End-to-End Probabilistic Inference for Nonstationary Audio Analysis

William J. Wilkinson [1 2] **Michael Riis Andersen** [2 3] **Joshua D. Reiss** [1] **Dan Stowell** [1] **Arno Solin** [2]

Abstract

A typical audio signal processing pipeline includes multiple disjoint analysis stages, including calculation of a time-frequency representation followed by spectrogram-based feature analysis. We show how time-frequency analysis and nonnegative matrix factorisation can be jointly formulated as a spectral mixture Gaussian process model with nonstationary priors over the amplitude variance parameters. Further, we formulate this nonlinear model's state space representation, making it amenable to infinite-horizon Gaussian process regression with approximate inference via expectation propagation, which scales linearly in the number of time steps and quadratically in the state dimensionality. By doing so, we are able to process audio signals with hundreds of thousands of data points. We demonstrate, on various tasks with empirical data, how this inference scheme outperforms more standard techniques that rely on extended Kalman filtering.

1. Introduction

Uncovering the high-resolution spectral and temporal information present in a natural auditory scene is a challenging task. Loosely following the approach taken by the human auditory system, we decompose a one-dimensional audio signal into its high-dimensional set of time-varying spectral components, and then utilise the statistical features of these components to perform some auditory task such as classification or source separation. The highly ill-posed nature of this decomposition necessitates the use of prior information about the behaviour of the spectral components, which strongly encourages a probabilistic modelling perspective.

[1]Centre for Digital Music, Queen Mary University of London, United Kingdom [2]Department of Computer Science, Aalto University, Finland [3]Department of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark. Correspondence to: William J. Wilkinson <william.wilkinson@aalto.fi>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

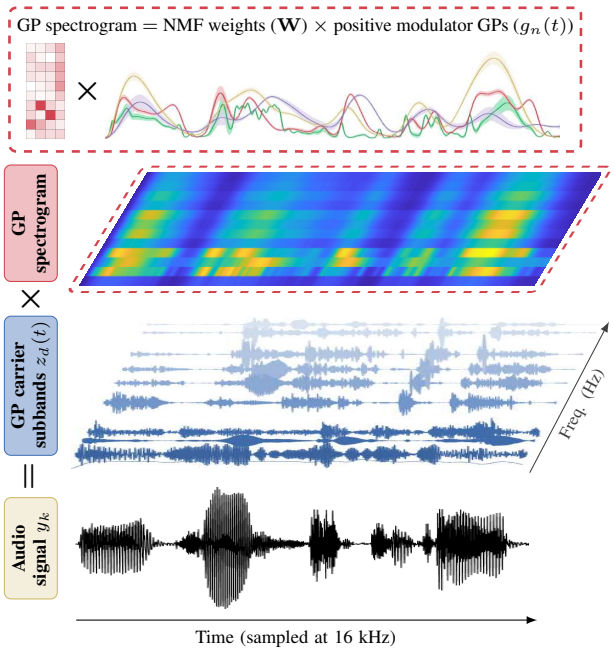


Figure 1. Nonstationary modelling of audio data: a recording of female speech (bottom). We decompose the signal into Gaussian process carrier waveforms (blue block) multiplied by a spectrogram (red block). The spectrogram is learned from data as a nonnegative weight matrix times positive modulators (top).

A typical (non-probabilistic) way to perform feature analysis on an audio signal is to apply nonnegative matrix factorisation (NMF) to the amplitude components of a time-frequency (TF) representation – the spectrogram. As outlined in (Turner & Sahani, 2014), this approach is limited since it discards phase information calculated during the TF stage, as well as dependencies between TF coefficients. It also fails to capture and share any uncertainty information between the analysis stages.

Moreover, the map that takes the waveform to the space of TF coefficients is not a bijection. This means that any function operating on the signal in the TF domain, e.g. noise removal, might push the signal outside the manifold of realisable waveforms (Turner, 2010). Hence, the modified TF representation must be projected back to the manifold of valid TF representations before the waveform can be resynthesized (e.g., Griffin & Lim, 1984). This projection might distort the signal and introduce undesirable artefacts.

These issues have motivated a large body of research on probabilistic models that operate directly on signal waveforms rather than on TF representations. Such models have been shown to outperform their spectrogram-based counterparts on several tasks, including source separation (Liutkus et al., 2011; Alvarado et al., 2019; Magron & Virtanen, 2019), audio inpainting and denoising (Badeau & Plumbley, 2014; Turner & Sahani, 2014). The limitations of spectrogram analysis have also motivated end-to-end machine learning algorithms for audio generation (Engel et al., 2017; Dieleman et al., 2018), generally based on neural networks that require large amounts of training data. In this paper we leverage prior knowledge to construct a probabilistic model that enables inference and learning for short- to medium-duration audio signals.

It has been shown that probabilistic TF analysis can be performed using a Gaussian process (GP) model whose kernel is a sum of quasi-periodic functions (Wilkinson et al., 2019). A GP formulation for combining TF analysis with nonnegative matrix factorization (NMF) has also been proposed (Turner & Sahani, 2014). However, the observation mechanism in this joint model is a nonlinear function of the latent components, making inference non-trivial. Previous work relies on a suboptimal inference scheme, where the separate model components are updated independently in an iterative fashion. Moreover, inference in GPs typically scales poorly in the number of time steps, making analysis infeasible for long audio signals. Hence, the full potential of probabilistic models for audio analysis has not yet been realised.

In this work, we propose a probabilistic model and an associated scalable inference algorithm that makes end-to-end audio analysis using GPs possible.[†] The contributions of this paper are as follows:

- We construct the state space form of a spectral mixture Gaussian process (GP) with nonstationary NMF priors over the amplitude variance parameters, showing that this model is equivalent to a Gaussian time-frequency NMF model (see Fig. 1 for an overview of the idea).
- We design an inference procedure for this nonlinear model based on power expectation propagation in the Kalman smoother setting.
- We construct the corresponding infinite-horizon GP (Solin et al., 2018) method for this model, which scales as $\mathcal{O}(M^2T)$ in time and $\mathcal{O}(MT)$ in memory, where M is the dimensionality of the state and T the number of time steps.
- We show performance of this approximate inference scheme on various tasks, and compare it to the classical signal processing approach: the iterated extended Kalman filter. By doing so, we demonstrate the flexibility of this generative model.

[†]Matlab code for all methods: https://github.com/AaltoML/nonstationary-audio-gp

In Sec. 2 we review the background material and related work on Gaussian process–based models for audio analysis. Sec. 3 introduces the proposed model and the associated inference algorithm. Sec. 4 demonstrates performance of the proposed method using a set of audio experiments.

2. Gaussian Process Time-Frequency Analysis

To specify a probabilistic end-to-end model for the audio processing pipeline, we must replace or remodel the standard processing stages with their probabilistic counterparts. Gaussian processes (GPs, Rasmussen & Williams, 2006) are a flexible tool for specifying probability distributions over functions, and can be deployed in many such cases.

