# Gaussian process audio segmentation

# Gaussian Process Audio Segmentation

**Benjamin C. Marshall**

A thesis presented for the degree of

Doctor of Philosophy

Birkbeck, University of London

2022

I, Benjamin Marshall, declare this thesis to be all my own work.

To Natalie and Hugh.

Many thanks to my supervisors Georgios Papageorgiou, Brad Baxter and Richard Widdess for all their support; thanks to the PhD panel who's valuable comments during the viva have helped shape the final version of this thesis; a thanks quite generally to Birkbeck College and the EMS department; a special thanks also to Anthony Brooms.

## *Abstract*

This thesis extends the monophonic probability model for music introduced by D. Mumford and A. Desolneaux [Mumford and Desolneux, 2010] to polyphonic music. An algorithm is described for finding the most likely interpretation of audio under this model, and the effectiveness of this method is assessed through examples and discussion.

My algorithm combines dynamic programming, recursive grouping rules and Bayesian inference. The underlying probability model combines older ideas from signal processing with newer ideas coming out of Bayesian inference in the form of hyperpriors over the parameters governing the audio generating process.

These techniques allow the model to adapt to novel audio sources (vocals, music instruments, background noises, special effects) and thus is not easily fooled by out-of-sample audio recordings. The reason for this is that the algorithm and model only relies on a small handful of parameters. In particular, it does not rely on a training data-set of audio-transcription pairs for the model to be learned. This is in contrast to the most popular methods employed today. Nevertheless the model is sufficiently rich to allow the transcribed audio to be re-synthesised, thus allowing the user to readily assess the effectiveness of the inferred representation.



## *Keywords*

Gaussian processes, Poisson processes, dynamic programming, automatic music transcription, circulant matrices, audio segmentation, Pattern Theory.

# Contents

# 1. Introduction

The objective of this thesis is to develop a probability model for polyphonic music based on the techniques and philosophy of Pattern Theory as described in [Mumford and Desolneux, 2010]. The thesis splits into three chapters, the first chapter is introductory, providing a short literature review and situates the present thesis within this context. The second chapter then focuses on the monophonic model and ideas from [Mumford and Desolneux, 2010], and discusses a number of elaborations on this model. The final chapter presents the extension of these ideas to polyphonic music and discusses an algorithm for applying the model to real music.

## 1.1. **Thesis contributions.**

The novel contributions of the thesis centre mainly around additions to the techniques and models developed in [Mumford and Desolneux, 2010]. Below is a list of these original contributions and the location in the thesis where they can be found.

Proof for the exact determinant of the precision matrix of periodic white noise (2.1.1). Definition of *periodic Brownian motion* and derivation of precision matrix and exact determinant (2.2.1). Discovery of a Gaussian model for signals with missing even harmonics (e.g. clarinet) (2.2.2). Sparse Gaussian treatment of the famous *source-filter* model for speech, with closed form determinants and an algorithm (inc. demonstration) for performing formant analysis with this model (2.2.3). Proposal of the compound Poisson process as a model for musical note sequences (with discussion) (2.3.1). Proof for the exact determinant of the *weak string model* (2.4.1). A probability model for chords with amplitudes which vary over the course of the note, plus algorithm for parsing such a chord - identifying the individual frequencies in the chord, along with a description of how the amplitude varied (3.1.2). A probability model for random chord sequences with notes that can overlap in time (3.2.3). The binding energy algorithm for segmenting a piece of polyphonic music under

the combination of the above two models (3.2.3). Proposal to use spectral hyper-priors to improve circulant Gaussian processes so they can handle complicated sounds (3.3).

By way of demonstration of the reasonableness of these ideas, I have provided example audio clips (`indian.wav`, `segovia.wav`, `bangbang.wav`) along with their re-synthesis by the model after inference. By comparing the original audio (which can be heard in the left audio channel) with a synthesised version, inferred via the model from the original audio (presented in the right channel), it is possible to build up a good impression of what the model can and cannot do. The examples have been chosen to demonstrate some strength or weakness of the methods and not as a representation of how the model performs in general.

As discussed in the review article [Benetos et al., 2019], there are many different approaches to modelling polyphonic music. These approaches compete with one another in the annual MIREX competition. The model we present at the end of Chapter 3 ought to be submitted and evaluated against these competitors. My impression is that it won't be competitive in specific domains, but it will give reasonable results over a very wide domain. The reason for this is that the model does not use any training data, being based on very simple and general stochastic models for audio. It does, however, require a small number of tuning parameters and these parameters must be given reasonable values for the model to succeed. Ideally I'd have found a way to set these parameters automatically from the current input data, but such a method never materialised.

1.2. **Literature review.** Perhaps the closest model in the current literature to that presented in Chapter 3 is the factorial hidden Markov model developed in [Bach and Jordan, 2005]. This model supposes the polyphonic music sequence develops according to a series of parallel Markov chains and combines this with a model for the spectrum of a segment of the windowed Fourier transformed audio. Inference with this model proceeds through dynamic

programming combined with a search algorithm for deciding if a particular segment is voiced/unvoiced and also what pitches (if any) are present. We borrow much from this basic setup but also fix a number of weaknesses. Firstly, we do not pre-segment the time domain before deciding what pitches are where, because it is impossible to separate the task of identifying the pitches from identifying the location of the boundaries. We believe these tasks should be solved together and doing so results in more accurate pitches and boundaries. The reason for this is simply that if a segment contains the end of one note and the beginning of another (something which unavoidably happens when the data a segmented prior to classification) it is impossible to detect the true pitch. The algorithm presented in Chapter 3 builds larger segments[1] out of smaller segments based on whether the amplitudes and pitches in the adjacent smaller segments are sufficiently alike to warrant grouping. Likeness being based on goodness-of-fit relative to an explicit stochastic model for sound waves with such pitches and amplitudes. We do however utilise a similar search-based approach to finding the best fitting pitches as used in [Bach and Jordan, 2005] and [Cemgil et al., 2006].

Another key difference of the model in Chapter 3 with that in [Bach and Jordan, 2005] is that it utilises a model over *audio* not over spectra. This allows new audio to be synthesised, and more importantly, lends itself to more robust modelling if there are features in the audio not described by the spectral templates. This is a key aspect of the Pattern Theory

[1]By a *segment* we mean a contiguous section of the audio recording. A segment will have a start time and an end time. Two adjacent segments (of any lengths) may thus be considered grouped into a third larger segment which inherits its start time from the earlier of the two segments and its end time from the later of the two segments. Starting from the smallest segments of a recording (the individual samples), through a process of pairwise grouping, a segmentation of the entire recording emerges. This segmentation consists of the segments which are not parts of a larger segment (i.e. have not themselves been paired and grouped). The models in Chapter 2 of this thesis, only try to group adjacent segments when they are detected to have the same period. Thus the final segmentation consists of a collection of non-overlapping segments which cover the entire the recording, thus implicitly also yielding the boundaries in time across which the period of the audio is perceived (by the computer) to have changed.
This is an unsatisfactory representation for most music as it doesn't permit the end of one note to overlap with the beginning of another note (a small instance of *polyphony*). Chapter 3 of this thesis is therefore concerned with extending these pairwise grouping operations to allow the inference of *overlapping* sections. This structure could be more precisely described using the concept of a *parse tree* (see Chapter 3 in [Mumford and Desolneux, 2010] for an application of this logic to the segmentation of images).

philosophy which is that the raw data should be directly modelled (and synthesised) rather than transformed or otherwise simplified/reduced (such as is done when pre-processing by windowed Fourier transform).

An important similarity with the model in [Bach and Jordan, 2005], is that we also employ Bayesian spectral hyperpriors to allow for a wide domain of possible audio to be accurately segmented and classified. It is impossible ahead of time to pre-judge what type of instrument or sounds will be present in any audio and so this technique (we would argue) is absolutely fundamental to the project of developing a fully general model for audio. The hyperpriors we use in this thesis (log-Gaussian and Gamma) are quite different to the model in [Bach and Jordan, 2005] and this is because I have chosen them specifically to work well with circulant Gaussian processes. These priors allows the model to adapt to complex sound effects and handle unique features such as missing fundamentals as commonly occur in piano music. However, these priors are still far too generic. They have essentially been lifted out of the statistical literature and applied here to audio (we believe for the first time). It would be better if these hyperpriors can be made more specific to the problem of modelling audio.

A final weakness of [Bach and Jordan, 2005] which we have sought to remove, is the dependency on large amounts of training data. Large data is a very common feature of audio processing algorithms and is indeed the hallmark of the current state of the art in this field which employs these big data-sets to tune neural network classifiers (see [Hawthorne et al., 2017]). I do not believe these data sets to be necessary. In just the same way that Newton's laws of motion were not inferred through a data set containing the trajectories of objects, resulting in a black-box which can predict the path of an object, but can not state the equation that underlies all the paths, I believe it is possible to write down a stochastic model for all audio (i.e. the Law for audio), which makes it clear how humans can

understand such a diverse and varied set of audio inputs in totally novel and unpredictable environments. I believe the work developed in [Mumford and Desolneux, 2010] shows that this is not a pipe-dream and the extensions explicated in this thesis further support this conjecture.

A key modelling tool in the thesis is the use of Gaussian processes to describe the patterns found in audio waves. Our work builds on the basic circulant model presented in [Mumford and Desolneux, 2010] (which we call later on *periodic white noise*), and this thesis is unique in pushing the circulant model as far as possible in the space of audio modelling. Other authors have sought to use Gaussian models to model audio. For example in [Wilkinson et al., 2019] and [Alvarado and Stowell, 2016], covariance matrices specifically designed to produce periodic signals with variable amplitudes are used to classify new audio. The magic in their approach is that they can use these priors for audio to fill in missing segments of an audio wave in the same way (perhaps) a human is able to fill in the gaps in sound produced by interruptions. It's quite remarkable that they are able to achieve this with Gaussian models and this to my mind confirms the Gaussian model as the core stochastic model for processing audio. However, working with the covariance matrix in the way these authors do, is generally too inefficient for the task of large scale simultaneous segmentation and classification. The reason for this is that to evaluate a Gaussian density formulated in terms of a covariance matrix requires the covariance matrix to be inverted and this is generally too slow given the variable lengths of the clips needing to be evaluated and the number of times they need evaluating in order to search for a well fitting model. The circulant approach taken in this thesis gets around this problem by formulating models in terms of the precision matrix and further exploiting the intimate connection between circulant matrices and the Fourier transform.

This idea is of course not novel. Some of the most well known Gaussian processes (e.g. Brownian motion) are most naturally written down in term of the precision rather than the covariance. But it is not at all trivial to find new expressions for Gaussian processes in terms of the precision matrix which model the thing you are interested in. For example, in the present case of audio, the challenge is to capture the essence of an audio signal, complete with random variations, with a simple closed form equation that can be rapidly evaluated many thousands of times on the computer. The genius of D. Mumford and A. Desolneux largely solved this with their model for a musical note. This model has an extremely simple formula and works remarkably well. The details of this formula is the subject of the next chapter. We first describe their model in full detail and then offer a number of elaborations and refinements to the basic equation.

1.3. **An introduction to Gaussian processes.** All the Gaussian models discussed in this report are examples of (or derivatives of) the circulant Gaussian model. What follows is a brief review of this topic based on Chapter 2 of [Mumford and Desolneux, 2010]. Complementary discussions can be found in [Grenander, 1952], where it is remarked that [Whittle, 1951] pioneered the use of these models for time-series analysis.

1.3.1. *Circulant Gaussian processes.* A circulant Gaussian density $p(s)$ is written

$$p(s) = \left(\frac{\det Q(a)}{(2\pi)^N}\right)^{\frac{1}{2}} e^{-(s-\mu)^T Q(a)(s-\mu)/2}, \qquad (s, \mu \in \mathbb{R}^N), \tag{1}$$

where $\mathbb{E}s = \mu$ and $\mathbb{E}(s-\mu)(s-\mu)^T = Q(a)^{-1}$ and $Q(a)$ is a symmetric circulant matrix with top row $a = (a_0, \ldots, a_{N-1}) \in \mathbb{R}^N$. The next couple of paragraphs review circulant matrices. See [Davis, 2013] for more details.

By definition, a circulant matrix $M(a)$ (with top row $a$) is such that each row is a right-shifted copy of the row above (with wrap-around). Therefore the top row $a$ defines the entire matrix $M(a)$. It follows that if a circulant matrix is symmetric then $a$ has the following reflection property: $a_k = a_{N-k}$ for $k = 0, \dots N-1$. To see this, consider the following decomposition of $M(a)$ into a weighted sum of permutation matrices:

$$M(a) = \sum_{k=0}^{N-1} a_k \pi^k, \qquad \pi \equiv \begin{pmatrix} 0 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \dots & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & \dots & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & \dots & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{2}$$

This decomposition is always available for circulant matrices. Then, by looking at $\pi^0$ it is clear that $a_0$ determines the constant diagonal of $M(a) = M$. Furthermore $\pi$ shows that $a_1$ occupies both the second element of the top row and the first element of the bottom row of $M$. Hence if $M$ is symmetric, this also places $a_1$ in the last position of the top row, which by definition is occupied by $a_{N-1}$, thus $a_1 = a_{N-1}$. Similar reasoning applies to $\pi^2$ etc... showing that $a_k = a_{N-k}$ for all $k$ when $M$ is symmetric.

A computationally useful consequence of decomposition in Equation 2 is that circulant matrices (both symmetric and non-symmetric) are diagonalised in the Fourier basis with eigenvalues $\lambda_k = p_a\left(\omega^k\right)$ (for $k = 0, \dots, N-1$), where $\omega \equiv e^{2\pi i/N}$ and $p_a(x) \equiv \sum_{k=0}^{N-1} a_k x^k$. Therefore when $\mu = 0$, Equation 1 can be written in terms of an inner-product between the square modulus of the Fourier transformed data $|\hat{s}_k|^2$ and the *spectrum* $\mathbb{E}|\hat{s}|^2 \in \mathbb{R}^N$. To see this, let $F^*_{k,j} = \frac{1}{\sqrt{N}}(\omega^{kj})$ (for $k, j = 0, \dots, N-1$) be the inverse Fourier transform

matrix and $^*$ mean conjugate transpose. Then the quadratic form $s^T Q(a)s$ is equal to $(Fs)^*\text{diag}(\lambda_0, \ldots, \lambda_{N-1})(Fs)$ which evaluates to $\sum_k \lambda_k |\hat{s}_k|^2$. To show that $\mathbb{E}|\hat{s}|^2 = \lambda^{-1}$ note that since $\hat{s} = Fs$

$$\mathbb{E}|\hat{s}|^2 = \text{diag}\left(\mathbb{E}Fs(Fs)^*\right) = \text{diag}(FQ^{-1}F^*) = \lambda^{-1}. \tag{3}$$

Circulant Gaussian densities (Equation 1) are finite dimensional distributions of a class of stochastic processes known as *random Fourier expansions*. A random Fourier expansion is a stochastic function $G(x) : \mathbb{R} \to \mathbb{R}$ of the form

$$G(x) = \sum_{n \in \mathbb{Z}} A_n e^{2\pi i n x}, \qquad (x \in \mathbb{R}), \tag{4}$$

where all $\bar{A}_n = A_{-n}$ are independent mean 0 complex Gaussian variables, with variances depending only on $n$ ([Mumford and Desolneux, 2010], Page 354). Writing $\sigma_n^2 = \mathbb{E}|A_n|^2$, the following useful facts hold.

- $G$ is 1-periodic: $G(x) = G(x+1)$ for all $x$.
- The covariance function is given by $\mathbb{E}G(x_1)G(x_2) = \sum_n \sigma_n^2 e^{2\pi i n(x_1 - x_2)}$.

The connection with circulant Gaussian distributions is found by sampling $G$ on an equal-spaced grid $s \equiv (G(0), G(1/N), \ldots, G((N-1)/N)^T$. It follows that $s$ is a zero mean Gaussian with circulant covariance matrix $C(b)$ where $b_{N-k} = \sum_n \sigma_n^2 e^{2\pi i n(N-k)/N} = \sum_n \sigma_n^2 e^{-2\pi i n k/N} = b_k$. Thus the precision matrix $Q(a) = C(b)^{-1}$ is also circulant with $p_a(\omega^k) = p_b(\omega^k)^{-1}$ (by diagonalization).

1.3.2. *Sparse Gaussian Processes.* This section provides a definition of the term *sparse* in the context of Gaussian processes as used in the present thesis. There are lots of different meanings of the term sparse in mathematics, so to avoid any confusion here I make explicit

what I mean, acknowledging that perhaps this is not standard and is not a term used in [Mumford and Desolneux, 2010] either.

By a *sparse* Gaussian process, we shall mean that the number of non-zero elements in $a$ does not grow with $N$. When this holds, there is always an $O(N)$ algorithm to evaluate $p(s)$. For example, assume $a_0 \neq 0$ and $a_1 = a_{N-1} \neq 0$ but that the rest of $a$ is 0. Then inspection of Equation 2 shows that the density $p(s)$ only depends on the data $s$ through two statistics $r_0 = s^T s$ and $r_1 = s^T \pi s$, which can be computed with cost $O(N)$. The quadratic $s^T Q(a) s$ is therefore simply $a_0 r_0 + 2a_1 r_1$. To calculate $\log\det Q(a)$, first evaluate the eigenvalues

$$\lambda_k = a_0 + 2a_1 \cos(2\pi k/N), \quad k = 0, \ldots N-1,$$

and finally $\log\det Q(a) = \sum_k \log \lambda_k$, both of which are $O(N/2)$ since $\lambda_k = \lambda_{N-k}$. Non-sparse circulant Gaussian densities $p(s)$ (Equation 1) have an $O(N \log N)$ algorithm for density evaluation based on $\sum_k \lambda_k |\hat{s}|^2$ or else $\sum_k a_k r_k$ since $r_k = p_{|\hat{s}|^2}(\omega^k)$. A general Gaussian density has $O(N^3)$ time complexity ([Wood, 2015]).

1.3.3. *Simulating a Gaussian Process.* It is often useful to be able to sample from a probability model to find out what data the model assigns high probability. With this in mind, `samplegau` is a function, written in the `R` programming language ([R Core Team, 2018]) which samples from a Gaussian density with inverse spectrum $\lambda =$ `lambda` (eigenvalues of $Q(a)$).

```
samplegau = function(lambda) {
    N = length(lambda)
    z = complex(N,rnorm(N),rnorm(N))/sqrt(lambda)
    Re(fft(z))/sqrt(N)
}
```

To prove that this samples the correct model, we need to recall the following fact about mean 0 complex random vectors $z \in \mathbb{C}^N$:

$$\mathbb{E}\Re z \Re z^T = \frac{1}{2}\Re\left(\mathbb{E}zz^T + \mathbb{E}z\bar{z}^T\right)$$

In this equation, $\bar{z} \in \mathbb{C}^N$ is the complex conjugate of $z$ and $\Re z \in \mathbb{R}^N$ is the real part of $z$. The above function works by letting $z_k = (x_k + iy_k)/\sqrt{\lambda_k}$ where $x$ and $y$ are length $N$ independent, multivariate standard normal vectors and $\lambda_k = p_a(\omega^k)$. It follows then that $\mathbb{E}z_i z_j = 0$ always and $\mathbb{E}z_i\bar{z}_j = 2/\lambda_i$ if $i = j$ and is otherwise 0. The real part of $F^*z$ thus has covariance matrix

$$F^*\mathrm{diag}(\lambda_0, \ldots, \lambda_{N-1})^{-1}F = Q(a)^{-1}.$$

The code evaluates $\Re Fz$ in the last line, but since $\bar{z}$ has the same distribution as $z$, $\Re F\bar{z} = \Re\bar{F}z$ and $\bar{F} = F^*$, it follows that

$$\mathbb{E}(\Re Fz)(\Re Fz)^T = \mathbb{E}(\Re F\bar{z})(\Re F\bar{z})^T = \mathbb{E}(\Re F^*z)(\Re F^*z)^T.$$

1.3.4. *The basic limitation of circulant processes.* If $s \in \mathbb{R}^N$ is a recorded sound wave, it is natural to write $\hat{s}_k = |\hat{s}_k|e^{i\theta_k}$, for phases $\theta_k \in [-\pi, \pi]$ such that $\theta_k = -\theta_{N-k}$. Then it is clear that the circulant Gaussian density is *phase insensitive*. This is because the circulant density $p$ is a function of the sound-wave $s$ only through the statistics $|\hat{s}|^2$:

$$p(s) \propto e^{-\frac{1}{2}\sum_k \lambda_k |\hat{s}_k|^2}.$$

A consequence of this is that one can take a short sound recording $s$, and sample a circulant Gaussian density with the same spectrum as $s$, that is by using `samplegau` with $\lambda = |\hat{s}|^{-2}$, and quite often this produces something totally unlike $s$, because the recognisable content in

$s$ is contained in the phases. Furthermore, it is currently understood that phase is important to accurate detection of note onsets, see [Benetos and Stylianou, 2010] for discussion. Better models for sounds could be found by moving away from the Gaussian distribution and using a measure on $\mathbb{R}^N$ with density

$$p(s) \propto e^{-\sum_k \lambda_k |\hat{s}_k|^2} f(\theta_0, \ldots, \theta_{N-1})$$

for some function $f \geq 0$. This idea is not explored in this essay and consequently when analysing audio in the sequel it must be remembered that we have thrown away all information contained in the phases.

