Supplementary material:
Learning spectrograms with convolutional spectral kernels

Zheyang Shen   Markus Heinonen   Samuel Kaski

Helsinki Institute for Information Technology, HIIT,
Department of Computer Science, Aalto University

In the supplementary material, we include additional derivations, figures and experiments regarding the technical details of our paper, including a detailed derivation of the CSK formulation, the derivation of spectrogram as well as a plot denoting the equivalence between two DGP formulations, and more experimental details.

1 Convolutional spectral kernels: a derivation

In this section, we derive the convolutional spectral kernel (CSK) in detail, which solves the following integral:

$$\begin{aligned}
K_{\mathbf{x}_i}(\mathbf{x}_i - \mathbf{u}) &= \mathcal{N}(\mathbf{x}_i - \mathbf{u} | \imath \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) = \frac{1}{|2\pi \boldsymbol{\Sigma}_i|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}_i - \mathbf{u} - \imath \boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x}_i - \mathbf{u} - \imath \boldsymbol{\mu}_i) \right), \qquad (1)\\
k(\mathbf{x}_i, \mathbf{x}_j) &= \int K_{\mathbf{x}_i}(\mathbf{x}_i - \mathbf{u}) \overline{K_{\mathbf{x}_j}(\mathbf{x}_j - \mathbf{u})} \, \mathrm{d}\mathbf{u}. \qquad (2)
\end{aligned}$$

Note that a transpose instead of Hermitian is used in the feature representation, making the function $K_{\mathbf{x}_i}(\mathbf{x}_i - \mathbf{u})$ an improper density. We denote the Fourier transform as an operator on functions: $\mathcal{F}[f](\boldsymbol{\omega}) = \int f(\mathbf{x}) e^{\imath \boldsymbol{\omega}^\top \mathbf{x}} \, \mathrm{d}\mathbf{x}$. The Fourier transform of $K_{\mathbf{x}_i}(\mathbf{u})$ takes a simple form, which can be used to formulate the Fourier transform of the kernel:

$$\begin{aligned}
\mathcal{F}\left[\mathcal{N}(\mathbf{v} | \imath \boldsymbol{\mu}, \boldsymbol{\Sigma})\right] &= \exp\left(-\frac{1}{2} \boldsymbol{\omega}^\top \boldsymbol{\Sigma} \boldsymbol{\omega} - \boldsymbol{\mu}^\top \boldsymbol{\omega} \right), \qquad (3)\\
k(\mathbf{x}_i - \mathbf{x}_j) &= \mathcal{F}^{-1}\left\{ \mathcal{F}\left[k(\mathbf{x}_i - \mathbf{x}_j) \right] \right\} \qquad (4)\\
&= \mathcal{F}^{-1}\left\{ \mathcal{F}\left[\int K_{\mathbf{x}_i}(\mathbf{x}_i - \mathbf{u}) \overline{K_{\mathbf{x}_j}(\mathbf{x}_j - \mathbf{u})} \, \mathrm{d}\mathbf{u} \right] \right\} \qquad (5)\\
&= \mathcal{F}^{-1}\left\{ \mathcal{F}\left[K_{\mathbf{x}_i}(\mathbf{x}_i - \mathbf{u}) \right] \mathcal{F}\left[\overline{K_{\mathbf{x}_j}(\mathbf{x}_j - \mathbf{u})} \right] \right\} \qquad (6)\\
&= \mathcal{F}^{-1}\left\{ \mathcal{F}\left[K_{\mathbf{x}_i}(\mathbf{x}_i - \mathbf{u}) \right] \overline{\mathcal{F}}\left[K_{\mathbf{x}_j}(\mathbf{x}_j - \mathbf{u}) \right] \right\} \qquad (7)\\
&= \mathcal{F}^{-1}\left\{ \mathcal{F}\left[K_{\mathbf{x}_i}(\mathbf{x}_i - \mathbf{u}) \right] \mathcal{F}\left[K_{\mathbf{x}_j}(\mathbf{x}_j - \mathbf{u}) \right] \right\} \qquad (8)\\
&= \mathcal{F}^{-1}\left\{ \exp\left(-\frac{1}{2} \boldsymbol{\omega}^\top \boldsymbol{\Sigma}_i \boldsymbol{\omega} - \boldsymbol{\mu}_i^\top \boldsymbol{\omega} - \frac{1}{2} \boldsymbol{\omega}^\top \boldsymbol{\Sigma}_j \boldsymbol{\omega} - \boldsymbol{\mu}_j^\top \boldsymbol{\omega} \right) \right\} \qquad (9)\\
&= \mathcal{F}^{-1}\left\{ \exp\left(-\frac{1}{2} \boldsymbol{\omega}^\top (\boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_j) \boldsymbol{\omega} - (\boldsymbol{\mu}_i + \boldsymbol{\mu}_j)^\top \boldsymbol{\omega} \right) \right\} \qquad (10)\\
&= \mathcal{N}\left(\mathbf{x}_i - \mathbf{x}_j | \imath(\boldsymbol{\mu}_i + \boldsymbol{\mu}_j), \boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_j \right). \qquad (11)
\end{aligned}$$

