_{i,j} \stackrel{\text{def}}{=} \exp -\frac{\mathbf{C}_{i,j}}{\varepsilon}\,.$$

We then have the unique solution as:

$$\mathbf{P}_\varepsilon = \mathrm{Proj}^{\mathrm{KL}}_{\mathbf{U}(\mathbf{a},\mathbf{b})}(\mathbf{K}) \stackrel{\text{def}}{=} \underset{\mathbf{P}\in\mathbf{U}(\mathbf{a},\mathbf{b})}{\arg\min}\, \mathrm{KL}(\mathbf{P}|\mathbf{K})$$

In the next section, the Sinkhorn algorithm used to solve this optimization problem is presented.

3.3.2 Sinkhorn's Algorithm

Another way of finding the solution is to write the Lagrangian Λ^ε_C associated to the problem $L^\varepsilon_C(\mathbf{a}, \mathbf{b})$. The two constraints are the mass conservation of $\mathbf{a}$ and $\mathbf{b}$, inherent to $\mathbf{U}(\mathbf{a}, \mathbf{b})$. We then obtain the following Lagrangian formulation:

$$\Lambda^\varepsilon_{\mathbf{C}}(\mathbf{P}, \mathbf{f}, \mathbf{g}) = \langle \mathbf{P}, \mathbf{C}\rangle - \varepsilon H(\mathbf{P}) - \langle \mathbf{f}, \mathbf{P}\mathbb{1}_m - \mathbf{a}\rangle - \langle \mathbf{g}, \mathbf{P}^\intercal\mathbb{1}_n - \mathbf{b}\rangle$$

And the unique solution has the form:

$$\mathbf{P}_{i,j} = e^{\frac{\mathbf{f}_i}{\varepsilon}} e^{-\frac{\mathbf{C}_{i,j}}{\varepsilon}} e^{\frac{\mathbf{g}_j}{\varepsilon}}, \qquad \forall (i,j) \in [\![n]\!] \times [\![m]\!]$$

By using nonnegative vectors u and v and the Gibbs kernel, we can write this solution as:

$$\mathbf{P}_{i,j} = \mathbf{u}_i \mathbf{K}_{i,j} \mathbf{v}_i, \qquad \forall (i,j) \in [\![n]\!] \times [\![m]\!]$$

And we can write this as a matrix scaling:

$$\mathbf{P}_{i,j} = \mathrm{diag}(\mathbf{u})\mathbf{K}\mathrm{diag}(\mathbf{v})$$

Under the mass conservation constraints, with $\circ$ the Hadamard-product:

$$\mathbf{u} \circ (\mathbf{K}\mathbf{v}) = \mathbf{a}\,,$$

$$\mathbf{v} \circ (\mathbf{K}^\intercal \mathbf{u}) = \mathbf{b}\,.$$

Finally, to find the solution, we are going to use the Sinkhorn algorithm, described as follows:

$\mathbf{u} \leftarrow \mathbf{1}_n$
$\mathbf{v} \leftarrow \mathbf{1}_m$
$\mathbf{P} \leftarrow \mathrm{diag}(\mathbf{u})\mathbf{K}\mathrm{diag}(\mathbf{v})$
while $\mathbf{P}$ change **do**
  $\mathbf{u} \leftarrow \frac{\mathbf{a}}{\mathbf{K}\mathbf{v}}$
  $\mathbf{v} \leftarrow \frac{\mathbf{b}}{\mathbf{K}^T\mathbf{u}}$
  $\mathbf{P} \leftarrow \mathrm{diag}(\mathbf{u})\mathbf{K}\mathrm{diag}(\mathbf{v})$
end while

The convergence of the Sinkhorn algorithm is ensured, as demonstrated by Franklin and Lorenz[6]. The complexity of each iteration of this algorithm is $O(n^2)$.

To conclude, by incorporating the entropy of the transport plan as a regularization factor within the Kantorovich minimization problem, it emerges that solving the discrete optimal transport problem involves utilizing iterative matrix multiplications and elementwise divisions. This algorithm is therefore very efficient because it is easily parallelizable and can be run on a GPU.

4 Suggested Kernel

4.1 A kernel based on optimal transport

The authors in the paper [1] use the Sinkhorn potentials to measure a distance between distributions and then create a kernel using this distance.

