Slope Estimation in Noisy Piecewise Linear Functions☆

Atul Ingle[a,b,*], James Bucklew[a], William Sethares[a], Tomy Varghese[b,a]

[a] *Department of Electrical and Computer Engineering, University of Wisconsin-Madison, Madison WI 53706, USA.*
[b] *Department of Medical Physics, University of Wisconsin-Madison, Madison WI 53705, USA.*

Abstract

This paper discusses the development of a slope estimation algorithm called MAPSlope for piecewise linear data that is corrupted by Gaussian noise. The number and locations of slope change points (also known as breakpoints) are assumed to be unknown *a priori* though it is assumed that the possible range of slope values lies within known bounds. A stochastic hidden Markov model that is general enough to encompass real world sources of piecewise linear data is used to model the transitions between slope values and the problem of slope estimation is addressed using a Bayesian maximum a posteriori approach. The set of possible slope values is discretized, enabling the design of a dynamic programming algorithm for posterior density maximization. Numerical simulations are used to justify choice of a reasonable number of quantization levels and also to analyze mean squared error performance of the proposed algorithm. An alternating maximization algorithm is proposed for estimation of unknown model parameters and a convergence result for the method is provided. Finally, results using data from political science, finance and medical imaging applications are

☆This work was supported in part by NIH-NCI grants R01CA112192-S103 and R01CA112192-05.

*Corresponding author. Sample MATLAB code is available on the author's website at homepages.cae.wisc.edu/~ingle/mapslope.html.

Email addresses: ingle@wisc.edu (Atul Ingle), bucklew@engr.wisc.edu (James Bucklew), sethares@ece.wisc.edu (William Sethares), tvarghese@wisc.edu (Tomy Varghese)

presented to demonstrate the practical utility of this procedure.

Keywords: piecewise linear function, MAP estimation, dynamic programming optimization, EM algorithm, alternating maximization

1. Introduction

The need for piecewise linear regression arises in many different fields, as diverse as biology, geology, and the social sciences. This paper addresses the problem of direct estimation of slopes from piecewise linear data. An important application of interest for this paper is ultrasound shear wave elastography, where ultrasonic echoes are used to track the motion of an externally generated mechanical shear wave pulse traveling through multiple tissue interfaces [19]. The time of arrival of this shear wave pulse is recorded as a function of spatial coordinates in the ultrasound imaging plane and the reciprocal of the slope of this function gives an estimate of the speed of the wave. Breakpoints (where the slope changes) indicate tissue interfaces. These estimates are useful from a clinical perspective because they provide a way to quantify mechanical properties of tissue, thereby adding value to subjective judgments about the location and size of cancerous tumors.

A similar issue in larger spatial dimensions occurs in seismology where the time of arrival of seismic waves is tracked at different locations relative to the epicenter of an earthquake. The velocity of these waves provides information about the mechanical properties of the geological medium. Piecewise linear data also occurs in the study of flow of soil through water streams and is referred to as bedload data [20].

1.1. Data Model

Assume that the piecewise linear data is generated by the following discrete time hidden Markov model (HMM). The underlying (unknown) function takes on values Z_n at each discrete index $1 \leq n \leq N$. This function value is obtained by accumulating slope values S_k up to the time index n. Zero mean Gaussian noise with variance σ^2 is added to each running sum resulting in the observed function value X_n. Also, suppose that for any n, the probability of maintaining the previous slope value is p and the probability of transitioning into a new slope value is $1 - p$. These relations can be written mathematically as follows:

$$\begin{aligned} Z_0 &= 0 \text{ with probability } 1, \\ Z_n &= Z_{n-1} + S_n, \\ X_n &= Z_n + w_n \end{aligned} \tag{1}$$

for $n = 1, \ldots, N$ where $w_n \overset{iid}{\sim} \mathsf{N}(0, \sigma^2)$. A Markov structure is imposed on the slope values as follows:

$$S_n = \begin{cases} S_{n-1} & \text{with probability } p \\ U_n & \text{with probability } 1 - p \end{cases}$$

for $n = 2, \ldots, N$ where $U_n \sim \mathcal{U}(\{0, \frac{1}{(M-1)}, \ldots, \frac{M-2}{M-1}, 1\} \setminus \{S_{n-1}\})$ denotes a discrete uniform random variable taking on one of $M - 1$ possible slope values and the initial slope value is drawn uniformly as $S_1 \sim \mathcal{U}(\{0, \frac{1}{(M-1)}, \ldots, \frac{M-2}{M-1}, 1\})$.

Another implicit assumption is that the slopes can take on values on a closed bounded interval $[s_l, s_u]$ with upper and lower limits $0 < s_l < s_u < \infty$ known *a priori*. For instance, in the ultrasound-based wave tracking application, the values of s_l and s_u can be obtained from the underlying physics which dictates that such mechanical waves travel with speeds between 0.5 to 10 m/s in homogeneous tissue. With the knowledge of s_l and s_u, the given data vector can be translated and rescaled so that all slope values lie in the interval $[0, 1]$. Hence, without loss of generality, it suffices to design a slope estimation algorithm that operates with a finite set of slopes $\mathcal{S} = \{0, \frac{1}{(M-1)}, \dots, \frac{M-2}{M-1}, 1\}$. Intuitively, this quantization step is justified because in the presence of noise it is impossible to detect the difference between slope values that differ only slightly.

1.2. Main Contributions

The main contributions of this paper are as follows:

(a) a hidden Markov model formulation of the slope estimation problem that is general enough to encompass different applications

(b) a procedure for MAP estimation of slopes from this Markov model

(c) a dynamic programming routine on a linearly growing trellis for fast MAP estimation

(d) an MSE optimality analysis of this routine via simulations and a comparison with reasonable upper and lower bounds

(e) an alternating maximization algorithm that alternately maximizes an objective function with respect to the unknown sequence of slope values and unknown model parameters to jointly estimate both of them from data

(f) a comparison of the performance of this algorithm with other methods in literature applied to real world data.

1.3. Related Work

In many real world applications, the local slope values of an observed noisy function have interesting physical interpretations. Most of the existing methods do not directly address slope estimation; rather, they attempt to fit a model to the data. For instance, standard regression or spline-based methods can be used to fit a smooth function to the data and local slopes can be estimated from this fit. However, even if the function-fitting algorithm generates optimal fits (according to a cost function such as the minimum MSE), there is no guarantee that the local slope estimates obtained from this fit are themselves optimal. This paper bypasses the need for such post-processing by directly estimating the slopes and breakpoints. This is particularly useful when the slopes correspond directly to the variables of interest and the breakpoints correspond to where those variables change.

The topic of slope estimation from noisy data is quite old; an early paper can be traced back to 1964 where the popular Savitzky-Golay differentiator [10] was introduced. Their main idea is to use a locally windowed least squares fit to estimate the slope at each data sample, where the window coefficients are chosen to satisfy a certain frequency response that mimics a high pass filter together with some level of noise averaging. Another similar technique that is used in statistics is called locally weighted least squares regression (LOWESS) [23]. However, these methods undesirably smooth out the breakpoint locations in when data has sharp transitions or jumps. In contrast, the algorithm in the present paper prevents blurring the transitions by explicitly allowing for sharp slope transitions using a Markov model.

In some situations, the raw data can be massaged using a preprocessing step so that it becomes piecewise linear. The simplest example is the case of piecewise constant data — the running sum (integral) of such data yields a piecewise linear function. Ratkovic and Eng [22] discuss a statistical spline fitting approach combined with the Bayesian information criterion (BIC) to detect abrupt transitions in political approval ratings. Data from their paper is used in Section 6.1. As a special case, their method can be applied when function values stay almost constant over long intervals and occasionally shift to a new value. In another application, Bai and Perron [16] use statistical regression techniques to detect multiple regime shifts in interest rate data. The algorithm developed in the present paper provides comparable numerical performance as the Bai-Perron algorithm as shown in Section 6.2.

Closely related problems of piecewise linear regression for noisy data have been addressed over the years. For example, an early paper by Hudson [1] focuses on a technique to obtain piecewise linear fits in a least-squares sense with only two segments. The break point location is included as a parameter in the least squares optimization problem. The method is extended by dealing with multiple breakpoint locations on a case-by-case basis which becomes combinatorially intractable as the number of breakpoints increases. Bellman [2] suggests a dynamic programming approach when the number of breakpoints is unknown. However, this method requires the use of a large number of grid points for accurate results. Gallant and Fuller [3] generalize this to the fitting of arbitrary polynomials with unknown breakpoints while requiring the function to be composed of segments with continuous first derivatives. They apply a nonlinear optimization routine (Gauss-Newton minimization) to fine-tune breakpoint locations while minimizing the squared error relative to the data. Another non-parametric approach involves use of edge preserving penalized optimization such as total variation minimization [4]. Denison *et al.* [5] use a Markov chain Monte Carlo approach to fit piecewise polynomials with different numbers and locations of knot points. Tishler and Frey [6] discuss a maximum likelihood approach to fit a convex piecewise linear function expressed as a point-wise maximum of a collection of affine functions with unknown coefficients. Maximum likelihood estimates are obtained by running a constrained optimization routine for a smoothed approximation of a mean squared error (MSE) cost function to bypass non-differentiability issues. A similar data model coupled with data clustering heuristics is utilized in a more recent paper by Magnani and Boyd [7] on fitting convex piecewise linear functions.