The pseudo-Gaussian density gives a convenient closed-form Fourier transform, which we use to obtain the solution to the Hermitian inner product.

2 The spectrogram derivations

Many kernels share the form of a mixture of generalized Gaussian characteristic functions (CFs). In this section, we address certain qualities of the Gaussian characteristic functions that justify our approximation of the kernel. Consider the CF of a Gaussian distribution (spectral mixture kernel [1]) the following generalized Gaussian CF:

$$\begin{aligned} k_{SM}(\mathbf{x}_i,\mathbf{x}_j) &= \sigma \exp\left(-\frac{1}{2}(\mathbf{x}_i-\mathbf{x}_j)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}_i-\mathbf{x}_j) + \imath\langle\boldsymbol{\mu},\mathbf{x}_i-\mathbf{x}_j\rangle\right), \qquad (12)\\ k(\mathbf{x}_i,\mathbf{x}_j) &= \sigma_{ij}\exp\left(-\frac{1}{2}D_{ij}+\imath U_{ij}\right). \qquad (13)\end{aligned}$$

The Wigner transform is tractable for the SM kernel (12), because D_{ij} and U_{ij} terms are linearized with respect to $\boldsymbol{\tau}=\mathbf{x}_i-\mathbf{x}_j=(\mathbf{x}+\boldsymbol{\tau}/2)-(\mathbf{x}-\boldsymbol{\tau}/2)$. We can similarly linearize the two terms for an approximation of kernel values, and get an approximate Wigner transform. Given by the property of a Gaussian CF, we give the following assumptions based on the convexity of D_{ij} and the zero diagonal of U_{ii} for k_{SM}.

$$\begin{aligned} D_{\mathbf{x},\mathbf{x}} &= 0, \qquad (14)\\ \mathcal{J}_{\mathbf{t}}D_{\mathbf{x}+\mathbf{t}/2,\mathbf{x}-\mathbf{t}/2} &= \mathbf{0}, \qquad (15)\\ \mathcal{H}_{\mathbf{t}}D_{\mathbf{x}+\mathbf{t}/2,\mathbf{x}-\mathbf{t}/2} &\succeq \mathbf{0}, \qquad (16)\\ U_{\mathbf{x},\mathbf{x}} &= 0. \qquad (17)\end{aligned}$$

We approximate a Taylor expansion based on those assumptions:

$$\begin{aligned} D_{\mathbf{x}+\boldsymbol{\tau}/2,\mathbf{x}-\boldsymbol{\tau}/2} &\approx D_{x,x} + \langle\, \mathcal{J}_{\mathbf{t}}D_{\mathbf{x}+\mathbf{t}/2,\mathbf{x}-\mathbf{t}/2}\big|_{\mathbf{t}=\mathbf{0}}, \boldsymbol{\tau}\rangle + \boldsymbol{\tau}^\top\, \mathcal{H}_{\mathbf{t}}D_{\mathbf{x}+\mathbf{t}/2,\mathbf{x}-\mathbf{t}/2}\big|_{\mathbf{t}=\mathbf{0}}\,\boldsymbol{\tau} \qquad (18)\\ &= \boldsymbol{\tau}^\top\, \mathcal{H}_{\mathbf{t}}D_{\mathbf{x}+\mathbf{t}/2,\mathbf{x}-\mathbf{t}/2}\big|_{\mathbf{t}=\mathbf{0}}\,\boldsymbol{\tau} \qquad (19)\\ &= \boldsymbol{\tau}^\top\boldsymbol{\Lambda}_{\mathbf{x}}\boldsymbol{\tau}, \qquad (20)\\ U_{\mathbf{x}+\boldsymbol{\tau}/2,\mathbf{x}-\boldsymbol{\tau}/2} &\approx U_{\mathbf{x},\mathbf{x}} + \langle\, \mathcal{J}_{\mathbf{t}}U_{\mathbf{x}+\mathbf{t}/2,\mathbf{x}-\mathbf{t}/2}\big|_{\mathbf{t}=\mathbf{0}}, \boldsymbol{\tau}\rangle \qquad (21)\\ &= \langle\, \mathcal{J}_{\mathbf{t}}U_{\mathbf{x}+\mathbf{t}/2,\mathbf{x}-\mathbf{t}/2}\big|_{\mathbf{t}=\mathbf{0}}, \boldsymbol{\tau}\rangle \qquad (22)\\ &= \langle\boldsymbol{\xi}_{\mathbf{x}},\boldsymbol{\tau}\rangle. \qquad (23)\end{aligned}$$