First of all, lets consider a measure $\mathcal{U}$ on Ω and two distributions P and Q. We then consider two transport problems, using the regularized optimal transport framework developed in 3.3. The regularized optimal transport from P to the reference measure $\mathcal{U}$ and from Q to $\mathcal{U}$. Lets denote by $\pi_{\mathcal{U}}^P$ and $\pi_{\mathcal{U}}^Q$ the optimal entropic plans of both problems. The Sinkhorn algorithm, then, gives us the optimal entropic potentials of both problems: $\left(f_{\mathcal{U}}^P, g_{\mathcal{U}}^P\right)$ for the first problem and $\left(f_{\mathcal{U}}^Q, g_{\mathcal{U}}^Q\right)$ for the second one. This can be summarized in the following way:

$$P \xrightarrow[Transport]{Reg.Opt.} \mathcal{U} \Longrightarrow \pi_{\mathcal{U}}^P \Longrightarrow \left(f_{\mathcal{U}}^P, g_{\mathcal{U}}^P\right) \tag{7}$$

$$Q \xrightarrow[Transport]{Reg.Opt.} \mathcal{U} \Longrightarrow \pi_{\mathcal{U}}^Q \Longrightarrow \left(f_{\mathcal{U}}^Q, g_{\mathcal{U}}^Q\right) \tag{8}$$

Those potentials are unique up to an additive constant. Therefore we can decide to use the centered potential. We redefine the potential as $g_{\mathcal{U}}^P = g_{\mathcal{U}}^P - \mathbb{E}\left(g_{\mathcal{U}}^P(U)\right)$ and also $g_{\mathcal{U}}^Q = g_{\mathcal{U}}^Q -$

$\mathbb{E}\left(g_{\mathcal{U}}^{Q}(U)\right)$. This leads to the following equality:

$$Var_{U\sim\mathcal{U}}\left(g_{\mathcal{U}}^{P}(U)-g_{\mathcal{U}}^{Q}(U)\right)=\|g_{\mathcal{U}}^{P}-g_{\mathcal{U}}^{Q}\|_{L^2(\mathcal{U})}^2 \tag{9}$$

This equality is used to introduce the following kernel:

Theorem 4. *Let $F\colon [0,+\infty[\to\mathbb{R}$ be a continuous function and $\mathcal{U}\in\mathcal{P}_{SG}(\Omega)$. If:*

1. *$F\circ\sqrt{\cdot}$ is completely monotonous on $[0,+\infty[$*
2. *There exist a nonnegative Borel measure ν on $[0,+\infty[$ such that for $t>0$, $F(t)=\int_0^{+\infty}e^{-ut^2}\,d\nu(u)$*

Then

$$\begin{aligned}K\colon \mathcal{P}_{SG}(\Omega)\times\mathcal{P}_{SG}(\Omega)&\longrightarrow\mathbb{R}\\(P,Q)&\longmapsto F\left(\|g_{\mathcal{U}}^{P}-g_{\mathcal{U}}^{Q}\|_{L^2(\mathcal{U})}\right)\end{aligned}$$

is a positive definite kernel on $\mathcal{P}_{SG}(\Omega)$.

The function F can be the usual kernel functions such as the Radial Basis Function kernel, the power exponential kernel or the Matérn kernel.

4.2 Theoretical proprieties

The paper then presents a few propositions. The first one assures that the distance on Sinkhorn potential can be dominated by a quantity that is controlled by the distance between the distributions P and Q.

Propositon 1. *Let $s\in\mathbb{N}$. Suppose Ω is compact and let $P,Q\in\mathcal{P}(\Omega)$. There exist a constant c, that depends on the dimension of Ω such that*

$$\|g_{\mathcal{U}}^{P}-g_{\mathcal{U}}^{Q}\|_{L^2(\mathcal{U})}\leq c\times\|P-Q\|_s$$

The second proposition ties the potentials to the probability distributions. It guarantees that the entropic potentials $g_{\mathcal{U}}^{P}$ and $g_{\mathcal{U}}^{Q}$ can be used to characterize the distributions P and Q.

We know that in practice we do not have access to real distributions such as P and Q. We only have a sample of those distributions, therefore:

$$P_n=\frac{1}{n}\sum_{i=1}^{n}\delta_{X_i}\quad\text{and}\quad Q_m=\frac{1}{m}\sum_{i=1}^{m}\delta_{Y_i} \tag{10}$$

It is shown that the kernel verifies the following proposition that states the consistency of the empirical kernel.

Propositon 2. *If F is continuous, then*

$$F\left(\|g_{\mathcal{U}}^{P_n}-g_{\mathcal{U}}^{Q_m}\|_{L^2(\mathcal{U})}\right)\xrightarrow[n,m\to+\infty]{a.s}F\left(\|g_{\mathcal{U}}^{P}-g_{\mathcal{U}}^{Q}\|_{L^2(\mathcal{U})}\right) \tag{11}$$

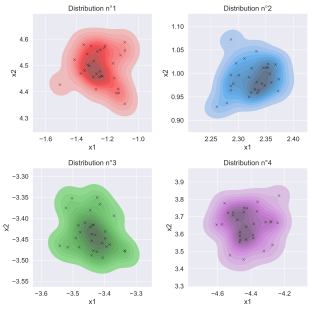
This last theorem is very important as it enable us to use this kernel in practice.