## 2. Monophonic probability models

In this chapter the main idea is to define *sparse* Gaussian process models and then apply them to the problem of analysing a sound-wave $s \in \mathbb{R}^N$ by segmenting it into non-overlapping regions according to the model which best describes the statistics of $s$ restricted to that region. Sparsity ensures density evaluation is $O(N)$ and therefore this methodology results in very fast analysis algorithms.

This chapter has four sections. The first section describes the foundational work of [Mumford and Desolneux, 2010]. In that book the authors define a sparse Gaussian model for approximately periodic sounds and then go on to show that a dynamic programming algorithm can be used to extract and classify the notes appearing in a monophonic[2] piece of music. The way this works is to assume the notes break up the soundwave according to a Poisson process and then within each region the periodic Gaussian model describes the

[2]Where only one sound can happen at a time.

statistics of the sound of each note. The method is so elegant that it cries out for further development.

The second section introduces new sparse models for audio. The main model combines Brownian motion with the periodic model just described to produce a Gaussian density which is sparse, smooth and approximately periodic. This smooth model is shown to be a more accurate model for music sounds. Resonance is then added to the mix. Resonance is used to classify vowel sounds in speech and also appears in music as well. We apply this model to the problem of formant analysis - a sub-problem of the much larger speech recognition problem. Also described in this section, is a model for musical sounds produced by tubes which are open at one end and closed at the other (e.g. a clarinet) and also it is shown how certain members of the popular '$1/f^{\alpha}$' noise models can be given a sparse treatment. The research project here is to generate a large family of these sparse Gaussian processes that can then be applied wholesale to all one-dimensional signals.

In the third section of this chapter, we return to the problem of segmenting monophonic signals and apply the periodic Brownian motion model to a piece of Indian classical music. An important development is that it is possible to allow each note in a piece of music to posses it's own amplitude and estimate this amplitude at no additional computational cost when transcribing music. This experiment is then followed by a discussion on the use of compound Poisson processes as more realistic models for music (for certain Levy measures) and a dynamic programming algorithm is derived for segmenting audio with these models.

For the final section, we shift gears somewhat and extend the models and methods to cases where the mean of the process being modelled is not assumed 0. We develop marginal maximum likelihood inference for a model which supposes the process has piecewise linear means and restored Brownian motion errors. In the process of developing this algorithm

we derive closed form expressions for the determinant of restored Brownian motion where previously only approximations have been known.

2.1. **The Mumford-Desolneux model for music.** Sounds produced by musical instruments, such as a flute or an oboe, are characterised by being strongly periodic. What this means is that if $s \in \mathbb{R}^n$ is a sound-wave with period $j$ then $s_k \approx s_{k+j}$ for all $k$. One popular approach to detecting this period automatically is based on looking for the $j$ for which the sum $\sum_k (s_k - s_{k+j})^2$ is smallest ([De Cheveigné and Kawahara, 2002]). For example, if $\Delta_t$ gives the time (in seconds) between adjacent samples $(s_k, s_{k+1})$, then a period of $j$ samples corresponds to a frequency of $1/(j\Delta_t)$ hertz. According to ISO 16:1975, the musical note $E_4$ has a frequency of about 329 hertz. Thus if $1/\Delta_t = 44100$, and $s$ is an $E_4$ then $\sum_k (s_k - s_{k+j})^2$ will have a local min at $j \approx 44100/329 \approx 134$.

In ([Mumford and Desolneux, 2010], Page 76) this idea is turned into a full probability model over musical sounds by assuming a Gaussian model, with precision $Q_j$ defined by

$$p_j(s) = \left(\frac{\det Q_j}{(2\pi)^n}\right)^{\frac{1}{2}} e^{-\alpha \sum_{k=0}^{n-1} (s_k - s_{k+j \pmod n})^2/2 - \beta \sum_{k=0}^{n-1} s_k^2/2}, \quad (s \in \mathbb{R}^n). \tag{5}$$

Identifying $Q_j$ with the circulant $Q(a)$, it is simple to read off the components of $a$: $a_0 = 2\alpha+\beta$, $a_j = a_{n-j}$ and $a_j + a_{n-j} = -2\alpha$ where $\alpha, \beta > 0$ and the rest of $a$ is 0. In what follows I will refer to the stochastic process defined by this density as *periodic white noise* (the authors did not give it a name). The name is appropriate because the magnitudes of $\alpha$ and $\beta$ control the amplitude of the signal and the strength of the periodicity. When $\alpha = 0$, the process reduces to white noise with variance (i.e. amplitude) proportional to $1/\beta$. As $\alpha$ increases away from 0, deviations away from periodicity (i.e. large values in the terms $(s_k - s_{k+j})^2$) reduce the density more intensely and thus the process takes on the characteristics of a noisy periodic wave.

The authors (ibid. Page 89) then generalise this model to melodies. A melody $s = (s_1, \ldots, s_N) \in \mathbb{R}^N$ contains some unknown number $m$ of notes, concatenated, but with varying duration $\{n_i\} = \{n_1, \ldots, n_m\}$ such that $\sum_i n_i = N$. Let $\{t_i\}$ denote the start times $1 = t_1 < t_2 < \ldots t_m \leq N$ of the notes and let $j_i \in J$ denote the $i$th period. The full probability model for a melody is written

$$p(s, m, \{n_i\}, \{j_i\}) = \frac{\theta^{m-1} e^{-\theta(N-1)\Delta_t} \sqrt{\prod_{i=1}^{m} \det Q_{j_i}}}{|J|^m (2\pi)^{N/2}} e^{-(2\alpha+\beta)s^T s/2 + \alpha \sum_{i=1}^{m} s(i)^T \pi^{j_i} s(i)}, \quad (6)$$

where $s(i) = s_{[t_i, t_i+n_i)}$. Hence the start times $\{t_i \cdot \Delta_t\}$ are a Poisson process with rate $\theta > 0$ and the the periods $\{j_i\}$ are distributed independent uniform over $J$. See [Cox and Lewis, 1966] for a derivation of the density of an ordinary Poisson process. Note that $n_m = N - t_m + 1$ is a lower bound on the last duration because the last event may have extended beyond the duration of the recording. Solving a model like this amounts to finding the best guesses for the note start times $\{t_i\}$ and periods $\{j_i\}$ given some fixed input recording $s = (s_1, \ldots, s_N)$. As proposed in [Mumford and Desolneux, 2010], and will be shown later, this can be done via dynamic programming.

2.1.1. *Exact and approximate determinants.* In this section we effectively solve some exercises in the Chapter 2 of the book [Mumford and Desolneux, 2010]. These exercises are offered without solutions (of course) and we provide our solution to them here because I believe they go beyond the requirements of the exercises. The exercises ask for various *approximate* results about the determinants of certain circulant matrices, what follows are derivations of the *exact* determinants.

For computations with $j-$periodic white noise $p_j$, we need the determinant of the matrix $Q$ defined by the quadratic form

$$(x_0, \dots, x_{N-1}) \longmapsto (2\alpha + \beta) \sum_{k=0}^{N-1} x_k^2 - 2\alpha \sum_{k=0}^{N-1} x_k x_{k+j \pmod N}$$

where $\alpha, \beta > 0$. Looking at this equation, it is clear there are $j$ chains of linked variables of the form $D_m = \{m + kj \pmod N : k \in \mathbb{Z}\}$ for $m = 0, \dots, j-1$. The index of every data point is in at least one chain. Moreover, if $D_m$ and $D_n$ share at least one index then they are identical. The determinant is a function of the size and number of distinct chains. For example, when $N = qj$ where $q$ is some integer, then there are $j$ distinct chains each of length $q$. Hence $\det Q = (\det C_q)^j$ where

$$C_q = \begin{pmatrix} 2\alpha+\beta & -\alpha & 0 & \dots & 0 & -\alpha \\ -\alpha & 2\alpha+\beta & -\alpha & 0 & \dots & 0 \\ 0 & -\alpha & 2\alpha+\beta & -\alpha & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \dots & 0 & -\alpha & 2\alpha+\beta & -\alpha & 0 \\ 0 & \dots & 0 & -\alpha & 2\alpha+\beta & -\alpha \\ -\alpha & 0 & \dots & 0 & -\alpha & 2\alpha+\beta \end{pmatrix} \in \mathbb{R}^{q \times q}.$$

More generally, if $N = qj + r$ (for $r = 0, 1, \dots, j-1$) then the determinant becomes

$$\det Q = \prod_k \det C_{\text{mult}(c_k)},$$

where $(c_1, c_2, \dots)$ are the cycles in the cyclic permutation that maps $r$ to 0 and $\text{mult}(c_k) = q|c_k| + |\{i \in c_k : i < r\}|$.

In [Mumford and Desolneux, 2010] $(103-104)$ it is shown that

$$\det C_q \approx q^2 \beta \alpha^{q-1} \approx \left( \frac{2\alpha + \beta + \sqrt{\beta^2 + 4\alpha\beta}}{2} \right)^q \tag{7}$$

when $\alpha \gg \beta > 0$. But in fact the exact determinant of $C_q$ can be found (in two very different ways). Indeed, we will now derive a closed form formula for the determinant of a matrix $\tilde{C}_q$ with same 0 elements as $C_q$ but with $x = 2|\alpha| + \beta$ on the main diagonal and $y = \alpha$ on the non-zero off diagonals, where $\alpha, \beta \in \mathbb{R}$. (This matrix $\tilde{C}_q$, is slightly more general than $C_q$ and it will crop up when we examine the case of open-closed tubes later on.) We shall show that

$$\det \tilde{C}_q = \left( \frac{2|\alpha| + \beta + \sqrt{\beta^2 + 4|\alpha|\beta}}{2} \right)^q + \left( \frac{2|\alpha| + \beta - \sqrt{\beta^2 + 4|\alpha|\beta}}{2} \right)^q - 2(-\alpha)^q. \tag{8}$$

The proof comes from noticing that $\tilde{C}_q$ is very nearly tri-diagonal. Then expanding along the top row: $\det \tilde{C}_n =$

$$x \det \begin{pmatrix} x & y & 0 \dots & 0 \\ y & x & y & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \dots & 0 & y & x & y \\ 0 & \dots & 0 & y & x \end{pmatrix} - y \det \begin{pmatrix} y & y & 0 \dots & 0 \\ 0 & x & y & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \dots & 0 & y & x & y \\ y & \dots & 0 & y & x \end{pmatrix} + \phi_n y \det \begin{pmatrix} y & x & y & 0 \dots \\ 0 & y & x & y \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 \dots & 0 & y & x \\ y & 0 & \dots & 0 & y \end{pmatrix},$$

where $\phi_n = 1$ when $n$ is odd and $-1$ when $n$ is even. The first matrix is tri-diagonal $T_{n-1}$. The second matrix is tri-diagonal $T_{n-2}$ in the bottom right and lower triangular in the top right. The third matrix is upper triangular in the top right and tri-diagonal $T_{n-2}$ in the bottom right. Hence we have the following

$$\det \tilde{C}_n = x \det T_{n-1} - y(y \det T_{n-2} + \phi_{n-1} y^{n-1}) + \phi_n y(y^{n-1} + \phi_{n-1} y \det T_{n-2}).$$

It is well known that the determinant of a tri-diagonal matrix satisfies the following second order recurrence relation

$$\det T_{n+2} = x \det T_{n+1} - y^2 \det T_n \tag{9}$$

with boundary conditions $\det T_0 = 1$ and $\det T_1 = x$. Hence the circulant determinant that we are trying to calculate is more neatly expressed as

$$\det \tilde{C}_n = \begin{cases} \det T_n & (n = 0, 1, 2), \\ \det T_n + 2\phi_n y^n - y^2 \det T_{n-2} & (n > 2). \end{cases} \tag{10}$$

To solve Equation 9, we can use the standard method (as say described in [Grimmett and Welsh, 2014]). First, the solutions of $\theta^2 - x\theta + y^2 = 0$ look like

$$\theta_1 = \frac{x + \sqrt{x^2 - 4y^2}}{2}, \qquad \theta_2 = \frac{x - \sqrt{x^2 - 4y^2}}{2},$$

which are both real since since $x^2 - 4y^2 = \beta^2 + 4|\alpha|\beta > 0$. Secondly, we have that $\det T_n = c_1\theta_1^n + c_2\theta_2^n$, where $c_1 + c_2 = 1$ and $c_1\theta_1 + c_2\theta_2 = x$ from the boundary conditions. Hence $c_1 = \frac{x-\theta_2}{\theta_1-\theta_2} = \frac{\theta_1}{\theta_1-\theta_2}$ and $c_2 = 1 - c_1$. Thus

$$\det T_n = \frac{\theta_1^{n+1} - \theta_2^{n+1}}{\theta_1 - \theta_2} = \sum_{k=0}^{n} \theta_1^{n-k}\theta_2^k.$$

Plugging this last expression into Equation 10 and noting that $\theta_1\theta_2 = y^2$ gives $\det \tilde{C}_n = \theta_1^n + \theta_2^n + 2\phi_n y^n$, completing the proof.

An alternative approach to the problem of finding the determinant of $Q$ is to re-parametrise the matrix as follows. Let $\beta_0 > 0$ and $\gamma \in [0, 1)$ and define the quadratic form

$$\begin{aligned} x^T Q x &= \beta_0 \sum_k x_k^2 + \frac{\beta_0 \gamma}{(1-\gamma)^2} \sum_k (x_k - x_{k+j \pmod N})^2 \\ &= \beta_0 \frac{1+\gamma^2}{(1-\gamma)^2} \sum_k x_k - \frac{2\beta_0 \gamma}{(1-\gamma)^2} \sum_k x_k x_{k+j \pmod N} \\ &= \frac{\beta_0}{(1-\gamma)^2} \sum_k (x_k - \gamma x_{k+j \pmod N})^2. \end{aligned}$$

This is equivalent to the original matrix when $\beta_0 = \beta$ and $f(\gamma) = \gamma/(1-\gamma)^2 = \alpha/\beta$. Furthermore $f' = \frac{1+\gamma^2}{(1-\gamma)^4}$ is strictly positive and thus $f$ is strictly increasing from 0 to $\infty$ as $\gamma$ ranges from 0 to 1, so that $f$ is invertible. Consequently we can always re-express $(\alpha, \beta)$ in terms of $(\gamma, \beta_0)$ and vice versa. But now it is easy to see that

$$\det Q = \beta_0^N \det Q(a)/(1-\gamma)^{2N},$$

where $a_0 = 1 + \gamma^2$, $a_j = a_{N-j} = -\gamma$ and the rest 0. Hence to find the determinant of $Q$, all that is needed is the determinant of $Q(a)$ and this is given by

$$\det Q(a) = (1 - \gamma^{N/j})^{2j} \tag{11}$$

whenever $N/j = q$ is an integer. To see this, write $\omega = e^{2\pi i/N}$ and $\omega_0 = e^{2\pi i/q}$ and consider the expression for the determinant in terms of the eigenvalues $\lambda_k = p_a(\omega^k)$ of $Q(a)$:

$$\begin{aligned} \det Q(a) &= \prod_{k=0}^{N-1} (1 + \gamma^2 - \gamma\omega^{kj} - \gamma\omega^{k(N-j)}) = \prod_{k=0}^{N-1} |1 - \gamma\omega^{jk}|^2 \\ &= \prod_{k=0}^{N-1} |1 - \gamma\omega_0^k|^2 = \left( \prod_{k=0}^{q-1} |1 - \gamma\omega_0^k|^2 \right)^j = (1 - \gamma^q)^{2j}. \end{aligned}$$

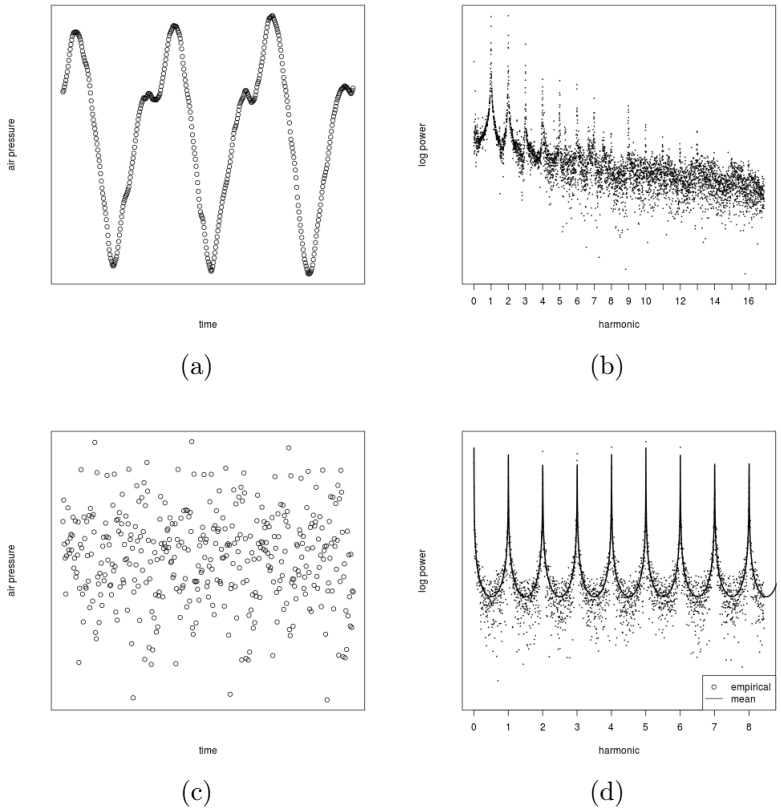



FIGURE 1. Plots comparing periodic white noise with a real musical sound. (a) A sound-wave of a flute which is approximately periodic (with period 134 samples) and locally very smooth. (b) The (log) empirical spectrum of the flute soundwave, where the x-axis marks off the location of the harmonics. The power in this spectrum is seen to rapidly decrease with increasing harmonic. (c) A sample from approximately periodic white noise with the same period as the recorded flute. This waveform is approximately periodic, but unlike the flute, is very noisy. (d) The empirical and mean spectrum of the sample from periodic white noise - these spectra posses harmonics and have constant power.

The last part follows from the fact that the numbers $\omega_0^k$ for $k = 0, \ldots, q-1$ solve $z^q = 1$ and thus $z^q - 1 = \prod_{k=0}^{q-1}(z - \omega_0^k)$ for all $z \in \mathbb{C}$.

2.2. **Sparse acoustic models.** Consider Figure 1. The top row displays the waveform and log empirical spectrum[3] of a recorded flute playing the note $E_4$ (329 Hz). The recording has

[3]The *empirical spectrum* of $s \in \mathbb{R}^N$ is the squared modulus $|\hat{s}|^2$ of the discrete Fourier transform $\hat{s} \in \mathbb{C}^N$ of $s$, defined by $\hat{s}_k = N^{-\frac{1}{2}} \sum_{j=0}^{N-1} s_j \omega^{-jk}$ for $k = 0, \ldots, N-1$.

a sampling rate of $1/\Delta_t = 44100$ samples per seconds. The empirical spectrum contains sharp peaks at integer multiples of $N/j_0$ where $j_0 = 134$ is the integer period. The locations $N/j_0, 2N/j_0, 3N/j_0, \ldots$ are called the harmonics of $s$ and $N/j_0$ is the fundamental frequency. The waveform is clearly highly periodic and locally very smooth. Note that here we are talking about temporal smoothness, which essentially means that adjacent samples $s_i$ and $s_{i+1}$ are near each other with high probably, as opposed to spectral smoothness, which would say that $\hat{s}_i$ and $\hat{s}_{i+1}$ are near each other with high probability. Temporal smoothness generally arises when power in the spectrum decreases rapidly as frequency increases.

