Computationally-efficient initialisation of GPs: The generalised variogram method

Felipe Tobar, Elsa Cazelles, Taco de Wolff

Computationally-efficient initialisation of GPs: The generalised variogram method

Felipe Tobar *ftobar@uchile.cl*
Initiative for Data & AI
Universidad de Chile

Elsa Cazelles *elsa.cazelles@irit.fr*
CNRS, IRIT
Université de Toulouse

Taco de Wolff *taco.dewolff@inria.cl*
Inria Chile

Abstract

We present a computationally-efficient strategy to find the hyperparameters of a Gaussian process (GP) avoiding the computation of the likelihood function. The found hyperparameters can then be used directly for regression or passed as initial conditions to maximum-likelihood (ML) training. Motivated by the fact that training a GP via ML is equivalent (on average) to minimising the KL-divergence between the true and learnt model, we set to explore different metrics/divergences among GPs that are computationally inexpensive and provide estimates close to those of ML. In particular, we identify the GP hyperparameters by matching the empirical covariance to a parametric candidate, proposing and studying various measures of discrepancy. Our proposal extends the Variogram method developed by the geostatistics literature and thus is referred to as the Generalised Variogram method (GVM). In addition to the theoretical presentation of GVM, we provide experimental validation in terms of accuracy, consistency with ML and computational complexity for different kernels using synthetic and real-world data.

1 Introduction

Gaussian processes (GPs) are Bayesian nonparametric models for time series praised by their interpretability and generality. Their implementation, however, is governed by two main challenges. First, the choice of the covariance kernel, which is usually derived from first principles or expert knowledge and thus may result in complex structures that hinder hyperparameter learning. Second, the cubic computational cost of standard, ML-based, training which renders the exact GP unfeasible for more than a few thousands observations. The GP community actively targets these issues, mainly by the development of robust and computationally efficient training strategies, and multiple accelerated, cost-efficient, training routines. Though these advances have facilitated the widespread use of GP models in realistic settings, its success heavily depends on the initialisation of the hyperparameters.

In practice, initialisation either follows from expert knowledge or time-consuming stochastic search. This is in sharp contrast with the main selling point of GPs, that is, being agnostic to the problem and able to freely learn from data. To provide researchers and practitioners with an automated, application-agnostic and cost-efficient initialisation methodology we propose to learn the hyperparameters by approximating the empirical covariance by a parametric covariance using divergences between covariances that are inexpensive to compute. This is certainly inspired by the common practice, such as when one computes some statistics (e.g., mean, variance, discrete Fourier transform) and identifies the hyperparameters from them. In the geostatistics literature, a method that follows this concept is the Variogram, however it is restricted to

particular cases of kernels and divergences. Therefore, we refer to the proposed methodology as Generalised Variogram Method (GMV) in the sense that it extends the application of the classic methodology to a broader scenario that includes general stationary kernels and metrics, in particular Fourier-based metrics.

2 Preliminaries

2.1 Motivation

Let us consider $y \sim \mathcal{GP}(0, K_0)$ and its realisation $\boldsymbol{y} = [y_1, \dots, y_n] \in \mathbb{R}^n$ at times $\boldsymbol{t} = [t_1, \dots, t_n] \in \mathbb{R}^n$. The kernel K_0 is usually learnt by choosing a family $\{K_\theta\}_{\theta \in \Theta}$ and optimising the log-likelihood

$$l(\theta) = -\frac{1}{2} \operatorname{Tr} \mathbf{K}_\theta^{-1} \boldsymbol{y}\boldsymbol{y}^\top - \frac{1}{2} \log |\mathbf{K}_\theta| - \frac{n}{2} \log 2\pi, \tag{1}$$

with respect to $\theta \in \Theta$, where we used the cyclic property of the trace, and defined $\mathbf{K}_\theta \overset{\text{def}}{=} K_\theta(\boldsymbol{t})$ according to $[\mathbf{K}_\theta]_{ij} = K_\theta(t_i - t_j), i, j \in \{1, \dots, n\}$. Since $\mathbb{E}\boldsymbol{y}\boldsymbol{y}^\top = K_0(\boldsymbol{t}) \overset{\text{def}}{=} \mathbf{K}_0$, we note that

$$\mathbb{E}l(\theta) = -\frac{1}{2} \operatorname{Tr} \mathbf{K}_\theta^{-1} \mathbf{K}_0 - \frac{1}{2} \log |\mathbf{K}_\theta| - \frac{n}{2} \log 2\pi. \tag{2}$$

Observe that, up to terms independent of θ, equation 2 is equivalent to negative Kullback-Leibler divergence (NKL) between the multivariate normal distributions $\mathcal{N}(0, \mathbf{K}_0)$ and $\mathcal{N}(0, \mathbf{K}_\theta)$ given by

$$D_{\text{NKL}}(\mathbf{K}_0 || \mathbf{K}_\theta) = -\frac{1}{2} \left(\operatorname{Tr} \mathbf{K}_\theta^{-1} \mathbf{K}_0 - n + \log \frac{|\mathbf{K}_\theta|}{|\mathbf{K}_0|} \right). \tag{3}$$

The above observation reveals that learning a GP by maximising $l(\theta)$ in equation 1 can be understood (in expectation) as minimising the KL between the $\boldsymbol{t}$-marginalisations of the true process $\mathcal{GP}(0, K_0)$ and a candidate process $\mathcal{GP}(0, K_\theta)$. This motivates the following remark.

Remark 1. *Since maximum-likelihood learning of GPs has a cubic computational cost but it is (on average) equivalent to minimising a KL divergence, what other divergences or distances, computationally cheaper than the likelihood, can be considered for learning GPs?*

2.2 Divergences over covariance functions

We consider zero-mean stationary GPs. The zero-mean restriction can be easily lifted by detrending the data with respect to a nonlinear parametric regression model to learn the mean function (and then substract it from the data). The stationary requirement allows us to i) aggregate observations in time when computing the covariance, and ii) compare covariances in terms of their (Fourier) spectral content. Therefore, we consider two types of distances over covariances: i) **temporal** ones, which operate directly to the covariances, and ii) **spectral** ones, which operate over the power spectral density (PSD), i.e., the Fourier transform of the covariances. Though we can use most metrics (e.g., L_1, L_2) on both domains, the advantage of the spectral perspective is that allows for using density-specific divergences as it is customary in signal processing (Basseville, 1989). Bregman divergences, which include the Euclidean, KL and Itakura-Saito (IS) (Itakura, 1968), are *vertical* measures, i.e, they integrate the point-wise discrepancy between densities across their support.

We also consider *horizontal* spectral measures, based on the minimal-cost to *transport* the mass from one distribution—across the support space—onto another. This concept, known as optimal transport (OT) (Villani, 2009) has only recently been considered for comparing PSDs using, e.g., the 1- and 2-Wasserstein distances, denoted W_1 and W_2 (Cazelles et al., 2021; Henderson & Solomon, 2019). See Appendix B for definitions of vertical and horizontal divergences.

2.3 Related work

Currently, two methodologies for accelerating GP training can be identified. The first one focuses directly on the optimisation procedure by, e.g., avoiding inverses (van der Wilk et al., 2020), or derivatives

(Rios & Tobar, 2018), or even by parallelisation; combining these techniques has allowed to process even a million datapoints (Wang et al., 2019). A second perspective is that of sparse GP approximations using *pseudo-inputs* (Quinonero-Candela & Rasmussen, 2005), with particular emphasis on variational methods (Titsias, 2009). This has allowed for fitting GPs to large datasets (Hensman et al., 2013), training non-parametric kernels (Tobar et al., 2015), and implementing deep GPs (Damianou & Lawrence, 2013; Salimbeni & Deisenroth, 2017).

The Wasserstein distance has been used to compare the laws of the GP (Masarotto et al., 2019; Mallasto & Feragen, 2017), and applied to kernel design, in particular to define GPs (Bachoc et al., 2018) and deep GPs (Popescu et al., 2020) over the space of probability distributions.