The linearization gives an approximate of kernel values where $\mathbf{x}$ and $\boldsymbol{\tau}$ are separate:

$$\begin{aligned} k(\mathbf{x}+\boldsymbol{\tau}/2,\mathbf{x}-\boldsymbol{\tau}/2) &\approx \sigma_{\mathbf{x}\mathbf{x}}\exp\left(-\frac{1}{2}\boldsymbol{\tau}^\top\boldsymbol{\Lambda}_{\mathbf{x}}\boldsymbol{\tau}+\imath\langle\boldsymbol{\xi}_{\mathbf{x}},\boldsymbol{\tau}\rangle\right) \qquad (24)\\ &= \widehat{k}(\mathbf{x}+\boldsymbol{\tau}/2,\mathbf{x}-\boldsymbol{\tau}/2), \qquad (25)\\ W(\mathbf{x},\boldsymbol{\omega}) &= \int k(\mathbf{x}+\boldsymbol{\tau}/2,\mathbf{x}-\boldsymbol{\tau}/2)e^{-2\imath\pi\langle\boldsymbol{\omega},\boldsymbol{\tau}\rangle}\,\mathrm{d}\boldsymbol{\tau} \qquad (26)\\ &\approx \int \widehat{k}(\mathbf{x}+\boldsymbol{\tau}/2,\mathbf{x}-\boldsymbol{\tau}/2)e^{-2\imath\pi\langle\boldsymbol{\omega},\boldsymbol{\tau}\rangle}\,\mathrm{d}\boldsymbol{\tau} \qquad (27)\\ &\approx \sigma_{\mathbf{x}\mathbf{x}}\mathcal{N}\left(\boldsymbol{\omega}|\,\frac{\boldsymbol{\xi}_{\mathbf{x}}}{2\pi},\frac{\boldsymbol{\Lambda}_{\mathbf{x}}}{2\pi^2}\right) \qquad (28)\\ &= \widehat{W}(\mathbf{x},\boldsymbol{\omega}), \qquad (29)\end{aligned}$$

which gives the spectrogram as the approximate Wigner distribution function. The rest of this subsection derives the corresponding $\boldsymbol{\xi}_{\mathbf{x}}$ and $\boldsymbol{\Lambda}_{\mathbf{x}}$ for some GP models.

2.1 The non-stationary quadratic (NSQ) kernel

The NSQ kernel [2] is a special case of CSK:

$$k_{NS}(\mathbf{x}_i, \mathbf{x}_j) = \frac{|\boldsymbol{\Sigma}_i|^{1/4}|\boldsymbol{\Sigma}_j|^{1/4}}{|(\boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_j)/2|^{1/2}} e^{-\frac{D_{ij}}{2}}, \tag{30}$$

$$D_{ij} = (\mathbf{x}_i - \mathbf{x}_j)^\top (\boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_j)^{-1} (\mathbf{x}_i - \mathbf{x}_j), \boldsymbol{\Sigma}_i \succeq \mathbf{0}, \tag{31}$$

$$U_{ij} \equiv 0. \tag{32}$$

It is straightforward that $\boldsymbol{\xi}_\mathbf{x} = \mathbf{0}$. Now we derive $\boldsymbol{\Lambda}_\mathbf{x}$:

$$\mathcal{H}_\mathbf{t} D_{\mathbf{x}+\mathbf{t}/2, \mathbf{x}-\mathbf{t}/2} = \mathcal{H}_\mathbf{t} \mathbf{t}^\top \left(\boldsymbol{\Sigma}_{\mathbf{x}+\mathbf{t}/2} + \boldsymbol{\Sigma}_{\mathbf{x}-\mathbf{t}/2}\right)^{-1} \mathbf{t} \tag{33}$$

$$= (\boldsymbol{\Sigma}_{\mathbf{x}+\mathbf{t}/2} + \boldsymbol{\Sigma}_{\mathbf{x}-\mathbf{t}/2})^{-1}. \tag{34}$$