The use of adaptive methods is an attractive way of handling the issue of unknown number of breakpoints. One of the first algorithms using this technique was proposed by Friedman [8] under the name "adaptive regression splines (ARES)." Recursive partitioning is used to obtain better partitions of the set of data points at each iteration. Either goodness of fit criteria or generalized cross validation is used to estimate the number of partitions. On similar lines, Kolaczyk and Nowak [9] apply the method of recursive dyadic partitioning and fit a smooth function in each partition using maximum likelihood estimation. A penalty term for the number of partitions is introduced to trade off model complexity and quality of fit. In recent work, Saucier and Audet [11] propose a different class of adaptively constructed basis functions that can capture the transition points in otherwise piecewise smooth functions.

In [15] Bai and Perron discuss the problem of detecting structural changes in data without requiring the estimated function to be piecewise linear or even continuous. Their related paper [16] discusses a dynamic programming approach to obtain a least sum of squares fit. The model order is determined by using the Akaike information criterion (AIC) [18] and they impose a minimum limit on the "run length" of each segment in the piecewise model. In contrast, the present paper proposes a dynamic program that generates optimum maximum a posteriori (MAP) estimates based on a stochastic finite state HMM.

In the signal processing literature, two kinds of paradigms have been applied to this problem — Bayesian estimation and pattern recognition approaches. Punskaya *et al.* [24] model the function using the number and locations of the breakpoints as free parameters with certain prior distributions. The posterior density of the parameters conditioned on the noisy data is estimated through Monte Carlo techniques. In response to this method, Fearnhead [25] proposes a direct method for estimating parameters of the same model without resorting to Monte Carlo simulations and exploiting a Markov property in the model that allows calculation of the probability of future data points conditioned on the most recent breakpoint location. In the present paper, a Markov structure is imposed on the underlying slope values that are chosen from a finite set and the MAP algorithm estimates these slopes at each data sample.

1.4. Organization and Notation

The rest of this paper is organized as follows. The problem statement is discussed further in Section 2. A computationally tractable algorithm that uses the principle of dynamic programming is presented in Section 3. The issue of automatic selection of model parameters from data is addressed in Section 4. The problem of choosing the right number of quantization levels is analyzed through simulations and MSE distortion bounds in Section 5. A series of diverse applications are presented in Section 6, followed by some closing remarks in Section 7.

The notation $v_{i:j}$ is hereafter used to denote a vector $(v_i, v_{i+1}, \ldots, v_j)$ and the model parameters are denoted by $\theta = (\sigma^2, p)$.

2. Problem Statement and Stochastic Formulation

A pictorial representation of the relationships between various random variables of the piecewise linear data model for this paper is shown in Fig. B.1. It can be seen that these relationships lead to an HMM with two hidden layers. Moreover, the cardinality of the state space of each of the random variables Z_n increases with n. The next two subsections show why standard inference and parameter learning algorithms for HMMs cannot be directly applied to this problem.

2.1. Inference

The probabilistic structure of the data generation process can be expressed using a conditional density function of the unknown function values conditioned on the observed data vector. Let $\mathsf{p}_{Z_{0:N}|X_{1:N},\theta}(Z_{0:N} = z_{0:N}|X_{1:N} = x_{1:N}, \theta)$ be the posterior probability density of the function values $z_{0:N}$ conditioned on the observed data points $x_{1:N}$. The goal of the inference problem is to unravel the most likely sequence of hidden states (slope values $s_{1:N}$) that produced the observed function values $x_{1:N}$. This can be posed as a MAP estimation problem where the posterior density of the unknown states conditioned on the observed data is maximized.

Proposition 1. *The states Z_n in the HMM described by (1) form a second-order Markov chain.*

Proof. Using (1), it is easy to switch between the random variable Z_n and S_n by invoking the recursive relationship $Z_n = Z_{n-1} + S_n$. The first two states are handled as special cases. Note that $\mathsf{p}(Z_0 = z_0) := 1$ for $z_0 \equiv 0$, and $\mathsf{p}(Z_1 = z_1|Z_0 = z_0) = \mathsf{p}(S_1 = z_1) = 1/M$ for $z_1 \in \mathcal{S}$ and 0 otherwise. Also,

$$\begin{aligned} \mathsf{p}(Z_2 = z_2|Z_1 = z_1, Z_0 = z_0) &= \mathsf{p}(S_1 + S_2 = z_2|S_1 = s_1) \\ &= \mathsf{p}(S_2 = z_2 - z_1|S_1 = z_1 - 0) \\ &= \mathsf{p}(S_2 = z_2 - z_1|S_1 = z_1 - z_0) \end{aligned}$$

which depends only on z_2, z_1 and z_0, since $Z_0 = 0$ with probability 1. In general, for $3 \leq n \leq N$,

$$\begin{aligned}
\mathsf{p}(Z_n = z_n | Z_{0:n-1} = z_{0:n-1}) &= \mathsf{p}(Z_{n-1} + S_n = z_n | Z_{0:n-1} = z_{0:n-1}) \\
&= \mathsf{p}(z_{n-1} + S_n = z_n | Z_{0:n-1} = z_{0:n-1}, Z_{n-1} - Z_{n-2} = z_{n-1} - z_{n-2}) \\
&= \mathsf{p}(S_n = z_n - z_{n-1} | Z_{0:n-1} = z_{0:n-1}, S_{n-1} = z_{n-1} - z_{n-2}) \\
&= \mathsf{p}(S_n = z_n - z_{n-1} | S_{n-1} = z_{n-1} - z_{n-2})
\end{aligned}$$

which only depends on z_n, z_{n-1} and z_{n-2}. Hence Z_n is a second-order Markov chain. The transition probabilities can be written explicitly as:

$$\mathsf{p}(Z_n = z_n | Z_{n-1} = z_{n-1}, Z_{n-2} = z_{n-2}) = \begin{cases} p & \text{for } z_n = 2z_{n-1} - z_{n-2} \\ 0 & \text{for } z_n < z_{n-1} \\ \dfrac{1-p}{M-1} & \text{otherwise} \end{cases} \tag{2}$$

for $3 \leq n \leq N$. □

The posterior density can be further simplified using Proposition 1 and Bayes' theorem as follows:

$$\begin{aligned}
\mathsf{p}(z_{0:N} | x_{1:N}, \theta) &= \frac{\mathsf{p}(x_{1:N} | z_{0:N}, \theta) \mathsf{p}(z_{0:N} | \theta)}{\mathsf{p}(x_{1:N} | \theta)} \\
&= \frac{\mathsf{p}(x_1 | z_1, \theta) \mathsf{p}(z_1 | z_0, \theta)}{\mathsf{p}(x_{1:N} | \theta)} \prod_{j=2}^{N} \left[\mathsf{p}(x_j | z_j, \theta) \cdot \mathsf{p}(z_j | z_{j-1}, z_{j-2}, \theta) \right]. \quad (3)
\end{aligned}$$

For MAP estimation, it convenient to work with the log posterior density derived using (2) and (3):

$$\begin{aligned}
\log \mathsf{p}(z_{0:N} | x_{1:N}, \theta) &= -\log \mathsf{p}(x_{1:N} | \theta) - \frac{N}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{n=1}^{N} (x_n - z_n)^2 \\
&\quad + \sum_{n=2}^{N} \log \left[p \, \chi_{\{0\}}(s_n - s_{n-1}) + \frac{1-p}{M-1} \chi^{c}_{\{0\}}(s_n - s_{n-1}) \right] (4)
\end{aligned}$$

where $\chi_A(t)$ is the indicator function for the set A. It is defined as $\chi_A(t) = 0$ when $t \notin A$ and $\chi_A(t) = 1$ when $t \in A$. Also, $\chi^c_A(t) := 1 - \chi_A(t)$.

In a standard HMM, this maximization problem is efficiently solved using a dynamic program called the Viterbi algorithm [12]. For the present model, the standard Viterbi approach cannot be applied because of the second-order transition structure coupled with the fact that the state space of the hidden state Z_n changes with n. Two algorithms to solve the MAP estimation problem for the present model are shown in Section 3, including a computationally tractable dynamic program discussed in Section 3.1.

