SPARSE GAUSSIAN PROCESS AUDIO SOURCE SEPARATION USING SPECTRUM PRIORS IN THE TIME-DOMAIN

Pablo A. Alvarado[⋆] *Mauricio A. Álvarez*[§∗] *Dan Stowell*[⋆†]

[⋆]Centre for Digital Music, Queen Mary University of London, London, UK
[§]Department of Computer Science, The University of Sheffield, Sheffield, UK

ABSTRACT

Gaussian process (GP) audio source separation is a time-domain approach that circumvents the inherent phase approximation issue of spectrogram based methods. Furthermore, through its kernel, GPs elegantly incorporate prior knowledge about the sources into the separation model. Despite these compelling advantages, the computational complexity of GP inference scales cubically with the number of audio samples. As a result, source separation GP models have been restricted to the analysis of short audio frames. We introduce an efficient application of GPs to time-domain audio source separation, without compromising performance. For this purpose, we used GP regression, together with spectral mixture kernels, and variational sparse GPs. We compared our method with LD-PSDTF (positive semi-definite tensor factorization), KL-NMF (Kullback-Leibler non-negative matrix factorization), and IS-NMF (Itakura-Saito NMF). Results show that the proposed method outperforms these techniques.

Index Terms— Time-domain source separation, Gaussian processes, spectral mixture kernels, variational inference.

1. INTRODUCTION

Single-channel audio source separation is a central problem in signal processing research. Here, the task is to estimate a certain number of latent signals or *sources* that were mixed together in one recorded *mixture* signal [1]. State of the art time-frequency methods for source separation include deep neural networks [2], non-negative matrix factorisation (NMF) [3], and probabilistic latent component analysis (PLCA) [4]. These approaches decompose the power spectrogram of the mixture into elementary components. Then, the components are used to calculate the individual source-spectrograms. Time-frequency methods often arbitrarily discard phase information. As a result, the phase of each source-spectrogram must be approximated, corrupting the reconstructed sources.

In contrast, time-domain source separation approaches can avoid the phase approximation issue of time-frequency methods [5, 6]. For example, Yoshii et al. [7] reconstructed source signals from the mixture waveform directly in the time domain. To this end, Gaussian processes (GPs) were used to predict each source waveform. GPs are probability distributions over functions [8]. A Gaussian process is completely defined by a mean function, and a kernel or covariance function. In fact, the kernel determines the properties of the functions sampled from a GP. A particularly influential work in time domain approaches is Liutkus et al. [1], who first formulated source separation as a GP regression task.

Although source separation Gaussian process (SSGP) models circumvent phase approximation, the computational complexity of GP inference scales cubically with the number of audio samples. Hence, different approximate techniques have been proposed to make the separation tractable. For instance, various authors partitioned the mixture signal into independent frames [1, 7]. Further, approximate inference in the frequency domain was used to learn model hyperparameters [1]. Alternatively, Adam et al. [9] recently proposed to use variational sparse GPs for source separation, however audio signals were beyond the scope of their study. Variational approaches rely on a set of *inducing variables* to build a low-rank approximation of the full covariance matrix. Here, the approximate distribution and hyperparameters are learned together by maximising a lower bound of the true marginal likelihood [10]. Moreover, variational inference has allowed the application of GPs models to large datasets [11, 12].

Although the kernel selection in SSGP models determines the properties of sources, only standard covariance functions have been used so far. For example, Adam et al. [9] considered stationarity, smoothness and periodicity, using *exponentiated quadratic* times *cosine* kernels. *Standard periodic* kernels [13] were applied in [1]. These kernels assume that the source spectrum is composed of a fundamental frequency and perfect harmonics. However, real audio signals have more intricate spectra [14], and so separating audio sources requires more flexible covariance functions. One such covariance, the spectral mixture (SM) kernel [15], is intended for intricate spectrum patterns. SM kernels approximate the spectral density of any stationary covariance function, using a Gaussian mixture. Alternatively, non-parametric kernels are implicitly considered when the covariance matrix of each source is directly optimised by maximum likelihood [7]. However, that

[∗]Partially financed by the EPSRC Research Project EP/N014162/1.
[†]Supported by EPSRC Fellowship EP/L020505/1.