In geostatistics, the variogram function (Cressie, 1993; Chiles & Delfiner, 1999) is defined as the variance of the difference of a process y at two locations t_1 and t_2, that is, $\gamma(t) = \mathrm{var}[y(t_1) - y(t_2)]$. The variogram is computed by choosing a parametric form for $\gamma(t)$ and then fit it to a cloud of points (sample variogram) using least squares. Common variogram functions in the literature include exponential and Gaussian ones, thus drawing a natural connection with GP models. Furthermore, when the process y is stationary and isotropic (or one-dimensional) as in the GP models considered here, the variogram and the covariance $K(t)$ follow the relationship $\gamma(t) = K(0) - K(t)$, therefore, given a kernel function the corresponding variogram function can be clearly identified (and vice versa). The way in which the variogram is fitted in the geostatistics literature is what inspires our proposed methodology: we fit parametric forms of the covariance and the PSD to their corresponding samples in order to find appropriate values for the kernel hyperparameters. Also, as we explore different distances for the covariance and PSD beyond the Euclidean one (least squares) we denote our method *Generalised Variogram Method* (GVM).

GVM complements the literature in a way that is orthogonal to the above developments. We find the hyperparameters of a GPs in a likelihood-free manner by minimising a loss function operating directly on the sample covariace or its Fourier transform. As we will see, GVM is robust to empirical approximations of the covariance or PSDs, admits arbitrary distances and has a remarkably low computational complexity.

3 A likelihood-free covariance-matching strategy for training GPs

Let us consider the zero-mean stationary process $y \sim \mathcal{GP}(0, K_\theta)$ with covariance K_θ and hyperparameter $\theta \in \Theta$. Recall that the following statistic is an estimator of K_θ.

Definition 1. *Let $y \in \mathbb{R}$ be a zero mean stochastic process over $\mathbb{R}$ with observations $\boldsymbol{y} = [y_1, \dots, y_n] \in \mathbb{R}^n$ at times $\boldsymbol{t} = [t_1, \dots, t_n] \in \mathbb{R}^n$. The empirical covariance of y is given by*

$$\hat{K}_n(t) = \sum_{i,j=1}^{n} \frac{y_i y_j \mathbf{1}_{t=t_i-t_j}}{Card\{t|t = t_i - t_j\}}. \tag{4}$$

We aim to find the hyperparameters of y by projecting $\hat{K}_n(t)$ onto the parametrised family $\mathcal{K} = \{K_\theta | \theta \in \Theta\}$. That is, by finding θ^* such that $K_\theta(\cdot)$ is *as close as possible* to $\hat{K}_n(t)$ in equation 4, i.e.,

$$\theta_n^* = \arg\min_{\theta \in \Theta} D(\hat{K}_n, K_\theta), \tag{5}$$

where the function $D(\cdot, \cdot)$ is the chosen criterion for similarity given by any distance or divergence.

3.1 Fourier-based covariance divergences

If the covariance K_θ is integrable and stationary, Bochner's Theorem (Bochner, 1959) states that K_θ and the process' power spectral density (PSD), denoted S_θ, are *Fourier pairs*, that is,

$$S_\theta(\xi) = \mathcal{F}\{K_\theta\} \stackrel{\text{def}}{=} \int_{\mathbb{R}} K_\theta(t) e^{-j2\pi\xi t} dt, \tag{6}$$

where j is the imaginary unit and $\mathcal{F}\{\cdot\}$ denotes the Fourier transform operator.

Remark 2. *Since zero-mean stationary GPs are uniquely determined by their PSDs, any distance defined on the space of PSDs can be "lifted" to the space covariances and then to that of GPs.*

In the same fashion as we proposed to learn the hyperparameters of the GP by matching the covariance solving the optimisation in equation 5, we can learn the hyperparameters by projecting now an estimator of the PSD, denoted $\hat{S}_n$ onto a parametric family $\mathcal{S} = \{S_\theta, \theta \in \Theta\}$, that is,

$$\theta_n^* = \arg\min_{\theta \in \Theta} D_F(\hat{S}_n, S_\theta), \tag{7}$$

where $D_F(\cdot, \cdot)$ is a divergence operating on the space of PSDs. However, since the map $K_\theta \to S_\theta$ is one-to-one, equation 5 and equation 7 are equivalent when $D_F(\cdot, \cdot) = D(\mathcal{F}\{\cdot\}, \mathcal{F}\{\cdot\})$ and $\mathcal{S} = \mathcal{F}\{\mathcal{K}\}$.

We will consider parametric families $\mathcal{S}$ with explicit inverse Fourier transform, since this way θ parametrises both the kernel and the PSD and can be learnt in either domain. These families include the Dirac delta, Cosine, Square Exponential (SE), Student's t, Sinc, Rectangle, and their mixtures.

3.2 A particular case with explicit solution

Of particular relevance to our work is the 2-Wasserstein distance (W_2) and *location-scale* PSDs.

Definition 2 (Location–scale)**.** *A family of one-dimensional integrable PSDs is said to be of location-scale type if it is given by*

$$\left\{ S_{\mu,\sigma}(\xi) = \frac{1}{\sigma} S_{0,1}\left(\frac{\xi - \mu}{\sigma}\right), \mu \in \mathbb{R}, \sigma \in \mathbb{R}_+ \right\}, \tag{8}$$

where $\mu \in \mathbb{R}$ is the location parameter, $\sigma \in \mathbb{R}_+$ is the scale parameter and $S_{0,1}$ is the prototype of the family.

For arbitrary prototypes $S_{0,1}$, location-scale families of PSDs are commonly found in the GP literature. For instance, the SE, Dirac delta, Student's t, Rectangular and Sinc PSDs, correspond to the Exp-cos, Cosine, Laplace, Sinc, and Rectangular kernels respectively. Location-scale families do not, however, include kernel mixtures, which are also relevant in our setting and will be dealt with separately. Though the prototype $S_{0,1}$ might also be parametrised (e.g., with a *shape* parameter), we consider those parameters to be fixed and only focus on (μ, σ) for the rest of this section.

Remark 3. *Let us consider a location-scale family of distributions with prototype $S_{0,1}$ and an arbitrary member $S_{\mu,\sigma}$. Their quantile (i.e., inverse cumulative) functions, denoted $Q_{0,1}$ and $Q_{\mu,\sigma}$ respectively, obey*

$$Q_{\mu,\sigma}(p) = \mu + \sigma Q_{0,1}(p). \tag{9}$$

The linear expression in equation 9 is pertinent in our setting and motivates the choice of the 2-Wasserstein distance W_2. This is because for one-dimensional distributions, $W_2^2(S, S_\theta)$ can be expressed in terms of the quantile functions of S and S_θ, denoted respectively Q and Q_θ, given by

$$W_2^2(S, S_\theta) = \int_0^1 (Q_\theta(p) - Q(p))^2 dp. \tag{10}$$

We are now in position to state the first main contribution of our work.

Theorem 1. *If $\mathcal{S}$ is a location-scale family with prototype $S_{0,1}$, the minimiser of $W_2(S, S_{\mu,\sigma})$ is unique and given by*

$$\mu^* = \int_0^1 Q(p)dp \qquad \text{and} \qquad \sigma^* = \frac{1}{\int_0^1 Q_{0,1}^2(p)dp} \int_0^1 Q(p)Q_{0,1}(p)dp \tag{11}$$

where Q is the quantile function of S. The PSD S does not need to be location-scale.

Proof Sketch. The proof follows from the fact that $W_2^2(S, S_{\mu,\sigma})$ is convex both on μ and σ, which is shown by noting that its Hessian is positive via Jensen's inequality. Then, the first order conditions give the solutions in equation 11. The details of the proof can be found in Appendix A. □

Remark 4. *Although integrals of quantile functions are not usually available in closed-form, computing equation 11 is straightforward. First, $Q_{0,1}(p)$ is known for a large class of prototypes including SE and rectangles. Second, $\mu^* = \mathbb{E}_{\mathrm{x}\sim S}[x]$ and $\int_0^1 Q_{0,1}^2(p)dp = \mathbb{E}_{\mathrm{x}\sim S}[x^2]$, where* x *is a random variable with PDF S. Third, both integrals are one-dimensional and supported on $[0, 1]$, thus numerical integration is inexpensive and precise, specially for S with compact support.*

As pointed out by Cazelles et al. (2021), $W_2^2(S, S_\theta)$ is in general non-convex, however, for the particular case of the location-scale family of PSDs, convexity holds. Therefore, identifying the result in Theorem 1 with the optimisation problem in equation 7, using the W_2 metric in the Fourier domain allows us to approximate the estimator $\hat{S}_n$ by replacing S by $\hat{S}_n$ in equation 11. Since this family includes usual kernel choices, the convexity of $W_2^2(S, S_{\mu,\sigma})$ promises to be instrumental to learning GPs.

