

Inference of Heartbeats in Cardiography Signals using Variable Rate Particle Filtering

Pete Bunch

December 2012

1 Introduction

Sometimes detecting heartbeats in a BCG or ECG signal is easy — one can simply pick the largest peaks at roughly 1 second intervals. Smoothing or differentiating may be required. At other times, more sophisticated algorithms may be required, because of low signal-to-noise ratios and inconsistency between heartbeats. Here, a particle filtering approach is described.

2 Preprocessing

Our BCG signals are corrupted by 50Hz mains interference, as well as a DC bias which occasionally makes large jumps. To remove these effects, the discrete-time signal is bandpass filtered between 0.2Hz and 15Hz with a 300 tap symmetric, non-causal FIR filter, and down-sampled to 30Hz. This reduces the processing burden with minimal loss of useful information.

3 Model

We model the cardiography signal as a sequence of similar, irregularly-spaced heartbeat waveforms in noise, using a variable rate framework. The heartbeat start times are represented by a sequence of changepoints, $\{\tau_k\}$, and the corresponding waveforms by a the vector sequence, $\{\mathbf{w}_k\}$. These vectors are sampled at the same rate as the signal. The sampled signal is composed of measurements, $\{y_n\}$, at times $\{t_n\}$ spaced by sampling period T_s .

We will need some useful index variables, $K(t) = \max\{k : \tau_k < t\}$, $K_n = K(t_n)$, which reference the most recent changepoint.

The signal is modelled as,

$$s_n = s(t_n) = \mathbf{h}(t, \tau_{K_n})^T \mathbf{w}, \quad (1)$$

$\mathbf{h}(t, \tau_{K_n})$ is a vector of interpolation coefficients. Because of the anti-aliasing filter, we can achieve perfect interpolation using,

$$\mathbf{h}(t, \tau_{K_n})_i = \text{sinc}\left(\frac{t - \tau_{K_n} - iT_s}{T_s}\right). \quad (2)$$

However, it might be computationally pragmatic to use something more sparse. Maybe some cubic-splines.

The waveform \mathbf{w}_k is expected to change only slowly over time. We use a Gaussian transition density with a tight covariance matrix,

$$p(\mathbf{w}_k | \mathbf{w}_{k-1}) = \mathcal{N}(\mathbf{w}_k | \mathbf{w}_{k-1}, Q_w). \quad (3)$$

We need a model for the evolution of τ_k . Examining the intervals between successive heartbeats, there is considerable variability. Over a short time, there appears to be a hard lower bound for the beat periods, while the upward variability is much greater. On a longer time scale, this lower bound varies slowly. Based on these observations, we adopt a shifted inverse-gamma prior for the beat periods, where the shift is itself a random variable, denoted Δ_k . The evolution of Δ_k is modelled as a gamma random walk.

Finally, we assume a Gaussian observation model,

$$y_n = \mathcal{N}(y_n | s_n, \sigma_y^2). \quad (4)$$

3.1 Clutter

The data is intermittently interrupted by high amplitude noise, caused by vibrations, the subject rolling over, or electrical effects. We need to be able to detect and correct for such clutter. An indicator variable, λ_n is introduced for each observation to indicate the presence of clutter. The new observation model is,

$$y_n = \begin{cases} \mathcal{N}(y_n | s_n, \sigma_y^2) & \lambda_n = 0 \\ \mathcal{N}(y_n | 0, \sigma_C^2) & \lambda_n = 1 \end{cases}. \quad (5)$$

λ_n is assumed to follow a Markov process with fixed transition probabilities.

4 Inference Algorithm

With this model we can do inference using a variable rate particle filter. In addition, because of the linear-Gaussian model assumptions, we can Rao-Blackwellise the waveform variable, \mathbf{w}_k , which will give us a huge dimensionality reduction. Hooray! The resulting algorithm will be a similar to those of [Morelande and Gordon, 2009] and [Whiteley, Johanssen, Godsill, 2011].

We want our particle filter to estimate the changepoint times and also the parameters. Additional parameters could be included.