Other terms involving the Hessian operator are $\mathbf{0}$ because of our assumptions. When $\mathbf{t} = \mathbf{0}$, we get $\boldsymbol{\Lambda}_\mathbf{x} = \frac{1}{2}\boldsymbol{\Sigma}_\mathbf{x}^{-1}$.

2.2 The generalized spectral mixture (GSM) kernel

The GSM kernel [3] augments the NSQ kernel with a cosine term:

$$k_{GSM}(\mathbf{x}_i, \mathbf{x}_j) = k_{NS}(\mathbf{x}_i, \mathbf{x}_j) \exp\left(\imath U_{ij}\right), \tag{35}$$

$$U_{ij} = \langle \boldsymbol{\mu}_i, \mathbf{x}_i \rangle - \langle \boldsymbol{\mu}_j, \mathbf{x}_j \rangle. \tag{36}$$

The lengthscale function conforms to the NSQ kernel: $\boldsymbol{\Lambda}_\mathbf{x} = \frac{1}{2}\boldsymbol{\Sigma}_\mathbf{x}^{-1}$. The frequencies are derived as:

$$\mathcal{J}_\mathbf{t} U_{\mathbf{x}+\mathbf{t}/2, \mathbf{x}-\mathbf{t}/2}\big|_{\mathbf{t}=\mathbf{0}} = \mathcal{J}_\mathbf{t} \left[\boldsymbol{\mu}_{\mathbf{x}+\mathbf{t}/2}^\top (\mathbf{x} + \mathbf{t}/2) - \boldsymbol{\mu}_{\mathbf{x}-\mathbf{t}/2}^\top (\mathbf{x} - \mathbf{t}/2)\right] \tag{37}$$

$$= \mathcal{J}_\mathbf{t} \left[\left(\boldsymbol{\mu}_{\mathbf{x}+\mathbf{t}/2} - \boldsymbol{\mu}_{\mathbf{x}-\mathbf{t}/2}\right)^\top \mathbf{x} + \left(\frac{\boldsymbol{\mu}_{\mathbf{x}+\mathbf{t}/2} + \boldsymbol{\mu}_{\mathbf{x}-\mathbf{t}/2}}{2}\right)^\top \mathbf{t}\right]\Bigg|_{\mathbf{t}=\mathbf{0}} \tag{38}$$

$$= \boldsymbol{\mu}_\mathbf{x} + (\mathcal{J}_\mathbf{x} \boldsymbol{\mu}_\mathbf{x})\, \mathbf{x}. \tag{39}$$

2.3 The Deep Gaussian process (DGP)

The kernel of DGP is a SE kernel with input warping $\mathbf{f}_i = \mathbf{f}(\mathbf{x}_i)$. Formally,

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{1}{2}(\mathbf{f}_i - \mathbf{f}_j)^\top \boldsymbol{\Lambda} (\mathbf{f}_i - \mathbf{f}_j)\right), \tag{40}$$

$$D_{ij} = (\mathbf{f}_i - \mathbf{f}_j)^\top \boldsymbol{\Lambda} (\mathbf{f}_i - \mathbf{f}_j). \tag{41}$$

The lengthscale function differ from a constant $\boldsymbol{\Lambda}$ because $\mathbf{f}_i$ is a function of $\mathbf{x}_i$.

$$\boldsymbol{\Lambda}_\mathbf{x} = (\mathcal{J}_\mathbf{x} \mathbf{f})^\top \boldsymbol{\Lambda} \left(\mathcal{J}_\mathbf{x} \mathbf{f}\right). \tag{42}$$

2.4 Equivalence between two DGPs

We provide an additional plot shown in Figure 1, which demonstrates the claim that a DGP is equivalent (up to second-order effects) to a GP with NSQ kernel with lengthscales defined via the derivatives of the input warping.