3.3 Learning from data

Our objective is to learn the ground truth kernel K_θ by matching an empirical (data-driven) estimate of the kernel K_θ with a parametric form. We achieve this using metrics in the temporal domain, i.e., solving equation 5, or in the spectral domain, i.e., solving equation 7. Learning the hyperparameters in this manner is possible provided that the statistics $\hat{K}_n$ and $\hat{S}_n$ converge to K_θ and S_θ respectively. We next provide theoretical results on the convergence of the optimal minimiser θ_n^*. The first result, Proposition 1, focuses on the particular case of the 2-Wasserstein distance and the location-scale family, presented in Section 3.2.

Proposition 1. *For S_θ in a location-scale family, S the ground truth PSD and $D_F = W_2^2$, $\mathbb{E}W_2^2(S, \hat{S}_n) \to 0$ implies that the empirical minimiser θ_n^* in equation 7 converges to the true minimiser $\theta^* = \arg\min_{\theta\in\Theta} D_F(S, S_\theta)$, meaning that $\mathbb{E}|\theta_n^* - \theta^*| \to 0$.*

Proof Sketch. First, in the location-scale family we have $\theta = (\mu, \sigma)$. Then, the solutions in Theorem 1 allow us to compute upper bounds for $|\mu^* - \mu_n^*|$ and $|\sigma^* - \sigma_n^*|$ via Jensen's and Hölder's inequalities, both of which converge to zero as $\mathbb{E}W_2^2(S, \hat{S}_n) \to 0$. The details of the proof can be found in Appendix A. □

The second result, Proposition 2, deals with the more general setting of arbitrary parametric families, the distances L_1, L_2, W_1, W_2 and either temporal and frequency based estimators. To cover both cases, we will denote $\hat{f}_n$ to refer to either $\hat{K}_n$ or $\hat{S}_n$. Let us also consider the parametric function $f_\theta \in \mathcal{H} := \{f_\theta | \theta \in \Theta\}$ and the ground truth function f which denotes either the groud truth covariance or ground-truth PSD.

Proposition 2. *For general parametric families $\mathcal{H}$, the empirical solution θ_n^* converges a.s. to the true solution θ^* under the following sufficient and stronger conditions: (i) $D = W_r$ or $L_r, r = 1, 2$, (ii) $D(\hat{f}_n, f) \xrightarrow[n\to\infty]{a.s.} 0$; (iii) $\theta_n \xrightarrow[n\to\infty]{} \theta \iff D(f_{\theta_n}, f_\theta) \to 0$; and (iv) the parameter space Θ is compact.*

Proof. This result follows as a particular case from Theorem 2.1 in Bernton et al. (2019), where the authors study general r-Wasserstein distance estimators for parametric families of distributions for empirical measures, under the notion of Γ-convergence or, equivalently, epi-convergence. The proof for the $L_r, r = 1, 2$ case is similar to that of W_r. □

With the above results, implementation of GVM only requires defining $\hat{K}_n$ and $\hat{S}_n$. In practice, we use equation 4 for $\hat{K}_n$ due to its known convergence properties. For $\hat{S}_n$, the usual choice is the Periodogram ($\hat{S}_{\mathrm{Per}}$) (Schuster, 1900). Though $\hat{S}_{\mathrm{Per}}$ is asymptotically unbiased ($\forall\xi, \mathbb{E}\hat{S}_{\mathrm{Per}}(\xi) \to S(\xi)$), it is inconsistent, i.e., its variance does not vanish when $n \to \infty$ (Stoica & Moses, 2005)[Sec. 2.4.2], meaning that Props. 1 & 2 do not guarantee $\theta_n^* \to \theta^*$. Luckily, the variance of $\hat{S}_{\mathrm{Per}}(\xi)$ can be reduced via *windowing* and the Welch/Bartlett techniques which produce (asymptotically) consistent and unbiased estimates of $S(\xi), \forall\xi$ (Stoica & Moses, 2005).

4 Practical considerations

4.1 Solving the general case: beyond the location-scale family

Recall that the proposed method has a closed-form solution only when the parametric family for the PSD is location-scale and we consider the W_2 distance over the spectral domain. However, in the general case

the corresponding minimisation is not convex in θ, therefore, iterative/numerical optimisation is needed. In particular, for horizontal Fourier distances the derivative of the loss depends on the derivative of the quantile function Q_θ which might not even be known in closed form in general, specially for mixtures. However, in most cases the lags of the empirical covariance or the frequencies of the Periodogram belong to a compact space and thus numerical computations are precise and inexpensive. Therefore, approximate derivatives can be considered (e.g., BFGS) though in our experiments the derivative-free Powell method also provide satisfactory results.

4.2 Linear computational complexity

First, for temporal divergences (operating on the covariance) we can modify the statistic in equation 4 using *binning*, that is, by averaging values for which the lags are similar; this process is automatic in the case of evenly-sampled data and widely used in discrete-time signal processing. This allows to reducing the amount of summands in $\hat{K}_n$ from $\frac{n(n+1)}{2}$ to an order n or even lower in the cases where the range of the data grows beyond the length of the correlations of interests.

Second, for the exact case of the spectral divergences in Sec. 3.2 the cost of the 2-Wasserstein loss is given by calculating i) the Periodogram $\hat{S}_n$, ii) its corresponding quantile function, and iii) the integrals in equation 11. The Periodogram has a complexity $\mathcal{O}(nk)$, where n is the number of observations and k the amount of frequency bins; in the evenly-sampled case, one could set $k = n$ and apply the fast Fourier transform at a cost $\mathcal{O}(n \log n)$. However, for applications where the amount of datapoints greatly exceeds the required frequency resolution, k can be considered to be constant, which results in a cost $\mathcal{O}(n)$. The quantile functions and their integrals also have a linear cost but only in the number of frequency bins $\mathcal{O}(k)$ since they are frequency histograms, therefore, computing the overall solution has a linear cost in the data. Second, for the general case using numerical optimisation methods, we need to calculate $\hat{S}_n$ or its quantile—which are $\mathcal{O}(n)$—only once, to then compute the chosen distance D, which is $\mathcal{O}(k)$ for discrete measures defined on a k-point grid, as many times as the optimisation routine requires it. Therefore, for the general case the cost is $\mathcal{O}(k)$ but with a constant that depends on the complexity of the parametric family $\{S_\theta, \theta \in \Theta\}$ and the optimiser of choice.

4.3 Noise variance and relationship to maximum likelihood

Following the assumptions of the Fourier transform, the spectral divergences considered apply for Lebesgue-integrable PSDs, which rules out the relevant case of noise-corrupted observations. This is because the observation noise, represented by a Dirac delta covariance, implies a PSD that includes a constant, positive, infinite-support, *spectral floor* that is non-integrable. These cases can be dealt with the temporal divergences, which are well suited (theoretically and in practice) to handle noise.

The proposed hyperparameter-search method is intended both as a standalone likelihood-free GP learning technique and also as a initialisation approach to fed initial conditions to a maximum likelihood (ML) routine. In this sense, we identify a relationship between the ML estimator $\hat{\theta}_{ML}$ and the proposed estimator θ^*, obtained from equation 7, for $D_F = L_2$. From equation 6 and Plancherel's theorem, we have $\|S - S_\theta\|_{L_2} = \|K - K_\theta\|_{L_2}$. Then, by definition of the estimators and Lemma 2 in Hoffman & Ma (2020), we obtain the following inequality

$$D_{\mathrm{KL}}(\mathbf{K}_0 || \mathbf{K}_{\hat{\theta}_{ML}}) \leq D_{\mathrm{KL}}(\mathbf{K}_0 || \mathbf{K}_{\theta^*}) \leq \frac{1}{2} \|\mathbf{K}_0^{-1}\|_2 \|\mathbf{K}_{\theta^*}^{-1}\|_2 \|\mathbf{K} - \mathbf{K}_{\theta^*}\|_F, \tag{12}$$

where $\|\cdot\|_F$ denotes the matrix Frobenius norm and recall that $\mathbf{K}_0$ is the kernel of the ground truth GP. Denoting the ball centred at 0 with radius M by $B(0, M)$, we present the following remark.