4.3 Sinkhorn kernel in practice

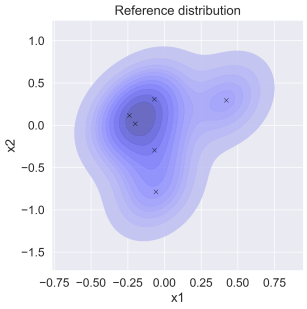
4.3.1 Reproducing the first experiment

To understand a little bit our such kernel works we will use it on a toy example from the paper. First of all lets understand our simulated data. We are going to consider a 100 two-dimensional normal distribution. Those distribution are going to have a mean vector (m_1, m_2) uniformly drawn from $[-0.3, 0.3]^2$ and the variance uniformly drawn from $[0.01^2, 0.02^2]$. We consider only isotropic distribution, therefore the covariance matrix of our distribution are going to be of the form $\sigma^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. For each of those distribution we compute the value $Y = \frac{(m_1+0.5-(m_2+0.5)^2)}{1+\sigma}$. Those values associated to each and every distributions are living in the target space $\mathcal{Y}$ in our regression framework.

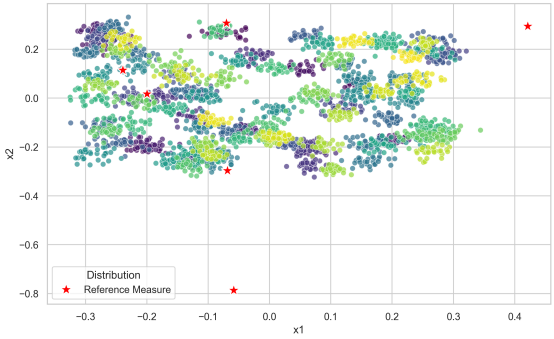
The data is going to be a random sample of 30 points from all of those 100 distributions. Figure 3a shows four of those distributions. All those sample are split into a training set 50% and a test set 50%. We need to specify our reference measure $\mathcal{U}$. For this toy experiment we are going to consider a centered two-dimensional isotropic Gaussian distribution, of variance 0.1. Once again, we are going to work with a sample of this distribution, a sample of size 6. We provide the plot of the reference measure sample in figure 3b. Note that this reference distribution can greatly influence our regression. We will see later how it impacts it.



(a) Samples from 4 different distributions.



(b) Sample points of the reference distribution.



(c) All the training samples and the reference measure sample.

Figure 3: Exploring all the distributions used in the toy experiment.

Next, we are going to perform a discrete-discrete optimal transport between our 100 samples and our reference measure. This is done using the Sinkhorn algorithm developed in section 3.3.2. We obtain the Sinkhorn potentials of all those problems, that is, we recover for each sample a

vector of size 6 (the number of sample in the reference measure). From here, those vectors define our observations and we need to compute a "classical" kernel ridge regression on those data points, using the Radial Basis Function kernel. We have two hyper-parameters to optimize in this kernel ridge regression, that are the parameter of the kernel and the strength of the regularization.

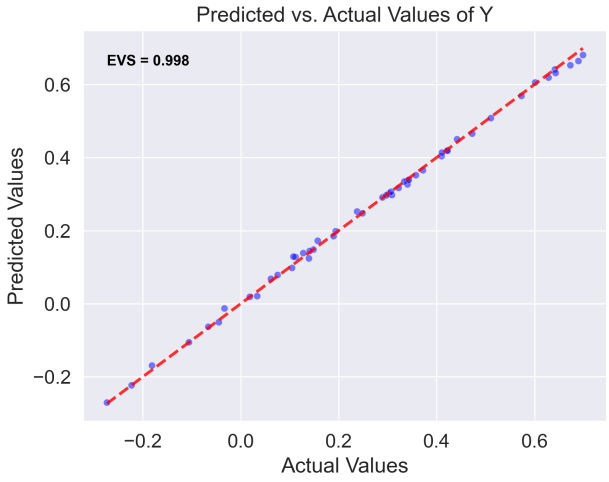


Figure 4: Comparing real objective values against predicted values for the test set.