[Figure 1 about here.]

2.2. Learning

Choosing the model parameters σ^2 and p is crucial to ensure a sensible fit to the noisy data and enable numerical evaluation of the log posterior probabilities. Unfortunately, these parameters that define the stochastic model are seldom known in advance. Although reasonable values can be guessed by manually preprocessing the data, an automatic method for choosing these parameter values is desirable. The Baum-Welch algorithm which is a special case of expectation-maximization (EM) algorithm is a standard method for parameter estimation [28, 29, 30] in HMMs. In general, it is difficult to prove that the EM procedure converges unless certain assumptions about the likelihood function are made, unimodality being a common assumption [33]. Other approaches include approximate EM iterations via Monte Carlo integration to approximate the expectation step [32].

For the present model, it is theoretically possible to perform iterative estimation of new parameter values θ from the old values θ' using EM:

$$\sigma^2 = \frac{\sum\limits_{n=1}^{N} \sum\limits_{s_{1:n} \in \mathcal{S}^n} \left(x_n - \sum_{j=1}^{n} s_j \right)^2 \mathsf{p}(x_{1:N}, s_{1:n} | \theta')}{N \mathsf{p}(x_{1:N} | \theta')}$$

and

$$p = \frac{\sum\limits_{n=2}^{N} \sum\limits_{i=j} \mathsf{p}(x_{1:N}, S_{n-1} = i, S_n = j | \theta')}{(N-1) \mathsf{p}(x_{1:N} | \theta')}.$$

A derivation can be found in Appendix A. Although it is easy to set up the EM iteration equations, observe that the inner summations in both equations run over slope sequences whose lengths grow exponentially in the cardinality of the set $\mathcal{S}$ making it intractable for larger values of M and N.

An alternating maximization scheme is presented in Section 4 as an alternative to EM. Although it has weaker theoretical properties than the standard EM algorithm, it gives good performance in practice, as will be seen from the applications of Section 6. Unlike the EM approach, which ascends the observed data likelihood function, the MAPSlope algorithm ascends the complete data likelihood function by alternately maximizing with respect to the slope sequence and the model parameters.

3. Maximum a Posteriori Estimation of Slope Values

The maximum a posteriori slope sequence can be obtained by maximizing (4) as a function of the slope sequence, which is equivalent to the following optimization problem:

$$\begin{aligned} s^*_{1:N} \quad = \quad & \arg\max_{s_{1:N}} \Bigg(-\frac{1}{2\sigma^2} \sum_{n=1}^{N} (x_n - z_n)^2 + \sum_{n=2}^{N} \log \Bigg[p \, \chi_{\{0\}}(s_n - s_{n-1}) \\ & \qquad + \frac{1-p}{M-1} \chi^c_{\{0\}}(s_n - s_{n-1}) \Bigg] \Bigg). \qquad (5) \end{aligned}$$

The term containing $\mathsf{p}(x_{1:N}|\theta)$ is dropped because it does not depend on $s_{1:N}$. Besides producing the MAP optimal solution, the objective function has certain intuitively appealing characteristics. The first summation on the right hand side of (5) is just the sum squared error, which must be minimized. The second summation acts like a penalty term that encourages longer runs of constant slope. In general, MAP estimation problems are non-trivial owing to the presence of multiple local maximizers and computational complexity associated with optimization of multi-variable functions. Fortunately, the present maximization problem can be handled efficiently by taking a piecemeal approach furnished by the dynamic programming principle [31]. The following algorithm differs from the standard forward-backward algorithm used with HMMs which only handles one level of hidden states.

3.1. Dynamic Program

[Figure 2 about here.]

The objective function in (5) is a sum of individual terms that depend only on z_n, s_n and s_{n-1}. One way of visualizing a maximization algorithm is using the trellis in Fig. B.2 where each depth n shows the possible values k that the state Z_n can realize. Each branch has an associated branch “reward” which corresponds to individual terms of the summation in (5). The peculiarity about this trellis is that each branch has a variable reward depending on the previous branch chosen along the optimizing path. A dynamic program called Fast-Trellis is shown in Fig. B.3. This algorithm works by storing the optimum path vector $\mathbf{\Pi}(n,k)$ and its associated reward $I(n,k)$ coming into each node k at every depth n of the trellis. The path that maximizes the partial sums in (5) is chosen as the optimal path for each node. For each $1 \leq n \leq N$, the path vectors $\mathbf{\Pi}(n,k)$ and cumulative rewards $I(n,k)$ are updated by appending new nodes to the optimal paths terminating at nodes at the previous depth $n-1$.

Note that the branch rewards referred to in the innermost loop of FastTrellis are composed of two terms: the first term is the negative squared error with respect to the data and the second term is an additional reward for maintaining the same slope value as the previous data sample. The final output of this algorithm contains the best path reaching the deepest level in the trellis and the value of the maximized sum shown in (5). Referring back to Proposition 1, it is also worth noting that the pseudocode shown in Fig. B.3 implicitly converts the second order Markov structure on the Z_n process into a first order Markov process by keeping track of a pair of values, viz., the recent slope value $s_n = z_n - z_{n-1}$ and the previous slope value $s_{n-1} = z_{n-1} - z_{n-2}$, when deciding the branch cost.

It is instructive to compare the computational complexity of this procedure *vis-á-vis* a standard Viterbi algorithm that operates on a constant height trellis. The basic unit of computation is assumed to consist of two floating point additions, one floating point multiply and one compare — these operations are needed for calculating the rewards for each branch in the trellis in Fig. B.2. The terms $\log p$ and $\log\left(\frac{1-p}{M-1}\right)$ can be precomputed, hence do not enter the complexity analysis. The height of the trellis at any depth n is $n(M-1)+1$. Therefore the worst case number of computations at depth n is of the order $\mathcal{O}(n^2M^2)$. Summing over a total trellis depth of N levels, the worst case computational complexity of FastTrellis is $\mathcal{O}(N^3M^2)$. Although this is worse than the $\mathcal{O}(NM^2)$ complexity [12] of a standard Viterbi algorithm with a constant trellis height, it is still an improvement over a brute force approach which will require searching over M^N slope value sequences.

[Figure 3 about here.]

3.2. Smooth Optimization

The indicator functions in (5) cause the objective function to be non-differentiable. This can be addressed by approximating an indicator using a narrow Gaussian pulse:

$$\chi_{\{0\}}(t) \approx \exp\left(-\frac{t^2}{\alpha^2}\right)$$

where α controls the width of the roll-off. As $\alpha \searrow 0$ a narrow spike of height 1 is obtained at $t = 0$ thus approximating the indicator function with increasing precision. After discarding terms that do not depend on $s_{1:N}$, the following approximation to the optimization problem in (5) is obtained:

$$\begin{aligned} s^*_{1:N} \quad = \quad & \underset{s_{1:N}}{\arg\max}\left(-\frac{1}{2\sigma^2}\sum_{n=1}^{N}\left(x_n - \sum_{j=1}^{n} s_j\right)^2\right. \\ & \left. + \sum_{n=2}^{N}\log\left[p\, e^{-\frac{(s_n - s_{n-1})^2}{\alpha^2}} + \frac{1-p}{M-1}\left(1 - e^{-\frac{(s_n - s_{n-1})^2}{\alpha^2}}\right)\right]\right). \qquad (6) \end{aligned}$$

This smooth approximation of the original problem can be solved using standard constrained optimization routines such as gradient descent [26, 27]. Although it does not guarantee that the slope values lie in the discrete set $\mathcal{S}$, this issue can be bypassed either by rounding the slopes to the nearest quantization bins, or by using integer-programming to solve the optimization problem. Algorithmic complexity of this method varies based on the optimization algorithm used. In general, for a gradient descent type approach that constructs a Hessian matrix of size $N \times N$, the worst case complexity of inverting the Hessian matrix is $\mathcal{O}(N^3)$. This is similar to the complexity of the dynamic programming method. It should, however, be noted that FastTrellis produces an exact solution, whereas there is no guarantee that the optimization method will converge to $s^*_{1:N} \in \mathcal{S}^N$.

4. Alternating Maximization Algorithm for Estimating Model Parameters

The following result provides a useful alternative re-estimation procedure that bypasses the computational issues with the full EM approach. Yet, it has a similar advantage as an EM algorithm in that it increases the likelihood at each iteration. Under an additional minor technical assumption, it can also be shown that the procedure converges.