GP models for time series typically admit the form:

$$f(t) \sim \mathrm{GP}(0, \kappa(t, t')), \tag{1a}$$

$$\mathbf{y} \,|\, \mathbf{f} \sim \prod_{k=1}^{T} p(y_k \,|\, f(t_k)), \tag{1b}$$

where the one-dimensional input t represents time, Eq. (1a) defines the Gaussian process prior and Eq. (1b) the likelihood (observation) model. The data $\boldsymbol{\mathcal{D}} = \{(t_k, y_k)\}_{k=1}^{T}$ consist of input–output pairs and $\kappa(t, t')$ is a covariance function encoding the prior assumptions of the latent (hidden) process $f(t)$.

Following the typical approach (see, e.g., Rasmussen & Nickisch, 2010, for an overview) we seek an approximate posterior of the form:

$$q(\mathbf{f} \,|\, \boldsymbol{\mathcal{D}}) = \mathrm{N}(\mathbf{f} \,|\, \mathbf{K}\boldsymbol{\alpha}, (\mathbf{K}^{-1} + \mathbf{V})^{-1}), \tag{2}$$

where the covariance matrix $K_{i,j} = \kappa(t_i, t_j)$ comes from the prior, $\boldsymbol{\alpha} \in \mathbb{R}^T$, and the likelihood precision matrix $\mathbf{V}$ is diagonal.

The predictive distribution for a test input t_* with training locations $\boldsymbol{t}$ is obtained by integrating the Gaussian latent marginal distribution $\mathrm{N}(f_* \,|\, \mu_{\mathrm{f},*}, \sigma^2_{\mathrm{f},*})$, where $\mu_{\mathrm{f},*} = \mathbf{K}(t_*, \boldsymbol{t})\boldsymbol{\alpha}$ and $\sigma^2_{\mathrm{f},*} = \mathbf{K}(t_*, t_*) - \mathbf{K}(t_*, \boldsymbol{t})(\mathbf{K} + \mathbf{V}^{-1})^{-1}\mathbf{K}(\boldsymbol{t}, t_*)$, against the likelihood $p(y_* \,|\, f_*)$ to obtain $p(y_*) = \int p(y_* \,|\, f_*)\, \mathrm{N}(f_* \,|\, \mu_{\mathrm{f},*}, \sigma^2_{\mathrm{f},*})\, \mathrm{d}f_*$, the predictive distribution describing the unknown y_*.

A probabilistic way of learning the hyperparameters $\boldsymbol{\theta}$ of the covariance function and the observation model is by maximising the log marginal likelihood function (Rasmussen & Williams, 2006) (or an approximation of it),

$$\log p(\mathbf{y} \,|\, \boldsymbol{\theta}) = \log \int \mathrm{N}(\mathbf{f} \,|\, \mathbf{0}, \mathbf{K}_{\boldsymbol{\theta}}) \prod_k p(y_k \,|\, f_k, \boldsymbol{\theta})\, \mathrm{d}\mathbf{f}. \tag{3}$$

Issues in dealing with the latent functions Given the well-established GP modelling framework, it may seem surprising that these methods are not widely used in audio modelling. However, the prohibitive computational cubic time-scaling in the number of data renders this naive approach useless for most audio applications where data samples are typically acquired at thousands of samples per second (say, 16 kHz).

Standard approaches for speeding up GP inference, such as inducing input (Quiñonero-Candela & Rasmussen, 2005; Snelson & Ghahramani, 2006; Titsias, 2009), interpolation approaches (Wilson & Nickisch, 2015), stochastic methods (Hensman et al., 2013; Krauth et al., 2017), basis function approximations (Lázaro-Gredilla et al., 2010; Hensman et al., 2018; Solin & Särkkä, 2014a) scale poorly in long (or potentially unbounded) time series models such as audio analysis. Band-structured or Toeplitz methods (Saatçi, 2012) work for data whose sampling is fixed, but would, for example, fail in missing data analysis and only be applicable in batch data scenarios.

Recent advances in combining GP models with efficient signal processing methods have lead to schemes that reformulate the GP prior in terms of a state space model and conduct inference by Kalman filtering in *linear* time complexity (Reece & Roberts, 2010; Särkkä et al., 2013). If the GP prior exhibits Markov structure, these models are exact and no approximations are needed. Recently, Nickisch et al. (2018) bridged the gap between the state space and kernel based GP methods, by providing a unifying framework for inference in non-Gaussian likelihoods with established inference schemes like the Laplace approximation, direct KL minimisation, variational Bayes, and single-sweep expectation propagation (EP). We build on these state space methods for linear-time inference for GP audio modelling.

Probabilistic time-frequency analysis It has been shown that standard approaches to probabilistic time-frequency analysis are equivalent to Gaussian process regression where the GP kernels are a sum of quasi-periodic components (Wilkinson et al., 2019). Such kernels, known as spectral mixtures (Wilson & Adams, 2013), can be written generally as

$$\kappa_{\mathrm{sm}}(t,t') = \sum_{d=1}^{D} \kappa_z^{(d)}(t,t'), \tag{4a}$$

$$\kappa_z^{(d)}(t,t') = \sigma_d^2 \cos(\omega_d(t-t'))\kappa_d(t,t'), \tag{4b}$$

and κ_d is free to be chosen, but is typically from the Matérn class. Parameters ω_d determine the periodicity of the kernel components, which can be interpreted as the centre frequencies of the filters in a probabilistic filter bank. By choosing the exponential kernel $\kappa_d(t,t') = \exp(|t-t'|/\ell_d)$ we recover exactly the probabilistic phase vocoder (Cemgil & Godsill, 2005), and the lengthscales ℓ_d control the filter bandwidths.

The drawback of this model for audio data is that it assumes independence across frequency channels. Correlation between amplitudes of harmonics or modes of vibration is crucial for representing audio signals and is a key component of auditory perception (Turner, 2010; McDermott et al., 2013). This motivates a model that explicitly captures these intra-channel correlations. However, such models no longer observe data through linear combinations of the latent functions, and typical techniques for dealing with these cases tend to fail due to the complex interactions present in audio data. This paper is concerned with addressing these issues.

Nonnegative matrix factorisation To capture the desired dependencies across channels, we follow Turner & Sahani (2014) by utilising nonnegative matrix factorisation (NMF) (Lee & Seung, 1999). NMF decomposes a high-dimensional matrix $\mathbf{A} \in \mathbb{R}^{D\times T}$, such as the spectrogram of an audio signal, into a product of two lower-rank nonnegative matrices: a temporal dictionary $\mathbf{G}$, and a spectral dictionary $\mathbf{W}$,

$$\mathbf{A} \simeq \mathbf{W}\mathbf{G}. \tag{5}$$

Typically $\mathbf{W} \in \mathbb{R}^{D\times N}$ and $\mathbf{G} \in \mathbb{R}^{N\times T}$ are learnt by minimising the divergence between the left and right hand sides of Eq. (5). In the next section, we place a GP prior over the rows of $\mathbf{G}$ and treat the elements of $\mathbf{W}$ as free parameters of our probabilistic model.