Contrast this with the same analysis run on a sample from approximately periodic white noise $p_{j_0}$ (Equation 5). The plots are displayed in the second row of the same figure. Since the top row $a$ of the precision matrix of $p_{j_0}$ is given by $a_0 = 2\alpha + \beta$, $a_{j_0} = a_{N-j_0} = -\alpha$, and as described in the introduction, the spectrum of a circulant Gaussian can be expressed as the complex polynomial $\mathbb{E}|\hat{s}_k|^2 = p_a(\omega^k)^{-1}$ it follows that the spectrum of periodic white noise is given by an inverted squared sine wave:

$$\mathbb{E}|\hat{s}_k|^2 = 1/\left(2\alpha + \beta - 2\alpha\cos(2\pi j_0 k/N)\right) \tag{12}$$

$$= 1/\left(\beta + 4\alpha\sin^2(\pi j_0 k/N)\right), \tag{13}$$

for $k = 0, \ldots, N-1$, where the above follows from the trigonometric relation $2\sin^2(x) = 1 - \cos(2x)$. The spectrum of the model is overlaid on the empirical spectrum. The shape of the harmonics captures *precisely*[4] the shape of the flute harmonics, but in contrast to the flute, the power in the harmonics is constant when it ought to rapidly decay with increasing frequency and consequently the waveform of a realisation from $p_j$ is not smooth. We therefore conclude that $p_j$ is not a wholly accurate model for flute notes. In fact, most harmonic

[4]An astonishing match between reality and mathematics.

sounds produced by actual musical instruments have a spectrum with decreasing power in the harmonics and thus much smoother wave-forms.

To provide a better model for musical sounds, in this section we define *approximately periodic Brownian motion*, which is sparse, smooth and approximately periodic. This model arises by passing approx. periodic white noise through the geometric filter:

$$s_k = (1-\gamma)\sum_u \gamma^u w_{k-u}, \quad k = 0, \ldots, n-1.$$

for $\gamma \in [0,1]$ where $w \sim p_j(w)$. We will denote the resulting density $b_j(s)$ and derive it's basic properties in the next section. After this we derive a sparse periodic model for the case of the clarinet which lacks certain harmonics and then derive a sparse model incorporating resonance and apply this to the problem of speech recognition.

2.2.1. *Periodic Brownian motion.* The standard approach to smoothing a signal $s(x) : \mathbb{R} \to \mathbb{R}$ is to pass it through a filter. The simplest such filter is (perhaps) the *exponential filter* $\delta_0 e^{-\delta_1 x}$, vanishing for $x < 0$ with $\delta_0, \delta_1 > 0$. Letting $G(x) : \mathbb{R} \to \mathbb{R}$ be a random Fourier expansion then a smooth Fourier expansion $s(x)$ can be defined by

$$s(x) = \delta_0 \int_{u \geq 0} e^{-\delta_1 u} G(x-u)du = \sum_n B_n e^{2\pi i n x},$$

where

$$B_n = \bar{B}_{-n} = \frac{\delta_0 A_n(\delta_1 - 2\pi i n)}{\delta_1^2 + (2\pi n)^2}.$$

Inspection of the variance

$$\mathbb{E}|B_n|^2 = \frac{\delta_0 \sigma_n^2}{\delta_1^2 + (2\pi n)^2} \tag{14}$$

shows that $s(x)$ retains the structure in $G$'s spectrum $\{\sigma_n^2\}$ while forcing the power to decrease like $1/(n^2 + \text{const.})$.

A useful class of noise models are $1/f^\alpha$ noises for $\alpha \geq 0$ ([Lindgren and Sandsten, 2014], Page 88 and [Mumford and Desolneux, 2010], Page 74). These models have a spectrum which goes down like $1/n^\alpha$. When $\alpha = 0$, the result is white noise, when $\alpha = 1$ the model is pink noise and when $\alpha = 2$ the result is Brownian motion. Hence exponential smoothing results in a signal whose waveform looks locally like a Brownian motion path.

To derive a sparse smooth Gaussian it is simpler to work in discrete time where the analog of the exponential filter is the *geometric smoother*[5]. Let $w = (w_0, \ldots, w_{N-1})$ be an input sequence, $\gamma$ a constant in $[0, 1)$. A geometric smooth output sequence $s_k$ is defined by

$$s_k = (1 - \gamma) \sum_{u=0}^{\infty} \gamma^u w_{k-u \pmod N} \tag{15}$$

for $k = 0, \ldots, N - 1$. Let's derive the spectrum of $s$. Since the $k$th entry in the Fourier transform of a $u$-shifted sequence $(w_{0-u}, w_{1-u}, \ldots, w_{N-1-u})$ (subscripts modulo $N$) is simply the Fourier transform of the original sequence pre-multiplied by $\omega^{-uk}$, for $k = 0, \ldots, N - 1$, we have

$$\mathbb{E}|\hat{s}_k|^2 = (1 - \gamma)^2 \mathbb{E}|\hat{w}_k|^2 | \sum_{u=0}^{\infty} \gamma^u \omega^{-uk}|^2 = \frac{(1 - \gamma)^2}{|1 - \gamma\omega^{-k}|^2} \mathbb{E}|\hat{w}_k|^2.$$

On the right hand side of the above equation we have the eigenvalues of $Q(a)^{-1}$ being multiplied by $(1 - \gamma)^2/(1 + \gamma^2 - \gamma\omega^k - \gamma\omega^{N-k})$ which are the eigenvalues of the matrix $Q(b)^{-1}$ where $b_0 = (1+\gamma^2)/(1-\gamma)^2$ and $b_1 = b_{N-1} = -\gamma/(1-\gamma)^2$ and the rest 0. Hence if $w$ is Gaussian with precision $Q(a)$ it must be that $s$ is Gaussian with precision $Q(c) = Q(b)Q(a)$

[5]See Chapter 6 in [Lindgren and Sandsten, 2014] for an account of the exponential and geometric linear filters applied to Gaussian processes

where

$$c_k = \frac{(1+\gamma^2)a_k - \gamma(a_{k-1} + a_{k+1})}{(1-\gamma)^2}, \qquad k = 0, \ldots, N-1, \tag{16}$$

and we define $a_{-1} = a_{N-1} = a_1$ and $a_N = a_0$. Furthermore, using Equation 11, geometric filtering modifies the determinant as follows

$$\det Q(c) = \det Q(b) \cdot \det Q(a) = \frac{(1-\gamma^N)^2}{(1-\gamma)^{2N}} \cdot \det Q(a). \tag{17}$$

Consequently, geometric smoothing preserves sparsity and closed form determinant calculations. This makes it a suitable for use in the Poisson process model for processing melodies previously described. Let's look at three examples.

Example (a): $w \sim$ white noise. When $w$ is white noise (precision $Q(a)$, $a_0 = \beta$, rest $a_k = 0$) and $\gamma \approx 1$ we get Brownian motion. To see this apply Equation 16. This gives the precision of $s$ as $Q(c)$ with $c_0 = \beta(1+\gamma^2)/(1-\gamma)^2$ and $c_1 = c_{N-1} = -\gamma/(1-\gamma)^2$ (the rest 0). Defining $\beta_0 = \beta/(1-\gamma)^2$, when $\gamma \approx 1$ the following approximation holds

$$\frac{1}{z} e^{-\beta_0 \sum_{k=0}^{N-1} (s_k - \gamma s_{k+1 \pmod N})^2/2} \approx \frac{1}{z} e^{-\beta_0 s_0^2/2 - \beta_0 \sum_{k=0}^{N-2} (s_k - s_{k+1})^2/2}.$$

On the left is the density of $s$, on the right is the Brownian motion density. The term $\sum_{k=0}^{N-2}(s_k - s_{k+1})^2$ can be understood as a Riemann sum approximation to the integral of the square of the derivative of the signal $\int (\nabla_x s(x))^2 dx$. Thus geometric smoothing works by constraining the variance of the first derivative.

Above we said that geometric smoothing is the analog of exponential smoothing. A nice way to see this is to calculate the spectrum of geometrically smoothed input $w$ (where

$\mathbb{E}|\hat{w}_k|^2 = \sigma_k^2$).

$$\mathbb{E}|\hat{s}_k|^2 = \frac{\sigma_k^2}{1+\gamma^2-2\gamma\cos(2\pi k/N)} \approx \frac{N^2\sigma_k^2/\gamma}{N^2(1-\gamma)^2/\gamma+(2\pi k)^2}, \tag{18}$$

for $k = 0, \ldots, N/2$ .This approximation makes use of the small angle formula $\cos(x) \approx 1-x^2/2$ for small $x$. Compare the resulting expression with the exponential smoothing spectrum Equation 14.

Example (b): $w \sim$ approximately periodic. When $w$ is a sample from $p_{j_0}$ so that $w$ has precision matrix $Q(a)$ for $a_0 = 2\alpha+\beta$ and $a_{j_0} = a_{N-j_0} = -\alpha$, where we will assume $j_0 > 1$ for simplicity, then geometric smoothing $w$ (Equation 15) results in *approximately periodic Brownian motion* $s$ - which has has precision matrix $Q(c)$ with $c_0 = \gamma_0(2\alpha+\beta)$, $c_1 = -\gamma_1(2\alpha+\beta)$, $c_{j_0} = -\gamma_0\alpha$ and $c_{j_0-1} = c_{j_0+1} = \gamma_1\alpha$ where $\gamma_0 = (1+\gamma^2)/(1-\gamma)^2$ and $\gamma_1 = \gamma/(1-\gamma)^2$. The quadratic form for this $j_0$-periodic Brownian motion looks like

$$s \mapsto (2\alpha+\beta)(\gamma_0 r_0 - 2\gamma_1 r_1) - 2\alpha\left(\gamma_0 r_{j_0} - \gamma_1 r_{j_0-1} - \gamma_1 r_{j_0+1}\right) \tag{19}$$

where $r_j = \sum_{k=0}^{N-1} s_k s_{k+j \mod N}$. The quadratic retains the same basic structure as $Q(a)$ : $s \mapsto (2\alpha+\beta)r_0 - 2\alpha r_{j_0}$ but differs in that weighted averages of $r_{j-1}$, $r_j$, $r_{j+1}$ are used in place of the original $r_j$ since $\gamma_0 - 2\gamma_1 = 1$. Another way of writing the quadratic would emphasise the presence of a first derivative penalty

$$\begin{aligned} s^T Q(c) s = &\sum_k \beta s_k^2 + \gamma_1(2\alpha+\beta)(s_k - s_{k+1})^2 + \gamma_0\alpha(s_k - s_{k+j})^2 \\ &- \gamma_1\alpha\sum_k (s_k - s_{k+j-1})^2 + (s_k - s_{k+j+1})^2. \end{aligned}$$

Combining the approximate and exact determinant results we have

$$\det Q(c) \approx \frac{\left(1-\gamma^N\right)^2}{(1-\gamma)^{2N}} \left(q_0^2 \beta \alpha^{q_0-1}\right)^{j_0} \tag{20}$$

where $q_0 = N/j_0$. A plot of the the spectrum of this model, fit to the flute sample, is shown in Figure 2. The model has captured the decreasing power in the harmonics. Sampling the fitted model results in a far smoother waveform also shown in the same figure. The (minimised) negative log likelihood - evaluated on the flute sample - without geometric smoothing is $251,842$ and the estimated parameters are $\log \alpha \approx -9$ and $\log \beta \approx -20$ (shown the second row of Figure 1). But with additional geometric smoothing, the negative log likelihood is brought closer to 0 by about 1 quarter: $190,182$. The parameter estimates are $\log \alpha \approx -15$, $\log \beta \approx -21$ and $\gamma = .98$. Despite this huge improvement in the fit, there are clear discrepancies between the model and the flute spectrum. In particular, the first two harmonics are significantly underestimated and the latter harmonics (above no. 7) are overestimated.

Example (c): $w \sim$ smooth and approximately periodic. Let $w = (w_0, \ldots, w_{N-1})$ be approximately $j_0-$periodic Brownian motion (with smoothing parameter $\gamma$). Geometric smoothing $w$ (with smoothing parameter $\epsilon$) results in a doubly smooth $j_0-$periodic signal $s$. The precision matrix $Q(c)$ of $s$ is sparse, with determinant

$$\det Q(c) \approx \frac{\left(1-\epsilon^N\right)^2 \left(1-\gamma^N\right)^2}{(1-\epsilon)^{2N} (1-\gamma)^{2N}} \left(q_0^2 \beta \alpha^{q_0-1}\right)^{j_0}.$$

The quadratic form of $Q(c)$ is not particularly nice to write down, so to understand what twice smoothing amounts to, let's analyse the simpler case where the input is white noise instead. In this case, the precision matrix of the output signal $x = (x_0, \ldots, x_{N-1})$ has

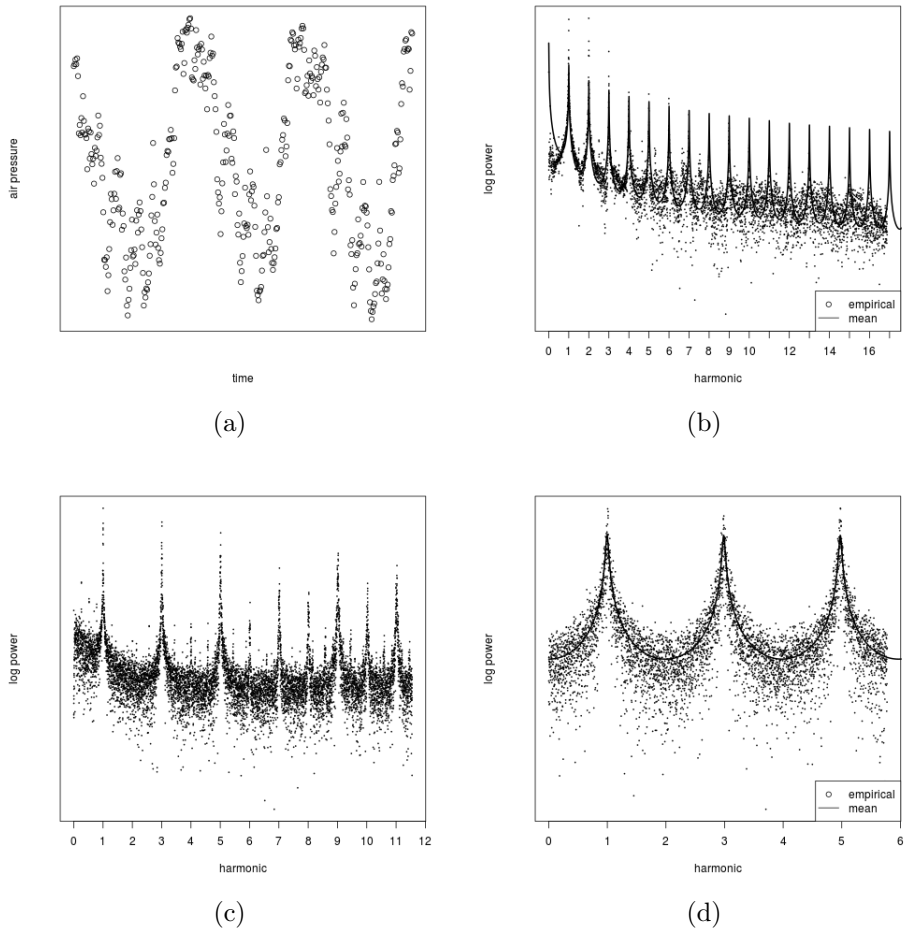



FIGURE 2. (a) A realisation from periodic Brownian motion. The path is still periodic but locally much smoother than periodic white noise. (b) The empirical spectrum of the flute note with the fitted periodic Brownian motion spectrum superimposed - note that the harmonics now rapidly decrease with frequency. (c) The empirical spectrum of a clarinet, note that the even harmonics are significantly weaker than the odd harmonics. (d) An empirical and mean spectrum of the sparse Gaussian process for sounds with missing even harmonics introduced here.

quadratic form

$$x \mapsto \beta \sum_k \left( x_k^2 + (\gamma_1 + \epsilon_1)(x_k - x_{k+1})^2 + \gamma_1 \epsilon_1 (x_{k+1} - 2x_k + x_{k-1})^2 \right) \tag{21}$$

where $\epsilon_0 = (1+\epsilon^2)/(1-\epsilon)^2$, $\epsilon_1 = \epsilon/(1-\epsilon)^2$ and the sums are modulo $N$. The term $\sum(x_{k+1} - 2x_k + x_{k-1})^2$ is a Riemann sum approximation to the integral of the square second derivative of the signal in continuous time:

$$\int \left(\nabla_t^2 x(t)\right)^2 dt \approx \sum \left( \frac{x_{k+1} - x_k}{\Delta_t^2} - \frac{x_k - x_{k-1}}{\Delta_t^2} \right)^2 \Delta_t$$
$$\propto \sum (x_{k+1} - 2x_k - x_{k-1})^2.$$

By considering a second application of the argument that lead to Equation 18 we see that the spectrum $\mathbb{E}|\hat{x}_k|^2$ of $(x_0, \ldots, x_{N-1})$ is proportional to $1/k^4$, placing this model within the family of $1/f^\alpha$ noise models. Repeated application of the geometric smoother therefore produces all $1/f^\alpha$ models with even integer $\alpha$.

In the spatial statistics literature the non circulant version of this model is referred to as a second order random walk ([Rue and Held, 2005], Page 110) and in the computer vision literature $(x_0, \ldots, x_{N-1})$ is referred to as a 'snake' ([Mumford and Desolneux, 2010], page 132). Snakes serve as the basic model for boundaries between objects in an image. The circulant version of this model derived above would only make sense if the boundary being modelled forms a (possibly self-intersecting) closed loop. See [Grenander, 1996] for examples of this idea applied to potatoes.

2.2.2. *Open-closed tubes.* The second row of Figure 2 shows the spectrum of a recorded clarinet. Unlike the flute, which possess all harmonics $N/j_0, 2N/j_0, 3N/j_0, \ldots$, the clarinet only has odd harmonics $N/j_0, 3N/j_0, 5N/j_0 \ldots$. Strictly speaking, the clarinet has significantly weaker even harmonics. In particular, the clarinet lacks completely a second harmonic $2N/j_0$ and barely has a fourth $4N/j_0$. Possessing weak even harmonics is a common property of periodic signals generated by tubes which are open at one end and restricted/closed at the

other The clarinet fits this description because the mouthpiece end is kept mostly closed by the presence of a reed[6]. The flute on the other hand, does not use a reed (or anything else) to restrict the mouthpiece-end and as can be seen in Figure 1, plot (b), a flute note contains all harmonics.

Is there a sparse Gaussian model for periodic signals lacking even harmonics? Some experimentation modifying Equation 13 gives $\cos^2(\pi j_0 k/N)$ as the correct basis terms for a period $2j_0$ signal with only odd harmonics:

$$\mathbb{E}|\hat{s}_k|^2 = 1/\left(\beta + 4\alpha\cos^2(\pi j_0 k/N)\right) \tag{22}$$

$$= 1/\left(2\alpha + \beta + 2\alpha\cos(2\pi j_0 k/N)\right) \tag{23}$$

using the fact that $2\cos^2(x) = \cos(2x) + 1$. This spectrum is shown in panel (d) of Figure 2. Notice that the spectrum only contains peaks at $1, 3, 5, \ldots$. Working backwards from the spectrum to the Gaussian quadratic yields a sparse precision $Q_{\text{odd}}(a)$ where $a_0 = 2\alpha + \beta$ and $a_{j_0} = a_{N-j_0} = \alpha$. Hence the quadratic form for a $2j_0$-periodic Gaussian with only odd harmonics looks like

$$(s_0, \ldots, s_{N-1}) \longmapsto \sum_k \alpha(s_k + s_{k+j_0})^2 + \beta s_k^2, \tag{24}$$

for $\alpha, \beta > 0$. Lacking even harmonics translates into asking for *negative* correlation between $s_k$ and $s_{k+j_0}$. This can be seen by noticing that a plus-sign appears in the wave-form coupling terms $(s_k + s_{k+j_0})^2$. In a model for a sound with a full set of harmonics these terms contain a minus-sign (see Equation 5). This results in positive correlation at lag $2j_0$ because the elements in the chains $D_m = \{s_m, s_{m+j_0}, s_{m+2j_0}, s_{m+3j_0}, \ldots\}$ will not want to align with

[6]If the restricted end of a clarinet were perfectly closed then the even harmonics would be totally absent. See `http://hyperphysics.phy-astr.gsu.edu/hbase/Waves/clocol.html` for the physics.

regards to being above zero (+) or below zero (−). Hence with high probability the resulting chain will have an alternating pattern like $\ldots, +, -, +, -, +, -, \ldots$ and thus values of the wave $(s_k, s_{k+2j_0})$ separated by $2j_0$ time steps will have a tendency to display the patterns $(+,+)$ or $(-,-)$.