Remark 5. *The inequality in equation 12 states that if the proposed estimator θ^* is such that $\|S - S_{\theta^*}\|_{L_2} \in B(0, M)$, then $D_{\mathrm{KL}}(\mathbf{K}_0 || \mathbf{K}_{\hat{\theta}_{ML}}) \in B(0, \frac{1}{2}\|\mathbf{K}_0^{-1}\|_2 \|\mathbf{K}_{\theta^*}^{-1}\|_2 M)$. Therefore, under the reasonable assumption that the function $\theta \mapsto \mathbf{K}_\theta$ only produces well-conditioned matrices, the factor $\|\mathbf{K}_0^{-1}\|_2 \|\mathbf{K}_{\theta^*}^{-1}\|_2$ is bounded and thus both balls have radius of the same order.*

5 Experiments

The following experiments (E) assess different aspects of the proposed GVM: ability for multi-input data (E1), robustness to initialisation and observations (E2), sensibility to ways of computing $\hat{S}_n$ (E3), computation time (E4), robustness to the chosen spectral metric (E5), and finding initial conditions for kernels with several hyperparameters (E6). The benchmarks were implemented on MOGPTK (de Wolff et al., 2021), and the Python codes (in Appendix C) will be released to the public.

5.1 E1: Learning an isotropic SE kernel (standard variogram, 5-dimensional inputs, L_2)

In geostatistics, the standard variogram method learns a parametric representation of the sample covariance and is thus a particular instance of our method. To replicate this setting, we sampled 1000 points from a GP with SE kernel $K(\tau) = \sigma^2 \exp(-\frac{1}{2l^2}||\tau||^2) + \sigma^2_{\text{noise}}\delta_t$, where $\tau \in \mathbb{R}^5$ and $\sigma^2 = 5$, $l = 1$ and $\sigma^2_{\text{noise}} = 1$. We chose the L_2 distance to match this kernel and the result is presented in Fig. 1. The implementation of GVM was executed in 0.8[s] and found hyperparameters $\sigma^2 = 5.85$, $l = 1.09$, and $\sigma_{\text{noise}} = 1.02$.

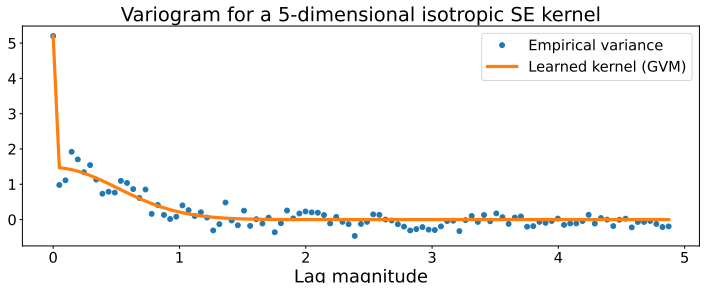


Figure 1: Empirical covariance and fitted covariance function. The data consisted of 1000 5-dimensional datapoints.

5.2 E2: Stability with respect to initial conditions (SM kernel, L_2, temporal)

This experiment assessed the stability of GVM with respect to random initial conditions and different realisations. We considered a GP with a 2-component spectral mixture kernel $K(t) = \sum_{i=1}^{2} \sigma_i^2 \exp(-\gamma_i \tau^2) \cos(2\pi\mu_i\tau) + \sigma^2_{\text{noise}}\delta_\tau$ with hyperparameters $\sigma_1 = 2$, $\gamma_1 = 10^{-4}$, $\mu_1 = 2 \cdot 10^{-2}$, $\sigma_2 = 2$,$\gamma_2 = 10^{-4}$, $\mu_2 = 3 \cdot 10^{-2}$, $\sigma_{\text{noise}} = 1$. We produced 4000-point realisations from the GP and 50 random initial conditions $\{\theta^r\}_{r=1}^{50}$ according to $[\theta^r]_i \sim \text{Uniform}[\frac{1}{2}\theta_i, \frac{3}{2}\theta_i]$, where θ_i is the i-th true hyperparameter.

We considered two settings: i) train from $\{\theta^r\}_{r=1}^{50}$ using ML, and ii) compute GVM from $\{\theta^r\}_{r=1}^{50}$ and then perform ML. Each procedure was implemented using a single realisation (to test stability wrt θ^r) and different realisations (to test stability wrt the data). Our estimates $\hat{\theta}_i$ were assesed in terms of the NLL and the relative mean absolute error (RMAE) of the parameters $\sum_{i=1}^{8} |\theta_i - \hat{\theta}_i|/|\theta_i|$.

Fig. 2 shows the NLL (left) and RMAE (right) versus computation time, for the cases of fixed (top) and multiple (bottom) observations; all times start from $t = 1$ to use the logarithmic scale. First, in all cases the GVM initialisation (in red) took about half a second and resulted in an NLL/RMAE virtually identical to those achieved by ML initialised by GVM, this means that GVM provides reliable parameters and not just initial conditions for ML. Second, for the fixed observations (top), the GVM was stable wrt θ^r unlike ML which in some cases diverged. Third, for the multiple observations (bottom) GVM-initialised ML diverged in two (out of 50) runs, which is far fewer than the times that random-initialised ML diverged.

5.3 E3: Sensibility of GVM wrt the Periodogram (exact case: W_2, location-scale)

We then considered kernels with location-scale PSDs and the W_2 metric over the PSDs. This case has a unique solution but requires us to compute $\hat{S}_n$ in equation 7; this experiment evaluates different ways of

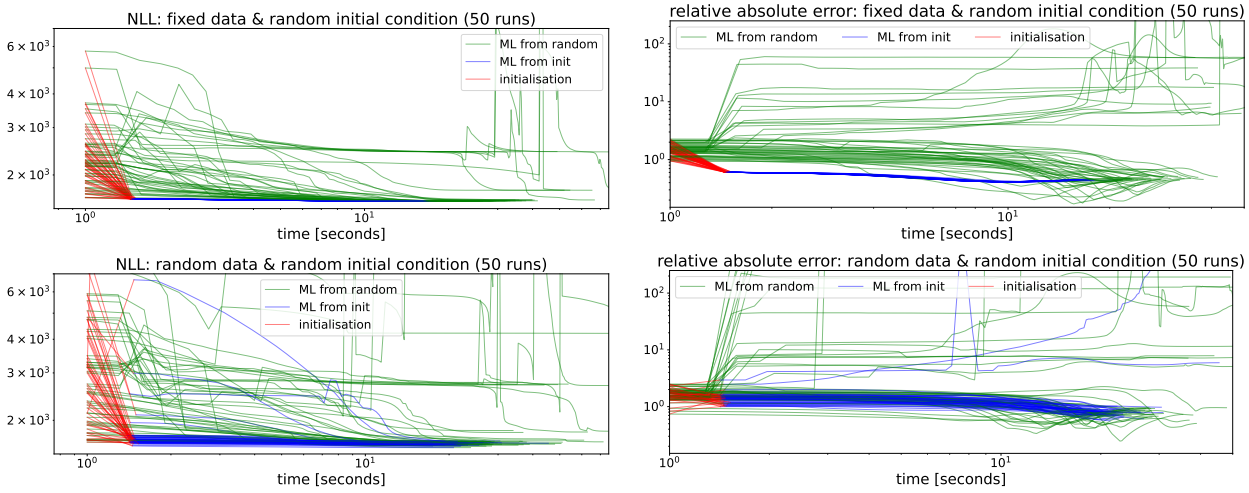


Figure 2: GVM as initialisation for ML starting form 50 random initial conditions: proposed GVM (red), standard ML (green) and ML starting from GVM (blue). The top plots consider a single dataset, while the bottom plots consider different realisations for each initial condition. The L-BFGS-B optimizer was used with the default gradient tolerance in order for the results to be comparable.

Table 1: Parameter identification for the Exp-cos and Sinc kernels under different sampling settings, Periodogram methods and windows using GVM. NB: estimates are multiplied by 100 for ease of presentation.