The model is performing very well on the test set. We obtain an explained variance score of 0.998, figure 4, which is slightly greater than what is obtain with Gaussian Processes in the paper, 0.997. This can be explained by the randomness of our samples, the cross-validation performed to optimize the hyper-parameters of the kernel ridge regression and the fact that we did not optimize the choice of the samples of the reference distribution as in the paper (see algorithm 1 in the paper). Nevertheless, our results are really strong. Taking a step back on this results we have multiple remarks to make. First of all, the reference measure chosen is slightly different from the one in the paper. Ours is certainly better as it is in the center of all the distributions considered, as we can see in figure 3c. The second point we want to make is that the problem is not that hard, the variances of the distribution are very small and thus it is not very hard to differentiate them. Multiple modifications that could be done to make the problem harder are changing the reference measure, changing the parameters of the training distributions and adding dimensions to the problem.

One last thing we want to talk about is the value of the optimal transport optimization parameter ϵ. Indeed, as we stated before, the bigger ϵ, the easier the problem but also the worst the solution is. In opposite, the smaller ϵ is, the harder the problem and the better the solution is. In the figure 5, we show the performance of the model against the value of ϵ for the first experiment case. We see that the best performing models are with the smallest ϵ. We do not encounter specific computing difficulties because we believe our problem is not difficult enough. Therefore, for all experiments we are going to use $\epsilon = 0.01$.

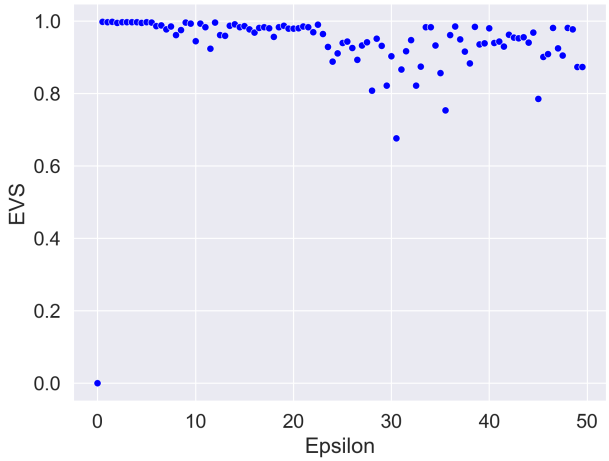


Figure 5: Explained Variance Score with regard to ϵ

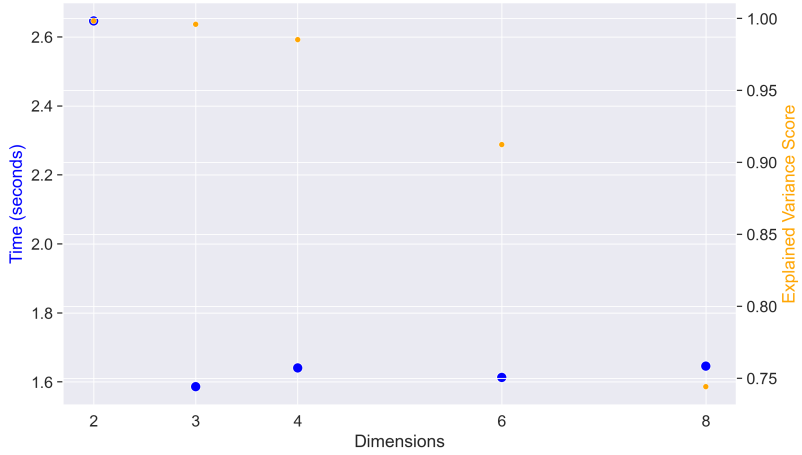


Figure 6: Scatter Plot of Time and Explained Variance Score vs. Dimensions

4.3.2 Changing the dimension of the problem

Because our applied case is for 3D models of hardware pieces, we thought trying with at least a 3D model would be interesting. We thus tested for 2, 3, 4 and 8 dimensions problems. All the rest of the experiment's parameter are being kept the same. To take into account the bigger dimensions in the problem we need to change our target variable, we did this as follow:

$$\sum_{i=1}^{\text{dimension}} \frac{(-1)^i (m_i + 0.5)^i}{1 + \sigma} \tag{12}$$

As we can see in figure 6, the performance of the kernel ridge regression diminishes greatly with the dimensions. Fortunately, the problem in dimension 3 still achieves a good explained variance score. We provide the performances on the test split for each problem in the appendix. The computing time increases a lot from 2 to 3 dimensions but not so much from there on. Again, our problem is very simple and the computing time might not be representative of the difficulty of the problem. More investigation is needed on this side but we can be quite hopeful for dimension 3.