To use this model in applications requiring high temporal resolution, such as is offered by the Poisson process prior discussed previously, we need the determinant of $Q_{\text{odd}}$ (Equation 24). Letting $\gamma/(1-\gamma)^2 = \alpha/\beta$ for $\gamma \in [0,1)$, then when $N/j_0 = q$ is an integer, from Equations 8 and 11 we get

$$\begin{aligned}\det Q_{\text{odd}} &= \frac{\beta^N (1-(-\gamma)^q)^{2j_0}}{(1-\gamma)^{2N}} \\ &= \left( \left( \frac{2\alpha+\beta+\sqrt{\beta^2+4\alpha\beta}}{2} \right)^q + \left( \frac{2\alpha+\beta-\sqrt{\beta^2+4\alpha\beta}}{2} \right)^q - 2(-\alpha)^q \right)^{j_0}.\end{aligned}$$

2.2.3. *Resonance in speech.* Along with a note's period, smoothness and whether it has a full set of harmonics, another audible property of a musical tone is whether there are any resonance frequencies. Resonance occurs because every physical object possesses frequencies at which it most easily vibrates. If an instrument has a resonance at $k$, then there will be a peak in the spectrum around $k$ in any sound produced by the instrument.

Resonances occur in speech as well as in music. The basic model of speech production is the *source-filter* model [Fant, 1970], where the resonance frequencies are called *formants*. The vocal tract is the filter and the source is either noise or else something approximately periodic. Changing the shape of the vocal tract changes the location of the formants and hence modifies the spectrum of the output sound. Some instruments, such as the guitar, also have a separate source (the string) and filter (the wooden body) but unlike a voice, the body of the guitar maintains a constant shape and thus the resonances are fixed. Figure 3 shows

an example of resonance appearing in a speech spectrum. The first three formants have been labelled. This figure will be discussed in more detail once the basic model is introduced.

Signal processing researches have known for a long time how to synthesise vowel sounds to a reasonable degree of accuracy. Each vowel sound can be characterised (approximately) by the location of its first two formants, which we denote $k_1, k_2$. Hence all that needs to be done to synthesise a person uttering a vowel is to filter a periodic source $w_0, \ldots, w_{N-1}$ twice; the first time to place a peak at $k_1$ and the second to place a peak at $k_2$. The auto-regressive equations for placing a peak at $k_1$ are

$$x_n = -\beta_1 x_{n-1 \pmod N} - \beta_2 x_{n-2 \pmod N} + w_n, \quad (n = 0, \ldots, N-1),$$

where $\beta_1 = -2\cos(2\pi k_1/N)$ and $\beta_2 = R^2$. The bandwidth $R \in [0, 1)$ controls the strength of the peak ([Smith, 2021][7]). The spectrum of the resulting signal is easily seen to be

$$\mathbb{E}|\hat{x}_k|^2 = \frac{\mathbb{E}|\hat{w}_k|^2}{|1 + \beta_1\omega^{-k} + \beta_2\omega^{-2k}|^2} = p_a(\omega^k)^{-1} p_b(\omega^k)^{-1}$$

for $k = 0, \ldots, N-1$, where $b_0 = 1 + \beta_1^2 + \beta_2^2$, $b_1 = b_{N-1} = \beta_1(1 + \beta_2)$, $b_2 = b_{N-2} = \beta_2$ (rest 0) and $a_0 = 2\alpha + \beta$, $a_{j_0} = a_{N-j_0} = -\alpha$ (rest 0). Hence resonance filtering acts to multiply the source's precision matrix $Q(a)$ with a sparse matrix $Q(b)$, the result $Q(c) = Q(a)Q(b)$ is circulant with top row $c \in \mathbb{R}^N$. Furthermore, the determinant is given by $\det Q(c) =$

[7] http://ccrma.stanford.edu/\~jos/fp/Formant_Filtering_Example.html

$\det Q(a) \det Q(b)$ where

$$\det Q(b) = \prod_{k=0}^{N-1} |1 + \beta_1 \omega^{-k} + \beta_2 \omega^{-2k}|^2 \tag{25}$$

$$= \prod_{k=0}^{N-1} |(1 - R\omega^{k_1}\omega^{-k})(1 - R\omega^{-k_1}\omega^{-k})|^2 \tag{26}$$

$$= (1 - R^N)^4, \tag{27}$$

by similar reasoning that lead to the proof of Equation 11. Hence so long as $R$ is not too close to 1 we can use the approximation $\det Q(c) \approx \det Q(a)$.

Now, $Q(c)$ can be computed by simple matrix multiplication, but in the application we have in mind, there is the need to modify $Q(b)$ and $Q(a)$ by changing the location of the resonance in the filter and value of the period in the source. It is therefore worthwhile taking advantage of the sparse circulant structure in these matrices to make computing $Q(c)$ as fast as possible. This can be done by noting that the eigenvalues of $Q(c)$ satisfy $p_c(\omega^k) = \sum_{i,j} a_i b_j \omega^{(i+j)k}$ for all $k$, hence

$$c_k = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} a_i b_j \mathbb{I}_{i+j=k \pmod N}, \quad k = 0, \ldots, N-1, \tag{28}$$

where $\mathbb{I}_A$ is 1 if the event $A$ occurs and is otherwise 0. Because both $a$ and $b$ are sparse, $c$ is also sparse and hence $Q(c)$ can be constructed quite quickly. Notice that Equation 16 is a special case of Equation 28.

Figure 3 shows the spectrum of a recording (of the author) uttering the vowel sound *ee* as in the word *speech*. The recording was made on a simple laptop microphone. By default the sampling rate is CD quality which has 44100 samples per second, but before carrying out any analysis, I first down-sampled the recording to have a sample rate of 8192 hertz

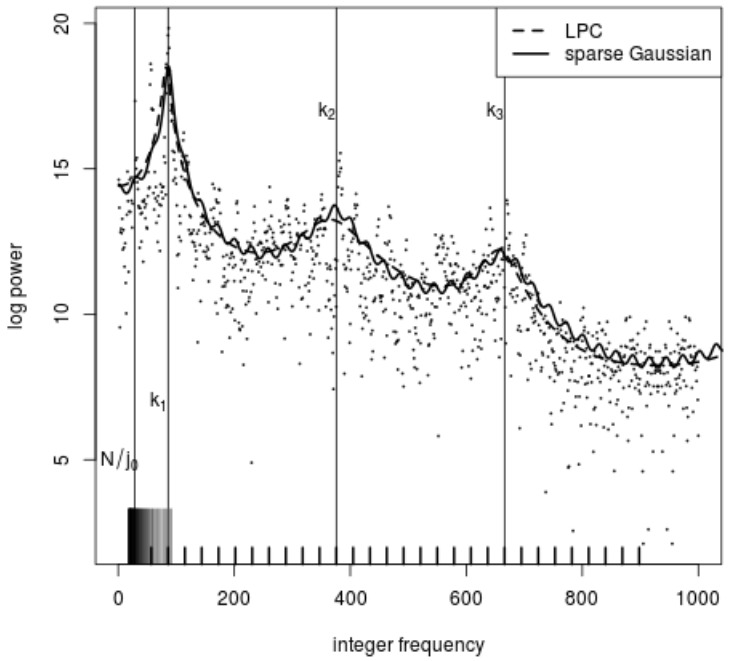



FIGURE 3. Comparison of Linear Predictive Coding with the sparse Gaussian model presented here applied to the problem of modelling the formants in speech. The figure shows the log spectrum of a recording of the vowel [i] and the three inferred formants labelled $k_1$, $k_2$, $k_3$. Also shown is the inferred fundamental frequency $N/j_0$. This shows that the sparse Gaussian model correctly locates the formants and the fundamental; and importantly also provides explicit variables for these hidden quantities (unlike LPC).

(telephone quality). The fitted spectrum is overlaid on the empirical spectrum. The period $j_0$ of the periodic source is estimated to be 65, or $8192/65 = 126$ hertz. Vowel frequencies for men typically fall in the range $[90 - 150]$, for women in the range $[150 - 250]$ and for children in the range $[250 - 350]$. The location of the first three formants are estimated to be $k_1 = 86$, $k_2 = 376$ and $k_3 = 666$. Looking at the quality of the fit in Figure 3, we see that the model is able capture the resonance peaks and simultaneously model the harmonics.

The methodology to find the hidden variables characterising a vowel $j_0$, $k_1$, $k_2$, $k_3$ and estimate the unknown acoustic parameters $(\alpha, \beta)$ and bandwidths $(R_1, R_2, R_3)$ goes as

follows. We first estimate the parameters $(\alpha, \beta)$ by marginal maximum likelihood. That is by numerically maximising the sum $\sum_{j \in J} p_j(s)$, where $p_j$ is a sparse Gaussian density $Q(a)$ with $a_0 = 2\alpha + \beta$, $a_j = a_{N-j} = -\alpha$ and the rest 0. Having obtained estimates for these parameters we then picked $j_0$ to be the $j \in J$ with the highest likelihood $p_j(s)$. For this experiment we chose $J = \{20, 21, \ldots, 102\}$, which corresponds to assuming the frequency falls in the range $[80, 400]$ hertz. This grid of possible source periods is indicated with tall tick-marks on Figure 3.

With $j_0$ so fixed, we used a second round of marginal maximum likelihood to estimate the bandwidths $R_1, \ldots, R_3$ and re-estimate the acoustic parameters. To do this requires summing over all possible formant locations $k_1 < k_2 < k_3$, where we restricted these to live at the harmonics $N/j_0, 2N/j_0, \ldots, 31N/j_0$. An upper limit of $31N/j_0$ restricts the frequency of all the formants to be located below 4000 hertz. This grid of possible values is labelled on the figure with short tick-lines. (It is actually a small mistake to force the first formant to be at least as high as the fundamental frequency like we have done here. Opera singers (apparently) can produce such a high pitch fundamental that it exceeds the location of the first formant. Needless to say, this wasn't an issue here.) The density required to do the second round of marginal maximum likelihood had a precision of the form $Q(a) \prod_{i=1}^{3} Q(b^{(i)})$, where $Q(a)$ has $a_0 = 2\alpha + \beta$ and $a_{j_0} = -\alpha$ and $b_0^{(i)} = 1 + \beta_{i,1}^2 + \beta_{i,2}^2$, $b_1^{(i)} = \beta_{i,1}(1 + \beta_{i,2})$, $b_2^{(i)} = \beta_{i,2}$, $\beta_{i,1} = -2\cos(2\pi k_i/N)$, $\beta_{i,2} = R_i^2$, for $i = 1, \ldots, 3$, all given up to sparsity and reflection.

The methodology just described is a new take on an old problem. The standard algorithm (called LPC [O'Shaughnessy, 1988]), works using an auto-regressive model:

$$s_k = \alpha_1 s_{k-1} + \alpha_2 s_{k-2} + \cdots + \alpha_p s_{k-p} + w_k, \quad k = p, \ldots, N-1,$$

where $\{w_k\}$ is white noise (sd. $\sigma$). LPC places no restriction on the the parameters $\alpha_i$, hence the model can be fit by a linear regression. The spectrum of $s_k$ is $\sigma^2|1-\sum_{j=1}^{p}\alpha_j\omega^{-jk}|^{-2}$. We set $p=10$ and ran LPC on the vowel recording. The resulting estimated spectrum is plotted on Figure 3. It can be quite hard to see because it is essentially identical to the sparse Gaussian solution. Although note LPC does not estimate a fundamental, hence the fitted spectrum lacks harmonics. By searching over the LPC fitted spectrum, it is possible to find the local maxima and these are taken to be the location of the formants. This methodology is used in the speech analysis program Praat ([Boersma and Weenink, 2009]).

2.3. **Segmenting monophonic audio.** In this section we are going to use these sparse Gaussian processes to transcribe monophonic music. The sample $s$ we will use is a piece of solo flute music taken from the Raga guide ([Bor, 1992], CD 3, track 11) with sampling rate 8000 hertz. The flute note discussed previously was extracted from this piece of music.

We will take as the sparse acoustic model periodic Brownian motion because as previously shown, this model fits the flute note much better than periodic white noise. The model has three parameters: $\alpha$ and $\beta$ which jointly control the amplitude and periodicity, and $\gamma$ which controls the degree of smoothness. In our experience, what varies most from note to note, both in the same piece of music and across pieces, is amplitude rather than periodicity or smoothness. Luckily, it is simple enough to estimate amplitude directly from the data[8]. This then allows each note in a recording to possess it's own amplitude when transcribing. The way this works is to multiply a sample from periodic Brownian motion $s' \in \mathbb{R}^n$ with a constant $\sigma \in \mathbb{R}$, so that $s = \sigma s'$ has density $b_j(s/\sigma)/|\sigma|^n$. With a uniform prior on $\log\sigma$, the joint distribution over $(s,\sigma)$ (for fixed $s$) has a mode at $\hat{\sigma}^2 = s^T Q_j s/(n+1)$ where $Q_j$ is given in Equation 19. Plugging this estimate for the amplitude back into the joint density gives

[8]This idea is demonstrated in [Mumford and Desolneux, 2010] in the case where the Gaussian process is white noise and $\sigma^2$ is then the variance. We apply this same idea here but now $\sigma^2$ is amplitude.

the following *energy* (negative log probability (ignoring constants)) for a variable amplitude musical note with period $j$

$$E(s, j) = (n + 1) \log(\hat{\sigma}^2) - \log(\det Q_j), \tag{29}$$

where the determinant is given in Equation 20. With amplitude inferred from the data, the other the parameters can be set to default values which work well in a wide range of situations. We have found that $\alpha = e^7$ and $\gamma = .5$ fit this description [9].

The prior we'll use to transcribe the Indian music is the ordinary Poisson process over the number of notes $m$, start times $\{t_i\}$ periods $\{j_i\}$. This was first proposed in [Mumford and Desolneux, 2010] and is written

$$j_i \sim \text{uniform on } J$$

$$n_i \sim \text{exponentially distributed}$$

$$t_{i+1} = t_i + n_i$$

for $i = 1, \ldots, m$ where $t_1 = 1$ and $t_{m+1} = N + 1$ and $J = \{5, \ldots, 40\}$. The joint distribution takes the form

$$p(s, \{\sigma_i\}, \{t_i\}, \{j_i\}, m) = \prod_{i=1}^{m} \frac{\theta}{|J|} e^{-\theta n_i} b_{j_i} \left( s_{[t_i, t_{i+1}]} / \sigma_i \right) / |\sigma_i|^{n_i + 1} \tag{30}$$

where $\theta > 0$ gives the tempo of the music. The combined energy of the sound-wave $s \in \mathbb{R}^N$ and hidden variables is written

$$E(s, \{t_i\}, \{j_i\}, m) = c_0 m + \sum_{i=1}^{m} E(s_{[t_i, t_{i+1})}, j_i)$$

[9] In the presence of $\sigma^2$ the parameter $\beta$ is redundant and is henceforth fixed at 1.

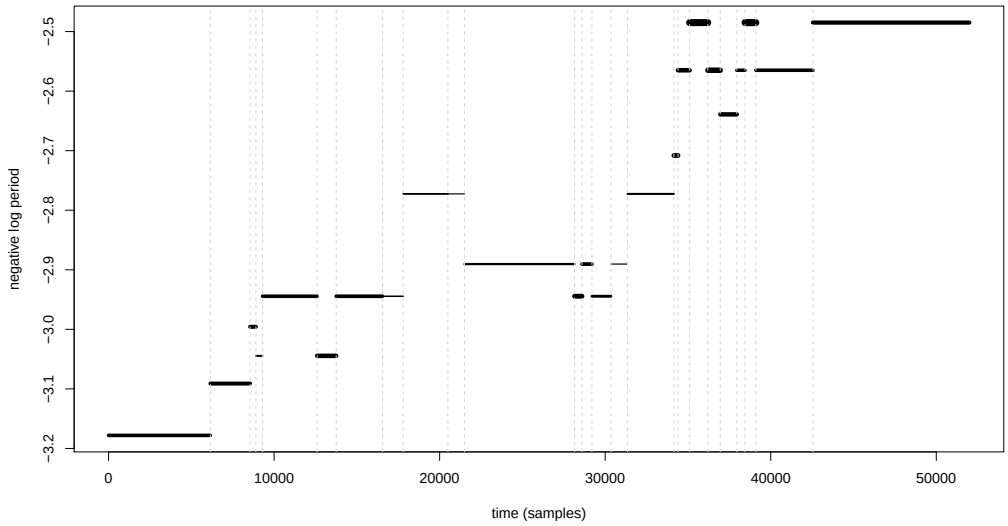



FIGURE 4. Transcription of the first 6.5 seconds of the Indian music ([Bor, 1992], CD 3, track 11) under the ordinary Poisson prior with periodic Brownian motion. The width of the bars indicate the inferred amplitudes and the horizontal lines mark the boundaries between notes. Note that some events have been split into more than one part to allow within-event amplitude variation. See `indian.wav`.

for constant $c_0 \in \mathbb{R}$. This can be minimised (for fixed $s$) with the following dynamic program. Let $M(t)$ be the minimum energy over $s_{(0,t]}$, then it follows that

$$M(t) = \min_{u,j} c_0 + E(s_{(t-u,t]}, j) + M(t-u), \tag{31}$$

where $M(0) = 0$ and the min is over $j \in J$ and $u = 1, \ldots, t$. This dynamic program has time complexity $O(N^2|J|)$.

We set $c_0 = 100$ and ran this dynamic program on the Indian music[10]. The results over the first 6.5 seconds are shown in Figure 4. In this plot the width of the bars are proportional to the square-root amplitudes $\{\sigma_i\}$ and the horizontal lines mark the location of the starts of new notes $\{t_i\}$. The transcription is accurate, except for two notes that have been incorrectly

[10]The hyper-parameter (like all future hyperparameters) was chosen quite casually by running the algorithm two or three times trying to arrive at decent looking results.

broken into multiple parts (both near $\sim 20000$ samples). This happens because sometimes the amplitude significantly changes over the duration of a note, starting off loud and then gradually getting quieter, following an *attack-decay* pattern. An audio comparison between the transcription and the original music is in the file `indian.wav`[11]. This has been prepared by sampling periodic Brownian motion conditioned on the inferred notes. It is surprising how life like this sample sounds, this is in part due to the modelling of amplitude variation, but also in part due to the simplistic sound of the flute which is well modelled by this sparse Gaussian.

On this music the results are quite good, however it is not hard to find music where this simple model fails. In particular, because notes are being split to model attack and decay, it very often happens that the two (or more) pieces of a single note are inferred as having different periods. This is because the model treats the periods as independent and so doesn't search for the best *single* period to describe all sections of a note. Hence to go further we need a new prior, one which allows the amplitude to change over the duration of a note. Furthermore, most music consists of multiple simultaneous sounds with more complex acoustic properties than the flute (e.g. synthesizers, voice etc..) and therefore the Gaussian processes need enhancing to deal with *superpositions* of signals and also require further hidden variables to enable them to produce a richer repertoire of sounds. It is to these enhancements we turn in the next chapter.

The authors of [Mumford and Desolneux, 2010] also describe in some detail how speech signals can also be segmented with the above dynamic program. Their basic example is the word 'sheep', which splits into 4 parts: (1) the coloured noise during *sh*, (2) the periodic vowel *ee*, (3) a short segment of near silence and finally (4) the plosive *p*. In this chapter

[11] When played on headphones, the right ear is playing the original and the left ear the reconstruction.

we have shown how geometric filtering white noise results in a sparse Gaussian process with known determinant, giving shades of coloured noise depending on the value of the smoothing parameter $\gamma$. This model would apply to noise parts of speech - (1),(3) and (4) - for various values of $\gamma$ that could be estimated directly from the data like we have done here for amplitude. For the vowel speech sounds - (2) - one can use the resonance filter (with periodicity) which we developed in section 2.2.3. In this case, a family of vowel models $\{p_{l,j}(s)\}$ is needed, where $j$ gives the periodicity of the voice (long for men, shorter for women) and $l$ is a label signifying the location of the resonance peaks, or equivalently, the type of vowel. These models can now be obtained by applying the estimation algorithm also described in that section.