Kernel	Window	Periodogram	Bartlett	Welch
Exp-cos	none	4.97±0.12/0.98±0.09	5.01±0.12/1.30±0.14	4.98±0.13/1.11±0.08
	hann	4.97±0.19/0.94±0.13	4.95±0.15/1.12±0.09	4.96±0.14/1.01±0.08
	hamm	4.97±0.18/0.94±0.12	4.95±0.15/1.10±0.09	4.96±0.14/1.00±0.08
Sinc	none	4.98±0.09/1.02±0.15	5.00±0.11/2.40±0.15	4.98±0.09/1.67±0.10
	hann	4.98±0.11/0.90±0.16	4.98±0.09/2.13±0.05	4.98±0.09/1.41±0.08
	hamm	4.98±0.11/0.90±0.15	4.88±0.09/1.99±0.05	4.98±0.09/1.73±0.09

doing so. We produced 2000 observations **evenly-sampled** in $[0, 4000]$ from GPs with square exponential and rectangular PSDs both with location $\mu = 0.05$ and scale $l = 0.01$, they correspond to the single component Exp-cos and Sinc kernels respectively. We computed $\hat{S}$ via the Periodogram, Welch and Bartlett methods with different windows. Table 1 shows the learnt parameters (multiplied by 100 for ease of presentation) with their standard deviation over 100 runs. The found estimates are consistent, in particular for the location parameter, for both kernels. In general, all approximations methods and windows give accurate estimates with the exception of the windowed version (Bartlett and Welch) for the Sinc kernel. This can be attributed to the spectral energy spread in the frequency domain due to the windows considered, then, by using the Wasserstein-2 metric, the scale parameter is overestimated. For both kernels, the case of the (windowless) Periodogram is shown in Fig. 3 and the remaining cases are all shown in Appendix D. In the light of these results, we considered the Periodogram (no window) for the following experiments.

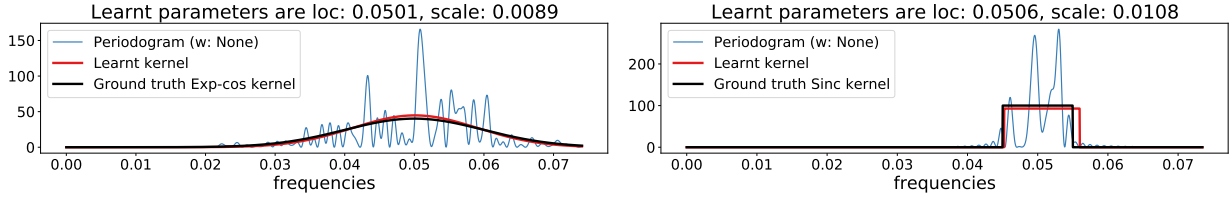


Figure 3: GVM estimates for Exp-cos (top) and Sinc (bottom) kernels shown in red against $\hat{S}$ (blue) and true kernels (black). This case: Periodogram, no window.

5.4 E4: Linear complexity (exact case)

We then evaluated the computation time for the exact case of GVM (W_2 distance and location-scale family) for an increasing amount of observations. We considered **unevenly-sampled** observations from an single component SM kernel ($\mu = 0.05, \sigma = 0.01$) in the range $[0, 1000]$. We compared GVM against i) the ML estimate starting from the GVM value (full GP, 100 iterations), and ii) the sparse GP using 200 pseudo inputs (Snelson & Ghahramani, 2006). Fig. 4 shows the computing times versus the number of observations, and validated the claimed linear cost of GVM and its superiority wrt to the rest of the methods. The (solid line) interpolation in the plot is of linear order for GVM, linear for sparse GP since the number of inducing points is fixed, and cubic for the full GP.

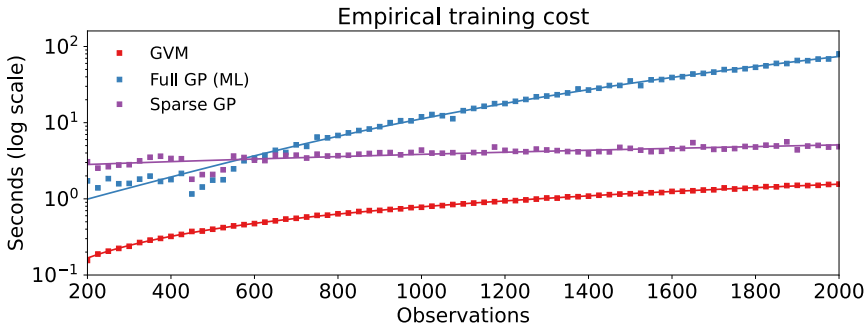


Figure 4: Training times vs number of datapoints.

5.5 E5: Fitting a 20-component spectral mixture (different spectral metrics)

This experiment shows the effect of different spectral distances in the GVM estimates using a real-world audio signal from the Free Spoken Digit Dataset[1]. We trained a 20-component SM kernel (Wilson & Adams, 2013), a kernel known to be difficult to train, and considered the distances L_1, L_2, W_1 and W_2 (spectral); Itakura-Saito and KL were unstable and left our of the comparison. Fig. 5 shows the results of the GVM: observe that under almost all metrics, the 20-component spectral mixture matches the Periodogram. The exception is W_2 which struggles to replicate the PSD peaks due to its objective of averaging mass *horizontally*.

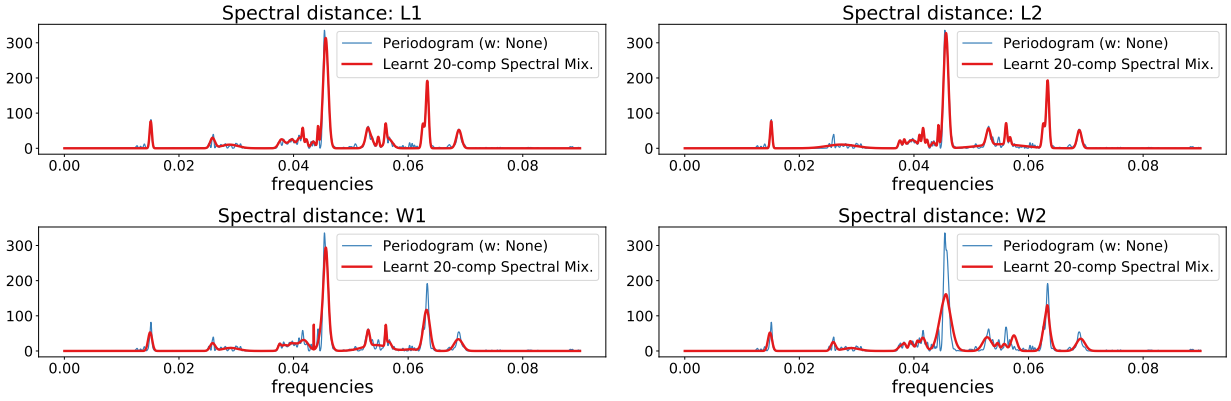


Figure 5: GVM matching Periodogram with a 20-component SM under different spectral metrics.

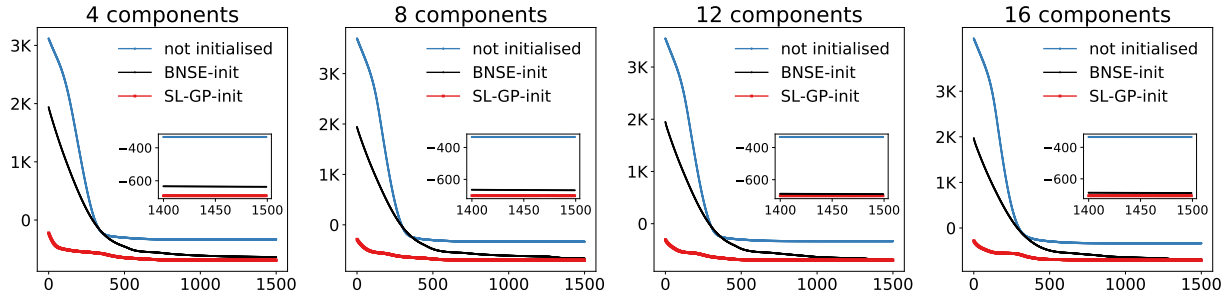


Figure 6: NLLs for the SE spectral mixtures (4,8,12 and 16 components) with different initialisation strategies.

5.6 E6: Learning spectral mixtures and parameter initialisation (L_2, multiple orders)

In this last experiment, GVM was implemented to find the initial conditions of a GP with SM kernel (4, 8, 12 and 16 components) to a real-world 1800-point heart-rate signal from the MIT-BIH database[2]. We considered the L_2 metric (spectral) minimised with Powell and then passed the hyperparameters to an ML routine for 1500 iterations (using Adam with learning rate = 0.1). This methodology was compared against the random initialisation and provided by MOGPTK based on Bayesian nonparametric spectral estimation (BNSE) (Tobar, 2018). Fig. 7 first shows the GVM approximations to the heart-rate PSD using 16-component spectral mixtures, for both both kernels.

Table 2: Computation times (secs) for fitting SMs.