Theorem 2. *Let $s_{1:N}$ be the current slope sequence estimate, $s^*_{1:N}$ be the new sequence estimate obtained by running* FastTrellis *with the current model parameter values $\theta = (\sigma^2, p)$. Let $\theta^* = ({\sigma^*}^2, p^*)$ be re-estimated from the new slope sequence estimate as:*

$$ {\sigma^*}^2 = \frac{1}{N} \sum_{n=1}^{N} \left(x_n - \sum_{j=1}^{n} s^*_j \right)^2 \tag{7} $$

and

$$ p^* = \frac{1}{N-1} \sum_{n=1}^{N-1} \chi_{\{0\}}(s^*_{n+1} - s^*_n). \tag{8} $$

Then the complete data likelihood function satisfies

$$ \mathsf{p}(x_{1:N}, s_{1:N}|\theta) \leq \mathsf{p}(x_{1:N}, s^*_{1:N}|\theta) \leq \mathsf{p}(x_{1:N}, s^*_{1:N}|\theta^*). $$

A similar idea for estimation of parameters in a generalized linear model can be found in a recent paper by Yen [17]. The re-estimation equations can be understood intuitively — the new estimate of σ^2 is just the sample variance of the residual signal after subtracting the current piecewise linear fit from the raw data; the new estimate of p is the relative frequency of occurrence of samples where slope remains unchanged in the current fit. A formal proof is presented in Appendix B.

This result suggests an alternating maximization algorithm [34] that iterates between the slope sequence and the unknown parameters. It guarantees that the complete data likelihood increases at each step through this iterative procedure. Note that this is different from the analysis of the EM algorithm [33, 28] where the observed data likelihood function values are non-decreasing. The alternating maximization steps can now be iterated until some termination criterion is met. In most practical examples this method provides a reasonable fit in a few iterations. The complete algorithm called MAPSlope is shown in Fig. B.4.

[Figure 4 about here.]

Assuming there is a lower bound on the noise variance, the following result relevant to the convergence of MAPSlope algorithm can be proved:

Corollary 3. *In addition to the hypotheses of Theorem 1, suppose there exists* $\delta > 0$ *such that* $\sigma \geq \delta$*. Then the sequence of likelihood function values obtained via alternating maximization iterations of (7) and (8) have a limit point.*

In order to make stronger claims about the properties of this limit point, further assumptions are required [35] that do not hold in this present scenario.

The re-estimation method may be applied when the smooth optimization method is used instead of MAPSlope, in which case the complete algorithm becomes a special case of alternating gradient ascent. However, the aforementioned convergence result does not apply because the slope sequence obtained using the smooth optimization method is only an approximate maximizer.

5. Mean Squared Error Optimality Analysis and Model Order Selection

The aim of this section is to propose an empirical method for selecting the right number of quantization levels M for the set of slope values. Simple arguments from source coding theory are used to obtain upper and lower bounds for the MSE performance of the algorithm. This method can be applied directly to the sequence of estimated slope values.

Assume that the slope values originate from a continuous amplitude Markov source with amplitude levels in the interval $[0, 1]$. The goal is to characterize the performance of a decoder that outputs discrete values from the set $\{0, \frac{1}{(M-1)}, \ldots, \frac{M-2}{M-1}, 1\}$ such that the MSE is minimized. With a slight abuse of notation, let S_n denote the true slope value at sample n and let $\widehat{S_n}$ be the slope estimated from noisy data. The MSE performance metric is defined as:

$$\mathsf{MSE}(M) = \frac{1}{N}\sum_{n=1}^{N}(\widehat{S_n} - S_n)^2 \tag{9}$$

where the dependence on M is due to the fact that the algorithm that generates the sequence $\widehat{S_n}$ depends on M. Other metrics such as difference between the actual and detected number of change points, and distance from the actual change points can be used in practice, but are harder to analyze theoretically.

Lower bound for MSE performance. Consider an omniscient decoder (oracle) that knows the exact slope values output by the source in advance. The MSE of this decoder can be used to obtain a lower bound on the MSE obtained using any decoder. The average error of this omniscient decoder is given by [36]:

$$\begin{aligned}
\varepsilon_{LB}(M) &= \int_{0}^{\frac{1}{2(M-1)}} x^2 dx + \sum_{n=1}^{M-1} \int_{\frac{2n-1}{2(M-1)}}^{\frac{2n+1}{2(M-1)}} \left(x - \frac{n}{M-1}\right)^2 dx + \int_{1-\frac{1}{2(M-1)}}^{1} (x-1)^2 dx \\
&= \frac{1}{24(M-1)^3} + \sum_{n=1}^{M-1} \frac{1}{12(M-1)^3} + \frac{1}{24(M-1)^3} \\
&= \frac{1}{12(M-1)^2}.
\end{aligned}$$

Upper bound for MSE performance. Consider an ignorant decoder that tries to minimize the MSE after discarding all input information. The MSE of this decoder provides an upper bound to the MSE performance. More "intelligent" decoders that do use the input information when deciding a slope value would have a lower MSE than this ignorant decoder. A natural strategy for the ignorant decoder is to assume that each input slope value is uniformly and randomly distributed over $[0, 1]$, and hence the best it can do is to announce a slope value that is nearest to $\frac{1}{2}$. When M is odd, the decoder has a bin at $\frac{1}{2}$ giving an MSE of $\frac{1}{12}$. If M is even, the nearest bin is always a distance of $\frac{1}{2(M-1)}$ away from $\frac{1}{2}$. So, the upper bound on the average error can be written as:

$$\begin{aligned}\varepsilon_{UB}(M) &= \int_0^1 \left(x - \frac{1}{2} - \frac{1}{2(M-1)}\right)^2 dx \\ &= \frac{1}{12} + \frac{1}{4(M-1)^2} \text{ when } M \text{ is odd,}\end{aligned}$$

and,

$$\varepsilon_{UB}(M) = \frac{1}{12} \text{ when } M \text{ is even.}$$

Fig. 5(a) and 5(b) show the MSE through simulation on 1000 randomly generated piecewise linear datasets each of length $N = 50$. The lower and upper bounds (LB and UB) derived above are also plotted. As one would expect, larger values of MSE are obtained for larger σ^2 values. Lower MSE values are obtained with larger values of p because the slope changes less often when p is closer to 1. Note that the true parameter values were provided as input to the FastTrellis routine during simulation.

[Figure 5 about here.]

Two empirical conclusions can be drawn from these simulation results. First, a value of M around 15–20 is sufficient to achieve the "flat regions" of the

$\mathsf{MSE}(M)$ curves over a reasonable range of p and σ^2 values. Using values of M larger than 20 does not give noticeable improvement in MSE. Secondly, it shows that the FastTrellis algorithm performs quite well when the underlying data is generated with p close to 1 and $\sigma^2 \leq 1$.

Other model order selection methods may be employed to choose M. The fit can be produced for a range of different values of M and the one that provides the smallest residual sum of squares can be selected. A classical parameter selection method such as leave-out-one cross-validation can also be applied. Use of information criteria such as AIC/BIC will require modification in the setup because the M slope values are already specified in the current model; they are not estimated as part of the function-fitting algorithm.

6. Applications

6.1. Presidential Approval Ratings

United States presidential job approval ratings have been published by various agencies over the years. These ratings are usually quoted as percentage values that are computed from the results of opinion polls administered to a sample of the country's population. An analysis of the approval ratings for President George W. Bush is presented by Ratkovic and Eng [22]. The result of applying the MAPSlope algorithm to a snippet from the same dataset is shown in Fig. B.6. A sudden transition in the dataset correlates with the 9/11 terrorist attack.

The raw data is assumed to be piecewise constant, which implies that the running-sum over this plot gives a piecewise linear curve. This noisy piecewise linear data can be interpreted as a "cumulative approval score" plotted as a function of time. For comparison, results from fitting a fourth order polynomial and a 3-point 1st order polynomial LOWESS filter [23] are also shown

Table 1: Numerical evaluation of MAPSlope for Bush's approval ratings

	Polynomial	Lowess	FastTrellis	Smooth Optimization
Squared residual	37.61	7.8	5.13	9.84
Average # breaks	-	-	4	3.63

Residual sum squared values are calculated with respect to the raw data. Note that the average number of breakpoints is similar for both the FastTrellis and smooth optimization methods but the residual is smaller with the former. The polynomial fit is of order 4, whereas the LOWESS smoother uses a sliding window of 3 samples.

in Fig. B.6. Observe that close to the breakpoint the absolute residual error from the MAPSlope algorithm is the lowest. Also, MAPSlope-FastTrellis detects a change point immediately following the 9/9–11 polling period. A similar result is obtained in [22] using a non-parametric segmented spline fitting method. Additional numerical evaluation is presented in Table 1. MAPSlope was run 1000 times with different initial guesses for the parameter values; σ^2 was drawn from $\mathcal{U}([0.05, 0.15])$ and p was drawn randomly from $\mathcal{U}([0.88, 0.93])$ A maximum limit of 6 alternating maximization iterations was used with $M = 15$ slope quantization levels. The smooth optimization algorithm was run with $\alpha = 10^{-3}$. Since there is no "ground truth" answer to the fitting problem in this application, a judgment is made using the residual squared error from the data. MAPSlope with the trellis dynamic program gives the lowest error as seen from Table 1.