4.3.3 Changing the reference measure

First, we are going to use a reference measure that is far from our data. Lets consider a 2-dimensional uniform distribution on $[-20, -10] \times [-20, -10]$. In the figure 7 we can see how different is this reference measure against our data. We obtain an expected variance score of

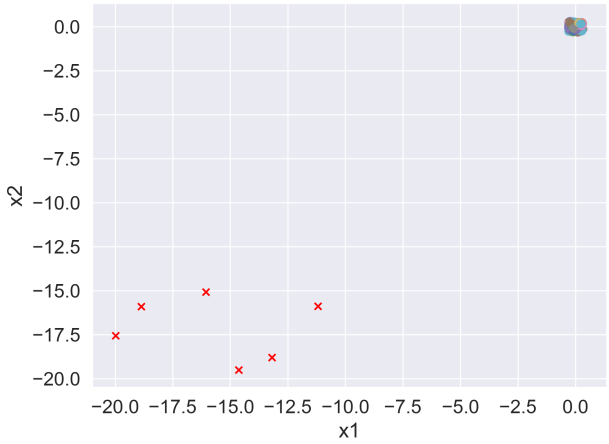
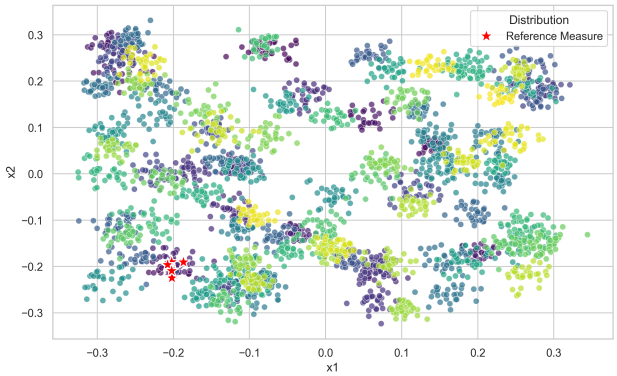


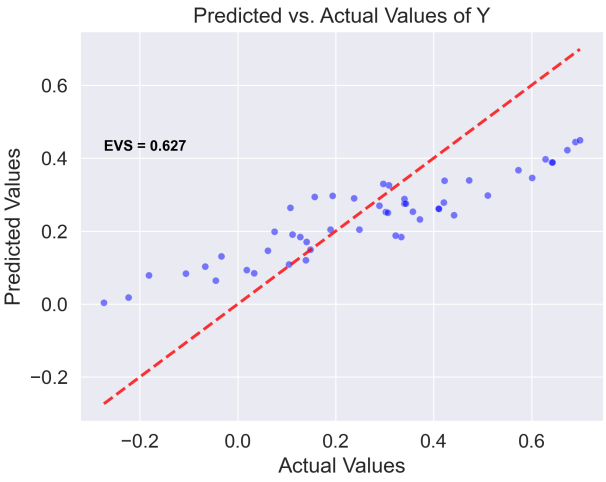
Figure 7: Uniform Reference Measure.

0.997. Which is really good and close to the score obtained with the previous reference measure. This is surprising to us as we expected the performance to deteriorate. What we can say is that this measure is far from the data but "equally" far, and thus the optimal transport problem is "equally" bad for all the data. Lets try to use a reference measure that is going to favor some distributions.

The next reference measure we are going to use is a normal distribution with mean vector $(-0.2, -0.2)$ and a very small variance of 0.0001. We show figure 8a the reference measure in the middle of our data. This time the performance decreases to an expected variance score of



(a) "Specific" Reference measure.



(b) Test set results with "specific" reference measure.

Figure 8: Tackling the problem with a "specific" reference measure.

0.627. The optimal transport is here not equal for all the distributions considered. We show

figure 8b the performance of the kernel ridge regression on the test set. This achieve to show that the reference measure is an important ”parameter” to optimize in this problem. Our believe is that it should be ”equally” distant from the training data, as in it should not favor some of the distributions, like seen in our last example. The paper suggests a way to optimize this reference measure, something we did not achieve to replicate here.

5 Conclusion

Kernel methods are incredibly powerful methods that present lots of advantages against more popular methods such as Neural Networks. They rely on a strong mathematical background and are versatile enough to be used in most cases. Kernel methods are known for some time now. On the other hand, the Optimal Transport theory is quite a recent field and not all possibilities of this theory have been explored. Recently the use of optimal transport between two distributions have been used to compare two distribution together.

The paper "Gaussian Processes on Distributions based on Regularized Optimal Transport" proposes a kernel based on the Regularized Optimal Transport between distributions. The optimal transport theory provides a nice framework to compare distributions and kernel methods allows to capitalise on this knowledge to perform machine learning tasks. We were able to reproduce a small toy experiment achieving similar performances as shown in the paper, even if we did not performed the optimization on the choice of the reference measure. We explored a little beyond this simple experiment and concluded that it may be too simple to showcase the difficulties one may encounter in practice using this method.

Nevertheless, using optimal transport to compare distributions in a kernel method is promising, although it needs more experiment. We are eager to perform more experiment and maybe find an application on real world data to see if the computation scales. It would also be really interesting to succeed into performing Gaussian Processes instead of Kernel Ridge Regression as it provides a nice framework, especially to compute confidence intervals.