3. Methods

In this section we will first write down the model along with its equivalent presentation as a nonstationary spectral mixture GP. We'll then discuss how it can be constructed as a stochastic differential equation in state space form, before outlining the potential inference methods available.

3.1. Gaussian Time-Frequency Nonnegative Matrix Factorisation Model (GTF-NMF)

We aim to decompose an input signal $\{y_k\}_{k=1}^T$ into D unknown frequency (oscillator) channels, whose relative amplitudes are modulated by N temporal NMF components. The GP *priors* for the $D+N$ latent model component functions are:

$$g_n(t) \sim \mathrm{GP}(0, \kappa_{\mathrm{g}}^{(n)}(t,t')), \quad n = 1,2,\ldots,N, \tag{6a}$$

$$z_d(t) \sim \mathrm{GP}(0, \kappa_{\mathrm{z}}^{(d)}(t,t')), \quad d = 1,2,\ldots,D, \tag{6b}$$

where $g_n(t)$ denotes the n^{th} temporal NMF component function and $z_d(t)$ the d^{th} frequency channel. The kernel $\kappa_z^{(d)}$ is chosen to be a quasi-periodic function, i.e. the d^{th} component of a spectral mixture, Eq. (4b). $\kappa_g^{(n)}$ should be determined by our assumptions about the behaviour of the amplitude modulators, such as their smoothness properties.

The *likelihood* model is given by:

$$y_k = \sum_d a_d(t_k)\, z_d(t_k) + \sigma_y\, \varepsilon_k, \tag{7}$$

for square amplitudes (the magnitude spectrogram):

$$a_d^2(t_k) = \sum_n W_{d,n}\, \psi(g_n(t_k)). \tag{8}$$

Positivity of the NMF components is enforced by a link function, the softplus $\psi(g_n) = \log(1 + \mathrm{e}^{g_n})$. $\mathbf{W} \in \mathbb{R}^{D\times N}$ are the NMF weights determining which modulators affect which oscillators. If we set $N < D$, then the model captures amplitude behaviour shared across frequency channels.

Note that if we set $a_d(t_k) = 1,\ \forall\, d, k$ then Eq. (7) reduces to standard probabilistic time-frequency analysis, the model given in Wilkinson et al. (2019). If we discard $z_d(t_k)$ by calculating a fixed spectrogram, such that $a_d^2(t_k)$ become our observations, then Eq. (8) is standard temporal NMF (Bertin et al., 2010). Further removing the GP prior over g_n brings us back to the NMF model in Eq. (5).

Fig. 1 shows the model diagrammatically – the frequency channel subbands z_d are D independent, unit variance GPs with quasi-periodic covariance functions. The modulators g_n and the NMF weights constitute a model for the spectrogram, the squared amplitudes of the frequency channels.

The inference methods we will next present allow for any choice of κ_g, κ_z, so long as they can be written in state space form, either approximately or exactly. See Särkkä & Solin (2019) for a guide to writing kernels in the appropriate way. We focus on the Matérn kernel class due to their strong connection to autoregressive filters, and because their parameters have convenient interpretations for our task – their lengthscales and variances relate to the bandwidth and scale of the filters in a filter bank (Wilkinson et al., 2019).

If we write down our model in its hierarchical form, we observe a striking similarity to the nonstationary spectral mixture GPs presented in Remes et al. (2017). This hierarchical form has a hyper-GP prior $g_n(t) \sim \mathrm{GP}(0, \kappa_{\mathrm{g}}^{(n)}(t, t'))$ for each component with an NMF-like positivity mapping $\alpha_d^2(t) = \sum_n W_{d,n}\, \psi(g_n(t))$, and the final model becomes:

$$z(t) \sim \mathrm{GP}\left(0, \sum_{d=1}^{D} \alpha_d(t)\alpha_d(t') \cos(\omega_d(t - t'))\kappa_d(t, t')\right), \tag{9a}$$

$$y_k = z(t_k) + \sigma_y\, \varepsilon_k. \tag{9b}$$

This is a nonstationary spectral mixture GP with fixed frequencies ω_d and lengthscales ℓ_d, with an NMF mapping in the GP prior over the time-varying amplitude variances $\alpha_d^2(t)$. This equivalence means that the inference methods laid out in Secs. 3.3 and 3.4 also apply to nonstationary spectral mixtures, as do their formulation as SDEs in Sec. 3.2.

3.2. State Space Methods for the Latent Functions

For scalable computation, we transform the GP model in Eq. (6) into state space form by mapping the associated covariance functions to stochastic differential equations (SDEs). If the GP priors admit (high-order) Markovian structure (as they do in our case), the model has an exact representation in terms of an SDE (see Solin, 2016, for examples and discussion). In continuous time, the system of independent GP priors is given by the following linear time-invariant SDE:

$$\dot{\mathbf{x}}(t) = \mathbf{F}\, \mathbf{x}(t) + \mathbf{L}\, \mathbf{w}(t), \tag{10}$$

where $\mathbf{F} \in \mathbb{R}^{M\times M}$ and $\mathbf{L} \in \mathbb{R}^{M\times S}$, for $S = 2D + N$, are the feedback and noise effect matrices, respectively. The driving process $\mathbf{w}(t) \in \mathbb{R}^S$ is a multivariate white noise process with spectral density matrix $\mathbf{Q}_c \in \mathbb{R}^{S\times S}$.

The state $\mathbf{x}(t)$ corresponds to a stacked multi-output stochastic process representing the GP priors $z_d(t), d = 1, \ldots, D$ and $g_n(t), n = 1, \ldots, N$. Each of the GP components have a representation in terms of submatrices of $\mathbf{F}$, $\mathbf{L}$, and $\mathbf{Q}_{\mathrm{c}}$.

The SDE representation of the $D + N$ Gaussian process priors can be written in the following block-Kronecker form:

$$\begin{aligned}\mathbf{F} = \mathrm{blkdiag}(&\mathbf{F}_{\mathrm{cos}}^{(1)} \oplus \mathbf{F}_{\mathrm{mat}}^{(1)}, \ldots, \mathbf{F}_{\mathrm{cos}}^{(D)} \oplus \mathbf{F}_{\mathrm{mat}}^{(D)},\\ &\mathbf{F}_{\mathrm{mat}}^{(1)}, \ldots, \mathbf{F}_{\mathrm{mat}}^{(N)}),\end{aligned} \tag{11a}$$

$$\begin{aligned}\mathbf{L} = \mathrm{blkdiag}(&\mathbf{L}_{\mathrm{cos}}^{(1)} \otimes \mathbf{L}_{\mathrm{mat}}^{(1)}, \ldots, \mathbf{L}_{\mathrm{cos}}^{(D)} \otimes \mathbf{L}_{\mathrm{mat}}^{(D)},\\ &\mathbf{L}_{\mathrm{mat}}^{(1)}, \ldots, \mathbf{L}_{\mathrm{mat}}^{(N)}),\end{aligned} \tag{11b}$$

$$\begin{aligned}\mathbf{Q}_{\mathrm{c}} = \mathrm{blkdiag}(&\mathbf{I}_2 \otimes \mathbf{Q}_{\mathrm{c,mat}}^{(1)}, \ldots, \mathbf{I}_2 \otimes \mathbf{Q}_{\mathrm{c,mat}}^{(D)},\\ &\mathbf{Q}_{\mathrm{c,mat}}^{(1)}, \ldots, \mathbf{Q}_{\mathrm{c,mat}}^{(N)}),\end{aligned} \tag{11c}$$

where '$\oplus$' and '$\otimes$' denote the Kronecker sum and product. The submatrices $\mathbf{F}_{\mathrm{mat}}^{(1)}$, $\mathbf{F}_{\mathrm{cos}}^{(1)}$, $\mathbf{L}_{\mathrm{mat}}^{(1)}$ etc. correspond to the matrices that make up the SDE representation for the Matérn and cosine kernels (Solin & Särkkä, 2014b). Here we have assumed a Matérn kernel for κ_d, κ_n, but this can be altered as necessary.