978-1-5386-4658-8/18/$31.00 ©2019 IEEE

study did not contemplate variational sparse GPs. To our knowledge, it has not been determined whether incorporating SM kernels together with variational sparse GPs into source separation models leads to more efficient and accurate audio source reconstructions.

In this paper we introduce a method that combines GP regression [8, 1], spectral mixture kernels [15], and variational sparse GPs [10]. We consider the mixture data as noisy observations of a function of time, composed as the sum of a known number of sources. Further, we assume that each source follows a different GP with a distinctive spectral mixture kernel. In addition, we adapt the kernels to reflect prior knowledge about the typical spectral content of each source. Also, we frame the mixture data, and for every frame we maximize a variational lower bound of the true marginal likelihood to learn the hyperparameters that control the amplitude of each source (variances). Finally, to separate the sources, we use the learned priors to calculate the true posterior over each source.

2. GAUSSIAN PROCESS SOURCE SEPARATION

We notate the mixture data as $\mathbf{y} = [y_i]_{i=1}^{n}$ at time instants $\mathbf{t} = [t_i]_{i=1}^{n}$. As mentioned previously, we consider each mixture audio sample y_i as an observation of a mixture function $f(t)$ corrupted by independent Gaussian noise. Further, we assume $f(t)$ as the sum of J independent source functions $\{s_j(t)\}_{j=1}^{J}$. These functions represent the sources to be reconstructed. Each source $s_j(t)$ follows a different GP with zero mean, and a distinctive spectral mixture kernel. That is, $y_i = f(t_i) + \epsilon_i$, where $f(t) = \sum_{j=1}^{J} s_j(t)$, and

$$s_j(t) \sim \mathcal{GP}\left(\, 0,\; k_j(t,t')\, \right) \quad \text{for } j = 1, 2, \ldots, J. \tag{1}$$

Here, the noise follows $\epsilon_i \sim \mathcal{N}(0,\ \nu^2)$, with variance ν^2. The kernel for the j-th source is represented by $k_j(t,t')$ (introduced shortly in section 2.1). In addition, it is a well known property that the sum of GPs is also a Gaussian process [8]. Therefore, the mixture function follows

$$f(t) \sim \mathcal{GP}\left(0,\ \sum_{j=1}^{J} k_j(t,t')\right), \tag{2}$$

where its kernel is the sum of source kernels, i.e. $k_f(t,t') = \sum_{j=1}^{J} k_j(t,t')$. We focus only on predicting the mixture function (2) as well as the sources (1) evaluated at $\mathbf{t}$.

Any finite set of evaluations of a GP function follows a multivariate normal distribution [8]. Therefore, the prior over the mixture function, and each source evaluated at $\mathbf{t}$, correspond to $\mathbf{f} \sim \mathcal{N}(\mathbf{0},\ \mathbf{K}_f)$, and $\mathbf{s}_j \sim \mathcal{N}\left(\mathbf{0},\ \mathbf{K}_{s_j}\right)$ respectively, where the column vectors $\mathbf{f} = [f(t_1), \ldots, f(t_n)]^\top$, $\mathbf{s}_j = [s_j(t_1), \ldots, s_j(t_n)]^\top$, and the covariance matrix $\mathbf{K}_f = \sum_{j=1}^{J} \mathbf{K}_{s_j}$. The matrices $\left\{\mathbf{K}_{s_j}\right\}_{j=1}^{J}$ are computed by evaluating the source kernels at all pairs of time instants. That is, $\mathbf{K}_{s_j}[l, l'] = k_j(t_l, t_{l'})$ for $l = 1, 2, \ldots, n$, and $l' = 1, 2, \ldots, n$. Also, when a Gaussian likelihood is assumed, the priors are conjugate to the likelihood [8]. Hence, the posterior distributions are also Gaussian. That is,

$$\mathbf{y} \mid \mathbf{f} \sim \prod_{i=1}^{n} \mathcal{N}\left(y_i \mid f_i,\ \nu^2\right), \tag{3}$$

$$\mathbf{f} \mid \mathbf{y} \sim \mathcal{N}\left(\mathbf{f}\ \mid \mathbf{K}_f^\top \mathbf{H}^{-1}\mathbf{y},\ \ \hat{\mathbf{K}}_f\right), \tag{4}$$

$$\mathbf{s}_j \mid \mathbf{y} \sim \mathcal{N}\left(\mathbf{s}_i \mid \mathbf{K}_{s_j}^\top \mathbf{H}^{-1}\mathbf{y},\ \ \hat{\mathbf{K}}_{s_j}\right). \tag{5}$$