3 Inference with covariance function DGPs

In this section, we summarize notable earlier works for scalable inference of sparse covariance function DGPs.

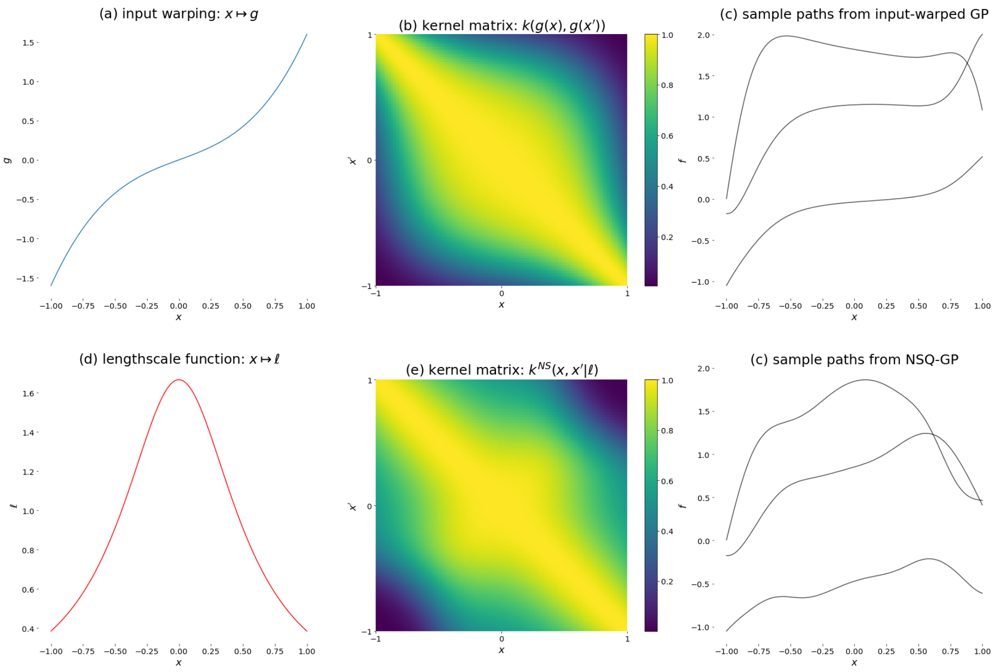


Figure 1: Equivalence between a GP with input warping function g (shown in **(a)**), and a NSQ kernel with corresponding lengthscale (shown in **(d)**). The kernel matrices (**(b)** and **(e)**) and sample paths (**(c)** and **(f)** are notably similar.

3.1 Small-scale datasets: maximum *a-posteriori* and Hamiltonian Monte Carlo

Previous work [2, 3, 4] have studied inference for the functional hyperparameters of nonparametric kernels. A maximum *a-posteriori* framework gets MAP estimates for values of the hyperparameters on the input points:

$$p(\boldsymbol{\theta}|\mathbf{X}) \propto p(\mathbf{X}, \boldsymbol{\theta}) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\boldsymbol{\theta})p(\boldsymbol{\theta}), \tag{43}$$

Maximizing the posterior likelihood (43), we obtain point estimates for $\boldsymbol{\theta}(\mathbf{X})$.

One point estimate with maximum likelihood does not effectively represent the posterior distribution, the Hamiltonian Monte Carlo (HMC) [4, 5] produces samples from $p(\boldsymbol{\theta}|\mathbf{X})$ using Hamiltonian dynamics. MAP and HMC are solid choices for small-scale data, but is infeasible in large-scale setting due to the necessity of inferring $O(N)$ parameters.

3.2 Our extension of the variationally sparse GPs

While sparse GP approximations [6, 7, 8, 9] typically use parametric variational distributions in terms of variational inference. The work of Hensman et al. [10] observes that the optimal form of variational distribution, while intractable, is proportional to an unnormalized likelihood. In the case of covariance function DGPs, it is

$$q^*(\mathbf{u}_f, \mathbf{u}_{\boldsymbol{\theta}}) \propto e^{\mathbb{E}_{p(\mathbf{f}|\mathbf{u}_f, \boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{u}_{\boldsymbol{\theta}})} \log p(\mathbf{y}|\mathbf{f})p(\mathbf{u}_f|\boldsymbol{\theta})} p(\mathbf{u}_{\boldsymbol{\theta}}).$$