References

[1] François Bachoc et al. *Gaussian Processes on Distributions based on Regularized Optimal Transport*. 2022. arXiv: 2210.06574 [stat.ML].

[2] Bernhard Schölkopf and Alexander J. Smola. *Learning with kernels : support vector machines, regularization, optimization, and beyond*. Adaptive computation and machine learning. MIT Press, 2002. URL: http://www.worldcat.org/oclc/48970254.

[3] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, 2006, pp. I–XVIII, 1–248. ISBN: 026218253X.

[4] Motonobu Kanagawa et al. *Gaussian Processes and Kernel Methods: A Review on Connections and Equivalences*. 2018. arXiv: 1807.02582 [stat.ML].

[5] Gabriel Peyré and Marco Cuturi. *Computational Optimal Transport*. 2020. arXiv: 1803.00567 [stat.ML].

[6] Joel Franklin and Jens Lorenz. "On the scaling of multidimensional matrices". In: *Linear Algebra and Its Applications, pages 717–735* (1989).

[7] Thomas Hofmann, Bernhard Schölkopf, and Alexander J. Smola. "Kernel methods in machine learning". In: *The Annals of Statistics* 36.3 (2008), pp. 1171–1220. DOI: 10.1214/009053607000000677. URL: https://doi.org/10.1214/009053607000000677.

A On the kernel methods

We present in this appendix a few details on kernel methods that we could not expose in our report.

We can represent the positive definite kernel as a matrix or a function interchangeably.

Definition 5. *For a given $k\colon \mathcal{X}\times\mathcal{X}\to\mathbb{R}$ and given inputs $(x_1,\dots x_n)\subset\mathcal{X}$, the $n\times n$ matrix:*

$$K \coloneqq \left(k\left(x_i,x_j\right)\right)_{1\leq i,j\leq n}$$

*is called the **Gram matrix** (or kernel matrix) of k with respect to $x_1,\dots x_n$.*

Propositon 3. *A symmetric function $k\colon \mathcal{X}\times\mathcal{X}\to\mathbb{R}$ is a positive definite kernel* if and only if *the associated Gram matrix K is positive definite for $n\in N$ and $(x_1,\dots x_n)\subset\mathcal{X}$.*

An important propriety of kernel is the following.

Propositon 4. *Let $\mathcal{X}$ be a nonempty set. And let $k_1,k_2,\dots$ be arbitrary positive definite kernels on $\mathcal{X}\times\mathcal{X}$.*

(i) The set of positive definite kernels is a closed convex cone:

(a) $\forall\alpha_1,\alpha_2\geq 0,\ \alpha_1k_1+\alpha_2k_2$ is a positive definite kernel.

(b) If $k\left(x,x'\right)\coloneqq\lim_{n\to\infty}k_n\left(x,x'\right)$ exists for all $(x,x')\in\mathcal{X}^2$, then k is a positive definite kernel.

(ii) The pointwise (Hadamard) product k_1k_2 is definite positive

(iii) For $i=1,2$, let k_i be a positive definite kernel on $\mathcal{X}_i\times\mathcal{X}_i$ ($\mathcal{X}_i$ nonempty), the tensor product $k_1\otimes k_2$ and direct sum $k_1\oplus k_2$ are positive definite kernel on the space $(\mathcal{X}_1\times\mathcal{X}_2)\times(\mathcal{X}_1\times\mathcal{X}_2)$

In Hofmann et al.[7] it is proven that those operations are the only one that preserves the positive definite property.

The next remark shows how to construct the RKHS from a given kernel.

Remark 10. *Let $\mathcal{X}$ a nonempty set and k a positive definite kernel on $\mathcal{X}$. We can construct the RKHS associated to k in the following way.*

We define $\mathcal{H}_0$ as the linear span of k. Meaning every function in $\mathcal{H}_0$ can be expressed as a linear combination of k at points $(x_1,\dots x_n)$:

$$\mathcal{H}_0 \coloneqq \operatorname{span}\{k\left(\cdot,x\right), x\in\mathcal{X}\}$$

Let $f=\sum_{i=1}^n a_ik\left(\cdot,x_i\right)\in\mathcal{H}_0$ and $g=\sum_{i=1}^m b_ik\left(\cdot,x_i\right)\in\mathcal{H}_0$ with $n,m\in\mathbb{N}$, $\left(a_1,\dots a_n\right),\left(b_1,\dots b_m\right)\in\mathbb{R}$ and $\left(x_1,\dots x_n\right),\left(y_1,\dots y_m\right)\in\mathbb{X}$. We equip $\mathcal{H}_0$ with the following inner product:

$$\langle f,g\rangle_{\mathcal{H}_0}\coloneqq\sum_{i=1}^n\sum_{j=1}^m a_ib_jk\left(x_i,y_j\right)$$