Here, the likelihood (3) factorizes across the mixture data, and the posterior over the mixture function (4) has covariance matrix $\hat{\mathbf{K}}_f = \mathbf{K}_f - \mathbf{K}_f^\top \mathbf{H}^{-1}\mathbf{K}_f$. Also, the posterior distribution over the i-th source (5) has covariance matrix $\hat{\mathbf{K}}_{s_j} = \mathbf{K}_{s_j} - \mathbf{K}_{s_j}^\top \mathbf{H}^{-1}\mathbf{K}_{s_j}$, where the matrix $\mathbf{H} = \mathbf{K}_f + \nu^2\mathbf{I}$, and $\mathbf{I}$ is the identity matrix. Further, the model hyperparameters are usually learned by maximizing the log-marginal likelihood

$$\log p(\mathbf{y}) = -\frac{1}{2}\left[\mathbf{y}^\top \mathbf{H}^{-1}\mathbf{y} + \log|\mathbf{H}| + n\log 2\pi\right], \tag{6}$$

where $\mathbf{H}$ needs to be inverted.

Although the source separation GP model introduced so far is elegant, its application to large audio signals becomes intractable. This is because the computational complexity of GP inference scales cubically with the number of audio samples. Specifically, learning the hyperparameters by maximizing the true marginal likelihood (6) is computationally demanding, as it requires the inversion of a $n \times n$ matrix. To overcome the limitations imposed by matrix inversion, we instead maximized a variational lower bound of the true marginal likelihood (6) (introduced shortly in section 2.2). In addition, we divided the mixture data into overlapping frames of size $\hat{n} \ll n$. Finally, to reconstruct the sources, we used the hyperparameters learned for each frame to calculate the true posterior distribution over the sources (eq. (5)). The rest of this section is structured as follows. Section 2.1 introduces the spectral mixture kernel used for each source. Then, section 2.2 presents the lower bound of the true marginal likelihood we maximized for learning the hyperparameters.

2.1. Spectral mixture kernels for isolated sources

The kernel $k_j(t,t')$ in (1) determines the properties of each source $s_j(t)$, that is, smoothness, stationarity, and more importantly, its spectrum. To model the typical spectral content of each isolated source, we used spectral mixture kernels [15]. These kernels approximate the spectral density of any stationary covariance function using a Gaussian mixture. Further, Alvarado et al. [16] assumed a Lorentzian mixture instead, resulting in the Matérn-$1/2$ spectral mixture (MSM) kernel

$$k_j(\tau) = \sigma_j^2 \exp\left(-\frac{\tau}{\ell_j}\right) \times \sum_{d=1}^{D} \alpha_{jd}^2 \cos(\omega_{jd}\, \tau), \tag{7}$$

where $\tau = |t - t'|$, the set of parameters $\left\{\alpha_{jd}^2,\ \omega_{jd}\right\}_{d=1}^{D}$ controls the energy distribution throughout all the harmonics/partials of the j-th source spectrum. In addition, the variance σ_j^2 controls the source amplitude, whereas the lengthscale ℓ_j determines how fast $s_j(t)$ evolves in time. We grouped all the kernel parameters in the set $\boldsymbol{\theta}_j = \left\{\sigma_j^2,\ \ell_j,\ \left\{\alpha_{jd}^2,\ \omega_{jd}\right\}_{d=1}^{D}\right\}$. We fitted a MSM kernel (7) to the spectrum of every source. For this purpose, we used training data consisting of one audio recording of each isolated source. We denoted the training data as $\left\{\mathbf{g}^{(j)}\right\}_{j=1}^{J}$, where $\mathbf{g}^{(j)} = [g^{(j)}(x_i)]_{i=1}^{\tilde{n}}$ is the training data vector for the j-th source, and $\mathbf{x} = [x_i]_{i=1}^{\tilde{n}}$ is the corresponding time vector. In addition, because only one single realization $\mathbf{g}^{(j)}$ was available for each source in $\{s_j(t)\}_{j=1}^{J}$, we assumed the sources to be covariance-ergodic processes with zero mean [17, 18, 19]. Therefore, their covariances $\{C_j(\lambda)\}_{j=1}^{J}$ were estimated as the time average

$$C_j(\hat{\tau}) = \frac{1}{T}\int_0^T g^{(j)}(x+\hat{\tau})\, g^{(j)}(x)\, \mathrm{d}x. \tag{8}$$