[Figure 6 about here.]

6.2. Interest Rates

Markov switching models have been used in the past to unravel piecewise constant trends in interest rate datasets. Hamilton [14] uses a 2-state Markov chain to model an economy that switches between fast and slow growth cycles. As an extension, Garcia and Perron [13] use a 3-state Markov switching model to analyze inflation adjusted quarterly interest rate data for the United States between 1961 to 1986. Assuming that this data follows a first-order Markov

Table 2: Numerical evaluation of MAPSlope for the interest rate data

	Bai-Perron	FastTrellis	Smooth Optimization
MSE (Eq. (9))	0	0.64	1.59
Squared residual	4.32	4.74	5.37
Average # breaks	3	3	6.98

The MSE values are obtained according to (9) by using the Bai-Perron fit as the ground truth. Residual sum squared values are calculated with respect to the raw data. Note that the average number of breakpoints is quite high in the smooth optimization method because the quantization step often leads to a sequence of small jumps.

property, the MAPSlope algorithm provides an easy generalization while providing a tractable algorithm for a moderate number of quantized interest rate levels.

The result of applying the MAPSlope algorithm to the dataset presented in [16] is shown in Fig. B.7. As in [16], it is assumed that this data is piecewise constant, with occasional changes in interest rate. The raw data is integrated to obtain a cumulative interest rate plot as a function of time. This plot is assumed to be piecewise linear and MAP estimates of the slope values are obtained. Initial parameter values are guessed from data; p is chosen close to 0.97 with the anticipation of around three regimes in the raw data vector of length 100, whereas the initial σ^2 is estimated from the sample variance after appropriate detrending. The slope values are quantized to 15 levels (which gives a resolution of about 0.067 on the unit interval).

Fig. B.7 also shows results obtained from the Bai-Perron algorithm[1] for comparison. The output of MAPSlope-FastTrellis agrees quite well with the Bai-Perron method as seen from the plot of absolute difference. Additional numerical evaluation results are shown in Table 2. Both FastTrellis and smooth optimization algorithms were simulated 1000 times with different initial

[1] Code available online at
`http://rss.acs.unt.edu/Rdoc/library/strucchange/html/RealInt.html`,
accessed Wed, Jan 23, 2013.

guesses for (σ^2, p) with σ^2 drawn randomly from $\mathcal{U}([0.2, 2])$ and p drawn from $\mathcal{U}([0.90, 0.98])$. A maximum iteration limit of MAXIT = 6 was used. For the smooth optimization algorithm, a sequential quadratic program [26] was used, followed by quantization to the discrete set of slopes. An initial guess of 0.5 was used for all the N slopes. The smooth optimization algorithm was run with $\alpha = 10^{-3}$. It was seen that MAPSlope converged to a local stationary point $(\sigma^2, p) = (0.12, 0.97)$ in 3–4 iterations when using FastTrellis but did not converge to a stable parameter value when the smooth optimization method was used. This is because the latter is not guaranteed to produce the optimal slope sequence, causing the hypotheses of Theorem 1 to be violated.

[Figure 7 about here.]

6.3. Shear Wave Elastography

Shear wave elastography is a medical imaging modality which aims at reconstructing tissue stiffness by tracking the propagation of a transverse mechanical wave in a region of interest. Since this wave travels faster in a stiffer medium, wave velocity maps can be used to distinguish cancerous tumors from healthy tissue which is typically softer than the tumor. Shear waves can be set up and imaged in a variety of ways; an overview of various techniques can be found in [38, 39]. The present section focuses on the analysis of experimental data from ultrasound electrode vibration elastography (EVE) [19, 42]

Radiofrequency tumor ablation procedures make use of a radiofrequency electrode inserted into the tumor to ablate cancerous cells. In EVE, the RF electrode is vibrated using an external actuator to set up a shear wave pulse in the surrounding tissue. Snapshots of this wave pulse are obtained using an ultrasound scanner operating at sufficiently high frame rate. Pixel level displacements are estimated as a function of time using the ultrasound echo data [37], which is then used to estimate the time of arrival of the shear wave pulse as a function of the distance from the ablation electrode. A cross-sectional view of the experimental setup is shown in Fig. B.8.

[Figure 8 about here.]

The location of this wave pulse is tracked along lines of constant depth to obtain a time-of-arrival plot. The slope of this plot of arrival time versus spatial location gives an estimate of the "slowness" of the shear wave pulse. Slowness is defined as the reciprocal of the wave speed, a term which has been used in the seismology literature [40] and has also been previously used in shear wave elastography studies [41]. The slowness of this shear wave pulse is related to the stiffness of the medium and hence a pictorial map of the slowness estimates can be used to locate stiff regions that may not be easily distinguishable on a traditional grayscale ultrasound scan (B-mode scan).

[Figure 9 about here.]

The dataset is acquired on a gelatin based tissue-mimicking (TM) phantom that consists of regions with three different stiffnesses. Since this time-of-arrival data is quite noisy, direct differentiation to obtain slopes is useless. The shear wave pulse propagation can be modeled as having constant speed in each medium which abruptly changes when the wave crosses an interface. Therefore, each noisy data vector is amenable to being processed using the MAPSlope-FastTrellis piecewise linear fitting algorithm. The parameters used for producing these images were $M = 15$ with an initial guess of $\theta = (\sigma^2, p) = (1, 0.95)$. In order to speed up processing time, the alternating maximization method was applied only at 4 different depths. Only FastTrellis was used at the remaining depths using the most recent parameter values obtained from MAPSlope.

Results of the complete procedure are shown alongside a B-mode ultrasound image in Fig. B.9. Note that the three regions of different stiffnesses are visible in the B-mode scan in Fig. B.9(a); this is done on purpose by altering the acoustic echogenicity of the TM material used. As mentioned previously, these stiffness variations are not easily visible in B-mode scans of real tissue. The slowness map of a shear wave pulse tracked in the same imaging plane is shown in Fig. B.9(b). There is good correlation between the boundary of the stiff ellipsoidal region The small irregular area on the left of the ellipsoid which has an intermediate stiffness is also visible in the slowness map. Similar visualization is possible through the shear wave velocity map in Fig. B.9(c). Additionally, a LOWESS filtered image is shown in Fig. B.9(d). This is similar to the LOWESS filter used in the application in Section 6.1, but uses a local quadratic model (instead of linear) with a sliding window of 15 samples. This kind of "least-squares smoothed" slope estimation is common in shear wave imaging literature [42]. It can be seen that the boundary details are sharper when the MAPSlope algorithm is used.

Numerical results obtained with three regions of interest (ROI), each of size 1 cm × 2 cm, fixed in the slowness and shear wave velocity maps are shown in Table 3. Standard image quality assessment metrics from the ultrasound elastography literature are used for this study [43]; Table 4 shows these evaluation metrics. For each region, the signal to noise ratio (SNR) is defined as:

$$SNR = \frac{\mu}{\sigma}$$

where μ and σ respectively denote the mean and the standard deviation values calculated over the ROI. The contrast (C) between a pair of regions is defined as:

$$C = \frac{\mu_1}{\mu_2}$$

where the subscripts indicate the two different ROIs. Similarly, the contrast to

Table 3: Slowness and Stiffness estimates

	Stiff	Intermediate	Soft
Slowness (s/m)	0.356 ± 0.1	0.537 ± 0.11	0.859 ± 0.18
Velocity (m/s)	3.09 ± 0.9	2.03 ± 0.35	1.21 ± 0.29
SNR (velocity)	12.6 ± 3.7	17.6 ± 3.4	12.7 ± 1.9
E (kPa)	30.4 ± 22	11.5 ± 3.8	4.86 ± 2.9

Values of shear wave slowness, shear wave velocity and Young's modulus for the three different regions in the experimental phantom obtained from the MAPSlope algorithm are indicated.

Table 4: Image quality metrics

	Stiff/Inter.	Inter./Soft	Stiff/Soft
C	3.6 ± 0.77	4.57 ± 0.72	8.17 ± 0.45
CNR	10.3 ± 6.4	18.1 ± 4	21.1 ± 6.6

Contrast (C), signal-to-noise ratios (SNR) and contrast-to-noise ratios (CNR) (in dB) obtained from shear wave velocity estimates for three pairs of regions are shown. See text for definitions of these quantities. The standard deviations are calculated from the dB values obtained over each ROI from individual datasets.

noise ratio (CNR) is defined as [44]:

$$CNR = \frac{2(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}.$$

Observe from Table 3 that the SNR values calculated from the shear wave velocity maps are around 10 dB in all three regions of the phantom, which is quite high given the noise levels of raw ultrasound echo data. It is worth noting in Table 4 that the best contrast and contrast-to-noise ratios are obtained for the pair of regions that differ the most in shear stiffness.