The audio data (observations) are evenly spaced in time, which simplifies the discrete-time solution to the SDE in Eq. (10). For discrete input values t_k, this translates into

$$\mathbf{x}_k \sim \mathrm{N}(\mathbf{A}\, \mathbf{x}_{k-1}, \mathbf{Q}) \tag{12}$$

with $\mathbf{x}_0 \sim \mathrm{N}(\mathbf{0}, \mathbf{P}_0)$. The discrete-time dynamical model is solved through a matrix exponential $\mathbf{A} = \exp(\mathbf{F}\, \Delta t)$. For stationary covariance functions, the process noise covariance is given by $\mathbf{Q} = \mathbf{P}_\infty - \mathbf{A}\, \mathbf{P}_\infty\, \mathbf{A}^\top$. The stationary state (corresponding to the initial state $\mathbf{P}_0$) is distributed by $\mathbf{x}_\infty \sim \mathrm{N}(\mathbf{0}, \mathbf{P}_\infty)$ and the stationary covariance can be found by solving the Lyapunov equation $\dot{\mathbf{P}}_\infty = \mathbf{F}\, \mathbf{P}_\infty + \mathbf{P}_\infty\, \mathbf{F}^\top + \mathbf{L}\, \mathbf{Q}_c\, \mathbf{L}^\top = \mathbf{0}$.

3.3. Linearisation-Based Inference

In classical signal processing, the most widely used technique for dealing with nonlinear/non-Gaussian inference problems in state space models is the *extended Kalman filter* (EKF, Jazwinski, 1970; Bar-Shalom et al., 2001). The EKF, together with the backward-pass known as the extended Rauch–Tung–Striebel smoother, provides a means of approximating the state distributions $p(\mathbf{x} \mid \mathbf{y}_{1:T})$ with Gaussians (corresponding to the time-marginals of Eq. 2):

$$q(\mathbf{x}_k \mid \boldsymbol{\mathcal{D}}) \simeq \mathrm{N}(\mathbf{x}_k \mid \mathbf{m}_k, \mathbf{P}_k). \tag{13}$$

In the EKF, these approximations are formed by first-order linearisations of the nonlinearities (see Särkkä, 2013, for a detailed presentation of the extended Kalman filtering recursion). For GPs, a related local linearisation scheme is known as the Laplace approximation, where the approximation is improved iteratively by mode-seeking. In signal processing, iterative versions of the EKF are known as iterated filters, where the iteration is typically in the inner update loop (local iterated EKF, Jazwinski, 1970; Maybeck, 1982). Outer-loop variants which—similar to the GP Laplace method—seek a global approximation are known as the global iterated EKF (Zhang, 1997).

In Alg. 2 in the supplementary material, we present an outer-loop extended Kalman filtering scheme for Laplace approximation-like inference. The local linearisation is done with respect to the measurement (likelihood) model in Eq. (7) by deriving its closed-form Jacobian $\mathbf{H}_{\mathbf{x}}(\mathbf{x})$. We consider this algorithm as the baseline for our experiments.

3.4. Expectation Propagation in the GTF-NMF Model

The signal processing community has provided linear-time algorithms for scaling linear state space models to huge, unbounded time series. While scalable, these methods are limited to systems that are well approximated by linear models and they are in general not capable of producing accurate inference in the presence of strong nonlinear dependencies such as in the model presented in Eq. (7). Nickisch et al. (2018) proposed to combine the classical methods with modern tools for approximate inference, e.g. variational Bayes and assumed density filtering (ADF), to overcome this issue. We generalise this work by extending the ADF algorithm to expectation propagation and thus combining the best methods from the signal processing and machine learning communities.

Expectation propagation (EP, Minka, 2001) and power expectation propagation are methods for approximating intractable probability distributions using tractable distributions from the exponential family. EP is a generalisation of ADF and works by minimising local Kullback-Leibler (KL) divergences in an iterative fashion. Power EP can be seen as a further generalisation of EP that minimises local α-divergences rather than KL divergences (Minka, 2005).

Using power EP, we approximate the intractable likelihood terms as follows:

$$p(y_k \mid \mathbf{g}_k, \mathbf{z}_k) \approx q_k(\mathbf{g}_k, \mathbf{z}_k), \tag{14}$$

where each site approximation q_k belongs to the exponential family. Specifically, we assume that q_k takes the form

$$q_k(\mathbf{g}_k, \mathbf{z}_k) = \prod_n \mathrm{N}(g_{n,k} \mid \nu^g_{n,k}, \tau^g_{n,k}) \prod_d \mathrm{N}(z_{d,k} \mid \nu^z_{d,k}, \tau^z_{d,k}), \tag{15}$$

where $\nu^g_{n,k}$ and $\tau^g_{n,k}$ are the precision-adjusted mean and precision, respectively, for $g_{n,k}$ etc. This choice leads to a joint Gaussian posterior approximation. Rather than simply matching the two distributions in Eq. (14), the EP algorithm iteratively refines the posterior approximation by updating each site approximation q_k in the context of the so-called *cavity distribution* q_{-k}. The cavity distribution for the k^{th} observation is defined by removing the contribution of the k^{th} site approximation from the posterior approximation $q(\mathbf{g}_k, \mathbf{z}_k \mid \boldsymbol{\mathcal{D}})$. That is,

$$q_{-k}(\mathbf{g}_k, \mathbf{z}_k) \propto \frac{q(\mathbf{g}_k, \mathbf{z}_k \mid \boldsymbol{\mathcal{D}})}{q_k(\mathbf{g}_k, \mathbf{z}_k)^\eta} \tag{16}$$

for $\eta \in (0, 1]$, where $\eta = 1$ corresponds to regular EP and $\eta < 1$ to power EP.