$$\mathbf{u}_k = [\Delta_k]. \quad (6)$$

As usual, we bundle all these into a single variable,

$$\theta_n = \{\tau_j, u_j \forall j : 0 \leq \tau_j < t_n\}. \quad (7)$$

4.1 Clutter-free

The target distribution is,

$$p(\theta_n | y_{1:n}). \quad (8)$$

A standard variable rate particle filter can now be applied, using bootstrap proposals. The likelihood is a little bit tricky,

$$\begin{aligned} p(y_n | \theta_n, y_{1:n-1}) &= \int p(y_n | \theta_n, \mathbf{w}_{K_n}) p(\mathbf{w}_{K_n} | \theta_n, y_{1:n-1}) d\mathbf{w}_{K_n} \\ &= \int p(y_n | s(t_n)) p(\mathbf{w}_{K_n} | \theta_n, y_{1:n-1}) d\mathbf{w}_{K_n} \\ &= \int \mathcal{N}(y_n | \mathbf{h}(t, \tau_{K_n})^T \mathbf{w}_{K_n}, \sigma_y^2) \mathcal{N}(\mathbf{w}_{K_n} | \mathbf{m}_{n-1}, \mathbf{P}_{n-1}) d\mathbf{w}_{K_n}. \end{aligned} \quad (9)$$

A Kalman filter is maintained for each particle to estimate the density over \mathbf{w}_{K_n} . If no changepoint occurs between t_{n-1} and t_n , then,

$$\begin{aligned} p(\mathbf{w}_{K_n} | \theta_n^-, y_{1:n}) &\propto p(y_n | \mathbf{w}_{K_n}, \theta_n^-) p(\mathbf{w}_{K_n} | \theta_n^-, y_{1:n-1}) \\ &= \mathcal{N}(y_n | \mathbf{h}(t, \tau_{K_n})^T \mathbf{w}_{K_n}, \sigma_y^2) \mathcal{N}(\mathbf{w}_{K_n} | \mathbf{m}_{n-1}, \mathbf{P}_{n-1}). \end{aligned} \quad (10)$$

Alternatively, if a changepoint does occur between t_{n-1} and t_n , then,

$$\begin{aligned} p(\mathbf{w}_{K_n} | \theta_n^-, y_{1:n}) &= p(y_n | \mathbf{w}_{K_n}, \theta_n^-) \int p(\mathbf{w}_{K_n} | \mathbf{w}_{K_n-1}) p(\mathbf{w}_{K_n-1} | \theta_n^-, y_{1:n-1}) d\mathbf{w}_{K_n-1} \\ &= \mathcal{N}(y_n | a_{K_n} b(t, \tau_{K_n})^T \mathbf{w}_{K_n}, \sigma_y^2) \int \mathcal{N}(\mathbf{w}_{K_n} | \mathbf{w}_{K_n-1}, Q_w) \mathcal{N}(\mathbf{w}_{K_n-1} | \mathbf{m}_{n-1}, \mathbf{P}_{n-1}) d\mathbf{w}_{K_n-1} \end{aligned}$$

So everything's Gaussian, and can be calculated in closed form. Jolly good.

4.2 With Clutter

The target distribution for the particle filter is now,

$$p(\theta_n, \lambda_{1:n} | y_{1:n}). \quad (12)$$

Note that if we weren't Rao-Blackwellising \mathbf{w} then we could calculate $p(\lambda_n | \theta_n, y_{1:n})$ in closed form, but since we are, we can't, because it would mess up the Kalman filters. We can however, propose exactly from $p(\lambda_n | \theta_n, y_{1:n})$.

4.3 Fixed Lag “Smoother”

The particle filter algorithm doesn’t work very well. This is because it takes many observations to observe a heartbeat, and often the right particles have been discarded before they are needed. The solution to this is to use a fixed-lag smoother. The target distribution is,

$$p(\theta_{n+L}, \lambda_{1:n+L} | y_{1:n+L}), \quad (13)$$

and we allow changes to $\theta_{n+L \setminus n}$ and $\lambda_{n+1:n+L}$. In addition, we can now move forward in batches rather than running a particle filter at every observation time. The batch size, S , will need to be smaller than the window size, L .

Such a fixed lag smoother uses the SMC sampler algorithm of [Doucet, Briers, Senecal, 2006], which makes it distinct from (although still pretty similar to) the method of [Whiteley, Johansen, Godsill, 2011].

```

for  $n = 0, S, 2S, \dots$  do
  for  $i = 1, \dots, N_F$  do
    Select a particle  $\{\theta_{n+L-S}, \lambda_{1:n+L-S}\}^{(i)}$  with probability  $w_{n-S}^{(i)}$ 
    Sample  $\theta'_{n+L \setminus n} \sim p(\cdot | \theta_n^{(i)})$ 
    for  $l = 1, \dots, L$  do
      Sample  $\lambda'_{n+l} \sim p(\cdot | \theta_{n+L}^{(i)}, \lambda_{n+l-1}^{(i)}, y_{1:n})$ 
    end for
     $w_n^{(i)} = p(y_{n+1:n+L} | \theta'_{n+L}, y_{1:n}) / p(y_{n+1:n+L-S} | \theta_{n+L-S}, y_{1:n})$ 
  end for
end for

```