7. Conclusion

This paper presented the MAPSlope method for estimating slopes from noisy piecewise linear data which applies techniques from Bayesian estimation. An alternating maximization routine was presented for estimating unknown model parameters, and its convergence properties were analyzed theoretically. Further, MSE performance of this algorithm was tested using simulated data for different parameter values. The MSE performance was compared with suitable upper and lower bounds, and it was shown that the error was only slightly worse than an oracle lower bound. Experimental evaluation was carried out on three different datasets from political science, finance and medical imaging, demonstrating the practical significance of this algorithm. A major strength of the MAPSlope approach is that it directly estimates the slopes and breakpoints (rather than inferring them from curve fits) and so more directly optimizes the parameters that may be of physical interest.

Appendix A. Derivation of EM Iterations

The complete data likelihood function is given by

$$\begin{aligned} \mathsf{p}(x_{1:N}, s_{1:N}|\theta) &= \prod_{i=1}^{N} \mathsf{p}(x_i|s_{1:N},\theta) \prod_{k=2}^{N} \mathsf{p}(s_k|s_{k-1},\theta) \\ &= \prod_{i} f\left(x_i; \sum_{j=1}^{i} s_j, \sigma^2\right) \prod_{k} \left[p\, \chi_{\{0\}}(s_k - s_{k-1}) + \frac{1-p}{M-1}\, \chi^c_{\{0\}}(s_k - s_{k-1})\right] \end{aligned}$$

where $f(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$ is the univariate Gaussian probability density function parametrized by its mean and variance.

Next, the auxiliary function for EM iterations is constructed as follows:

$$\begin{aligned}
Q(\theta,\theta') &= \sum_{s_{1:N}\in\mathcal{S}^N} [\log \mathsf{p}(x_{1:N}, s_{1:N}|\theta)]\ \mathsf{p}(x_{1:N}, s_{1:N}|\theta') \\
&= \sum_{s_{1:N}\in\mathcal{S}^N} \sum_i \log f\left(x_i, \sum_{j=1}^{i} s_j, \sigma^2\right) \mathsf{p}(x_{1:N}, s_{1:N}|\theta') \\
&\quad + \sum_{s_{1:N}\in\mathcal{S}^N} \sum_k \log\left[p\ \chi_{\{0\}}(s_k - s_{k-1}) + \frac{1-p}{M-1}\ \chi^c_{\{0\}}(s_k - s_{k-1})\right] \\
&\qquad \cdot \mathsf{p}(x_{1:N}, s_{1:N}|\theta') \qquad \text{(A.1)} \\
&=: T_1 + T_2
\end{aligned}$$

where $\mathcal{S}^N$ is the set of all valid slope value sequences of length N, and the primes denote old values of the parameters from the previous iteration.

The two terms can now be independently optimized due to decoupling of parameters σ^2 and p.

Estimating σ^2*.* Setting the gradient of T_1 with respect to σ^2 to zero yields

$$\sigma^2 = \frac{\sum\limits_{n=1}^{N} \sum\limits_{s_{1:n}\in\mathcal{S}^n} \left(x_n - \sum_{j=1}^{n} s_j\right)^2\ \mathsf{p}(x_{1:N}, s_{1:n}|\theta')}{N\mathsf{p}(x_{1:N}|\theta')}$$

The constraint $\sigma^2 > 0$ is automatically met since all the terms in the expression on the right hand side are positive.

Estimating p*.* Setting the gradient of T_2 with respect to p to zero yields

$$p = \frac{\sum\limits_{k=2}^{N} \sum\limits_{i=j} \mathsf{p}(x_{1:N}, S_{k-1} = i, S_k = j|\theta')}{(N-1)\mathsf{p}(x_{1:N}|\theta')}.$$

The constraint $0 < p < 1$ is automatically met since all the quantities in the expression on the right hand side of this equation are positive and

$$\sum_{i=j} \mathsf{p}(x_{1:N}, S_{k-1} = i, S_k = j|\theta') \leq \sum_{i,j} \mathsf{p}(x_{1:N}, S_{k-1} = i, S_k = j|\theta') = \mathsf{p}(x_{1:N}|\theta').$$

Appendix B. Convergence of Alternating Maximization

Appendix B.1. Proof of Theorem 2

It suffices to prove the two inequalities for $\log \mathsf{p}(x_{1:N}, s_{1:N}|\theta)$ because log is monotonic increasing. The first inequality follows from the fact that Fast-Trellis solves the maximization problem

$$s^*_{1:N} = \arg\max_{s_{1:N}} \log \mathsf{p}(x_{1:N}, s_{1:N}|\theta).$$

The second inequality can be obtained maximizing $\log \mathsf{p}(x_{1:N}, s^*_{1:N}|\theta)$ as a function of θ. Setting the derivative with respect to σ to zero yields,

$$-\frac{N}{\sigma} + \frac{1}{\sigma^3} \sum_{i=1}^{N} \left(x_i - \sum_{j=1}^{i} s^*_j \right)^2 = 0$$

which gives

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^{N} \left(x_i - \sum_{j=1}^{i} s^*_j \right)^2 .$$

Also note that the second derivative with respect to σ is negative implying that this is in fact a maximum.

Next, setting the derivative with respect to p to zero yields,

$$\sum_{\substack{k:\, s_k = s_{k-1} \\ 2 \leq k \leq N}} \frac{1}{p} = \sum_{\substack{k:\, s_k \neq s_{k-1} \\ 2 \leq k \leq N}} \frac{1}{1-p}$$

which gives

$$p = \frac{1}{N-1}\sum_{k=2}^{N}\chi_{\{0\}}(s_k^* - s_{k-1}^*).$$

Again, note that the second derivative with respect to p is negative which implies that this is a maximizer. □

Appendix B.2. Proof of Corollary 3

Assuming there exists $\delta > 0$ such that $\sigma \geq \delta$, the following upper bound is obtained:

$$\log \mathsf{p}(x_{1:N}, s_{1:N}|\theta) \leq -\frac{N}{2}\log(2\pi\delta^2) + (N-1)\max\left(\log p, \log\frac{1-p}{M-1}\right)$$

for all $s_{1:N}$ and θ. Since every bounded non-decreasing sequence has a limit point by the monotone convergence theorem [45, Theorem 3.14], it follows from Theorem 1 that the alternating maximization iterations must converge. □

References

[1] D. Hudson, Fitting Segmented Curves Whose Join Points Have to be Estimated, J. Amer. Stat. Asso., 61 (316) (1966), 1097–1129.

[2] R. Bellman, R. Roth, Curve Fitting by Segmented Straight Lines, J. Amer. Stat. Asso., 64 (327) (1969), 1079–1084.

[3] A. Gallant, W. Fuller, Fitting Segmented Polynomial Regression Models Whose Join Points have to be Estimated, J. Amer. Stat. Asso., 68 (341) (1973), 144–147.

[4] A. Gholami, S. M. Hosseini, A balanced combination of Tikhonov and total variation regularizations for reconstruction of piecewise-smooth signals, Signal Processing, 93 (7) (2013), 1945–1960.

[5] D. Denison, B. Mallick, A. Smith. Automatic Bayesian curve fitting, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 60 (2) (1998), 333–350.

[6] A. Tishler, I. Zang, A New Maximum Likelihood Algorithm for Piecewise Regression, J. Amer. Stat. Asso., 76 (376) (December 1981), 980–987.

[7] A. Magnani, S. Boyd, Convex piecewise-linear fitting, Optim. Eng., 10 (1) (March 2009), 1–17.

[8] J. Friedman, Multivariate Adaptive Regression Splines, Ann. of Stats., 19 (1) (March 1991), 1–67.

[9] E. Kolaczyk, R. Nowak, Multiscale Generalized Linear Models for Nonparametric Function Estimation, Biometrica, 92 (1) (March 2005), 119–133.

[10] A. Savitzky , M. J. E. Golay, Smoothing and Differentiation of Data by Simplified Least Squares Procedures, Analy. Chem., 36 (8), 1627–1639 (July 1964).

[11] A. Saucier, C. Audet, Construction of sparse signal representations with adaptive multiscale orthogonal bases, Signal Processing, 92 (6) (June 2012), 1446–1457.

[12] L. Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Proceedings of the IEEE, 77 (2) (February 1989), 257–286.