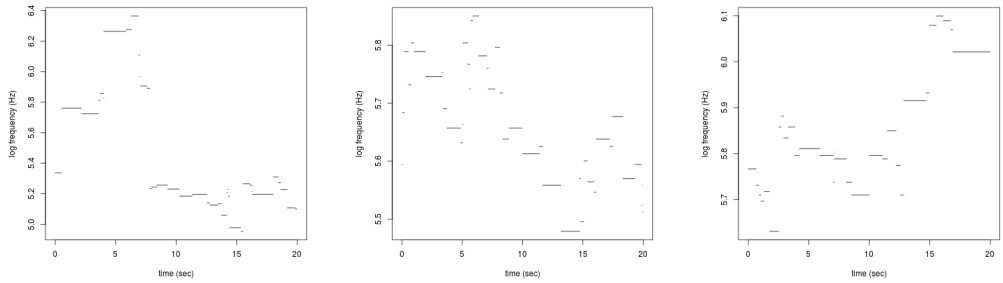
2.3.1. *The compound Poisson prior.* It is interesting to note that the transcription of the Indian music doesn't look like the periods $\{j_i\}$ have been independently sampled from $J$. The path traced by the melody has an upward trend and each note is near its two neighbours. This suggests a better prior for music would be a *compound Poisson process* (see e.g. [Çınlar, 2011]), which can be written

$$t_{i+1} = t_i + \text{exponentially distributed duration}$$

$$\log(j_{i+1}) = \log(j_i) + x_i$$

$$x_i \sim d\nu(x)/\nu(\mathbb{R})$$

where $\nu$ is any positive measure on $\mathbb{R}$, known as the *Levy measure*. Thus the log periods/frequencies follow a random walk where the jumps are independently sampled from the Levy measure. Three examples of Levy measures are the Cauchy, Gaussian and Laplace. Simulations from these three models are shown in Figure 5.




(a) Cauchy (b) Gaussian (c) Laplace

FIGURE 5. Simulated music from three compound Poisson processes. These simulations look far more like the Indian music transcription than independent sampling under the ordinary Poisson process.

The Gaussian density looks like

$$d\nu(x) = \exp(-x^2/\kappa), \quad (x \in \mathbb{R}).$$

Using this Levy measure, the total energy takes the form

$$E(s, \{t_i\}, \{j_i\}, m) = c_0 m + c_1 \sum_{i=1}^{m} \left(\log(j_i) - \log(j_{i-1})\right)^2 + \sum_{i=1}^{m} E(s_{[t_i, t_{i+1})}, j_i), \tag{32}$$

where $\log(j_0) = 0$. This can also be minimised by dynamic programming. Let $M(t, j)$ be the minimum energy over $s_{(0,t]}$ assuming the right boundary note has period $j$, then it is easy to see from Equation 32 that

$$M(t, j) = \min_{u,k} c_0 + c_1 \left(\log(j) - \log(k)\right)^2 + E(s_{(t-u,t]}, j) + M(t - u, k), \tag{33}$$

where $M(0, j) = 0$ for all $j$. This dynamic program has complexity $O(N^2|J|^2)$, which is an increase by a factor of $|J|$ over the ordinary Poisson process due to the need to track the period of the last note.

It will be seen therefore that the Gaussian Levy measure acts to penalise the sum of the *square* of the jumps in log frequency space. Thus the Gaussian makes use of of the $L_2$ penalty to regularise the transcription. An alternative to this is to penalise on the sum of the *absolute values* of the jumps and thus make use of an $L_1$ norm. The distribution associated with this norm is the Laplace distribution, which has functional form

$$d\nu(x) = \exp(-|x|/\kappa), \quad (x \in \mathbb{R}).$$

It will be seen in Figure 5 that the simulations for the Laplace and Gaussian are quite similar. This is to be expected because the way these simulations are done is to first sample the location of the boundaries from an ordinary Poisson prior and then sample the random walk melody. Thus under sampling, the Gaussian, Laplace and Cauchy are only going to differ in the distribution over jumps not boundaries. The Laplace distribution is not so dissimilar to the Gaussian for this to show up in the plots.

The difference between the Laplace and the Gaussian measure is more clear when one considers their behaviours at the mode rather than under (prior) sampling. This can be illustrated with a simple problem. Suppose you need to travel in a straight line from $a$ to $b$ ($b > a$). This journey can be taken in one step of length $b - a$, or else two steps of length $(b - a)/2$, or in general $n$ steps of length $(b - a)/n$. Using the Laplace measure, each step, a price proportional to the distance travelled is paid, plus an additional penalty for stopping $c > 0$. Thus for any $n$, the total price paid is $n|(b - a)/n| + nc = b - a + nc$. This is minimised for $n = 1$. On the other hand, paying a price proportional to the square of the distance travelled,i.e. using the Gaussian measure, then using $n$ steps, the total price paid is $n((b - a)/n)^2 + nc = (b - a)^2/n + nc$. This is minimised by taking $n = (b - a)/\sqrt{c}$ steps.

In the context of transcribing music there will be strong evidence that the frequency is say $a$ at some time $t_0$ and $b$ at some other time $t_1$ but it may be less obvious what's happening in between these times. The Laplace prior will prefer to take the jump from $a$ to $b$ in as few steps possible and thus will want to continue $a$ until sometime $t_0 < t < t_1$ and make a single jump to $b$; whereas the Gaussian measure will prefer to interpolate between $a$ and $b$ by inserting a number of grace notes $a < a_1 < a_2 < \cdots < a_k < b$, where the analysis above suggests $k \propto b - a$.[12]

An alternative to both these Levy measures is the Cauchy distribution

$$\nu(dx) = \frac{1}{\kappa + x^2} dx, \quad (x \in \mathbb{R}).$$

It is well known that this distribution permits orders of magnitude differences in the size of the jumps. This can be seen in Figure 5 (plot (a)) where a big drop in frequency is followed by a sequence of smaller changes.

2.4. **Piecewise linear Brownian motion with restoring force.** Up to now we have focused exclusively on mean 0 processes, this being an appropriate assumption when analysing audio. However, often the interest is focused on the mean, and the methodology and models discussed in the first part of this chapter can be used in these cases as well.

In this section we allow the mean to be given by an unknown piecewise linear function of time. This model is then suitable for time-series where the mean changes at unknown times but is linear over each segment. The most vexing question when analysing data using these sorts of change-point models is whether some sequence of values which appears to suggest a change in the mean is really just a consequence of short term correlation. The only way to

[12]This is the music version of the well-known difference between the $L_1$ and $L_2$ norms when used to regularise statistical regression parameters. In this case, the use of the $L_2$ norm is known as *ridge regression* and the use of the $L_1$ norm goes by the name of *The Lasso* (see e.g. [Efron and Hastie, 2016]).

deal with this is to fit both the mean and the correlation structure and let the model decide which is the better explanation.

2.4.1. *Is the climate warming?* As an example, consider Figure 6. This plot shows the annual average global temperature from $1850-2021$[13]. The overall time average has been subtracted from each data-point and this is why the series is centred on 0. A question of some interest is whether there is any change in the average temperature, or rather, whether the apparent rise in average temperature over the last $\sim 100$ years is merely a consequence of year on year correlation.

Overlaid on the raw data is the best piecewise linear fit under the model to be introduced in this section. The model and segmentation algorithm have decided these data are best described by three sections spanning $1850-1901$, $1902-1963$ and $1964-2021$. During the first section the linear trend is essentially flat, suggesting no rise or fall in average temperature, however the next two sections show periods where the average temperature is increasing, with the second section increasing at a faster pace.

The correlation structure assumes that the errors about these linear trends follow a mean restored Brownian motion. This correlation structure has two parameters, the first $\sigma^2$ controls the variance and the second $\gamma$ controls the strength of the correlation. On these data the estimates of these parameters are $\sigma^2 \approx 0.01$ and $\gamma \approx 0$, suggesting that there is very little correlation and thus the pattern seen in the time series is better explained almost entirely with changes in the mean. A further parameter included in this model is the rate at which the climate is changing $\lambda$. Here this was estimated to be $1/\lambda \approx 57$, suggesting that the climate shifts into a different regime every 57 or so years.

[13] `www.metoffice.gov.uk/hadobs/hadcrut4/data/current/time_series/HadCRUT.4.6.0.0.annual_ns_avg.txt`

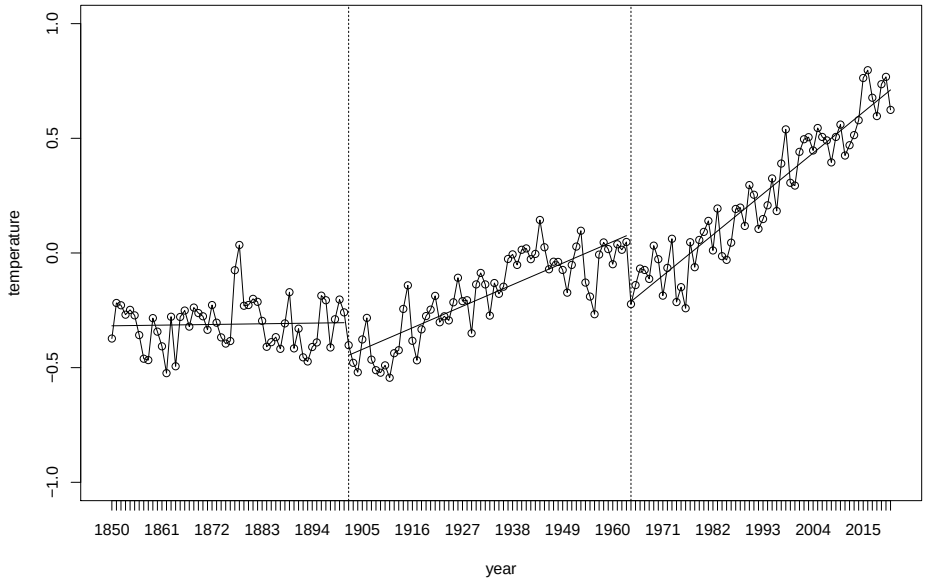



FIGURE 6. Global average yearly temperature raw data series along with the best piecewise linear fit under the model described in this section. According to the model, the climate dramatically shifted in years 1902 and 1964.

2.4.2. *The piecewise linear model.* Consider the following density for a time series $\{x_i\} = \{x_0, \ldots, x_{n-1}\}$ with linear trend $\alpha + \beta i$:

$$p_{\alpha,\beta}(x) = \left(\frac{\det Q_n}{(2\pi)^n}\right)^{\frac{1}{2}} e^{-\frac{1}{2}\sum_{i=0}^{n-1}(x_i-\alpha-i\beta)^2/\sigma^2 - \frac{\epsilon}{2}\sum_{i=1}^{n-1}(x_i-x_{i-1}-\beta)^2/\sigma^2} \tag{34}$$

where $\sigma^2 > 0$ and $\epsilon = \gamma/(1-\gamma)^2 \geq 0$ for $\gamma \in [0, 1)$. The parameter $\sigma^2$ controls how close to the linear trend $\alpha + \beta i$ the Brownian motion path $x_0, \ldots, x_{n-1}$ is likely to be, and then $\epsilon/\sigma^2$ controls the strength of the serial correlation between adjacent time-points after correcting for the deterministic change of $\beta$ per time-step. This density, or rather the version with $\alpha = \beta = 0$, i.e. $p_{0,0}(x)$, under a different parameterisation, is called the *weak string* model in [Mumford and Desolneux, 2010]. The precision matrix $Q_n$ under the $(\sigma^2, \epsilon)$ parameterisation

used here, is given in matrix form as

$$Q_n = \frac{1}{(1-\gamma)^2\sigma^2}\begin{pmatrix} 1+\gamma^2-\gamma & -\gamma & 0 & \dots & 0 & 0 \\ -\gamma & 1+\gamma^2 & -\gamma & 0 & \dots & 0 \\ 0 & -\gamma & 1+\gamma^2 & -\gamma & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \dots & 0 & -\gamma & 1+\gamma^2 & -\gamma & 0 \\ 0 & \dots & 0 & -\gamma & 1+\gamma^2 & -\gamma \\ 0 & 0 & \dots & 0 & -\gamma & 1+\gamma^2-\gamma \end{pmatrix} \in \mathbb{R}^{n\times n}.$$

We will shortly derive the following exact expression for the determinant of this matrix:

$$\det Q_n = \frac{1-\gamma^{2n}}{(1-\gamma^2)\sigma^{2n}(1-\gamma)^{(n-1)2}}. \tag{35}$$

Real world time series are often better described not as having a single trend, but as being the concatenation of multiple time series each with a differing trend. This can be modelled with a Poisson process over the location of these discontinuities coupled with a uniform distribution over the trend parameters $\{\alpha_i\}$ and $\{\beta_i\}$. Let $x = x_1, \dots, x_N$ is be one of these time series and let $\{t_i\}$ be the jump times, then the joint density of this model is written

$$p(x, m, \{t_i\}, \{\alpha_i\}, \{\beta_i\}) = \prod_{i=1}^{m} \frac{\theta e^{-\theta(t_{i+1}-t_i)}}{|D|^2} p_{\alpha_i,\beta_i}(x_{[t_i,t_{i+1})]}), \tag{36}$$

where $t_{m+1} = N+1$, $D \subset \mathbb{R}$ is a finite domain on which both $\alpha$ and $\beta$ are uniformly distributed and $m$ gives the number of segments.

2.4.3. *A parsing algorithm.* One of the most vexing questions in any analysis of a time series is whether some sequence of values is due to trend or rather correlation. The above model can be used to answer this question by finding the optimal segmentation of the data into

piecewise linear regions, with the following dynamic program.

$$M(t) = \min_u \theta u + \log(|D|^2/\theta) + M(t-u) - \max_{\alpha,\beta} \log p_{\alpha,\beta}(x_{(t-u,t]}), \tag{37}$$

for $t = 1, \ldots, N$, $M(0) = 0$ and then $M(N)$ is the negative log probability of the series under the optimal segmentation.

The maximisation of $\log p_{\alpha,\beta}$ in this dynamic program is quite straightforward as it is a simple instance of regression with correlated errors, see [Diggle et al., 2013]. Writing the quadratic in $p_{\alpha,\beta}(x)$ in the more general form $(x - zb)^T A(x - zb)$, where $z$ is an $n \times p$ matrix of known constants and $b$ is a length $p$ vector of parameters, differentiation of the quadratic gives $\hat{b} = (z^T A z)^{-1} z^T A x$ as the maximising parameter values. When $A = Q_n$ and $z = z_1 : z_2$ where $z_1 = (1, \ldots, 1)$ and $z_2 = (0, \ldots, n-1)$, this formula gives the following estimates of $\alpha$ and $\beta$

$$\hat{\alpha} = \bar{x} - \hat{\beta}(n-1)/2, \qquad \hat{\beta} = \frac{\sum_{i=0}^{n-1}(x_i - \bar{x})i + \epsilon \sum_{i=1}^{n-1}(x_i - x_{i-1})}{\sum_{i=0}^{n-1} i^2 - \frac{n-1}{2}\sum_{i=0}^{n-1} i + \epsilon(n-1)}. \tag{38}$$

Although note that it is far simpler to derive this by directly differentiating the density. Furthermore the expression for $\hat{\beta}$ can be simplified using $\sum_{i=0}^{n-1} i = (n-1)n/2$ and $\sum_{i=0}^{n-1} i^2 = (n-1)n(2n-1)/6$. Hence the terms appearing in the expression for $M(t)$ are differences of cumulative sums over $x_1, \ldots, x_N$, or else known quantities (assuming $\sigma^2$ and $\epsilon$ are known). For example, to compute $\hat{\beta}$ relevant for $x_{[t_0,t_1]}$, then we need the sum

$$\sum_{i=0}^{t_1-t_0} i x_{t_0+i} = \sum_{i=t_0}^{t_1} (i - t_0) x_i = \sum_{i=t_0}^{t_1} i x_i - t_0 \sum_{i=t_0}^{t_1} x_i.$$

These cumulative sums should be pre-computed at a one-off cost of $O(N)$ and therefore the calculation of $M(N)$ has time complexity $O(N^2)$.

2.4.4. *Derivation of* $\det Q_n$. The derivation given here builds on the approximation to $\det Q_n$ given in [Mumford and Desolneux, 2010]. The parameterisation of the weak string model in terms of $(\sigma^2, \epsilon)$ allows their approximation to be made exact. Unlike in previous chapters, the matrix $Q_n$ is not circulant, but it is very closely related to the precision matrix of circulant Brownian motion hence by the matrix determinant lemma, it follows that

$$\det Q_n = \frac{\det Q_n(\gamma) \cdot \left(1 + e_1^T Q_n(\gamma)^{-1} e_2\right)}{(1-\gamma)^{2n}\sigma^{2n}},$$

where $Q_n(\gamma)$ is the precision matrix of circulant Brownian motion[14] and $e_1 = (-\sqrt{\gamma}, 0, \ldots, 0, \sqrt{\gamma})^T$ and $e_2 = -e_1$. This is the key insight from [Mumford and Desolneux, 2010] which we apply here largely unchanged.

The next step is to spot that $Q_n(\gamma)^{-1}$ is also circulant $Q(c)$ for some top row $c \in \mathbb{R}^n$, with $c_{n-j} = c_j$ and thus the quadratic in the above equation can be written

$$e_1^T Q_n(\gamma)^{-1} e_2 = \sum_{i=0}^{n-1} c_i e_1^T \pi^i e_2,$$

where $\pi$ is the cyclic permutation matrix of size $n \times n$. The resulting permuted inner-products between $e_1$ and $e_2$ makes this come out quite simply as $2\gamma(c_1 - c_0)$. Furthermore, $Q_n(\gamma)^{-1} Q_n(\gamma) = I$, hence it must also be that $(1+\gamma^2)c_0 - 2\gamma c_1 = 1$ and thus $c_1 = ((1+\gamma^2)c_0 - 1)/(2\gamma)$ and so the determinant can be written

$$\det Q_n = \frac{\det Q_n(\gamma) c_0 (1-\gamma)^2}{(1-\gamma)^{2n}\sigma^{2n}}, \tag{39}$$

[14] That is, the top row of $Q_n(\gamma)$ is sparse with non zero first element $1+\gamma^2$ and where the second and last elements of the top row are identical and sum to $-2\gamma$.

where $c_0$ is the element of the diagonal in the covariance matrix of circulant Brownian motion, that is to say

$$c_0 = \text{variance}\left(\sum_{u=0}^{\infty} \gamma^u w_{-u \pmod n}\right) \tag{40}$$

for $\{w_0, \ldots, w_{n-1}\}$ white noise. Rearranging this sum into independent parts gives this variance as the product of two geometric series

$$\begin{aligned} c_0 &= \left(\sum_{k=0}^{\infty} (\gamma^n)^k\right)^2 \times \sum_{k=0}^{n-1} (\gamma^2)^k \\ &= \frac{1}{(1-\gamma^n)^2} \times \frac{1-\gamma^{2n}}{1-\gamma^2}. \end{aligned}$$

Now, the determinant of circulant Brownian motion is $(1-\gamma^n)^2$, hence plugging this and $c_0$ into Equation 39 results in Equation 35, completing the proof.