Here, T denotes the size (in seconds) of the window used to compute the correlation. We used the discrete version of eq. (8). Finally, for every source we then minimized the mean square error (MSE) between the covariance estimator (8) and the corresponding MSM kernel (7). That is,

$$L(\boldsymbol{\theta}_j) = \frac{1}{N_c}\sum_{i=1}^{N_c}\left[k_j(\hat{\tau}_i) - C_j(\hat{\tau}_i)\right]^2, \tag{9}$$

where N_c is the number of points where (8) was approximated, and $\boldsymbol{\theta}_j$ is the set of kernel parameters in (7).

2.2. Preprocessing and inference

To reduce the computational complexity of learning the hyperparameters by maximizing the true marginal likelihood (6), we divided the mixture data $\{t_i, y_i\}_{i=1}^{n}$ into W overlapping frames of size $\hat{n} \ll n$. Therefore, the set of frames corresponded to $\left\{\hat{\mathbf{t}}^{(w)}, \hat{\mathbf{y}}^{(w)}\right\}_{w=1}^{W}$. In addition, for each mixture frame $\hat{\mathbf{y}}^{(w)}$, we instead maximized the lower bound of the true marginal likelihood, proposed by Titsias [10] for variational sparse GPs. This method depends on a smaller set of *inducing variables* $\mathbf{u} \in \mathbb{R}^m$, where $m \leq \hat{n}$. The set $\mathbf{u}$ represents the values of the function $f(t)$ (eq. (2)) evaluated at a set of *inducing points* $\mathbf{z} = [z_i]_{i=1}^{m}$. Thus, $\mathbf{u} = \left[f(z_1), \ldots, f(z_m)\right]^{\top}$. The inducing points $\mathbf{z}$ lie on the same domain as $\mathbf{t}$, i.e. time. Moreover, the inducing points, together with the model hyperparameters are learned by minimizing the Kullback-Leibler (KL) divergence between the Gaussian approximate distribution $q(\mathbf{u})$, and the true posterior $p(\hat{\mathbf{f}} \mid \hat{\mathbf{y}}^{(w)})$. This approach leads to the following bound

$$\mathcal{L} \triangleq \log\mathcal{N}\left(\hat{\mathbf{y}}^{(w)} \middle|\, \mathbf{0},\ \mathbf{Q}_{\hat{n}\hat{n}} + \nu^2\mathbf{I}\right) - \frac{1}{2\nu^2}\mathrm{tr}\left(\mathbf{K}_{\hat{n}\hat{n}} - \mathbf{Q}_{\hat{n}\hat{n}}\right), \tag{10}$$

where the matrix $\mathbf{Q}_{\hat{n}\hat{n}} = \mathbf{K}_{\hat{n}m}\mathbf{K}_{mm}^{-1}\mathbf{K}_{m\hat{n}}$. Here, the cross covariance $\mathbf{K}_{\hat{n}m}[i,j] = k_f(t_i^{(w)}, z_j)$. Similarly, $\mathbf{K}_{mm}[i,j] = k_f(z_i, z_j)$. Where $t_i^{(w)} = \mathbf{t}^{(w)}[i]$. Recall that $k_f(t, t')$ is the kernel of the mixture function (eq. (2)). In brief, the computational complexity of learning hyperparameters in each frame was reduced from $\mathcal{O}(\hat{n}^3)$, to $\mathcal{O}(\hat{n}m^2)$.

Method	SDR	SIR	SAR	Opt. time
KL-NMF	17.7	22.2	19.7	–
IS-NMF	19.1	24.0	21.0	–
LD-PSDTF	23.0	27.7	25.1	–
SSGP (proposed)	**24.1**	**31.4**	**25.1**	**5.33**
SSGP-full	22.9	22.3	24.6	284.2

Table 1. Separation metrics (dB). Optimization time (min).