	4-comp	8-comp	12-comp	16-comp
GVM init	**2.6**	**7.6**	**10.9**	**23.9**
BNSE init	74.6	68.9	72.0	74.1
ML	420.3	464.6	529.2	582.6

Fig. 6 shows NLL for the cases considered. Observe that: i) the non-initialised ML training becomes trapped in local minima in all four cases, ii) the initialisation provided by GVM provides a dramatic reduction of the NLL, even wrt to the BNSE initialisation, iii) the "elbow" at the beginning of the GVM-initialised case suggests that the ML training could have run for a a few iterations (e.g., 100) and still reach a sound solution. Table 2 shows the execution times and reveals the superiority of GVM also in computational time.

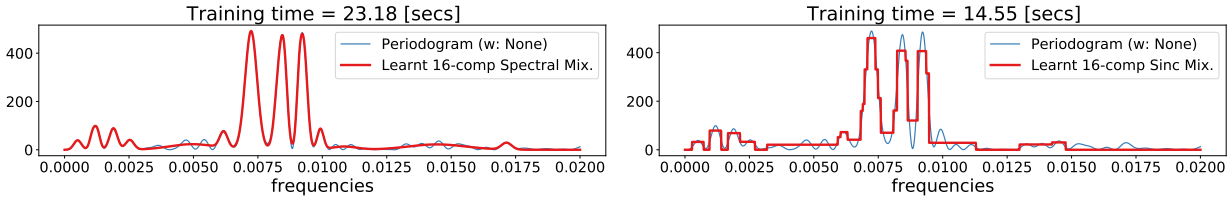


Figure 7: GVM approximations of the PSD of a 1800-sample heart-rate signal using 16-components SE (top) and rectangular (bottom) mixtures. Training time shown above each plot.

6 Conclusions

By direct minimisation of the discrepancy among covariance functions and their Fourier transforms, we have proposed a novel method for training Gaussian processes, which avoids computation of the (cubic cost)

[1] https://github.com/Jakobovski/free-spoken-digit-dataset
[2] http://ecg.mit.edu/time-series/

likelihood function, and is thus well suited for stationary data. Our approach, termed Generalised Variogram Method (GVM), represents a critical improvement in terms of computational complexity: we have shown, both theoretically and empirically, that for the particular case of the 2-Wasserstein spectral distance and location-scale PSDs, GVM is convex and its solution can be computed in a single step. In experimental terms, we showed the following properties of GVM in the general case: i) applicability to multi-input data, ii) stability wrt to different ways of computing the Periodogram, iii) consistency under different realisations of the GP unlike ML, iv) computational efficiency wrt ML and sparse GPs, v) a realistic alternative to compute initial conditions for ML resulting in considerable reduction of ML iterations, and lastly, vi) ability to train kernels of large number of components that are challenging to train from random initial conditions using ML.

We hope that our work paves the way for further research. In theoretical terms, we envision extensions towards non-stationary data using, e.g., time-frequency representations or mini-batches. In practical terms, our developed companion software (to be converted into a standalone toolbox) will help others make use of this initialisation method.

References

S. Amari. *Information geometry and its applications*, volume 194. Springer, 2016.

F. Bachoc, F. Gamboa, J.-M. Loubes, and N. Venet. A Gaussian process regression model for distribution inputs. *IEEE Transactions on Information Theory*, 64(10):6620–6637, 2018.

M. Basseville. Distance measures for signal processing and pattern recognition. *Signal processing*, 18(4): 349–369, 1989.

E. Bernton, P. Jacob, M. Gerber, and C. Robert. On parameter estimation with the Wasserstein distance. *Information and Inference: A Journal of the IMA*, 8(4):657–676, 2019.

S. Bochner. *Lectures on Fourier integrals*, volume 42. Princeton University Press, 1959.

E. Cazelles, A. Robert, and F. Tobar. The Wasserstein-Fourier distance for stationary time series. *IEEE Transactions on Signal Processing*, 69:709–721, 2021.

J. P. Chiles and P. Delfiner. *Geostatistics, Modelling Spatial Uncertainty*. Wiley, 1999.

N. Cressie. *Statistics for spatial data*. Wiley, 1993.

A. Damianou and N.D. Lawrence. Deep Gaussian processes. In *Proc. of the International Conference on Artificial Intelligence and Statistics*, volume 31 of *PMLR*, pp. 207–215, 2013.

T. de Wolff, A. Cuevas, and F. Tobar. MOGPTK: The multi-output Gaussian process toolkit. *Neurocomputing*, 424:49 – 53, 2021.

T. Henderson and J. Solomon. Audio transport: A generalized portamento via optimal transport. *arXiv preprint arXiv:1906.06763*, 2019.

J. Hensman, N. Fusi, and N.D. Lawrence. Gaussian processes for big data. In *Proc. of the Conference on Uncertainty in Artificial Intelligence*, pp. 282–290, 2013.

M. Hoffman and Y.-A. Ma. Black-box variational inference as distilled Langevin dynamics. In *Proc. of the International Conference on Machine Learning*, pp. 9267–9277, 2020.

F. Itakura. Analysis synthesis telephony based on the maximum likelihood method. In *The 6th international congress on acoustics, 1968*, pp. 280–292, 1968.

A. Mallasto and A. Feragen. Learning from uncertain curves: The 2-Wasserstein metric for Gaussian processes. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

V. Masarotto, V. M Panaretos, and Y. Zemel. Procrustes metrics on covariance operators and optimal transportation of Gaussian processes. *Sankhya A*, 81(1):172–213, 2019.

G. Peyré and M. Cuturi. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

S. Popescu, D. Sharp, J. Cole, and B. Glocker. Hierarchical Gaussian processes with Wasserstein-2 kernels. *arXiv preprint: 2010.14877*, 2020.

J. Quinonero-Candela and C.E. Rasmussen. A unifying view of sparse approximate Gaussian process regression. *The Journal of Machine Learning Research*, 6:1939–1959, 2005.

G. Rios and F. Tobar. Learning non-Gaussian time series using the Box-Cox Gaussian process. In *International Joint Conference on Neural Networks*, pp. 1–8, 2018.

H. Salimbeni and M. Deisenroth. Doubly stochastic variational inference for deep Gaussian processes. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

A. Schuster. The periodogram of magnetic declination as obtained from the records of the Greenwich Observatory during the years 1871–1895. *Trans. Cambridge Philos. Soc*, 18:107–135, 1900.

E. Snelson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems*, volume 18. MIT Press, 2006.

P. Stoica and R. Moses. *Spectral analysis of signals*. Pearson Prentice Hall Upper Saddle River, NJ, 2005.

M. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5, pp. 567–574, 2009.

F. Tobar. Bayesian nonparametric spectral estimation. In *Advances in Neural Information Processing Systems 31*, pp. 10148–10158, 2018.

F. Tobar, T. Bui, and R. Turner. Learning stationary time series using Gaussian processes with nonparametric kernels. In *Advances in Neural Information Processing Systems 28*, pp. 3483–3491, 2015.

M. van der Wilk, ST John, A. Artemev, and J. Hensman. Variational Gaussian process models without matrix inverses. In *Proceedings of The 2nd Symposium on Advances in Approximate Bayesian Inference*, volume 118 of *Proceedings of Machine Learning Research*, pp. 1–9. PMLR, 08 Dec 2020.

C. Villani. *Optimal transport: old and new*, volume 338. Springer, 2009.

K. Wang, G. Pleiss, J. Gardner, S. Tyree, K.Q. Weinberger, and A.G. Wilson. Exact Gaussian processes on a million data points. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

A. Wilson and R. Adams. Gaussian process kernels for pattern discovery and extrapolation. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pp. 1067–1075, Atlanta, Georgia, USA, 2013. PMLR.