[13] R. Garica, P. Perron, An Analysis of the Real Interest Rate Under Regime Shifts, The Review of Economics and Statistics, 78 (1) (February 1996), 111–125.

[14] J. Hamilton, A New Approach to the Economic Analysis of Nonstationary Time Series and the Business Cycle, Econometrica, 57 (2) (March 1989), 357–384.

[15] J. Bai, P. Perron, Estimating and Testing Linear Models with Multiple Structural Changes, Econometrica, 66 (1) (January 1998), 47–78.

[16] J. Bai, P. Perron, Computation and Analysis of Multiple Structural Change Models, J. Appl. Econ., 18 (1) (January 2003), 1–22.

[17] T.-J. Yen, A Majorization-Minimization Approach to Variable Selection using Spike and Slab Priors, Ann. of Stat., 39 (3) (June 2011), 1748–1755.

[18] H. Akaike, A New Look at the Statistical Model Identification, IEEE Trans. Autom. Control, 16 (6) (December 1974), 716–723.

[19] S. Bharat, T. Varghese, Radiofrequency electrode vibration-induced shear wave imaging for tissue modulus estimation: a simulation study, J. Acoust. Soc. Am., 128 (4) (October 2010), 1582–1585.

[20] S. Ryan, L. Porth, A Tutorial on the Piecewise Regression Approach Applied to Bedload Transport Data, U.S. Dept. of Agriculture Forest Service, Rocky Mountain Research Station, Tech. Rep. RMRS-GTR-189, May 2007.

[21] L. Wasserman, Nonparametric Regression, in: All of Nonparametric Statistics (Springer Texts in Statistics), 1st ed., New York: Springer-Verlag, 2006, Ch. 5, Sec. 5.3, pp. 68–71.

[22] M. Ratkovic, K. Eng, Finding Jumps in Otherwise Smooth Curves: Identifying Critical Events in Political Processes, Political Analysis, 18 (1) (January 2010), 57–77.

[23] W. Cleveland, Robust Locally Weighted Regression and Smoothing Scatterplots, Journal of the American Statistical Association 74 (368) (December 1979), 829–836.

[24] E. Punskaya, C. Andrieu, A. Doucet, W. Fitzgerald, Bayesian Curve Fitting Using MCMC With Applications to Signal Segmentation, IEEE Trans. Sig. Proc., 50 (3) (March 2002), 747–758.

[25] P. Fearnhead, Exact Bayesian Curve Fitting and Signal Segmentation, IEEE Trans. Sig. Proc. 53 (6) (June 2005), 2160–2166.

[26] S. Boyd, L. Vandenberghe, Convex Optimization Problems, in: Convex Optimization, 1st ed., Cambridge, UK: Cambridge University Press, 2004.

[27] J. Nocedal and S. Wright, Numerical Optimization, 2nd ed., New York: Springer, 2006.

[28] A. Dempster, N. Laird, D. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. Royal Stat. Soc., 39 (1977).

[29] W. Qian, D. Titterington, Estimation of parameters in hidden Markov models, Philos. Trans. Roy. Soc. London Ser. A, 337 (1991), 407–428.

[30] G. Archer, D. Titterington, "Parameter estimation for hidden Markov chains," J. Stat. Planning and Inference, 108 (2002), 365–390.

[31] R. Bellman, Dynamic Programming, Mineola, NY: Dover, 2003.

[32] G. Goodwin, J. Agüero, Approximate EM Algorithms for Parameter and State Estimation in Nonlinear Stochastic Models, in Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005, Seville, Spain, December 12–15, 2005.

[33] C. F. J. Wu, On the Convergence Properties of the EM Algorithm, Ann. of Stat., 11 (1) (March 1983), 95–103.

[34] I. Csiszar, G. Tusnady, Information geometry and alternating minimization procedures, Statistics and Decisions, Suppl. 1 (1984), 205–237.

[35] R. Yeung, The Blahut-Arimoto Algorithms, in: Information Theory and Network Coding, 1st Ed., Springer, 2008, Ch. 9, Sec. 9.3, pp. 222–226.

[36] J. Bucklew, Two results on asymptotic performance of quantizers, IEEE Trans. Info. Theory, 30 (2) (March 1984).

[37] M. Bilgen, M. Insana, Covariance analysis of time delay estimates for strained signals, IEEE Trans. on Sig. Proc., 46 (10) (October 1998), 2589–2600.

[38] L. Gao, K. Parker, R. Lerner, S. Levinson, Imaging of the elastic properties of tissue—A review, Ultrasound in Medicine and Biology, 22 (8) (May 1996), 959–977.

[39] K. Nightingale, S. McAleavey, G. Trahey, Shear-wave generation using acoustic radiation force: in vivo and ex vivo results, Ultrasound in Medicine and Biology, 28 (12) (December 2003), 1715–1723.

[40] S. Stein, M. Wysession, Basic Seismological Theory, in: An Introduction to Seismology, Earthquates, and Earth Structure, 1st ed., Malden, MA: Blackwell, 2003, Ch. 2, Sec. 2.5.7, pp. 69.

[41] R. DeWall, Shear wave velocity imaging using transient needle vibration for monitoring thermal ablative therapies, Ph.D. dissertation, Dept. of Biomed. Eng., Univ. of Wisc., Madison, WI, 2011.

[42] R. DeWall, T. Varghese, Shear wave velocity imaging using transient electrode perturbation: phantom and ex vivo validation, IEEE Trans. Med. Imag., vol. 30, no. 3, pp. 666–678, Mar. 2011.

[43] S. Bharat, T. Varghese, E. Madsen, J. Zagzebski Radio-frequency ablation electrode displacement elastography: a phantom study, Medical Physics, 35 (6) (June 2008), 2432–2442.

[44] T. Varghese, J. Ophir, An analysis of elastographic contrast to noise ratio, Ultrasound in Medicine and Biology, 24 (6) (July 1998), 915–924.

[45] W. Rudin, Numerical Sequences and Series, in: Principles of Mathematical Analysis, 3rd ed., New York, NY: McGraw Hill, 1976, Ch. 3, pp. 55.

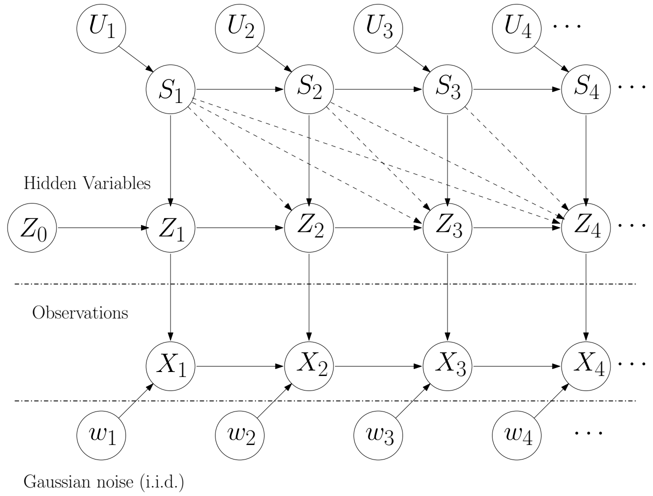


Figure B.1: A pictorial representation of different random variables involved in the Markov model for piecewise linear data. Solid arrows indicate conditional dependence. A dotted arrow indicates redundant dependency link. For instance, Z_4 depends on S_1, S_2, S_3 through Z_3.