3. EXPERIMENTAL EVALUATION

We tested the proposed SSGP method on the same dataset analysed in [7]. That is, three different mixture audio signals sampled at 16KHz, corresponding to piano, electric guitar, and clarinet. Each mixture lasts 14 seconds, and consists of the following sequence of music notes (C4, E4, G4, C4+E4, C4+G4, E4+G4, and C4+E4+G4). Thus, for each mixture, the aim was to reconstruct three source signals, each with a corresponding note, C4, E4, and G4. The metrics used to measure the separation performance were: source to distortion ratio (SDR), source to interferences ratio (SIR), source to artifacts ratio (SAR) [20], and root mean square error (RMSE). We compared with LD-PSDTF (positive semi-definite tensor factorization), KL-NMF (Kullback-Leibler NMF), and IS-NMF (Itakura-Saito NMF) with rank three [7]. The code was implemented using GPflow [21].

We determined the performance of the proposed method in mixtures of three sources. That is, $J = 3$ in eq. (2). To this end, we first divided the mixtures into frames of 125 milliseconds ($\hat{n} = 2001$) with 50% overlap, and initialized the kernel for each source (eq. (7) with $D = 15$), by minimizing eq. (9). Then, for each mixture frame, we maximized eq. (10) to learn the variance of each source, i.e., $\left\{\sigma_j^2\right\}_{j=1}^{J}$. We used two separate criteria to select $\mathbf{z}$: either the inducing points were located at the extrema of the mixture data (sparse GP), or the inducing points were equal to the time vector (full GP). We compared the time required for learning the hyperparameters in these two scenarios. Finally, we used eq. (5), and the learned hyperparameters to calculate the true posterior over each source $p\left(\mathbf{s}_i^{(w)} | \mathbf{y}^{(w)}\right)$. We recovered the sources applying the *overlap-add* method to the frame-wise predictions

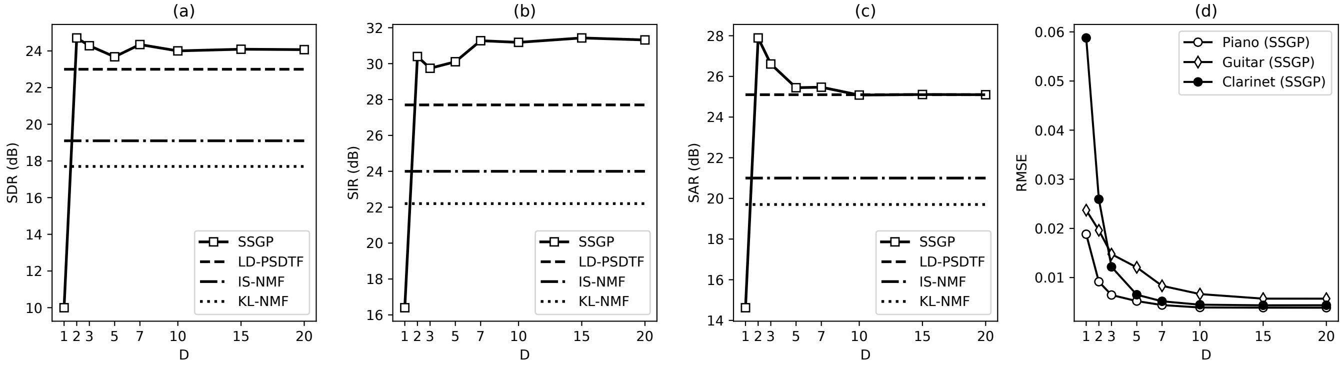

Fig. 1. Source separation metrics. SDR (a), SIR (b), SAR (c), RMSE (d).