A Proofs

A.1 Convexity of spectral loss for W_2 and location-scale family

Proof of Theorem 1. We recall that

$$W_2^2(S, S_{\mu,\sigma}) = \int_0^1 (Q_{\mu,\sigma}(p) - Q(p))dp. \tag{13}$$

From a direct application of the rule of differentiation under the integral sign, we get the gradient for the location-scale family:

$$\begin{aligned}\nabla_{\mu,\sigma} W_2^2(S, S_{\mu,\sigma}) &= 2\int_0^1 (Q_{\mu,\sigma}(p) - Q(p))\nabla_{\mu,\sigma} Q_{\mu,\sigma}(p)dp \\ &= 2\int_0^1 (\mu + \sigma Q_{0,1}(p) - Q(p))\nabla_{\mu,\sigma}(\mu + \sigma Q_{0,1}(p))dp \\ &= 2\int_0^1 (\mu + \sigma Q_{0,1}(p) - Q(p))\begin{pmatrix}1\\ Q_{0,1}(p)\end{pmatrix}dp.\end{aligned} \tag{14}$$

Hessian for the location-scale family:

$$\boldsymbol{H}_{\mu,\sigma} W_2^2(S, S_{\mu,\sigma}) = 2\int_0^1 \begin{pmatrix}1 & Q_{0,1}(p)\\ Q_{0,1}(p) & Q_{0,1}(p)^2\end{pmatrix}dp. \tag{15}$$

Determinant of the Hessian (via Jensen's inequality):

$$|\boldsymbol{H}|/2 = \int_0^1 Q_{0,1}(p)^2 dp - \left(\int_0^1 Q_{0,1}(p)dp\right)^2 > \int_0^1 Q_{0,1}(p)^2 dp - \int_0^1 Q_{0,1}(p)^2 dp = 0, \tag{16}$$

where the inequality is strict due to the strict convexity of $(\cdot)^2$.

Therefore, the first order conditions are given by:

$$\int_0^1 (\mu + \sigma Q_{0,1}(p) - Q(p))dp = 0 \iff \mu = \int_0^1 (Q(p) - \sigma Q_{0,1}(p))dp = \int_0^1 Q(p)dp \tag{17}$$

$$\text{and} \quad \int_0^1 (\mu + \sigma Q_{0,1}(p) - Q(p))Q_{0,1}(p)dp = 0 \tag{18}$$

$$\iff \sigma = \frac{\int_0^1 (Q(p) - \mu)Q_{0,1}(p)dp}{\int_0^1 Q_{0,1}(p)^2 dp} = \frac{\int_0^1 Q(p)Q_{0,1}(p)dp}{\int_0^1 Q_{0,1}^2(p)dp}, \tag{19}$$

where in the last expression we have used the fact that the location of the prototype $S_{0,1}$ is zero and so is its mean, meaning that if $\mathrm{x} \sim S_{0,1}$ we can write $\int_0^1 \mu Q_{0,1}(p)dp = \mu \mathbb{E}_{\mathrm{x}\sim S_{0,1}}[x] = 0$. □

A.2 Learning from data (W_2 distance and location-scale family)

Proof of Proposition 1. First, recall that in the location-scale family $\theta = (\mu, \sigma)$. We denote by Q and $\hat{Q}_n$ the respective quantile functions of S and $\hat{S}_n$. Then, following the solutions in Theorem 1 and Jensen's inequality we can compute the following upper bound for the location parameter μ^*:

$$\begin{aligned}(\mathbb{E}|\mu^* - \mu_n^*|)^2 \leq \mathbb{E}|\mu^* - \mu_n^*|^2 &= \mathbb{E}\left|\int_0^1 Q(p)dp - \int_0^1 \hat{Q}_n(p)dp\right|^2 \\ &\leq \mathbb{E}\left[\int_0^1 |Q(p) - \hat{Q}_n(p)|^2 dp\right] = \mathbb{E}[W_2^2(S, \hat{S}_n)],\end{aligned}$$

and by hypothesis $\mathbb{E}W_2^2(S, \hat{S}_n) \to 0$.

In the same sense, now using Hölder's inequality, we obtain the following bound for the scale parameter σ^*:

$$\begin{aligned}
\mathbb{E}|\sigma^* - \sigma_n^*| &= \mathbb{E}\left|\int_0^1 Q(p)Q_{0,1}(p)dp - \int_0^1 \hat{Q}_n(p)Q_{0,1}(p)dp\right| \\
&\leq \mathbb{E}\left[\int_0^1 |(Q(p) - \hat{Q}_n(p))Q_{0,1}(p)|dp\right] \\
&\leq \mathbb{E}\left[\left(\int_0^1 |Q(p) - \hat{Q}_n(p)|^2 dp\right)^{\frac{1}{2}}\right]\left(\int_0^1 |Q_{0,1}(p)|^2 dp\right)^{\frac{1}{2}} \\
&= \mathbb{E}[W_2(S, \hat{S}_n)]\left(\int_0^1 |Q_{0,1}(p)|^2 dp\right)^{\frac{1}{2}},
\end{aligned}$$

which tends to 0 by again Jensen's inequality, $(\mathbb{E}[W_2(S, \hat{S}_n)])^2 \leq \mathbb{E}[W_2^2(S, \hat{S}_n)] \to 0$. □

B Definition of the distances and divergences

For two functions f_1 and f_2, we have the following general distances

- 1-Euclidean : $L_1(f_1, f_2) = \int_{\mathbb{R}} |f_1(\xi) - f_2(\xi)|d\xi$
- 2-Euclidean : $L_2(f_1, f_2) = \int_{\mathbb{R}} (f_1(\xi) - f_2(\xi))^2 d\xi$

Furthermore, when f_1 and f_2 are densities with quantile functions Q_1 and Q_2 respectively, we have the additional divergences.

- 1-Wasserstein (Villani, 2009; Peyré & Cuturi, 2019) :

$$W_1(f_1, f_2) = \int_0^1 |Q_1(p) - Q_2(p)|dp$$

- 2-Wasserstein (Villani, 2009; Peyré & Cuturi, 2019) :

$$W_2(f_1, f_2) = \int_0^1 (Q_1(p) - Q_2(p))^2 dp$$

- Kullback-Leibler : $D_{\mathrm{KL}}(f_1 \| f_2) = \int_{\mathbb{R}} \log\left(\frac{f_1(\xi)}{f_2(\xi)}\right) f_1(\xi)d\xi$
- Itakura-Saito (Itakura, 1968) :

$$D_{\mathrm{IS}}(f_1 \| f_2) = \int_{\mathbb{R}} \left(\frac{f_1(\xi)}{f_2(\xi)} - \log\frac{f_1(\xi)}{f_2(\xi)} - 1\right) d\xi$$

- Bregman divergences (Amari, 2016) : for a function $G : \mathbb{R} \to \mathbb{R}$ that is differentiable and strictly convex,

$$D_G(f_1, f_2) = G(f_1) - G(f_2) - \langle \nabla G(f_2), f_1 - f_2 \rangle$$

Here, we have assumed that both f_1 and f_2 integrate unity, in the cases where this condition is not met, the densities can be normalised before computing the distance.

C Code

We have developed a short self-contained toolbox included in the attached Supplementary Material. The purpose of the code is to facilitate the use of the proposed method by the community and in particular to replicate all our results. The components included are as follows

- The files `waflgp.py` and `gpinit.py` defines the class, constructor and methods for the spectral and temporal distances respectively.
- The file `utils.py`, which contains simple auxiliary functions
- The Jupyer Notebook `Exp0_minimal_ex`, a minimal working example of our toolbox
- Notebooks `Exp1, Exp2, Exp3, Exp4, Exp5, Exp6` replicate the paper experiments.
- The data used for the experiments (heart-rate and audio)

For the reader's conveniece, Jupyter Notebook `Exp0_minimal_ex` is attached here.

Minimal working example of provided code

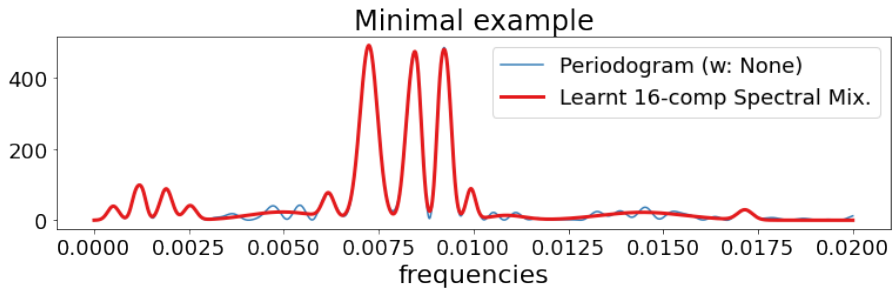
[1]:
```
#general imports
import numpy as np
import matplotlib.pyplot as plt
#our package
from waflgp import *
import utils
```

[2]:
```
#load data
signal = np.loadtxt('Data/hr2.txt')
```

[3]:
```
#instantiate model, sum of 16 Gaussians
q = 16
gp = waflgp(space_output=signal, aim = 'learning', kernel = 'qSM')#Spectral Mix
#set frequencies (optional)
freqs = np.linspace(0,0.02,2000)
gp.set_freqs(freqs)
```

[4]:
```
#train with periodogram, L2 metric and q components
gp.train_WL(method = 'periodogram', metric = 'L2', order=q)
#plot Periodogram and best PSD fit
gp.plot_psd(title = f'Minimal example')
```

```
Optimization terminated successfully.
         Current function value: 0.002876
         Iterations: 33
         Function evaluations: 21261
L2-ok
```



D Additional figures for E3

Experiment E3 shows the sensibility of the choice of Periodogram method and window for two kernels. Here, we provide all the figures corresponding to the estimates in Table 2 in the paper. The caption of each set of figures explains the setting considered.