HMC methods is suitable for extracting samples from a distribution with an intractable normalizing constant, as $\nabla \log p(\mathbf{u}) = \nabla \log p'(\mathbf{u})$ whenever $p \propto p'$. Therefore, we have

$$U(\mathbf{u}_f, \mathbf{u}_{\boldsymbol{\theta}}) = \log q^*(\mathbf{u}_f, \mathbf{u}_{\boldsymbol{\theta}}) + \log C = \mathbb{E}_{p(\mathbf{f}|\mathbf{u}_f, \boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{u}_{\boldsymbol{\theta}})} \log p(\mathbf{y}|\mathbf{f})p(\mathbf{u}_f|\boldsymbol{\theta}) + \log p(\mathbf{u}_{\boldsymbol{\theta}}). \tag{44}$$

While it is still intractable to directly compute the expectation term, Hensman et al. [10] proposed to sum the expectation terms over data points, and obtain one dimensional integrals tractable in the case of Gaussian or Poisson likelihoods, or approximated via quadrature methods. In the work of deep GPs, Salimbeni and Deisenroth [9] and Havasi et al. [11] observed that the sum can be further approximated with a random subsample of the data and Monte Carlo sums $\tilde{\boldsymbol{\theta}} \sim p(\boldsymbol{\theta}|\mathbf{u}_{\boldsymbol{\theta}})$, and thus can be approximated with stochastic gradient HMC [12, 13].

3.3 Moving window MCEM

For hyperparameter learning along with SGHMC, we use the moving window Monte Carlo expectation maximization (MCEM) [11], an extension of the expectation maximization algorithm to learn maximum likelihood estimate of hyperparameters. The moving window MCEM keeps track of a fixed-length window of recent samples from HMC sampler, and each optimization action of hyperparameters is done given the parameters randomly drawn from the window. This algorithm has shown better optimization of hyperparameter for deep GP models [11].

In practice, we use the auto-tuning approach given by [13], which has shown to work well for Bayesian neural networks, as well as compositional DGPs [11]. While we cannot directly compute the conditional likelihood $p(\mathbf{y}, \mathbf{u}_f, \mathbf{u}_{\boldsymbol{\theta}}|\xi)$, where λ denotes all hyperparameters, we can optimize the lower bound of its logarithm

$$\log p(\mathbf{y}, \mathbf{u}_f, \mathbf{u}_{\boldsymbol{\theta}}|\lambda) = \log \mathbb{E}_{p(\boldsymbol{\theta}|\mathbf{u}_{\boldsymbol{\theta}})} p(\mathbf{y}, \boldsymbol{\theta}, \mathbf{u}_f, \mathbf{u}_{\boldsymbol{\theta}}|\lambda) \geq \mathbb{E}_{p(\boldsymbol{\theta}|\mathbf{u}_{\boldsymbol{\theta}})} \log p(\mathbf{y}, \boldsymbol{\theta}, \mathbf{u}_f, \mathbf{u}_{\boldsymbol{\theta}}|\lambda). \tag{45}$$

The right hand side of (45) can be approximated by same Monte Carlo samples used for approximating the HMC objective (44).

4 Experiment details

The baseline results reported in our paper are implemented based on the works of variational sparse Gaussian process regression [7], stochastic variational Gaussian processes [8], spectral mixture kernel [1], harmonizable mixture kernel [14] and stochastic gradient HMC inference for compositional deep GPs [10, 11].