The k^{th} site approximation q_k is then updated by minimising the KL-divergence between the *tilted distribution* $\hat{p}_k = \frac{1}{Z_k} p(y_k \mid \mathbf{g}_k, \mathbf{z}_k)^\eta q_{-k}(\mathbf{g}_k, \mathbf{z}_k)$ and the power EP approximation $q_k(\mathbf{g}_k, \mathbf{z}_k)^\eta q_{-k}(\mathbf{g_k}, \mathbf{z_k})$ such that

$$q^*_k(\mathbf{g}_k, \mathbf{z}_k \mid \boldsymbol{\mathcal{D}}) = \arg\min_{q_k} \mathrm{D}_{\mathrm{KL}}\left[\hat{p}_k \,\|\, q_k^\eta q_{-k}\right], \tag{17}$$

or equivalently, by matching the moments of the two distributions. The normalisation constant Z_k is given by

$$Z_k = \mathbb{E}_{q_{-k}}\left[p(y_k \mid \mathbf{g}_k, \mathbf{z}_k)^\eta\right]. \tag{18}$$

The moments of the tilted distribution can be obtained from the first two partial derivatives of $\log Z_k$ with respect to two sets of cavity mean parameters $\{\mu^g_{n,-k}\}_{n=1}^N$ and $\{\mu^z_{d,-k}\}_{d=1}^D$. For a full derivation of the normalisation constant and its derivatives, see the supplementary material.

The resulting expectations are analytically intractable because the likelihood is a nonlinear function of $\mathbf{g}_k$ and $\mathbf{z}_k$. We numerically approximate the N-dimensional integrals required to calculate the expectations with 9^{th}-order sigma-point methods (McNamee & Stenger, 1967; Kokkala et al., 2016). However, the number of sigma-points required in this 9^{th}-order approximation scales poorly with the number of NMF components, $\frac{1}{2}(2N^4 - 4N^3 + 22N^2 - 8N + 3)$, which slows down inference for large N.

Algorithm 1 EP using Kalman smoothing

Input: $\{t_k, y_k\}_{k=1}^T$ training inputs and targets
$\mathbf{A}, \mathbf{Q}, \mathbf{H}, \mathbf{P}_0$ discretised state space model
$\boldsymbol{\tau} \leftarrow \mathbf{0}$, $\boldsymbol{\nu} \leftarrow \mathbf{0}$ likelihood eff. precision and location
while not converged **do** EP loop
  for $k = 1$ **to** T **do** forward pass
    if $k == 1$ **then**
      $\mathbf{m}_k \leftarrow \mathbf{0}$; $\mathbf{P}_k \leftarrow \mathbf{P}_0$ init
    else
      $\mathbf{m}_k \leftarrow \mathbf{A}\mathbf{m}_{k-1}$; $\mathbf{P}_k \leftarrow \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^\top + \mathbf{Q}$ predict
    end if
    if has label y_k **then**
      $\boldsymbol{\mu} \leftarrow \mathbf{H}\mathbf{m}_k$; $\mathbf{U} \leftarrow \mathbf{P}_k\mathbf{H}^\top$; $\boldsymbol{\sigma}^2 \leftarrow \mathrm{diag}\,(\mathbf{H}\mathbf{U})$
      if first EP iteration **then**
        $\boldsymbol{\tau}_{-k} \leftarrow \boldsymbol{\sigma}^2$; $\boldsymbol{\nu}_{-k} \leftarrow \boldsymbol{\mu}$ cavity
        set $(\boldsymbol{\nu}_k, \boldsymbol{\tau}_k)$ to minimise the KL div. in Eq. (17) by calculating Z_k in Eq. (18) and its gradients
      end if
      $\mathbf{c}_k \leftarrow \boldsymbol{\mu} \oslash \boldsymbol{\tau}_k - \boldsymbol{\nu}_k$
      $\mathbf{K}_k \leftarrow \mathbf{U}\left(\boldsymbol{\sigma}^2 + \mathbf{1} \oslash \boldsymbol{\tau}_k\right)^{-1}$
      $\mathbf{P}_k \leftarrow \mathbf{P}_k - \mathbf{K}_k\mathbf{U}^\top$ variance
      $\mathbf{m}_k \leftarrow \mathbf{m}_k + \mathbf{K}_k\mathbf{c}_k$ mean
    end if
  end for
  for $k = T - 1$ **to** 1 **do** backward pass
    $\mathbf{G}_k \leftarrow \mathbf{P}_k\,\mathbf{A}^\top\,(\mathbf{A}\,\mathbf{P}_k\,\mathbf{A}^\top + \mathbf{Q})^{-1}$ gain
    $\mathbf{m}_k \leftarrow \mathbf{m}_k + \mathbf{G}_k\,(\mathbf{m}_{k+1} - \mathbf{A}\,\mathbf{m}_k)$
    $\mathbf{P}_k \leftarrow \mathbf{P}_k + \mathbf{G}_k\,(\mathbf{P}_{k+1} - \mathbf{A}\,\mathbf{P}_k\,\mathbf{A}^\top - \mathbf{Q})\,\mathbf{G}_k^\top$
    $\boldsymbol{\mu} \leftarrow \mathbf{H}\mathbf{m}_k$; $\boldsymbol{\sigma}^2 \leftarrow \mathrm{diag}\left(\mathbf{H}\mathbf{P}_k\mathbf{H}^\top\right)$ latent
    $\boldsymbol{\tau}_{-k} \leftarrow \mathbf{1} \oslash \boldsymbol{\sigma}^2 - \eta\boldsymbol{\tau}_k$; $\boldsymbol{\nu}_{-k} \leftarrow \boldsymbol{\mu} \oslash \boldsymbol{\sigma}^2 - \eta\boldsymbol{\nu}_k$ cavity
    set $(\boldsymbol{\nu}_k, \boldsymbol{\tau}_k)$ to minimise the KL div. in Eq. (17) by calculating Z_k in Eq. (18) and its gradients
  end for
end while
Return: $\mathbb{E}[g_n(t_k)] = \mathbf{h}_n^{\mathrm{g}}\mathbf{m}_k; \mathbb{V}[g_n(t_k)] = \mathbf{h}_n^{\mathrm{g}}\mathbf{P}_k\mathbf{h}_n^{\mathrm{g}\top}$
$\mathbb{E}[z_d(t_k)] = \mathbf{h}_d^{\mathrm{z}}\mathbf{m}_k; \mathbb{V}[z_d(t_k)] = \mathbf{h}_d^{\mathrm{z}}\mathbf{P}_k\mathbf{h}_d^{\mathrm{z}\top}$
$\log p(\mathbf{y} \mid \boldsymbol{\theta}) \simeq \sum \log Z_k$

Notation: $\mathbf{a} \circ \mathbf{b}$ and $\mathbf{a} \oslash \mathbf{b}$ denote the element-wise multiplication and element-wise divison of the vectors $\mathbf{a}$ and $\mathbf{b}$, respectively. $\mathbf{H}$ is the measurement model with rows $\mathbf{h}$.

The proposed algorithm is prone to convergence issues. To prevent EP from oscillating, we use *damped* updates for the site parameters (Minka & Lafferty, 2002). That is, the site parameters are updated as a convex combination of the current parameter values and the new parameters values. Given the large amount of damping required, we generally had to run EP for 20 iterations to reach convergence, more than the 5-10 that is often reported in simpler models.

Standard EP scales cubicly in the number of observations. However, by using the Rauch–Tung–Striebel smoother to approximate the marginal posterior distributions $q(\mathbf{g}_k, \mathbf{z}_k \mid \boldsymbol{\mathcal{D}})$ in Eq. (16), we can reduce the complexity of the algorithm to be linear in the number of observations. The EP algorithm is summarised in Alg. 1.