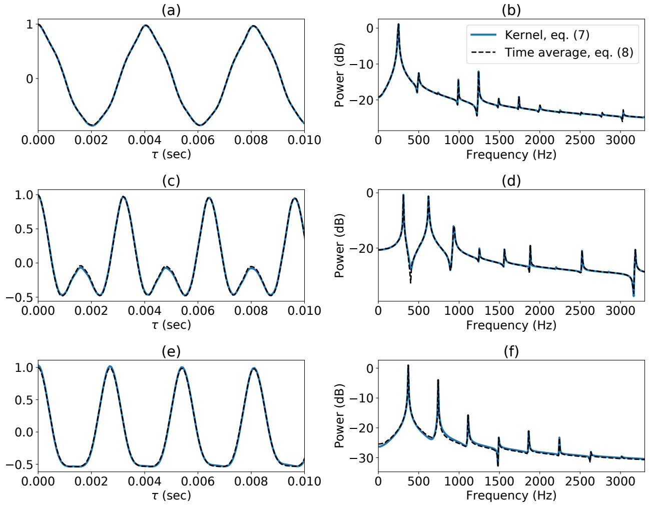

Fig. 2. Kernels learned for each piano source (left column). Corresponding log-spectral density (right column).

[22]. We found that our method (SSGP) presented the highest SDR and SIR metrics, and reduced the optimization time by 98.12% compared to the full GP (Table 1), indicating that our method is efficient, robust to interferences between sources (highest SIR), and it introduces less distortion (highest SDR). Further, we observed that the kernels learned for each source presented distinctive spectral patterns (Fig 2), which demonstrates that SM kernels are appropriate for learning the rich frequency content found in audio sources. Moreover, we observed that the proposed approach reconstructed accurately the sources (Fig 3), showing the variances learned by maximizing the lower bound were consistent with the true sources. In addition, to establish the effect of kernel selection on the separation performance, we carried out the same previous experiment, but changing the number of components D in the kernel eq. (7). We found that SDR, SIR and SAR metrics stabilized when $D > 3$ (Fig. 1(a-c)), indicating that the proposed model is less affected by kernel selection when more than three components are used. Further, RMSE decreased exponentially with D (Fig. 1(d)), suggesting that increasing the number of components in the kernel leads to more accurate waveform reconstructions.

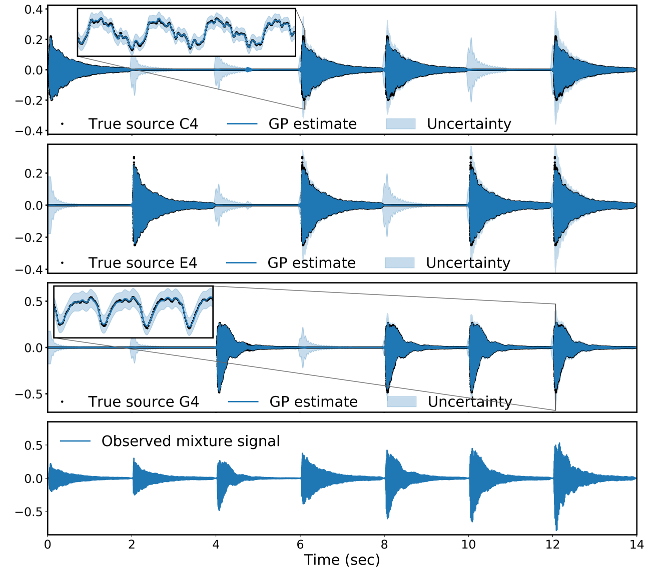

Fig. 3. Source reconstruction on piano mixture signal.

4. CONCLUSIONS

Our findings indicate that combining variational sparse GPs together with SM kernels enables time-domain source separation GP models to reconstruct audio sources in an efficient and informed manner, without compromising performance. Also, RMSE results imply that suitable spectrum priors over the sources are essential to improve source reconstruction. Moreover, SDR, SIR, and SAR results suggest the proposed method can be used for other applications such as multipitch-detection, where low interference between sources (SIR) is more relevant than reconstruction artifacts (SAR). We proposed an alternative method that circumvents phase approximation by addressing audio source separation from a variational time-domain perspective. The code is available at [23].

5. REFERENCES

[1] Antoine Liutkus, Roland Badeau, and Gäel Richard, "Gaussian processes for underdetermined source separation," *IEEE Transactions on Signal Processing*, vol. 59, no. 7, pp. 3155–3167, July 2011.

[2] Daniel Stoller, Sebastian Ewert, and Simon Dixon, "Adversarial semi-supervised audio source separation applied to singing voice extraction," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2391–2395.