4.1 Solar irradiance

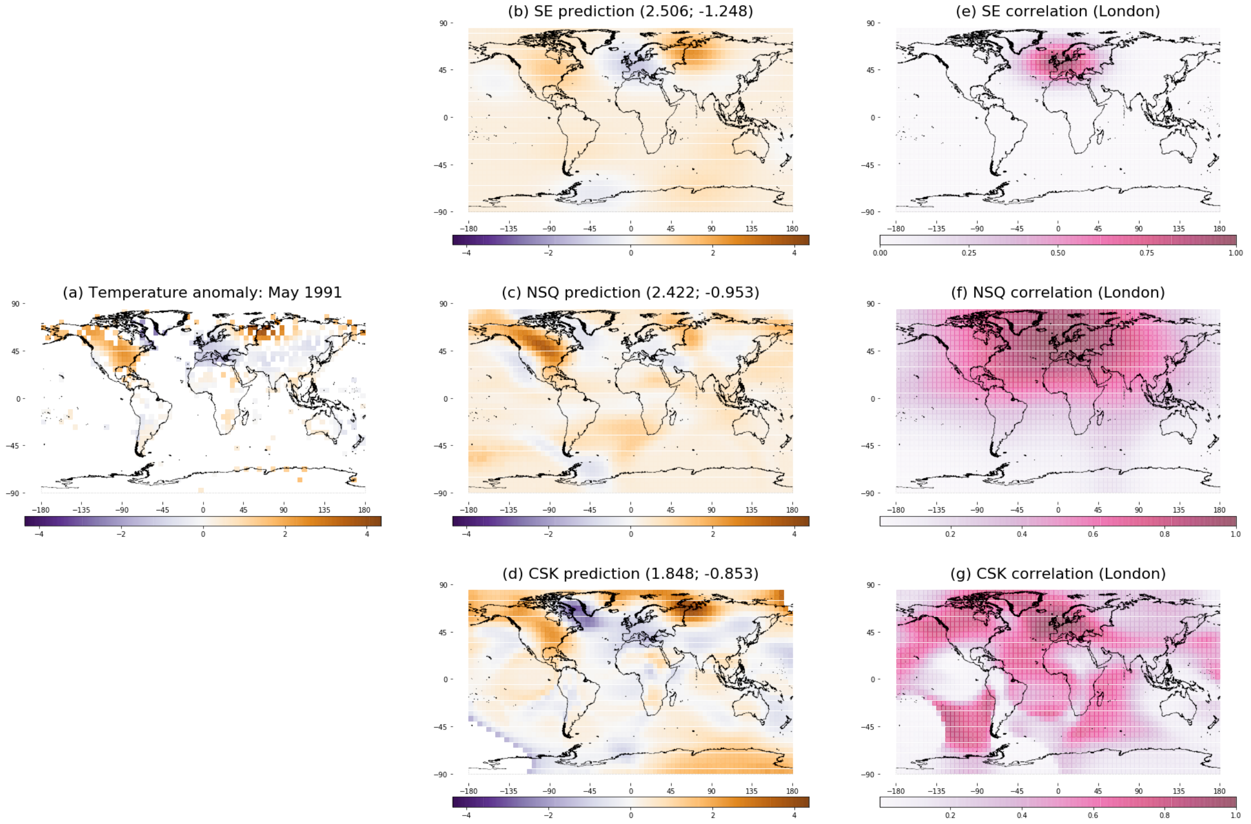
With the solar irradiance dataset [15][1], we follow the same partition of training and test set as Gal and Turner [16] and Hensman et al. [17]. This 1-dimensional dataset has 281 training points and 110 test points. We standardize both $\mathbf{X}$ and $\mathbf{Y}$ before the regression.

We use 3 components to fit the SM kernel as well as CSK. The HMK kernel has 6 centroids, each with three frequency terms. We initialize all parameters (including kernel hyperparameters for parametric kernels, kernel hyperparameters for latent GPs of NSQ and CSK, variational parameters, inducing points, and parameters sampled from HMC) at random. And we do 5000 iterations for each models, with a learning rate of 10^{-3}.

4.2 Air temperature anomaly: additional experiments

The DGP-NSQ model is a good frame of reference to compare against the CSK result, and we visualize the predictive mean and correlation with the coordinates of London, shows in Figure 2. While NSQ is notably more flexible with better predictive performance compared to the SE kernel, the correlation is notably monotonic, and thus cannot capture the intricate correlation pattern of CSK.

[1] `https://github.com/jameshensman/VFF/blob/master/experiments/solar/solar_data.txt`



(b) SE prediction (2.506; -1.248)

(e) SE correlation (London)

(a) Temperature anomaly: May 1991

(c) NSQ prediction (2.422; -0.953)

(f) NSQ correlation (London)

(d) CSK prediction (1.848; -0.853)

(g) CSK correlation (London)

Figure 2: Air temperature anomaly dataset. **(a)** demonstrates the temperature anomaly readings from May 1991. **(b)**, **(c)** and **(d)** display the posterior predictive mean of May 1991 on a grid of global locations, with the numbers in parentheses denoting mean squared error (MSE) and mean log-likelihood, respectively. **(e)**, **(f)** and **(g)** depict the correlation between London and other geographical locations: it is worth noting that the SE kernel **(e)** only captures positive correlation on a small elliptical region, and NSQ kernel **(f)** captures an irregularly shaped pattern monotonic with respect to the distance from London.