2.4.5. *Statistical Inference via marginal maximum likelihood.* The *marginal likelihood* is obtained by summing the joint density in Equation 36 over all possible realisations of the Poisson process while integrating out the unknown regression coefficients $\{\alpha_i, \beta_i\}$. Let $L(t)$ be this quantity over $x_1, \ldots, x_t$ for $t = 1, \ldots, N$, where $L(0) = 1$, then the following recursion follows straightforwardly from the expression for the joint density in Equation 36:

$$L(t) = \sum_{0<u\leq t} L(t-u) \frac{e^{-\theta u}\theta}{|D|^2} \int_{D^2} p_{\alpha\,\beta}(x_{(t-u,t]}) d(\alpha, \beta). \tag{41}$$

In applications, $D$ should be quite large and centred on 0, so that the inner integrals can be approximated by assuming $D = \mathbb{R}$ so that

$$\begin{aligned}\int_{D^2} p_{\alpha,\beta}(x)d(\alpha,\beta) &\approx \int_{\mathbb{R}^2} p_{\alpha,\beta}(x)d(\alpha,\beta)\\ &= \int_{\mathbb{R}^p} \frac{1}{Z} e^{-\frac{1}{2}(x-zb)^T Q_n (x-zb)} db\\ &= \frac{1}{Z} e^{-x^T Q_n x/2} e^{\hat{b}^T z^T Q z \hat{b}/2} \int_{\mathbb{R}^p} e^{-\frac{1}{2}(b-\hat{b})^T z^T Q_n z (b-\hat{b})} db\\ &= p_{0,0}(x) e^{\frac{1}{2}\sum_{i=0}^{n-1}(\hat{\alpha}+i\hat{\beta})^2/\sigma^2 + \frac{\epsilon}{2}\sum_{i=1}^{n-1}\hat{\beta}^2/\sigma^2} \left(\frac{\det z^T Q_n z}{(2\pi)^2}\right)^{-\frac{1}{2}},\end{aligned}$$

where the third line follows from the second by the standard method of completing the square. Furthermore, the simplified form of $z = z_1 : z_2$ means that the $2 \times 2$ matrix $z^T Q_n z$ has a nice equation

$$z^T Q_n z = \begin{pmatrix} z_1^T Q_n z_1 & z_1^T Q_n z_2 \\ z_2^T Q_n z_1 & z_2^T Q_n z_2 \end{pmatrix} = \frac{1}{\sigma^2} \begin{pmatrix} n & \sum_{i=0}^{n-1} i \\ \sum_{i=0}^{n-1} i & \sum_{i=0}^{n-1} i^2 + \epsilon(n-1) \end{pmatrix}.$$

Therefore the determinant of this matrix is

$$\det z^T Q_n z = \frac{1}{\sigma^4}\left(n\left(\sum_{i=0}^{n-1} i^2 + \epsilon(n-1)\right) - \left(\sum_{i=0}^{n-1} i\right)^2\right). \tag{42}$$

The above inference scheme was used to obtain Figure 6. A weakness in the model is that the end points of the piecewise linear trend do not have to meet and at present the models are limited to linear means and restored Brownian motion error structure. Future work would be to extend the methods to polynomials (that meet at the end points) plus second order restored Brownian motion (based on a non-circulant version of Equation 21).

## 3. Polyphonic probability models

As discussed in the previous chapter, audio consists of (1) multiple simultaneous acoustic events, where (2) the amplitude changes over the course of the event and (3) these events are typically complex and so not well modelled by a simple Gaussian. In this chapter we define what appears to us to be the most basic stochastic model for polyphonic audio which incorporates these three properties.

This statistical model contains a prior over the location, duration and class label of all the sounds in the audio. Then, for each region of time where the labels are constant, a Poisson process provides a piecewise constant model for the amplitudes. Lastly, a random Fourier expansion equipped with hyper-priors describes the statistics of the sound wave for fixed labels and amplitude. The description of this model appears in stages throughout sections $1-3$.

Naturally then, inference with this model proceeds in reverse: first, short segments of the sound-wave where the amplitudes and labels are constant are found. Then these small sections are grouped into larger sections where the amplitude varies but the labels are still fixed. Finally, the start and end times of each of the events is inferred by incorporating Markov adjacency relations induced by the fact that events typically only partially overlap in time. This algorithm, which we call *Inference via the Binding Energy*, is described in section 2. What is most unique about this algorithm is that it exists entirely in the time domain and so doesn't rely on a pre-segmentation of the soundwave such as is offered by a windowed Fourier transform.

The quality of this analysis can be readily assessed by sampling a version of the audio from the fitted model and listening to see what the model has recovered and what has been missed out. Section 3 describes an application of the model to Nancy Sinatra's Bang-Bang; the model

accurately transcribes both the guitar and vocals and remarkably has even reconstructed much of Nancy Sinatra's voice (see `bangbang.wav`[15]).

3.1. **Superpositions of Gaussian processes.** Figure 7 shows the waveform of about six seconds of Bach's Chaconne (on classical guitar, sampled at 8000 hertz), broken into segments according to the groups of periods that are being played and the time domain is labelled with these periods. This segmentation and labelling has been prepared by listening to the original recording which comes from the cd *Segovia – Complete Bach Recordings 1927-47*. Looking at this segmentation, to a first approximation, polyphonic music consists in a sequence of groups of periods $(j_1, \ldots, j_k)$ (i.e. *chords*), where the number of periods $k$ changes from group to group. This suggests the *random chord* prior model for polyphonic music which is described next.

3.1.1. *The random chord model for polyphonic music.* Let $\{t_n\}$ be the start times and $\{j_{n,i}\}$ the periods of the chords, then the random chord model says

$$t_{n+1} = t_n + \text{exponential chord duration}$$

$$(j_{n,1}, \ldots, j_{n,k_n}) \sim \text{sampled without replacement uniformly from } J,$$

which is the natural generalisation to polyphonic music of the ordinary Poisson process model for monophonic music discussed in the previous chapter (Equation 30). The joint prior density takes the form

$$p\left(\{t_n\}, \{j_{n,i}\}, m\right) = \prod_{n=1}^{m} \theta e^{-\theta(t_{n+1}-t_n)} \pi^{k_n} (1-\pi)^{|J|-k_n},$$

[15]As in the previous experiment, the left ear is playing the reconstruction, the right ear the original.

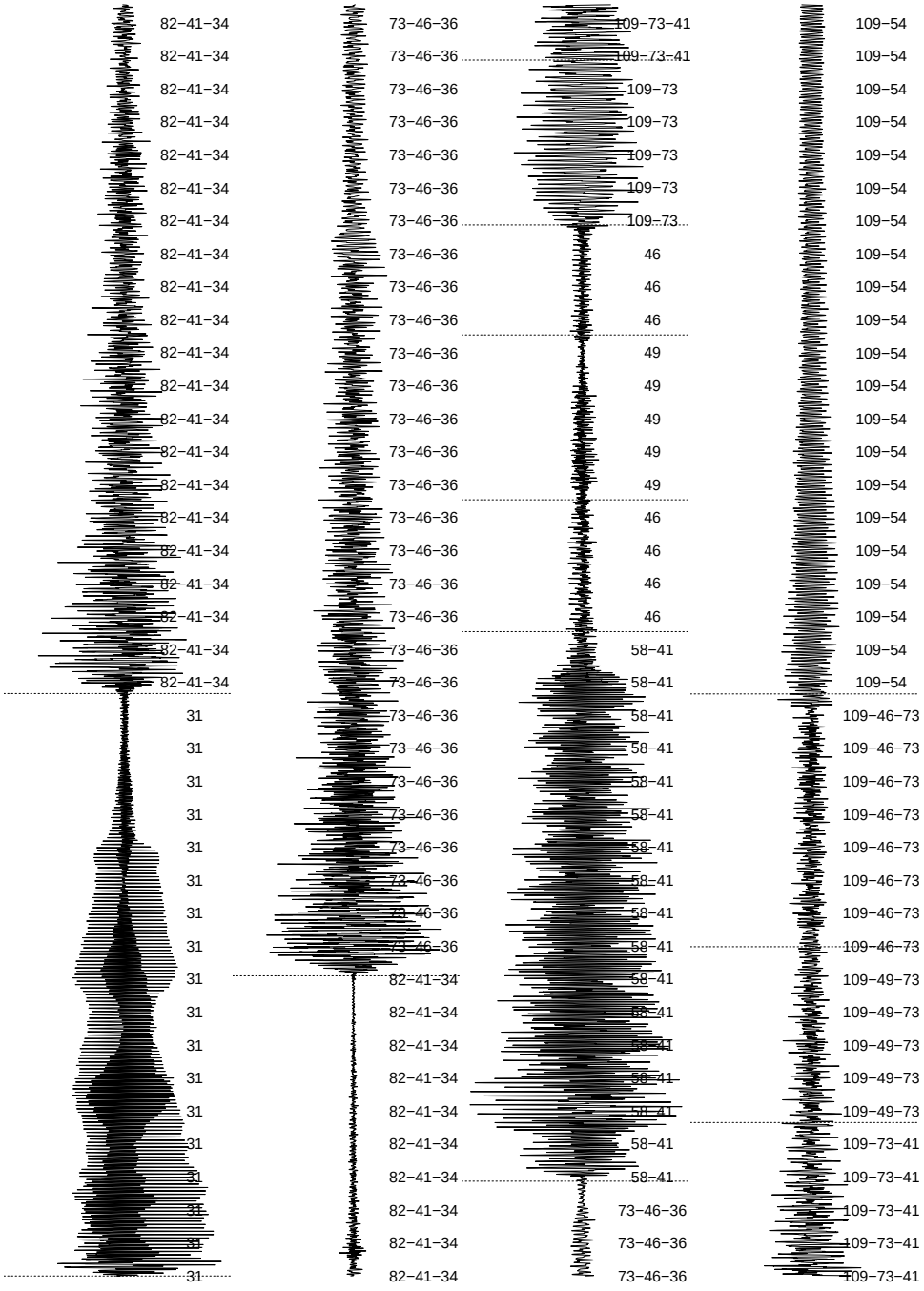



FIGURE 7. Six seconds of Bach's Chaconne played on classical guitar, broken (by hand) into segments where the chords are constant. The time domain has been labelled with the periods making up the chord at that moment in time. Notice how the amplitude varies over the duration of each segment, following an *attack-decay-sustain-release* pattern - a programmable feature on many synthesizers.

where $m$ is the number of chords, $\theta$ gives the tempo, and $\pi$ is the probability that each $j \in J$ appears in a chord. The prior energy of a transcription therefore looks like

$$E\left(\{t_n\}, \{j_{n,i}\}, m\right) = c_1 m + c_2 \sum_{n=1}^{m} k_n \tag{43}$$

for $c_1$ and $c_2$ parameters in $\mathbb{R}$. Hence this prior penalises both the number of chords $m$ and the total number of periods appearing in all chords $\sum_{n=1}^{m} k_n$.

3.1.2. *The Poisson process acoustic event model.* Conditional on these prior variables, in this chapter we take the actual sound of each chord to be characterised by a Poisson process over the times $x_i$ within the chord at which the amplitude $\sigma_i \in \mathbb{R}$ changes. This then allows each acoustic event to have it's own particular attack and decay. No prior instrument specific knowledge of the characteristics of this attack-decay are built into the model, so this is quite generic and will work for all instruments and also other acoustic events, such as speech acts.

Letting $\beta(j) \in \mathbb{R}^{u_i}$ be a periodic Brownian motion spectrum with period $j$ (see section 2.2.1), and $s_i$ the $i$th chord fragment, the generic model for a chord $s = s_1 \dots s_d$ is written

$$\begin{aligned}
&x_{i+1} = x_i + u_i \\
&u_i \sim \text{exponential event } \textit{fragment} \text{ duration} \\
&\log \sigma_i \sim \text{uniform on some finite subset of } \mathbb{R} \\
&s_i \sim \text{circulant Gaussian with spectrum } \sum_{j \in (j_1, \dots, j_k)} \sigma_i^2 \cdot \beta(j),
\end{aligned}$$

for $i = 1, \dots, d$. If the periods $j_1, \dots, j_k$ are known the spectrum $\mu \propto \sum_{j \in (j_1, \dots, j_k)} \beta(j)$ is known at all times. Hence it is possible to compute the minimum energy, piecewise constant amplitude fit to a chord $s$ under this model using the dynamic program associated with the

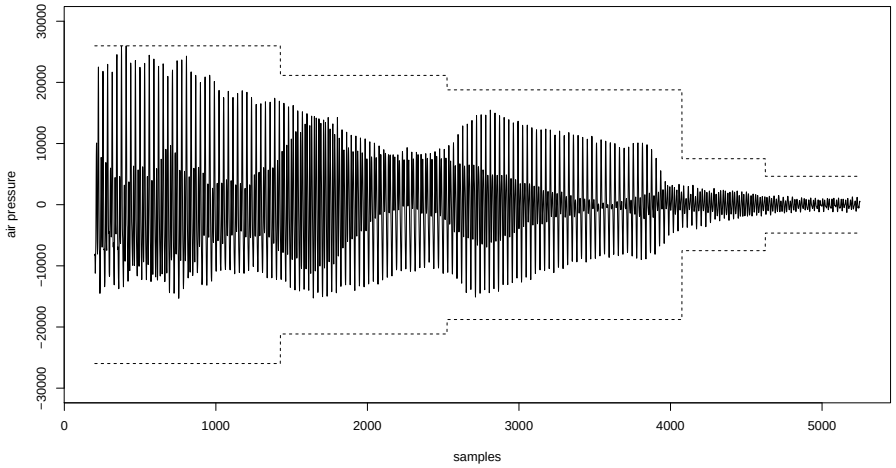



FIGURE 8. The best piecewise constant amplitudes (shown by the bounding lines), under the Poisson process event model, fit to the first note in the Bach music (which has a period of $j = 31$), where $c_0 = 100$. This generic model correctly describes the attack-decay pattern of the classical guitar.

ordinary Poisson process

$$P(s_{[1,t]}, \mu) = c_0 + \min_u \left( E(s_{(t-u,t]}, \mu) + P(s_{[1,t-u]}, \mu) \right), \tag{44}$$

for $t = 1, \ldots, n$, where the minimum is over $u = 1, \ldots, t$ with boundary condition

$$P(s_{[1,0]}, \mu) = 0.$$

An example of this model, for $c_0 = 100$ with energy $E(s, 31)$ (Equation 29), fit to the first note in the Bach music is shown in Figure 8. The square-root amplitudes are shown (positively and negatively) by the bounding lines. In the next section we derive the variable amplitude energy $E(s, \mu)$ of a sound wave $s$ where $\mu$ is a sum of periodic Brownian motion spectra. This will allow the Poisson process event model $P$ to be fit to the rest of the events in the Bach music.

3.1.3. *Polyphonic Gaussian energies.* Recall that the acoustic models developed thus far have all been sparse - meaning density evaluation is $O(n)$. It turns out that it is not possible to maintain sparsity in the presence of polyphony. The reason for this is that the inverse of a sparse matrix is in general not sparse. Let $s = s_1 + s_2$ where $s_i \in \mathbb{R}^n$ for $i = 1, 2$. Thus $s$ is the superposition of two signals, for example, $s_1$ could be the sound produced by a piano key and $s_2$ the sound produced by a second key and $s$ is the total sound of the two note piano chord. Following the development in the last chapter, we'll model this as $s_1 \sim$ Gaussian with circulant precision $Q(a)$ and independently $s_2 \sim$ Gaussian with precision $Q(b)$. Due to independence, the covariance matrix of the chord $s$ will be $Q(a)^{-1} + Q(b)^{-1}$. Thus the precision matrix of $s$ is $Q = Q(a)Q(b)Q(a+b)^{-1}$, since

$$\begin{aligned}\left(Q(a)^{-1} + Q(b)^{-1}\right) Q &= (Q(b) + Q(a))Q(a+b)^{-1} \\ &= Q(b+a)Q(a+b)^{-1} \\ &= I.\end{aligned}$$

Hence even if $Q(a)$ and $Q(b)$ are sparse, $Q$ will not usually be sparse because $Q(a+b)^{-1}$ isn't likely to be sparse. This example shows that when working with superpositions, covariance matrices appear as factors in the precision matrices. Hence the advantage of parameterising Gaussian models in terms of sparse precision matrices, as opposed to dense covariance matrices, is lost when it comes to superpositions.

None of this is a proof that $Q$ could never be sparse. But when $s_1$ is approximately $j$ periodic white noise and $s_2$ is independent white noise, the precision matrix $Q = Q(c) \in \mathbb{R}^{n \times n}$ has a top row $c$ with non-zero elements at $c_0$, $c_j$, $c_{2j}$, $c_{3j}$ etc... (and then reflected). Hence the number of non-zero elements of $c$ grows with $n$ and $Q(c)$ is therefore not sparse. If a

sparse signal in the presence of white noise is no longer sparse it's unlikely adding something more complex than white noise would be sparse.

The consequence of the above observation is that in the case of polyphonic audio, density evaluation is best based on the random Fourier representation:

$$p(s) = e^{-\frac{1}{2\sigma^2}s^T Q s}/z = e^{-\frac{1}{2\sigma^2}\sum_k \lambda_k |\hat{s}_k|^2} \cdot \left(\frac{\prod_k \lambda_k/\sigma^2}{(2\pi)^n}\right)^{\frac{1}{2}}, \tag{45}$$

where $\sigma^2 > 0$ is amplitude and $\lambda_k = \mu_k^{-1}$ are the eigenvalues of $Q$, which for the above example are given by

$$\mu_k = 1/p_a(\omega^k) + 1/p_b(\omega^k), \quad (k = 0, \ldots, n-1),$$

and as usual $\mu_k = \mu_{n-k}$ and $\omega = e^{2\pi i/n}$.

Without a prior on $\sigma^2$, the 'energy' (negative log likelihood, ignoring constants) is given by $E(s,\mu) = \sum_{k=0}^{n-1} \log(\sigma^2 \mu_k)$ where $\sigma^2 = \frac{1}{n}\sum_{k=0}^{n-1} \lambda_k |\hat{s}_k|^2$. With a uniform prior on $\log \sigma$ the energy and amplitude are given by

$$E(s,\mu) = (n+1)\log(\sigma^2) + \sum_{k=0}^{n-1} \log(\mu_k), \quad \sigma^2 = \frac{1}{n+1}\sum_{k=0}^{n-1} \lambda_k |\hat{s}_k|^2. \tag{46}$$

We take $E\left(s, \sum_l \mu(l)\right)$ as the basic energy for $s = \sum_l s_l$ a superposition of signals $\{s_l\}$ each with spectrum $\mu(l)$.[16]

3.2. **Inference via the Binding Energy.** Combining the periodic Brownian motion polyphonic Gaussian energy, the Poisson process acoustic event model and the random chord

[16]A small modification can be made to this energy by letting $\mu_0 = \mathbb{E}|\hat{s}_0|^2$ be a free parameter. The reason for doing this is that $|\hat{s}_0|^2 = \left(\sum_{k=0}^{n-1} s_k\right)^2/n$, which isn't something the spectral templates really ought to try and predict. The maximum likelihood estimator for $\mu_0$ is $|\hat{s}_0|^2/\sigma^2$, which means the estimator for the amplitude, without priors, becomes $\sigma^2 = \frac{1}{n-1}\sum_{k=1}^{n-1} \lambda_k |\hat{s}_k|^2$. Not much is gained in doing this, but we mention it merely to point out that should only part of the spectrum $\mu$ of a signal be known, then it is easy to allow the unknown part to be estimated from the data.

prior, we get a total energy

$$E(s, \{t_n\}, \{j_{n,i}\}, m) = c_1 m + \sum_{n=1}^{m} P\left(s_{[t_n, t_{n+1}), \sum_{j \in (j_{n,1}, \ldots, j_{n,k_n})} \beta(j)}\right) + c_2 k_n. \tag{47}$$

To minimise this energy (for fixed $s$) we here propose an algorithm based on finding the boundaries between chords using a general quantity known as the *binding energy* ([Mumford and Desolneux, 2010], p.g. 40). The idea is to start by assuming boundaries between all the samples in a recording $s \in \mathbb{R}^N$, so that the initial segmentation is $\{s_1, \ldots, s_N\}$. Then look over adjacent segments $(s, s')$ and consider if a better segmentation can be gotten by joining $(s, s')$ into a single event $ss'$. This is measured using the binding energy $B(s, s')$ between $s$ and $s'$, which we define as

$$B(s, s') = c_1 + P(s) + P(s') - P(ss'). \tag{48}$$

In this equation,

$$P(s) = \min_{\mu} P(s, \mu) + c_2 \cdot \text{width}(\mu), \tag{49}$$

where width($\mu$) is equal to the number of signals making up the superposition. In the case that $\mu$ is the spectrum of a chord, then this will simply be $k$, the number of periods. But we use this more general notation because the superposition may be non-musical (e.g. sports commentary over the sound of a crowd) and the algorithm would still apply.