3.5. Infinite-Horizon Gaussian Processes

The inference in Alg. 1 has linear time complexity, $\mathcal{O}(TM^3)$ (with $M \ll T$), with respect to the number of data points T, and state dimensionality M. The memory scaling is $\mathcal{O}(TM^2)$ due to the need for storing the state covariances at every time step. However, in the case of audio data T can be tens or hundreds of thousands even for short audio segments. This is mainly problematic with regards to the required memory (M typically in the range of 100–1000). For example, for $M = 100$, the required memory is in the range of 1.2 Gb per second of data.

To mitigate the memory bottleneck, we use the infinite-horizon GP (IHGP) framework proposed by Solin et al. (2018), where the GP is approximated by finding an associated posterior steady state of the filter for each of the $D + N$ latent functions. This way the propagation of the covariance terms in Alg. 1 can be simplified, leading to a computational time-scaling of $\mathcal{O}(TM^2)$ and memory scaling $\mathcal{O}(TM)$. Solin et al. (2018) derived their method to work with ADF, but the EP formulation given in Alg. 1 directly lends itself to the approach by using the cavity parameters for updating the likelihood variance terms. With these changes, the required memory drops by orders of magnitude to 12.2 Mb per second of data.

3.6. Hyperparameter Tuning

Model learning is difficult in this setting due to the highly correlated nature of the kernel hyperparameters and the non-identifiability of the NMF mapping. We initialise the parameters via frequency domain fitting with the standard probabilistic TF model, as outlined in Wilkinson et al. (2019), which is fast and gives an accurate estimate of the subband frequencies and lengthscales. We initialise the NMF weights using standard NMF applied to a spectrogram calculated with the subband model. Further tuning is then carried out by direct optimisation of the (log) marginal likelihood, $\log p(\mathbf{y} \mid \boldsymbol{\theta})$, which is calculated during Kalman smoothing as shown in Alg. 1. We leave development of a more robust learning scheme to future work.

4. Experiments

In this section we compare the proposed inference methods, showing that fully iterated EP is absolutely necessary for inference in the GTF-NMF model, since the iterated EKF and single-sweep EP approaches fail to uncover the latent functions with sufficient accuracy. Our generative model is extremely flexible, and we demonstrate here how it can be applied to three different real world tasks (and one simulated task) with no adjustment of the model or algorithm: missing data synthesis, denoising and source separation. The GTF-NMF performs on a similar level to application spe-

Table 1. Performance measures for each inference scheme. *'sim.'* shows fit to observed data $\mathbf{y}$ in the simulated data experiment (likelihood noise variance is $\sigma_y^2 = 10^{-4}$). *'mis.'* shows mean missing data imputation results on a dataset of 10 musical instrument sounds, with segments of 20ms removed. Signal-to-noise ratio (in dB, larger is better) and root mean square error (smaller is better). Based on predictive mean. MP is the matching pursuit baseline.

	EP1	EP20	IHGP1	IHGP20	EKF1	EKF20	MP
RMSE (sim.)	0.044	**0.003**	0.042	0.029	0.124	0.128	—
SNR (mis.)	7.494	**8.087**	4.520	4.591	3.716	3.735	5.232
RMSE (mis.)	0.590	**0.551**	0.720	0.716	0.746	0.743	0.761

cific algorithms (better in missing data imputation, worse in denoising), whilst being much more general.

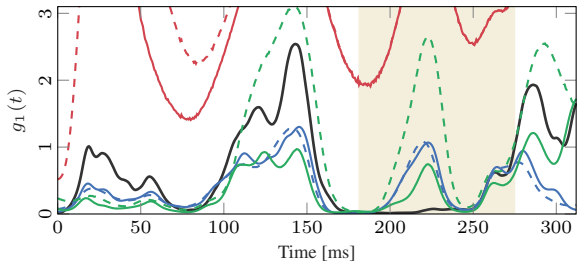
For ease of comparison, in all the real-world experiments we set $D = 16$, $N = 3$ and tune the parameters via single-sweep EP (ADF), with $\eta = 0.75$ and damping of 0.1. We use these parameters to directly compare the different inference methods (with the exception of the simulated data experiment where we use the known parameters). We use the exponential and Matérn-$5/2$ kernels for κ_d and κ_g. The advantages of the infinite-horizon approach become clear when we consider the source separation problem, in which the mixture signal contains multiple sources (leading to a very high-dimensional state space $M = 123$), and is 6 seconds in duration ($T = 96{,}000$).

Simulated Data Experiment We set $D = 5$, $N = 2$ and fix the hyperparameters by hand, before sampling from the generative model to create synthetic data. Fig. 2 shows how each of the proposed inference methods estimates the hidden subband signals and NMF modulators. Uncovering the latents is a highly non-identifiable problem, especially due to the ambiguous nature of the model in which amplitude variation can occur due to variance in the subbands or the modulators. However, EP finds a much better match to the ground truth than EKF, and we see that iterating the IHGP method resolves part of the ambiguity. Table 1 shows how closely the approximate inference methods are able to fit the training data. Since $\sigma_y^2 = 10^{-4}$, we would hope the RMSE to be below $\sigma_y = 0.01$, a feat which only full EP manages.

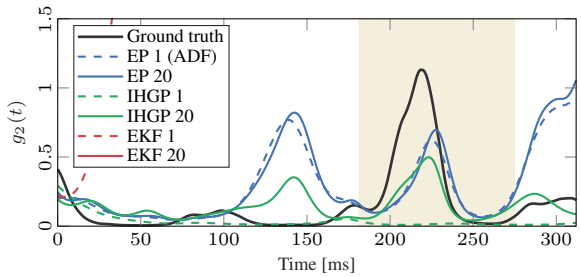
Missing Data Imputation The generative model handles missing data synthesis naturally by treating the time steps where there are missing data as test locations and making predictions as usual. Table 1 shows the results of the prediction task on a dataset of 10 musical instrument recordings. Fig. 3 shows an example segment. As a baseline we compare our methods to a well known matching pursuit algorithm (Adler et al., 2012), which was outperformed by the iterated EP scheme, performing roughly in line with IHGP.

Denoising Assuming a signal is corrupted by Gaussian noise of known variance, the GTF-NMF model can be adapted to a denoising task by setting the measurement noise variance σ_y^2 to the appropriate level. Fig. 5 is an example of denoising a speech recording, where the clean signal is corrupted with $\sigma_y^2 = 0.3$. Fig. 4 shows the denoising results for the various inference methods for five different noise levels. Here we also compare against a spectral subtraction algorithm (Ephraim & Malah, 1984). GP models are expected to deal with Gaussian noise well, however the approximate nature of inference in the GTF-NMF prevents it from outperforming this application-specific approach.