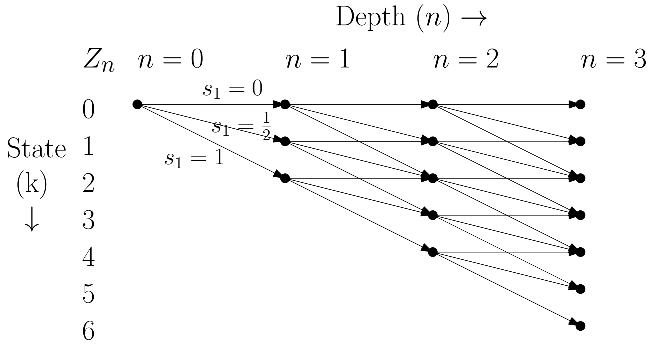


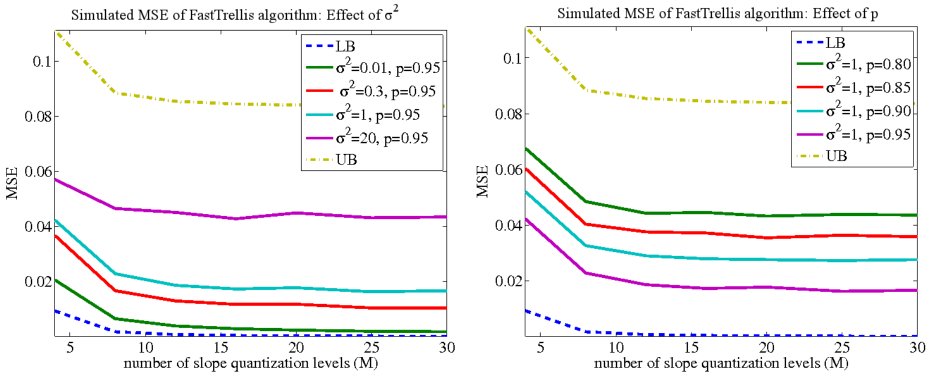
Figure B.2: A representative diagram of a trellis structure used for finding the MAP optimal slope sequence. This trellis is shown with slope values quantized into three bins and a data vector length of 4.

```
Input: $x_{1:N}$: noisy data vector of length $N$
       $M$: number of slope quantization levels
       $\sigma^2$: Gaussian noise variance
       $p$: probability of staying in the same slope value
Output: $s^*_{1:N}$: MAP slope sequence
       $L^*$: log-likelihood for MAP slope sequence
procedure FastTrellis($x_{1:N}, M, \sigma^2, p$)
    $I(0,0) \leftarrow 0$
    for $n = 1$ to $N$ do
        for $k = 0$ to $(M-1)n$ do
            if $n == 1$ then
                $I(1,k) \leftarrow -\frac{1}{2\sigma^2}\left(x_1 - \frac{k}{M-1}\right)^2 + \log \frac{1}{M}$
                $\mathbf{\Pi}(1,k) \leftarrow [0,k]$.
            else
                $I(n,k) \leftarrow \max_j [I(n-1,j) + \text{branch reward of } (n-1,j) \text{ to } (n,k)]$
                $\widehat{j} \leftarrow \arg\max_j [I(n-1,j) + \text{branch reward of } (n-1,j) \text{ to } (n,k)]$
                $\mathbf{\Pi}(n,k) \leftarrow$ Append($\mathbf{\Pi}(n-1,\widehat{j}), k$)
            end if
        end for
    end for
    $L^* \leftarrow \max_k I(N,k)$
    $s^*_{1:N} \leftarrow \mathbf{\Pi}(N, \arg\max_k I(N,k))$
    return $(s^*_{1:N}, L^*)$
end procedure
```

Figure B.3: Fast dynamic program for searching an optimal route through a linearly growing trellis (like the one shown in Fig. B.2).

```
Input: $x_{1:N}$: noisy data vector of length $N$
       $M$: number of slope quantization levels
       $\sigma^2$: initial guess for the Gaussian noise variance
       $p$: guess probability of staying in the same slope value
       $\tau$: threshold for likelihood value convergence test
       MAXIT: maximum number of iteration
Output: $s^*_{1:N}$: MAP slope sequence
       ${\sigma^*}^2$: estimated noise variance
       $p^*$: estimated value of $p$
procedure MAPSlope($x_{1:N}, M, \sigma^2, p, \tau$, MAXIT)
    loop $\leftarrow 1$
    $L \leftarrow -\infty$
    CONVERGED $\leftarrow$ $False$
    repeat
        if using FastTrellis then
            $(s^*_{1:N}, L^*) \leftarrow$ FastTrellis$(x_{1:N}, M, \sigma^2, p)$
        else if using another optimization routine then
            $s^*_{1:N} \leftarrow \arg\max_{s_{1:N}} \log \mathsf{p}(x_{1:N}, s_{1:N} | \sigma^2, p)$
            $L^* \leftarrow \log \mathsf{p}(x_{1:N}, s^*_{1:N} | \sigma^2, p)$
        end if
        ${\sigma^*}^2 \leftarrow \frac{1}{N} \sum_{n=1}^{N} \left( x_n - \sum_{j=1}^{i} s^*_j \right)^2$
        $p^* \leftarrow \frac{1}{N-1} \sum_{n=1}^{N-1} \chi_{\{0\}}(s^*_{n+1} - s^*_n)$
        loop $\leftarrow$ loop $+ 1$
        if $(|L^* - L| < \tau) \vee (\text{loop} > \text{MAXIT})$ then
            CONVERGED $\leftarrow$ $True$
        end if
        $p \leftarrow p^*$, $\sigma^2 \leftarrow {\sigma^*}^2$, $L \leftarrow L^*$
    until $\neg$CONVERGED
    return $(s^*_{1:N}, {\sigma^*}^2, p^*)$
end procedure
```

Figure B.4: MAPSlope algorithm for maximum a posteriori slope estimation in piecewise linear functions with alternating maximization parameter estimation.



(a) Varying noise variance

(b) Varying jump probability

Figure B.5: MSE of the FastTrellis algorithm on randomly generated simulated data with (a) different noise variances and (b) slope change probabilities.

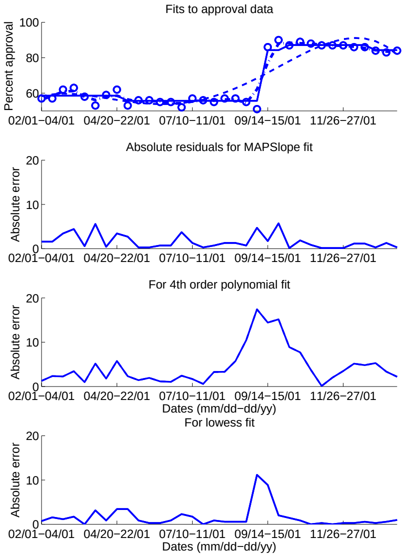


Figure B.6: Fits to Presidential approval data for President George W. Bush around 9/11. Absolute residuals obtained from MAPSlope-FastTrellis (solid line), a 4th order polynomial fit (dashed line) and a 3-point 1st order LOWESS fit (dash-dotted line) are also shown. Observe that the 4th order polynomial fit has a large error at the breakpoint. The 3-point LOWESS smoother performs slightly better than the näive polynomial fit, but still fails to capture the sharp jump.

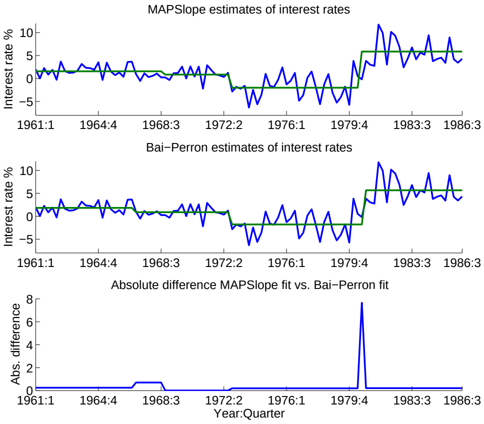


Figure B.7: Piecewise constant estimates of quarterly interest rates, data obtained from the paper by Bai and Perron [16]. The MAPSlope-FastTrellis fit was obtained using $(\sigma^2, p) = (0.12, 0.97)$ which was found to be a local convergence point of the alternating maximization scheme.

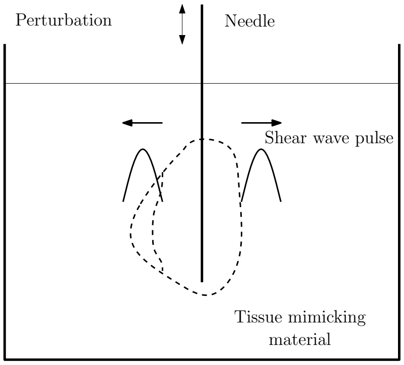
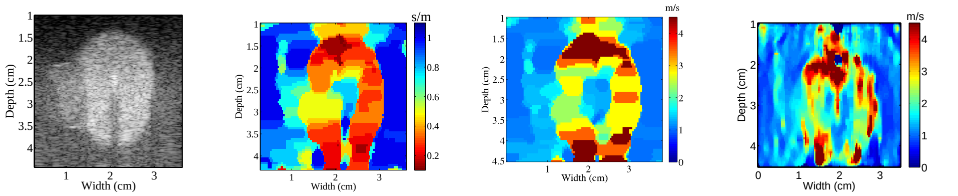


Figure B.8: Cross section view of the experimental setup showing a needle that sets up a shear wave pulse in the underlying tissue mimicking gelatin material.



(a) B-mode (b) Slowness (c) Velocity (d) Velocity (LOWESS)

Figure B.9: Results from shear wave tracking using the MAPSlope algorithm. (a) Ultrasound B-mode image of the phantom, (b) slowness map generated by applying the MAPSlope algorithm to denoise the arrival time data and obtain MAP slope estimates. Wave velocities are shown in (c) by calculating the reciprocal of the slowness values. The velocity values estimated using a 2nd order LOWESS filter with a 15 sample sliding window is are shown in (d).