Now, if $B(s, s') < 0$, it means the probability associated with the concatenated wave $ss'$ is smaller when $s$ and $s'$ are considered parts of the same acoustic event than when considered independent. In such a situation it would be preferable to retain the boundary between $s$ and $s'$ otherwise the two segments should be joined together to form a larger segment. It is possible to merge precisely those pairs leading to the greatest combined binding energy

$\mathcal{B} = \sum_{i \text{ left bind}} B(s_{i-1}, s_i)$ using the following recursion

$$\mathcal{B}(i,0) = \max\{\mathcal{B}(i-1,0), \mathcal{B}(i-1,1)\}$$

$$\mathcal{B}(i,1) = B(s_{i-1}, s_i) + \mathcal{B}(i-1,0)$$

$$\mathcal{B} = \max\{\mathcal{B}(N,0), \mathcal{B}(N,1)\}$$

where $\mathcal{B}(i,0)$ gives the maximum binding energy over $s_1, \ldots, s_i$ assuming $s_{i-1}$ and $s_i$ are not grouped together and $\mathcal{B}(i,1)$ is the maximum binding energy assuming they are. This calculation is necessary because it might be the case that the triple $(s_{i-1}, s_i, s_{i+1})$ has positive binding energy across both its boundaries $(s_{i-1}, s_i)$ and $(s_i, s_{i+1})$. This calculation decides which is the best to make, taking into account all the other binding decisions at the current stage. This whole grouping process is then repeated using the new, larger segments, and so on, until negative binding energy exists between all adjacent segments.

3.2.1. *Forward-Loop-Reduce.* In the case of polyphonic music, the main complication with the methodology is to overcome the curse of dimensionality and search over the $\sim 2^{|J|}$ possible combinations of periods in order to calculate $P(s)$. This search can only be done approximately and we try out here a simple method based on the forward/backward variable selection technique used extensively in statistics ([Wasserman, 2004]). There are three parts to this algorithm (1) forward, (2) loop, (3) reduce. Letting $K$ be some fixed upper limit on the size of a chord, the forward part consists in iteratively solving for the next best period

$$j_{k+1} = \arg\min_{j \in J} P\left(s, \sum_{i=1}^{k} \beta(j_i) + \beta(j)\right)$$

for $k = 0, \ldots K-1$, avoiding duplicates. A co-ordinate descent routine then loops over each period in turn and locally optimises it

$$j_k = \arg\min_{j \in J} P\left(s, \sum_{i \neq k} \beta(j_i) + \beta(j)\right)$$

again avoiding duplicate periods. Finally a backward sweep considers the best single deletions, to give a nested sequence of solutions of size $k = K, K-1, \ldots, 1$. The best of these is selected after incorporating the width penalty $c_2 \cdot k$. It is necessary to not stop after the forward sweep because a common error (similar in nature to *period doubling*[17]) is found with polyphonic music where the best fitting single period $j_1$ is (if possible) the lowest common multiple of the (most dominant) true periods.

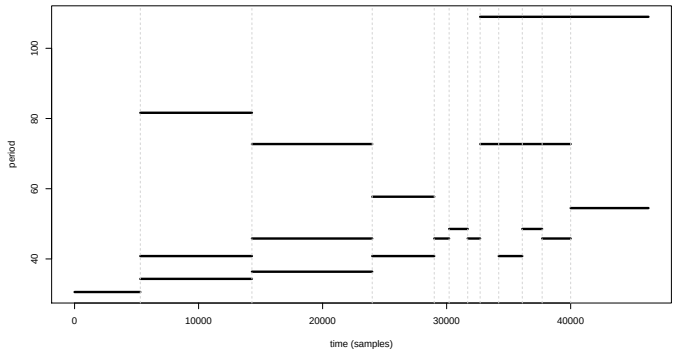
3.2.2. *Segmenting Bach's Chaconne.* We now return to the Bach music discussed at the beginning of the chapter and see if our methods can reproduce the hand segmentation shown in Figure 7 or equivalently the transcription shown in plot (a) of Figure 9.

The classical guitar has only six strings, so it makes sense to limit the number of simultaneous sounds at $K = 6$. Using a set of periodic Brownian motion spectral templates $\{\beta(j)\}$ for $j \in J$, where $J$ corresponds to a guitar tuned according to equal temperament[18] (and $\alpha = e^7$ and $\gamma = .5$), we parsed the music under the random chord prior with $c_0 = 100$, $c_1 = -300$ and $c_2 = 350$. Since each chord must contain at least one note, these prior parameters make the cost of inserting a boundary 50 and the cost 350 for each note in a chord after the first.

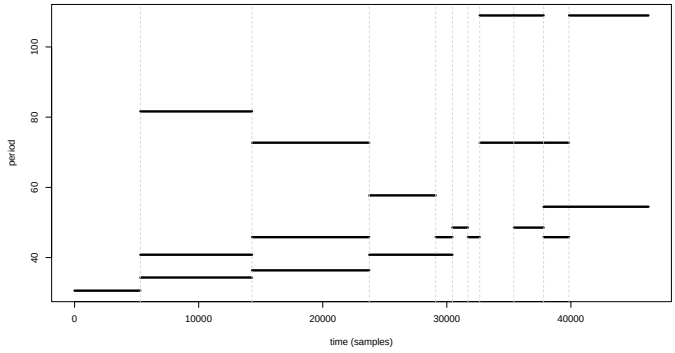
The resulting transcription (see plot (b) of Figure 9) is correct over the first seven chords but there are errors during the final five segments. It is quite telling that the errors emerge

[17]Period doubling is a common error in these sorts of algorithms whereby instead of the true period $j$ having the minimum energy, a period of the form $dj$ for integer $d \geq 2$ has minimum energy. See [Mumford and Desolneux, 2010], page 106

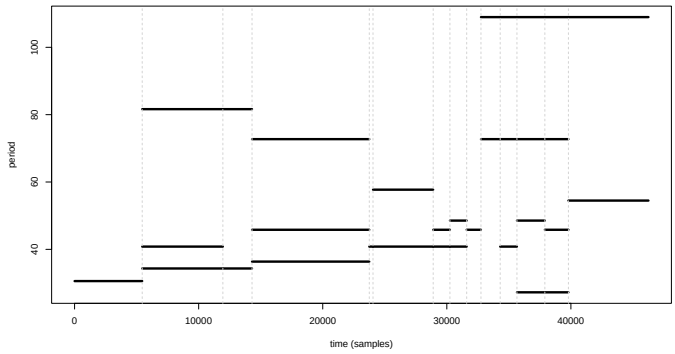
[18]That is, the periods are of the form $8000/\left(440 \cdot 2^{(k-49)/12}\right)$ for $k = 1, \ldots, 88$.




(a) By hand transcription




(b) Random chord transcription




(c) Overlapping chords transcription

FIGURE 9. (a) The transcription of the Bach music produced by the author. There are twelve segments, the first seven correspond to chords, the remaining segments consist of some short notes overlapping some longer notes and chords. (b) The transcription using the random chord prior. Over the first seven segments the transcription is accurate, but it breaks down when it meets the overlapping notes. (c) The transcription produced by the overlapping chord prior. This transcription contains both chords and preserves detail of individual notes and also fixes broken-notes. See `segovia.wav`.

as soon as the music breaks the assumption of the random chord model. The model does not have any notion of a note continuing beyond the start and end times of other notes. When transcribing with the random chord prior, this leads to broken-notes, e.g. the long period 109 note has a gap in it. Ideally we'd want the inference to look to the left and right and see that in both adjacent segments there is a 109 period and so infer that it is likely to be one long note rather than two notes with a gap in the middle. Note that the missing middle piece appears further down the plot at period 54.5. This is no accident since $2 \times 54.5 = 109$ because as mentioned earlier periods that are integer multiples of some common period sound nearly the same.

Another problem with the random chord prior is that it is quite hard to tune the hyper-parameters to simultaneously preserve all the chord segments $m$ and in each segment contain the correct number of periods $k_n$. The settings used here incorrectly merge segments eight and nine and additionally fail to detect one of the periods in the ninth segment. Reducing $c_2$ to allow more periods brings in additional false periods elsewhere in the transcription before this missing note is recovered. Ideally the model should constrain not the number of periods in each segment $k_n$, but rather the number of true notes (i.e. recognising that notes extend beyond segment boundaries). This would allow the transcription to be regularised to be free of extraneous notes, without this also leading to this over-merging behaviour

A better transcription can be achieved with a more faithful prior - the *overlapping chords* model - which is described in the next section. This model fixes broken notes and penalises the true number of notes rather than the number of periods in a segment.

3.2.3. *Overlapping chords prior and inference algorithm.* The variables in this prior are the number of chords $m$, the chord start times and periods $\{t_n\}$, $\{j_{n,i}\}$, along with each note's

own particular duration $\{d_{n,i}\}$. These variables are given the following density

$$p(m, \{t_n\}, \{j_{n,i}\}, \{d_{n,i}\}) = \prod_{n=1}^{m} \theta e^{-\theta(t_{n+1}-t_n)} \pi^{k_n} (1-\pi)^{|J|-k_n} \lambda^{k_n} e^{-\lambda \sum_{i=1}^{k_n} d_{n,i}}. \tag{50}$$

Hence the chords are a Poisson process in time, with the notes uniformly sampled from $J$ (with probability $\pi$) and of exponential duration. In this model $\theta$ controls the tempo of the music and $\lambda$ controls the sustain of the instrument.

The overlapping notes prior has the following energy

$$E(m, \{t_n\}, \{j_{n,i}\}, \{d_{n,i}\}) = c_1 m + c_2 \sum_n k_n + c_3 \sum_{n,i} d_{n,i}, \tag{51}$$

which can be seen to generalise the random chord energy (Equation 43) by the addition of a penalty on the total duration of all the notes and hence this model contains an additional tuning parameter $c_3 \in \mathbb{R}$. But also what's changed is the very definition of a note. In this model a new note is signalled by a period in a segment which is not matched to a period in the previous segment - a Markov type dependence.

What follows is a generalisation the binding energy segmentation algorithm to deal with this new way of carving up a piece of music. Begin by supposing that we have some initial segmentation $\{s_1, \ldots, s_N\}$ with guesses for the periods at each segment $\{J_1, \ldots, J_N\}$ where $J_i \subset J$. Then the total combined energy of this initial segmentation can be written

$$E(\{s_i\}, \{J_i\}) = \sum_{i=1}^{N} c_1 \mathbb{I}_{J_i - J_{i-1} \neq \varnothing} + c_2 |J_i - J_{i-1}| + c_3 \cdot \text{len}(s_i) \cdot |J_i| + P_{J_i}(s_i), \tag{52}$$

where $J_0 = \varnothing$ and $P_{J_i}(s_i) = P\left(s_i, \sum_{j \in J_i} \beta(j)\right)$.

As before, we'll consider reductions to this energy by left-merging adjacent segments $(s_{i-1}, s_i)$ while simultaneously fitting the best periods $J_{i-1}^{\text{new}}$ using the forward-reduce algorithm[19] applied to the following equation

$$J_{i-1}^{\text{new}} = \arg\min_{J' \subset J} P_{J'}(s_{i-1}s_i) + Q_i(J') + c_3 \cdot \text{len}(s_{i-1}s_i) \cdot |J'| \tag{53}$$

where $1 \leq |J'| \leq K$[20] and $Q_i$ contains the prior coupling terms between adjacent sets of periods which from Equation 52 is given by

$$Q_i(J') = c_1 \left( \mathbb{I}_{J' - J_{i-2} \neq \varnothing} + \mathbb{I}_{J_{i+1} - J' \neq \varnothing} \right) + c_2 \left( |J' - J_{i-2}| + |J_{i+1} - J'| \right), \tag{54}$$

with $J_{N+1} = \varnothing$.

Thus associated with each pair $(i-1, i)$ is a binding energy given as the *difference in energy* between the old segmentation where $s_{i-1}$ and $s_i$ are considered separate and the new segmentation where these two pieces of the sound wave are grouped together. Hence the binding energy $B(s_{i-1}, s_i)$ is given by

$$B(s_{i-1}, s_i) = e_{i-1} + e_i + q_{i-1} + q_i + q_{i+1} - \left( e_{i-1}^{\text{new}} + q_{i-1}^{\text{new}} + q_i^{\text{new}} \right), \tag{55}$$

[19]The loop stage hes been dropped in the overlapping chords prior to speed up the algorithm.

[20]An alternative to a fixed upper limit $K$ on the number of periods is to restrict $|J'|$ to be at most $D$ more than the maximum number of periods in either of the two segments being bound together (for some small integer $D$), that is $1 \leq |J'| \leq D + \max(|J_{i-1}|, |J_i|)$. This can greatly speed up the algorithm by allowing the search space to adapt to the number of simultaneous sounds in each region.

where

$$e_i = P_{J_i}(s_i) + c_3 \cdot \text{len}(s_i) \cdot |J_i|$$

$$q_i = c_1 \mathbb{I}_{J_i - J_{i-1} \neq \varnothing} + c_2 |J_i - J_{i-1}|$$

$$e_{i-1}^{\text{new}} = P_{J_{i-1}^{\text{new}}}(s_{i-1}s_i) + c_3 \cdot \text{len}(s_{i-1}s_i) \cdot |J_{i-1}^{\text{new}}|$$

$$q_{i-1}^{\text{new}} = c_1 \mathbb{I}_{J_{i-1}^{\text{new}} - J_{i-2} \neq \varnothing} + c_2 |J_{i-1}^{\text{new}} - J_{i-2}|$$

$$q_i^{\text{new}} = c_1 \mathbb{I}_{J_{i+1} - J_{i-1}^{\text{new}} \neq \varnothing} + c_2 |J_{i+1} - J_{i-1}^{\text{new}}|.$$

At each stage the pairs leading to the greatest total binding energy $\mathcal{B}$ are grouped together using a recursion similar to the one given earlier except now if $(s_{i-1}, s_i)$ are grouped then we must be careful not to also group either $(s_{i-3}, s_{i-2})$ or $(s_{i+1}, s_{i+2})$ due to the occurrence of $J_{i-2}$ and $J_{i+1}$ in $Q_i$.

The transcription produced by this prior and algorithm on the Bach music is shown in plot (c) of Figure 9. The hyper-parameters were set at $c_0 = 100$, $c_1 = 5$, $c_2 = 90$ and $c_3 = .07$. These settings add a small bias in favour of connecting notes which start at nearly the same time into chords and then regularises the number of notes and also their duration. The transcription is quite similar to the hand transcription except it has allowed different notes in the same chord to have differing durations. The hand transcription is only an approximation to the truth and so these additional details about the precise timing of the notes may well be closer to the truth. However, what is certainly an error is the insertion of an extra period 27 note spanning the second to last two segments.

Mistakes like these occur for a range of reasons. One of these being if the boundaries are slightly off then a segment contains a soundwave with non-constant periods. The weakness with the binding energy algorithm is that the early grouping decisions determine how precisely

the true boundaries are recovered, but these early decisions are based on very little data. There are lots of ways to try and arrive at lower energy solutions: (1) loop over each *complete* note and consider if changes to it's value (including deletion) reduces the energy; (2) consider refining the boundaries by moving them a little to the right or left; (3) using some more basic stochastic acoustic model during the early groupings focused on properties of the soundwave other than periodicity such as strong continuity. However, even with perfect boundaries, the search algorithm (forwards-backwards) can fail to arrive at the true periods.

A comparison between the original music and the model's reconstruction is contained in the file `segovia.wav`. Listening to this comparison is the easiest way to check that the segmentation and classification is close to the original, but also it shows up how much of the statistics of the real world sounds have been captured by the model. Now that attack and decay have been properly modelled, the reconstructed music is fairly plausible classical guitar but there are many audible discrepancies. (a) The reconstruction contains 'pops'. These are caused by discontinuities in the sampled sound-wave at points where the amplitude abruptly changes (a consequence of the simplistic piecewise constant model). (b) At the times when two or more notes overlap they are given the same amplitude; but in reality, some simultaneous notes are quieter than others. (c) The rich harmonics and vibrato of the guitar and playing are missing from the reconstruction and (d) fret-board squeaks and the background acoustics are also not reproduced.

3.3. **Complex sounds and random filters.** In this section we attempt to account for some of the more complex sonic details appearing in audio by equipping the Gaussian models with additional variables describing deviations from their mean Fourier transforms. These new models are then used to analyse an acoustically rich piece of music - *Bang-Bang* by Nancy Sinatra - containing vocals accompanied by an electric guitar. The core idea is to

make periodic Brownian motion less prescriptive of what the sound-wave should be like. This then enables the binding energy segmentation algorithm handle the varied sounds created by the entire range of possible instruments.

3.3.1. *Random filters I - the conjugate prior.* Recall the discussion on the lack of fit to the flute sample (section 2.2.1). There, the observed spectrum $|\hat{s}|^2 \in \mathbb{R}^n$ had visibly greater first and second harmonics than that predicted by the model. Also, the model underestimated many of the later harmonics. It would be possible to correct this by modifying the spectral templates to contain extra power at certain frequencies. For example, to add extra power at integer frequency $k$, one could multiply some constant $0 < \tau_k \leq 1$ to $\lambda_k = \mu_k^{-1}$. Clearly this is going to require a large number of constants $\{\tau_k\}$ to make a real difference to the spectrum. These constants could come by designing a filter (like the resonance filter for modelling vowels - see section 2.2.3). Designing a filter is the correct thing to do when the event classification depends on the presence/absence of the property specified by the filter (as is the case for vowels). However more often than not the precise shape of the spectrum, which gives each sound its unique sound quality, is not of interest per-se. In these circumstances it is impractical to model all the nuances of the spectrum, but also, just like with amplitude variations, it isn't always safe to simply ignore them. This leads to the idea of using a *random filter*. That is, instead of specifying exactly what the filter is ahead of time, we only need specify a distribution over possible filters and then maximise the resulting distribution to find the best fitting filter in each new situation. We do this here by taking advantage of the exponential family form of the circulant Gaussian and use a Gamma *conjugate* prior over (multipliers of) the eigenvalues of the precision matrix $Q$. The use of a prior here is absolutely essential because if the filter $\{\tau_k\}$ is not constrained, the random Fourier density with the maximum likelihood estimates of the filter plugged

back in, no longer depends on how well the spectral model $\lambda$ matches the data $|\hat{s}|^2$ (a case of over-fitting).

The *random Fourier filter* density is defined as

$$p(s \mid \{\tau_k\}, \sigma) = e^{-\frac{1}{2\sigma^2}\sum_{k=0}^{n-1}\tau_k\lambda_k|\hat{s}_k|^2} \cdot \prod_{k=0}^{n-1}\left(\frac{\tau_k\lambda_k}{2\pi\sigma^2}\right)^{\frac{1}{2}}. \tag{56}$$

The idea is to place a Gamma prior on the filter $\{\tau_k\}$, which means the joint distribution is written

$$p(s, \{\tau_k\}, \sigma) \propto \frac{1}{\sigma^{n+1}}\prod_{k=0}^{n-1}\tau_k^a\lambda_k^{\frac{1}{2}}e^{-\tau_k\left(\frac{1}{2\sigma^2}\lambda_k|\hat{s}_k|^2+b\right)},$$

which places a Gamma($a+1/2$, $b$) density on $\tau_0$ (and $\tau_{\frac{n}{2}}$ if $n$ is even) and a Gamma($2a$,$2b$) density on the rest. The mode of this density, for fixed $s$, is located at the solution to the following equations

$$\tau_k = \frac{a}{\lambda_k|\hat{s}_k|^2/(2\sigma^2)+b}; \qquad \sigma^2 = \frac{1}{n+1}\sum_{k=0}^{n-1}\tau_k\lambda_k|\hat{s}_k|^2.$$

Combining these estimators gives a fixed point iteration scheme for finding $\sigma^2$ given some initial guess (say $\frac{1}{n+1}\sum_{k=1}^{n-1}\lambda_k|\hat{s}_k|^2$):

$$\sigma^2 = \frac{a}{n+1}\sum_{k=0}^{n-1}\left(\frac{1}{2\sigma^2}+\frac{b}{\lambda_k|\hat{s}_k|^2}\right)^{-1}.$$

Multiplying these two estimates together and then with the model spectrum, gives the following heuristic equation for the inferred spectrum

$$\hat{\mathbb{E}}|\hat{s}_k|^2 = \sigma^2\mu_k/\tau_k = \frac{|\hat{s}_k|^2/2+b\sigma^2\mu_k}{a}, \tag{57}$$

which is a weighted average of the empirical spectrum with the mean spectrum. Substituting this new spectrum back into the joint distribution gives the model's energy

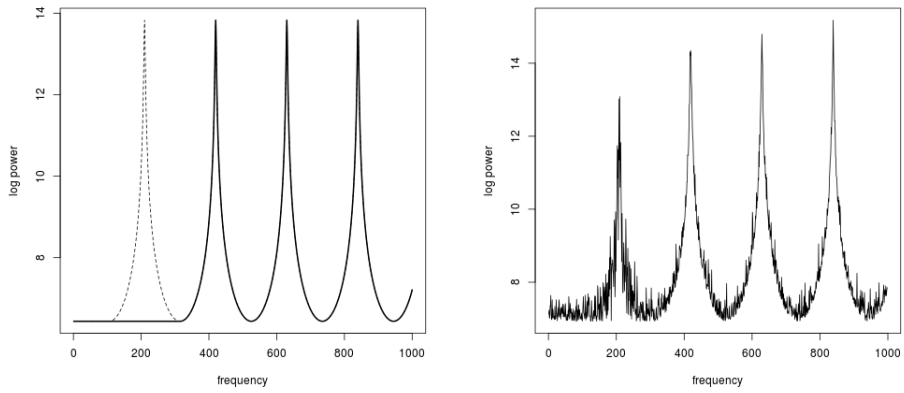
$$E(s,\mu) = (n+1)\log(\sigma^2) - \sum_{k=0}^{n-1} \log\left(\lambda_k \tau_k^{2a}\right), \tag{58}$$

which we will refer to as the *Gamma-Gaussian* energy.