4.3 New York Yellow Taxi dataset

Due to limited time and CSK's sensitivity to higher dimensions, we use 6 subsets of the New York Taxi dataset[2], randomly subsampling 25% of all taxi trips over the span of two months. The dataset has 5 dimensions (longitudes and latitudes of the pickup and dropoff locations, and the date and time of the beginning of the taxi trip). We combine the date and time to a "date value". The training set includes 100671 data points, and the test set includes 45260 data points. We also normalize both input and output into zero mean and unit standard deviation before regression.

We use minibatch of 10000 given the medium-large scale of data, with baselines stochastic variational GP [8] for GPs with parametric kernels (SE kernel and SM kernel), and SG-HMC [11] for compositional DGP, and the proposed SG-HMC for GP with NSQ kernel and CSK. For CSK and SM, we use 4 frequency components. The parameters are initialized at random, and the inducing point locations are randomly initialized with a K-Means algorithm.

[2] https://www.kaggle.com/c/nyc-taxi-trip-duration/overview

References

[1] A. G. Wilson and R. Adams. Gaussian process kernels for pattern discovery and extrapolation. In *International Conference on Machine Learning*, pages 1067–1075, 2013.

[2] C. J. Paciorek and M. J. Schervish. Nonstationary covariance functions for Gaussian process regression. In *Advances in Neural Information Processing Systems*, pages 273–280, 2004.

[3] S. Remes, M. Heinonen, and S. Kaski. Non-stationary spectral kernels. In *Advances in Neural Information Processing Systems*, pages 4642–4651, 2017.

[4] M. Heinonen, H. Mannerström, J. Rousu, S. Kaski, and H. Lähdesmäki. Non-stationary Gaussian process regression with Hamiltonian Monte Carlo. In *Artificial Intelligence and Statistics*, pages 732–740, 2016.

[5] R. M. Neal. Probabilistic inference using Markov Chain Monte Carlo methods. Technical report, University of Toronto, 1993.

[6] E. Snelson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. In *Advances in neural information processing systems*, pages 1257–1264, 2006.

[7] M. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *Artificial Intelligence and Statistics*, pages 567–574, 2009.

[8] J. Hensman, N. Fusi, and N. D. Lawrence. Gaussian processes for big data. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence, UAI 2013, Bellevue, WA, USA, August 11-15, 2013*, 2013.

[9] H. Salimbeni and M. Deisenroth. Doubly stochastic variational inference for deep Gaussian processes. In *Advances in Neural Information Processing Systems*, pages 4588–4599, 2017.

[10] J. Hensman, A. G. Matthews, M. Filippone, and Z. Ghahramani. MCMC for variationally sparse Gaussian processes. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1648–1656. Curran Associates, Inc., 2015.

[11] M. Havasi, J. M. Hernández-Lobato, and J. J. Murillo-Fuentes. Inference in deep Gaussian processes using stochastic gradient Hamiltonian Monte Carlo. In *Advances in Neural Information Processing Systems*, pages 7517–7527, 2018.

[12] T. Chen, E. Fox, and C. Guestrin. Stochastic gradient Hamiltonian Monte Carlo. In *International Conference on Machine Learning*, pages 1683–1691, 2014.

[13] J. T. Springenberg, A. Klein, S. Falkner, and F. Hutter. Bayesian optimization with robust Bayesian neural networks. In *Advances in Neural Information Processing Systems*, pages 4134–4142, 2016.

[14] Z. Shen, M. Heinonen, and S. Kaski. Harmonizable mixture kernels with variational Fourier features. In *Artificial Intelligence and Statistics*, pages 3273–3282, 2019.

[15] J. Lean. Solar irradiance reconstruction. *IGBP PAGES/World Data Center for Paleoclimatology Data Contribution Series*, 35, 2004.

[16] Y. Gal and R. Turner. Improving the Gaussian process sparse spectrum approximation by representing uncertainty in frequency inputs. In *International Conference on Machine Learning*, pages 655–664, 2015.

[17] J. Hensman, N. Durrande, and A. Solin. Variational fourier features for Gaussian processes. *Journal of Machine Learning Research*, 18:151–1, 2017.