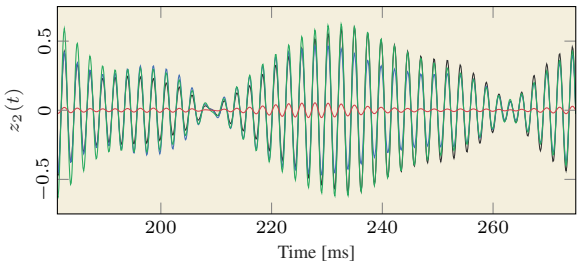
Source Separation As a further demonstration, we follow the approach taken in Alvarado et al. (2019) by training the model on musical instrument notes (sources), and then



(a) First NMF component, $g_1(t)$



(b) Second NMF component, $g_2(t)$



(c) Short segment of one of the subband signals, $z_2(t)$

Figure 2. A simulated data experiment examining the ability of various inference methods to uncover the spectral components z_d and NMF components g_n when the true parameters are known. Due to the ambiguity inherent in the model, (multiple sources of amplitude modulation), uncovering the latents is a difficult task. Standard EP and the IHGP methods far outperform EKF. "EP 1" relates to inference with 1 EP iteration (ADF). The iterated methods (dashed lines, each using 20 iterations) resolve the ambiguity better than the single sweep approach, except in the EKF case. Only the mean of the predictive distributions are shown.

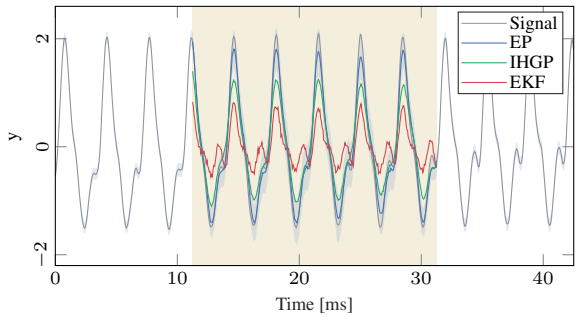


Figure 3. An example of missing data imputation with the GTF-NMF model for each inference method with 20 iterations. Grey signal is the ground truth, a recording of a bamboo flute. The yellow shaded region indicates where the data is missing. Blue shaded area is the 95% confidence region for the EP method.

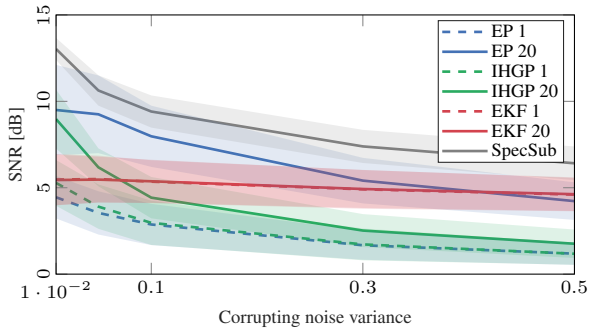


Figure 4. Denoising with various inference methods across five levels of corruption noise variance (0.01–0.5). y-axis is the signal-to-noise ratio of the recovered waveform. Mean values across 10 speech signals are shown. Shaded areas are standard error. SpecSub is the spectral subtraction baseline.

attempting to uncover these sources when they are mixed via summation of their waveforms in a series of two-note chords. The only inference method capable of processing these series of notes is IHGP, due to the computation and memory requirements of stacking the sources in a state space model for 6 seconds of data (sampled at 16 kHz, $T = 96{,}000$, $M = 123$). Therefore we cannot compare performance on this task, but we show an example separation result in Fig. 6.

5. Discussion and Conclusions

We have constructed a novel scheme for inference in the Gaussian time-frequency NMF model based on expectation propagation and infinite-horizon GPs, leading to an end-to-end probabilistic approach for audio modelling. By outlining how this model is similar to a nonstationary spectral mixture GP, we have further unified the theory connecting probabilistic machine learning and signal processing.

We demonstrated that our inference scheme consistently outperforms the extended Kalman filtering approach. This suggests that it is indeed necessary to go beyond classical

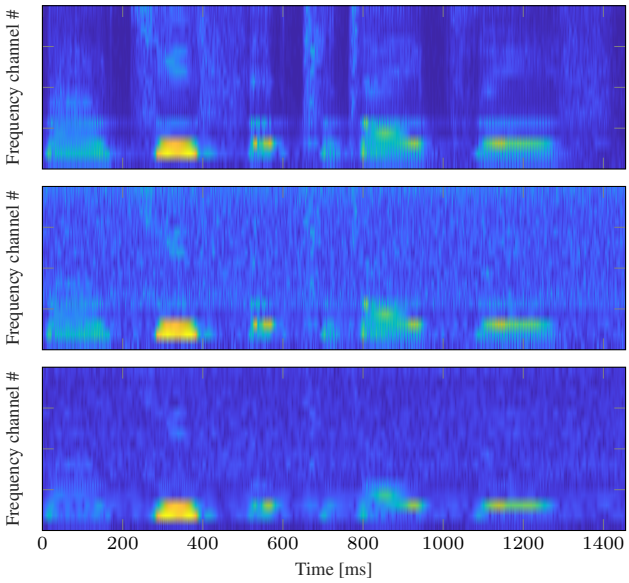


Figure 5. Spectrograms of a clean, corrupted, and reconstructed signal (from top to bottom) for audio denoising in the GTF-NMF model with inference via EP, applied to a speech signal.

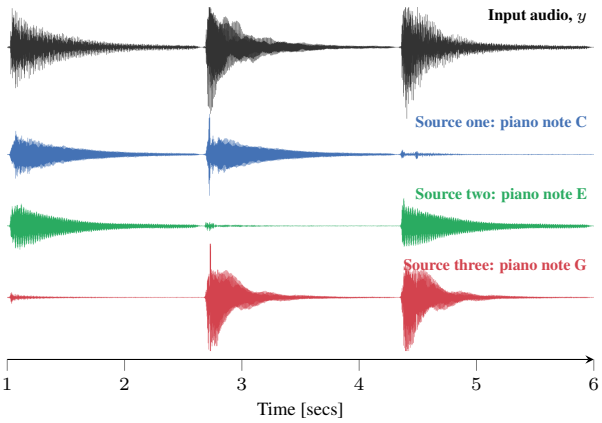


Figure 6. Infinite-horizon GP source separation example showing three piano notes (sources) recovered from a mixture signal (top), where two notes are played at a time in the original recording.

signal processing techniques if we are to build more in-depth nonstationary methods for audio analysis, and that probabilistic modelling has much potential in this domain. By applying it to various real world tasks, we have shown the flexibility of such end-to-end generative models.

For future work, it is necessary to further reduce the inherent computational burden, and to develop more efficient and robust parameter learning schemes to allow these models to become more widely used.

Acknowledgements

MRA and AS acknowledge funding from Academy of Finland grants 298742 and 308640. DS was supported by EPSRC Early Career research fellowship EP/L020505/1.

References

Adler, A., Emiya, V., Jafari, M. G., Elad, M., Gribonval, R., and Plumbley, M. D. Audio inpainting. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(3): 922–932, 2012.

Alvarado, P. A., Álvarez, M. A., and Stowell, D. Sparse Gaussian process audio source separation using spectrum priors in the time-domain. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 995–999, 2019.