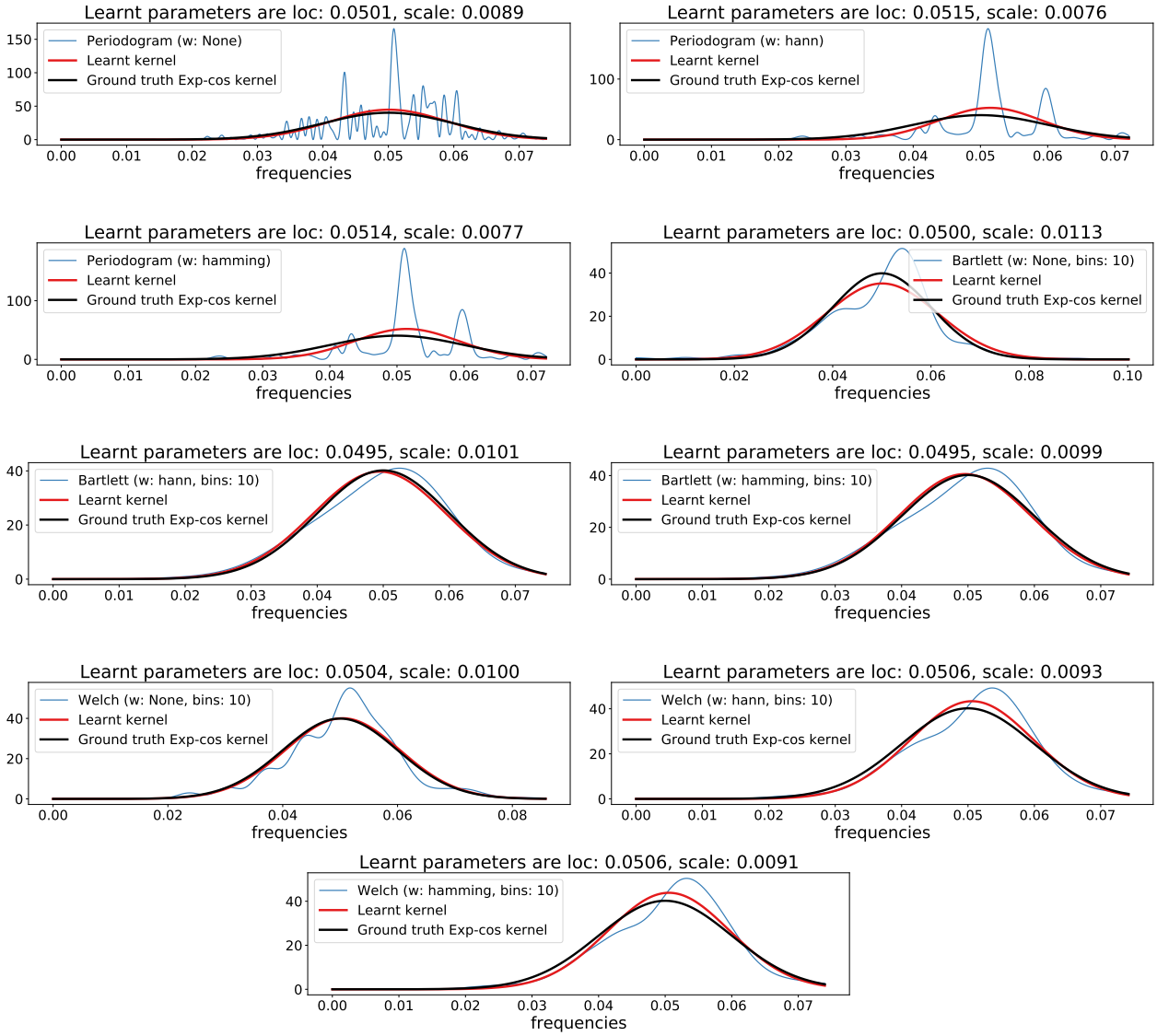


Figure 8: Kernel: Exp-cos.

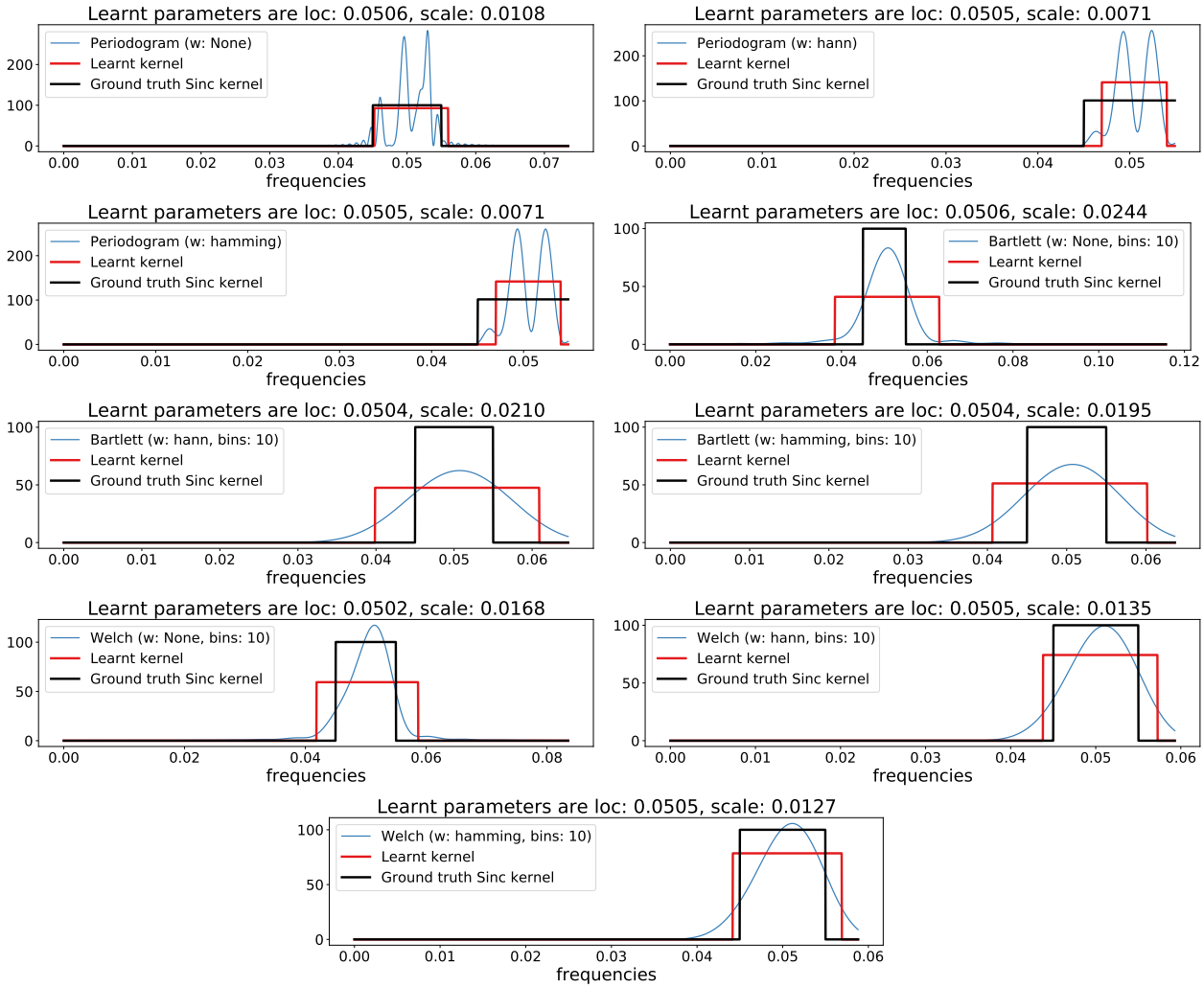


Figure 9: Kernel: Sinc.