A situation of some interest relates to Piano notes, because these are well known to often lack their first harmonic (or *fundamental*). It is simple enough to model this by flattening the spectral templates up-to the midpoint between the first and second harmonic. Figure 10, plot (a), shows the flattened fundamental. However, when the instrument being modelled does not lack the first harmonic, these flattened templates are less accurate. To show this we sampled 1000 monophonic periodic white noise signals, with random periods, and looked to see where the minimum of the plain Gaussian energy (Equation ) with flattened white noise templates is located. The minimum is located at the correct period only 40% of the time. We then gave the Gamma-Gaussian energy the wrong templates, the model still scored 100%[21]. The reason for this is that the filter is filling in the missing fundamental using the data as a guide. This can be seen in the inferred template shown in plot (b), which is calculated using equation 57.

When we switch the templates around, so that now the model contains the fundamental, but the data lacks the fundamental, the Gaussian energy still accurately classifies the sampled notes. This property also holds for the Gamma-Gaussian energy, however the inferred spectrum *incorrectly* contains the fundamental, when really $\{\tau_k\}$ ought to have flattened it out. Changing the hyper-parameters can solve this problem, but not without significantly degrading the period-classifying accuracy of the Gamma-Gaussian model.

[21]Both models returned the true period with 100% accuracy when given the correct (un-flattened) templates.




(a) (b)

FIGURE 10. Plot (a) shows a periodic white noise template with the fundamental removed. This emulates the spectrum of a piano note. Plot(b) shows the Gamma prior's best reconstruction of the spectrum, with hyper-parameters $a = b = 5$. Notice that the prior has recovered the missing fundamental and this leads to a more accurate classification.

3.3.2. *Random filters II - the log Gaussian prior.* Ideally we want $\tau_k$ to be centred on 1 with symmetric flexibility to both inflate and deflate and thus arguably $\log \tau_k$ ought to be a zero mean Gaussian. This suggests the use of the *log-Gaussian* prior on $\tau_k$, in combination with the random Fourier filter density and a uniform prior on $\log \sigma$, this has joint density

$$p(s, \{\tau_k\}, \sigma) \propto \frac{\sqrt{\prod_k \lambda_k}}{\sigma^{n+1}} \prod_{k=0}^{n-1} e^{-\frac{1}{2}\left(\log^2 \tau_k / b + \tau_k \lambda_k |\hat{s}_k|^2 / \sigma^2\right)} \tag{59}$$

where $b > 0$ is the variance of $\log \tau_0$ (and $\log \tau_{\frac{n}{2}}$ if $n$ is even) and $b/2$ the variance of $\log \tau_k$ ($k \neq 0, \frac{n}{2}$). Jointly maximising with respect to $\{\tau_k\}$ and $\sigma^2$, gives the following set of simultaneous equations

$$e^{\frac{b}{2\sigma^2} \tau_k \lambda_k |\hat{s}_k|^2} \tau_k = 1; \qquad \sigma^2 = \frac{1}{n+1} \sum_{k=0}^{n-1} \tau_k \lambda_k |\hat{s}_k|^2;$$

Unlike with the Gamma prior, there is no closed form formula for $\{\tau_k\}$, however they can be expressed in terms of the principle branch of the Lambert $W$ function:

$$\tau_k = W\left(\frac{b}{2\sigma^2}\lambda_k|\hat{s}_k|^2\right)\left(\frac{b}{2\sigma^2}\lambda_k|\hat{s}_k|^2\right)^{-1} \in [0,1]$$

since $\frac{b}{2\sigma^2}\lambda_k|\hat{s}_k|^2 > 0$. These can be substituted into the equation for $\sigma^2$ to give an iterative algorithm for the amplitude. The energy of the model is

$$E(s,\mu) = \log(\sigma^2) + \sum_{k=0}^{n-1}\log\sigma^2\mu_k + \frac{1}{b}\sum_{k=0}^{n-1}\log^2\tau_k \tag{60}$$

which we refer to as the *log-Gaussian* energy. This model correctly both inflates and deflates the model templates so long as $b$ is large enough. and the energy effectively always has a minimum located at the true period. However, there is something of a puzzle with this energy in that our motivation of the log-Gaussian prior was to allow $\tau_k$ to be free to vary above and below 1, but the above maximum-a-posteriori estimate forces $\tau_k \leq 1$.

Iteratively solving for $\{\tau_k\}$ and $\sigma^2$ by successive substitution can be quite slow to converge. In our experiments, it required about 200 iterations to achieve full convergence. This is hugely time consuming given the cost of evaluating the Lambert W function. During the early iterations the filter $\{\tau_k\}$ takes on its overall shape and latter iterations are mere refinements. Thus it is reasonable to limit the number of iterations, say to about 10, and accept a solution somewhat short of the mode. Further speed improvements can be obtained by using an approximation to Lambert's $W$. In our implementation we made use of the fourth approximation given in [D'Angelo et al., 2019].

Our conclusions are that (a) the Gamma prior can be used to fill in missing power in the spectrum and that this can improve accuracy in certain situations. (b) The Gamma

prior cannot be used to delete energy from the spectrum without losing accuracy. (c) The log-Gaussian can both fill-in and delete power and still retain accuracy. (d) Piano notes ought not be any more challenging to accurately classify than notes which do possess a fundamental, even when using the (incorrect for piano) non-flattened spectrum. However it is conjectured that flexible models fit to piano notes may not generalise well to non piano notes because they will have modelled the missing fundamental.

In [Leonard and Hsu, 1992] Gaussian priors for the matrix logarithm of general precision matrices were first introduced and shown to have certain statistical advantages over the conjugate Wishart prior. The energies developed over the previous two sections are maximum-a-posteriori versions of this applicable when the precision matrices are circulant. Specifically, the Gamma prior arises as a re-normalised Wishart and since circulant matrices have a fixed (and known) eigenbasis only the eigenvalues need be given a log Gaussian distribution.

3.3.3. *Nancy Sinatra's Bang-Bang.* Bang-Bang by Nancy Sinatra is a good example of music which is not well modelled by the ordinary Gaussian likelihood. The electric guitar has some very heavy effects on it and the singing contains all the complexity of speech. In-particular there are strong resonance frequencies which articulate the various vowel sounds and make the voice recognisable. Furthermore, both these things are happening at the same time. To my mind, by far the simplest way to deal with this is not by developing specific models for guitar, guitar plus X effect, guitar plus X effect plus Nancy Sinatra etc... because this isn't going to scale. Instead, a general purpose statistical model which is able to adapt to the acoustics of the situation should be used.

We ran the overlapping chords prior and binding energy algorithm, with parameters $c_0 = 100$, $c_1 = 10$, $c_2 = 90$ and $c_3 = .08$, using the log-Gaussian energy, with $b = 1$ and capping the number of iterations in the estimation of the filter $\{\tau_k\}$ to 4. The results on the

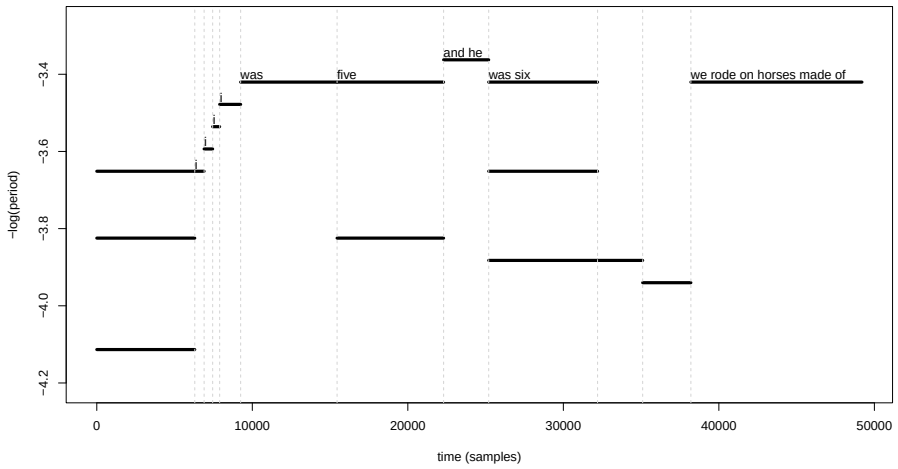



FIGURE 11. Six seconds of Nancy Sinatra's Bang-Bang transcribed using the log-Gaussian energy. The lyrics are not part of the output of the model, but they have (to some degree) been reproduced in the reconstruction `bangbang.wav`.

first 6 seconds (starting just after the solo guitar intro) of the music are shown in Figure 11. The highest melodic line corresponds to the singing and we have indicated lyrics on the plot to make this easier to track. The bottom notes and chords are the accompanying electric guitar. The reconstruction `bangbang.wav` has picked up the effects on the guitar and the quality of the voice and it's possible to make out some of the words being sung.

## 4. Conclusion

This thesis has introduced a stochastic model for audio events with four main features: (1) the event can be built up out of multiple simultaneous individual sounds, each sound being described by a spectral template model (see Section 2 for a range of templates that can be used); (2) the amplitude of the sound can change over the course of the event and the location and values of the amplitude changes are readily inferred (see Section 3.1.2 for the amplitude model and it's inference algorithm); (3) the event itself can contain features not

explicitly present in the spectral templates but which must be acknowledged for successful classification, this is accomplished through the use of spectral hyper-priors (see Section 3.3 for two proposed forms for these priors); and (4) the likely prior structure of the audio is determined by the *overlapping chord model* (introduced in Section 3.2.3).

This event model has been used to infer a discrete representation of long and quite complex music and to then re-synthesise this music from scratch. The reader should refer to the audio clips provided to judge for themselves how successful these representations are. The results are far from perfect and our limited examples have merely scratched the surface of a full and complete investigation into the properties of this system. However we feel able to make the following general statements which serve to summarise this thesis and suggest directions for future work.

The basic audio template we have introduced - (*periodic Brownian motion*) - extends the one introduced in [Mumford and Desolneux, 2010] (which captures periodicity) to further capture the fact that most sounds have decreasing power in the spectrum with increasing frequency. We have found these templates to be remarkably effective both in classifying and simulation. However it is likely in full scale applications that the additional flexibility provided by the ability to model missing harmonics and resonance peaks will be needed (see Sections 2.2.2 and 2.2.3). When combined with the spectral hyper-priors to mop-up any remaining details missed by these models, the resulting audio models are likely to be very high performing. However there is a catch. As the number of templates increases, the time it takes to search for the best fitting set of templates to any section dramatically increases. The *forward-loop-reduce* (see Section 53)) method we have devised for our simple investigations does not search through all possible combinations, as even with a limited number of periods, and without the additional possibility of missing harmonics and resonance peaks, a full

search would be impossible (*curse of dimensionality*). This is the fundamental weakness in this approach, and furthermore there does not seem to be a simple solution... Except to point out that this problem will vanish with the passage of time. It is a simple fact that humans can't distinguish say more than a few hundred periods, and thus there really is a clear upper bound on the number of distinguishable possibilities. For this reason I believe these discrete search methods, which while currently not in vogue, will make a comeback once computers catch up.

Aside from the templates, another important feature in the preceding algorithm is the amplitude model. By not assuming the amplitude is constant over the duration of the musical event we found the classification accuracy in the context of polyphony to increase markedly. However, the model is currently missing a crucial feature. This is the prior knowledge that the amplitude for many instruments (such as piano and guitar) rapidly increases at the start of the event and then decays. This attack-decay structure needs putting into the model because then it would allow the identification of repeat notes, whereas the current model will classify repeat notes as one long note with two sharp amplitude peaks. This prior knowledge should be applied as a second-stage analysis step to the output of the amplitude-model. The amplitude model provides the amplitude of the chord at each moment in time, and thus it ought to be a simple matter to detect attack-decay and thus correct any mistakenly grouped double-notes. See Figure 8 for an image of the output of the amplitude model, which should make it obvious what is meant here.

Further, it will be recalled that in the introduction it was noted that phase information has been thrown away in the above models. This has implication for the detection of note onsets because in the algorithm of the preceding chapter onsets are only identified by the absence of a particular period $j$ followed by the introduction of $j$. As already noted, with

two adjacent $j$s the amplitudes must be analysed to detect the second $j$ - but even this is not totally guaranteed to detect the second $j$ if the second onset amplitude is commensurate with the amplitude of the first $j$ (at the time the second $j$ occurs). The solution to this problem, as investigated in [Benetos and Stylianou, 2010], is to use phase. The second $j$ will (in-all-likelihood) be out of phase with the first $j$. This can be detected by looking for phase differences associated with relevant frequencies of the two segments. It seems then that for state of the art results (like in [Benetos and Stylianou, 2010]) a second stage of the algorithm needs designing, incorporating both amplitude and phases information.

Another aspect of the algorithm I'd like to discuss here is the use of the binding energy recursion to parse the entire piece of audio. This algorithm was first described in section 3.2 and then generalised in section 3.2.3 to deal with overlap. The unusual feature of this algorithm is that the entire piece of audio is first reduced to individual samples, and then these samples are pairwise merged if they are judged to belong to the same event. For example, consider the sequence of letters *t-h-e-d-o-g*. After the first stage of grouping this might become *th-e-d-og*, then after a second stage *th-e-dog* then after a further stage *the-dog*, hopefully the grouping would stop here, but perhaps may proceed to group into *thedog* if the model isn't sufficiently realistic. In this case we are making an analogy between words and complete notes, and letters as the individual samples. Thus the entire audio is processed as a whole (albeit in stages), thus it is not processed sequentially. This is then totally unlike how a human would parse a long piece of audio, however it does open up the door to the use of parallel computation which would significantly speed up the search process.

A crucial issue I have admittedly been quite vague on is where did the values for the hyper-parameters $(c_1, c_2, \dots)$ come from? Well, these were chosen by running the algorithm multiple times and trying a bunch of values and picking the one that gave the best results

(according to me). This is of course cheating. The difficulty here is that the energies as currently written lack normalizing constants and thus there is no objective measure of goodness of fit across different values of the hyper-parameters. If these normalizing constants were put in, then with enough compute, the best fitting hyper-parameters for each audio recording could be found. This would open the door to full and complete performance evaluation and get rid of the task of hand tuning these numbers. I believe this extension of the current model would be a significant step in the right direction.

## References

[Alvarado and Stowell, 2016] Alvarado, P. A. and Stowell, D. (2016). Gaussian processes for music audio modelling and content analysis. In *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE.

[Bach and Jordan, 2005] Bach, F. R. and Jordan, M. I. (2005). Discriminative training of hidden markov models for multiple pitch tracking [speech processing examples]. In *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 5, pages v–489. IEEE.

[Benetos et al., 2019] Benetos, E., Dixon, S., Duan, Z., and Ewert, S. (2019). Automatic music transcription: An overview. *IEEE Signal Processing Magazine*, 36(1):20–30.

[Benetos and Stylianou, 2010] Benetos, E. and Stylianou, Y. (2010). Auditory spectrum-based pitched instrument onset detection. *IEEE Transactions on Audio, Speech & Language Processing*, 18(8):1968 – 1977. © 2010 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other users, including reprinting/ republishing this material for advertising or promotional purposes, creating new collective works for resale or redistribution to servers or lists, or reuse of any copyrighted components of this work in other works.

[Boersma and Weenink, 2009] Boersma, P. and Weenink, D. (2009). Praat: doing phonetics by computer (version 5.1.13).

[Bor, 1992] Bor, J. (1992). *The Raga Guide: A Survey of 74 Hindustani Ragas*. Nimbus Records.

[Cemgil et al., 2006] Cemgil, A., Kappen, H., and Barber, D. (2006). A generative model for music transcription. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(2):679–694.

[Çınlar, 2011] Çınlar, E. (2011). *Probability and Stochastics*. Graduate Texts in Mathematics. Springer New York.

[Cox and Lewis, 1966] Cox, D. and Lewis, P. (1966). *The Statistical Analysis of Series of Events*. Methuen's Monographs on Applied Probability and Statistics. Springer Netherlands.

[Davis, 2013] Davis, P. J. (2013). *Circulant matrices*. American Mathematical Soc.

[De Cheveigné and Kawahara, 2002] De Cheveigné, A. and Kawahara, H. (2002). Yin, a fundamental frequency estimator for speech and music. *The Journal of the Acoustical Society of America*, 111(4):1917–1930.

[Diggle et al., 2013] Diggle, P., Heagerty, P., Liang, K., and Zeger, S. (2013). *Analysis of Longitudinal Data*. Oxford Statistical Science Series. OUP Oxford.

[D'Angelo et al., 2019] D'Angelo, S., Gabrielli, L., and Turchet, L. (2019). Fast approximation of the lambert w function for virtual analog modelling. *In practice*, 100:8.

[Efron and Hastie, 2016] Efron, B. and Hastie, T. (2016). *Computer Age Statistical Inference*. Institute of Mathematical Statistics Monographs. Cambridge University Press.

[Fant, 1970] Fant, G. (1970). *Acoustic theory of speech production*. Walter de Gruyter.

[Grenander, 1952] Grenander, U. (1952). On Toeplitz forms and stationary processes. *Arkiv för Matematik*, 1(6):555 – 571.

[Grenander, 1996] Grenander, U. (1996). *Elements of pattern theory*. JHU Press.

[Grimmett and Welsh, 2014] Grimmett, G. and Welsh, D. (2014). *Probability: An Introduction*. Oxford University Press. Oxford University Press.

[Hawthorne et al., 2017] Hawthorne, C., Elsen, E., Song, J., Roberts, A., Simon, I., Raffel, C., Engel, J., Oore, S., and Eck, D. (2017). Onsets and frames: Dual-objective piano transcription. *arXiv preprint arXiv:1710.11153*.

[Leonard and Hsu, 1992] Leonard, T. and Hsu, J. S. (1992). Bayesian inference for a covariance matrix. *The Annals of Statistics*, 20(4):1669–1696.

[Lindgren and Sandsten, 2014] Lindgren, G. Rootzen, H. and Sandsten, M. (2014). *Stationary Stochastic Processes for Scientists and Engineers*. New York: Chapman and Hall/CRC.

[Mumford and Desolneux, 2010] Mumford, D. and Desolneux, A. (2010). *Pattern Theory: The Stochastic Analysis of Real-World Signals*. A K Peters.

[O'Shaughnessy, 1988] O'Shaughnessy, D. (1988). Linear predictive coding. *IEEE Potentials*, 7(1):29–32.

[R Core Team, 2018] R Core Team (2018). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

[Rue and Held, 2005] Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. CRC Press.

[Smith, 2021] Smith, J. O. (April 2021). *Introduction to Digital Filters with Audio Applications*. `http://ccrma.stanford.edu/~jos/filters///ccrma.stanford.edu/~jos/filters/`. online book.

[Wasserman, 2004] Wasserman, L. (2004). *All of Statistics: A Concise Course in Statistical Inference*. Springer Texts in Statistics. Springer.

[Whittle, 1951] Whittle, P. (1951). *Hypothesis Testing in Time Series Analysis*. Statistics / Uppsala universitet. Almqvist & Wiksells boktr.

[Wilkinson et al., 2019] Wilkinson, W., Andersen, M., Reiss, J. D., Stowell, D., and Solin, A. (2019). End-to-end probabilistic inference for nonstationary audio analysis. In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6776–6785. PMLR.

[Wood, 2015] Wood, S. N. (2015). *Core Statistics*. Institute of Mathematical Statistics Textbooks. Cambridge University Press.