Therefore $(\mathcal{H}_0,\langle\cdot,\cdot\rangle)$ is a pre-Hilbert space. Considering the associated norm $\|f\|_{\mathcal{H}_0}^2=\langle f,f\rangle_{\mathcal{H}_0}$, one can construct the RKHS associated to k as the closure of $\mathcal{H}_0$ with respect to the norm above:

$$\mathcal{H}_k\coloneqq\overline{\mathcal{H}_0}=\{f=\sum_{i=1}^\infty c_ik\left(\cdot,x_i\right),\ \text{with}\ (c_1,c_2,\dots)\subset\mathbb{N},(x_1,x_2,\dots)\subset\mathcal{X}\ \text{such that}\ \|f\|_{\mathcal{H}_k}^2<\infty\}$$

Example 5. *We present in detail the **Kernel Ridge Regression**:*

- *The data is :* $\mathcal{D}_n = ((x_i, y_i))_{1 \leq i \leq n} \in (\mathcal{X} \times \mathbb{R})^n$
- *The kernel here is* k*. We denote by* $\mathcal{H}_k$ *the feature space associated to* k*.*
- *The regularization function is*

$$\begin{aligned} \Omega \colon [0, +\infty[&\longrightarrow \mathbb{R} \\ \|f\|_{\mathcal{H}_k} &\longmapsto \|f\|_{\mathcal{H}_k}^2 \end{aligned}$$

  which is indeed a strictly monotonic increasing function.

- *Let* l *be a loss function. In our regression settings:*

$$\begin{aligned} l \colon (\mathcal{X} \times \mathbb{R}) \times \mathcal{H}_k &\longrightarrow \mathbb{R} \\ ((x_i, y_i), f) &\longmapsto (y_i - f(x_i))^2 \end{aligned}$$

- *Let* $\lambda > 0$*, the weight of the regularization. This parameter should be tuned by cross-validation.*

The optimization problem thus writes:

$$\min_{f \in \mathcal{H}_k} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \|f\|_{\mathcal{H}_k}^2$$

We know with the Representer Theorem seen in 2 that we dispose of $\alpha_i \in \mathbb{R}, i = 1, \ldots, n$ *such that* $f = \sum_{i=1}^{n} \alpha_i k(\cdot, x_i)$ *therefore the norm in the feature space writes:*

$$\begin{aligned} \|f\|_{\mathcal{H}_k}^2 &= \langle f, f \rangle_{\mathcal{H}_k} \\ &= \langle \sum_{i=1}^{n} \alpha_i k(\cdot, x_i), \sum_{i=1}^{n} \alpha_i k(\cdot, x_i) \rangle_{\mathcal{H}_k} && \text{(Representer theorem)} \\ &= \sum_{i=1}^{n} \sum_{i=j}^{n} \alpha_i \alpha_j \langle k(\cdot, x_i), k(\cdot, x_j) \rangle_{\mathcal{H}_k} && \text{(bilinearity of dot product)} \\ &= \sum_{i=1}^{n} \sum_{i=j}^{n} \alpha_i \alpha_j k(x_i, x_j) && \text{(from remark 3)} \\ &= \alpha^\intercal K \alpha && \text{(as a matrix formulation)} \end{aligned}$$

Writing the total loss in a matrix form we have:

$$\begin{aligned} \sum_{i=1}^{n} (y_i - f(x_i))^2 = \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{n} \alpha_j k(x_j, x_i) \right)^2 && \text{(Representer theorem)} \\ = \|Y - K\alpha\|^2 && \text{(as a matrix formulation)} \end{aligned}$$

Therefore, the optimization problem writes in a matrix formulation:

$$\min_{\alpha \in \mathbb{R}^n} \|Y - K\alpha\|^2 + \lambda \alpha^\intercal K \alpha$$

To solve this optimization problem, one must derive this expression with regard to α:

$$\begin{aligned}\frac{\partial}{\partial\alpha}\left[\|Y-K\alpha\|^2+\lambda\alpha^\intercal K\alpha\right] &= \frac{\partial}{\partial\alpha}\left[(Y-K\alpha)^\intercal(Y-K\alpha)+\lambda\alpha^\intercal K\alpha\right]\\ &= \frac{\partial}{\partial\alpha}\left[Y^\intercal Y-Y^\intercal K\alpha-(K\alpha)^\intercal Y+(K\alpha)^\intercal(K\alpha)+\lambda\alpha^\intercal K\alpha\right]\\ &= 0-K^\intercal Y-K^\intercal Y+2K^\intercal K\alpha+\lambda\left(K+K^\intercal\right)\alpha\end{aligned}$$