Badeau, R. and Plumbley, M. D. Multichannel high-resolution nmf for modeling convolutive mixtures of non-stationary signals in the time-frequency domain. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(11):1670–1680, 2014.

Bar-Shalom, Y., Li, X.-R., and Kirubarajan, T. *Estimation with Applications to Tracking and Navigation*. Wiley-Interscience, 2001.

Bertin, N., Badeau, R., and Vincent, E. Enforcing harmonicity and smoothness in bayesian non-negative matrix factorization applied to polyphonic music transcription. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(3):538–549, 2010.

Cemgil, A. T. and Godsill, S. J. Probabilistic phase vocoder and its application to interpolation of missing values in audio signals. In *European Signal Processing Conference (EUSIPCO)*, pp. 1–4, 2005.

Dieleman, S., van den Oord, A., and Simonyan, K. The challenge of realistic music generation: modelling raw audio at scale. In *Advances in Neural Information Processing Systems 31 (NeurIPS)*, pp. 8000–8010. 2018.

Engel, J., Resnick, C., Roberts, A., Dieleman, S., Norouzi, M., Eck, D., and Simonyan, K. Neural audio synthesis of musical notes with WaveNet autoencoders. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *PMLR*, pp. 1068–1077, 2017.

Ephraim, Y. and Malah, D. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. *IEEE Transactions on acoustics, speech, and signal processing*, 32(6):1109–1121, 1984.

Griffin, D. and Lim, J. Signal estimation from modified short-time fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(2):236–243, 1984.

Hensman, J., Fusi, N., and Lawrence, N. D. Gaussian processes for big data. In *Uncertainty in Artificial Intelligence (UAI)*, pp. 282–290, 2013.

Hensman, J., Durrande, N., and Solin, A. Variational Fourier features for Gaussian processes. *Journal of Machine Learning Research (JMLR)*, 18(151):152, 2018.

Jazwinski, A. H. *Stochastic Processes and Filtering Theory*. Academic Press, New York, 1970.

Kokkala, J., Solin, A., and Särkkä, S. Sigma-point filtering and smoothing based parameter estimation in nonlinear dynamic systems. *Journal of Advances in Information Fusion*, 11(1):15–30, 2016. ISSN 15576418.

Krauth, K., Bonilla, E. V., Cutajar, K., and Filippone, M. AutoGP: Exploring the capabilities and limitations of Gaussian process models. In *Uncertainty in Artificial Intelligence (UAI)*, 2017.

Lázaro-Gredilla, M., Quiñonero-Candela, J., Rasmussen, C. E., and Figueiras-Vidal, A. R. Sparse spectrum Gaussian process regression. *Journal of Machine Learning Research (JMLR)*, 11:1865–1881, June 2010.

Lee, D. D. and Seung, H. S. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755): 788, 1999.

Liutkus, A., Badeau, R., and Richard, G. Gaussian processes for underdetermined source separation. *IEEE Transactions on Signal Processing*, 59(7):3155–3167, 2011.

Magron, P. and Virtanen, T. Complex ISNMF: a phase-aware model for monaural audio source separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(1):20–31, 2019.

Maybeck, P. S. *Stochastic Models, Estimation and Control*, volume 2. Academic Press, 1982.

McDermott, J. H., Schemitsch, M., and Simoncelli, E. P. Summary statistics in auditory perception. *Nature Neuroscience*, 16(4):493, 2013.

McNamee, J. and Stenger, F. Construction of fully symmetric numerical integration formulas of fully symmetric numerical integration formulas. *Numerische Mathematik*, 10(4):327–344, Nov 1967.

Minka, T. Divergence measures and message passing. Technical report, 2005.

Minka, T. and Lafferty, J. Expectation-propagation for the generative aspect model. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 352–359, 2002.

Minka, T. P. Expectation propagation for approximate Bayesian inference. In *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence (UAI)*, pp. 362–369, 2001.

Nickisch, H., Solin, A., and Grigorievskiy, A. State space Gaussian processes with non-Gaussian likelihood. In *International Conference on Machine Learning (ICML)*, volume 80 of *PMLR*, pp. 3789–3798, 2018.

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

Rasmussen, C. E. and Nickisch, H. Gaussian processes for machine learning (GPML) toolbox. *Journal of Machine Learning Research (JMLR)*, 11:3011–3015, 2010.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Reece, S. and Roberts, S. An introduction to Gaussian processes for the Kalman filter expert. In *Proceedings of the 13th Conference on Information Fusion (FUSION)*. IEEE, 2010.

Remes, S., Heinonen, M., and Kaski, S. Non-stationary spectral kernels. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*, pp. 4642–4651, 2017.

Saatçi, Y. *Scalable Inference for Structured Gaussian Process Models*. PhD thesis, University of Cambridge, UK, 2012.

Särkkä, S. *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.

Särkkä, S. and Solin, A. *Applied Stochastic Differential Equations*. Cambridge University Press, 2019.

Särkkä, S., Solin, A., and Hartikainen, J. Spatiotemporal learning via infinite-dimensional Bayesian filtering and smoothing. *IEEE Signal Processing Magazine*, 30(4): 51–61, 2013.

Snelson, E. and Ghahramani, Z. Sparse Gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1257–1264, 2006.

Solin, A. *Stochastic Differential Equation Methods for Spatio-Temporal Gaussian Process Regression*. Doctoral dissertation, Aalto University, Helsinki, Finland, 2016.

Solin, A. and Särkkä, S. Hilbert space methods for reduced-rank Gaussian process regression. *arXiv preprint arXiv:1401.5508*, 2014a.

Solin, A. and Särkkä, S. Explicit link between periodic covariance functions and state space models. In *Artificial Intelligence and Statistics*, pp. 904–912, 2014b.

Solin, A., Hensman, J., and Turner, R. E. Infinite-horizon Gaussian processes. In *Advances in Neural Information Processing Systems 31 (NeurIPS)*, pp. 3490–3499. 2018.

Titsias, M. K. Variational learning of inducing variables in sparse Gaussian processes. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 5 of *PMLR*, pp. 567–574, 2009.

Turner, R. E. *Statistical Models for Natural Sounds*. PhD thesis, University College London, UK, 2010.

Turner, R. E. and Sahani, M. Time-frequency analysis as probabilistic inference. *IEEE Transactions on Signal Processing*, 62(23):6171, 2014.

Wilkinson, W. J., Riis Andersen, M., Reiss, J. D., Stowell, D., and Solin, A. Unifying probabilistic models for time-frequency analysis. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 3352–3356, 2019.

Wilson, A. and Adams, R. Gaussian process kernels for pattern discovery and extrapolation. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, 2013.

Wilson, A. G. and Nickisch, H. Kernel interpolation for scalable structured Gaussian processes (KISS-GP). In *International Conference on Machine Learning (ICML)*, volume 37 of *PMLR*, pp. 1775–1784, 2015.

Zhang, Z. Parameter estimation techniques: A tutorial with application to conic fitting. *Image and Vision Computing*, 15(1):59–76, 1997.