[3] Daniel D. Lee and H. Sebastian Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems 13 (NIPS)*, pp. 556–562. MIT Press, 2001.

[4] Paris Smaragdis and Bhiksha Raj, "Shift-invariant probabilistic latent component analysis," Tech. Rep., Mitsubishi Electric Research Laboratories, 2007.

[5] Cédric Févotte and Matthieu Kowalski, "Low-rank time-frequency synthesis," in *Advances in Neural Information Processing Systems 27 (NIPS)*, pp. 3563–3571. Curran Associates, Inc., 2014.

[6] Daniel Stoller, Sebastian Ewert, and Simon Dixon, "Wave-U-Net: A multi-scale neural network for end-to-end source separation," in *International Society for Music Information Retrieval Conference (ISMIR)*, 2018, vol. 19, pp. 334–340.

[7] Kazuyoshi Yoshii, Ryota Tomioka, Daichi Mochihashi, and Masataka Goto, "Beyond NMF: Time-domain audio source separation without phase reconstruction," in *14th International Society for Music Information Retrieval Conference (ISMIR)*, 2013, pp. 369–374.

[8] Carl Edward Rasmussen and Christopher K.I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*, The MIT Press, 2005.

[9] Vincent Adam, James Hensman, and Maneesh Sahani, "Scalable transformed additive signal decomposition by non-conjugate Gaussian process inference," in *26th IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2016, pp. 1–6.

[10] Michalis K. Titsias, "Variational learning of inducing variables in sparse Gaussian processes," in *12th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2009, pp. 567–574.

[11] James Hensman, Nicoló Fusi, and Neil D. Lawrence, "Gaussian processes for big data," in *20th Conference on Uncertainty in Artificial Intelligence (UAI)*, 2013, pp. 282–290.

[12] James Hensman, Nicolas Durrande, and Arno Solin, "Variational Fourier features for Gaussian processes," *Journal of Machine Learning Research*, vol. 18, no. 151, pp. 1–52, 2018.

[13] David J. C. MacKay, "Introduction to Gaussian processes," in *Neural Networks and Machine Learning*, C. M. Bishop, Ed., NATO ASI Series, pp. 133–166. Kluwer Academic Press, 1998.

[14] Taylor Berg-Kirkpatrick, Jacob Andreas, and Dan Klein, "Unsupervised transcription of piano music," in *Advances in Neural Information Processing Systems 27 (NIPS)*, pp. 1538–1546. Curran Associates, Inc., 2014.

[15] Andrew Gordon Wilson and Ryan Prescott Adams, "Gaussian process kernels for pattern discovery and extrapolation," *30th International Conference on Machine Learning (ICML)*, pp. 1067–1075, 2013.

[16] Pablo A. Alvarado and Dan. Stowell, "Efficient learning of harmonic priors for pitch detection in polyphonic music," *arXiv preprint arXiv:1705.07104*, 2017.

[17] Papoulis Athanasious, *Probability, Random Variables, and Stochastic Process*, McGraw-Hill, Inc, 1991.

[18] K. Sam Shanmugan and Arthur M. Breipohl, *Random Signals: Detection, Estimation and Data Analysis*, Wiley, 1988.

[19] M. Goulard and M. Voltz, "Linear coregionalization model: Tools for estimation and choice of cross-variogram matrix," *Mathematical Geosciences*, vol. 24, no. 3, pp. 269–286, 1992.

[20] Emmanuel Vincent, Rémi Gribonval, and Cédric Févotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, July 2006.

[21] Alexander G. de G. Matthews, Mark van der Wilk, Tom Nickson, Keisuke. Fujii, Alexis Boukouvalas, Pablo León-Villagrá, Zoubin Ghahramani, and James Hensman, "GPflow: A Gaussian process library using TensorFlow," *Journal of Machine Learning Research*, vol. 18, no. 40, pp. 1–6, apr 2017.

[22] J. B. Allen and L. R. Rabiner, "A unified approach to short-time Fourier analysis and synthesis," *Proceedings of the IEEE*, vol. 65, no. 11, pp. 1558–1564, Nov 1977.

[23] `https://github.com/PabloAlvarado/ssgp`.