In order to find the minimum one must set the derivative equal to zero:

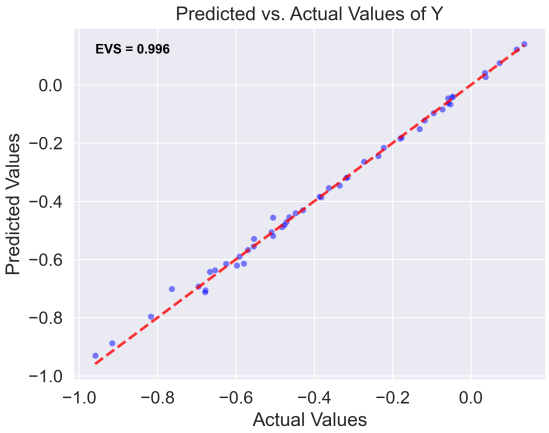
$$\begin{aligned}\frac{\partial}{\partial\alpha}\left[\|Y-K\alpha\|^2+\lambda\alpha^\intercal K\alpha\right]=0 &\iff -K^\intercal Y-K^\intercal Y+2K^\intercal K\alpha+\lambda\left(K+K^\intercal\right)\alpha=0\\ &\iff 2K^\intercal K\alpha+2\lambda K\alpha=2K^\intercal Y\\ &\iff \left(K+\lambda I_n\right)\alpha=Y\\ &\iff \hat{\alpha}=\left(K+\lambda I_n\right)^{-1}Y\end{aligned}$$

Writing $\hat{\alpha}=(\hat{\alpha}_i,\ldots,\hat{\alpha}_n)\in\mathbb{R}^n$ we have the Kernel Ridge estimator for all x in $\mathcal{X}$:

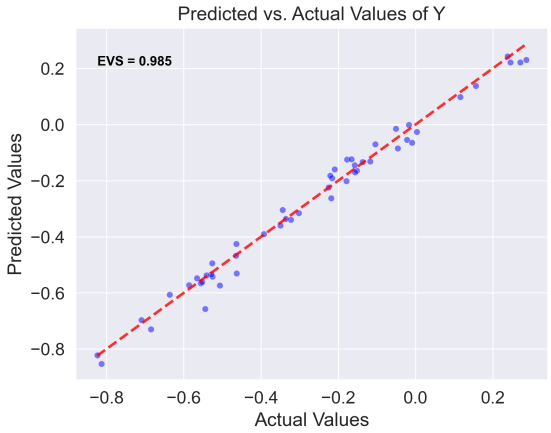
$$\hat{f}=\sum_{i=1}^{n}\hat{\alpha}_i k\left(\cdot,x_i\right)$$

B Looking at the performances of KRR in multiple dimensions problem

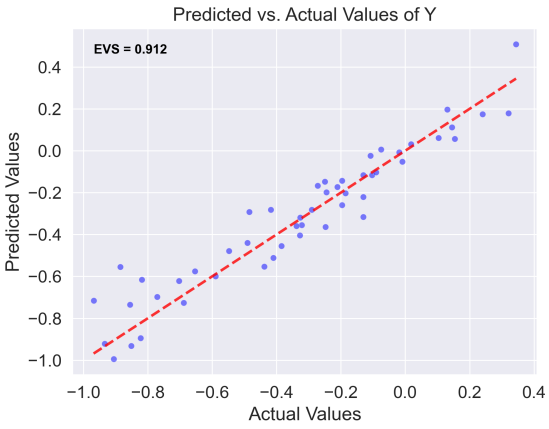
Here we show the performances of the kernel ridge regression on the 3, 4, 6 and 8 dimensions problems on the test set.



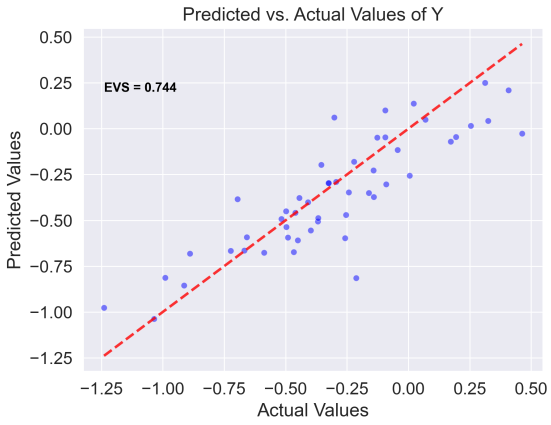
(a) On the 3D problem



(b) On the 4D problem



(c) On the 6D problem



(d) On the 8D problem

Figure 9: Performances on the test set for bigger dimension problems.