Laplace State Space Filter with Exact Inference and Moment Matching - Slides

Julian Neri[1] Roland Badeau[2] Philippe Depalle[1]

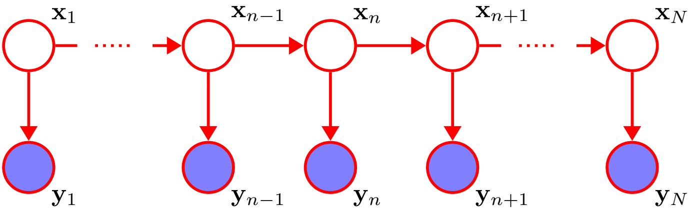
[1]McGill University, CIRMMT, Montréal, Canada
[2]LTCI, Télécom Paris, Institut Polytechnique de Paris, France

May 6, 2020

Overview

Introduction

State space models are probabilistic models for **sequential (time-series) data** that are used in many applications like target tracking, finance, audio processing, and neuroscience.



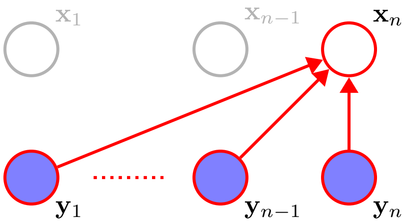
The joint distribution of a state space model is given by

$$p(\mathbf{Y}, \mathbf{X}) = \underbrace{p(\mathbf{x}_1)}_{\text{Prior}} \prod_{n=2}^{N} \underbrace{p(\mathbf{x}_n | \mathbf{x}_{n-1})}_{\text{Transition}} \prod_{n=1}^{N} \underbrace{p(\mathbf{y}_n | \mathbf{x}_n)}_{\text{Emission}}$$

Introduction - Filtering

Filtering infers the marginal posterior distribution of latent state $\mathbf{x}_n$ given all the data up to that point, $\mathbf{y}_{1:n} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$.

$$p(\mathbf{x}_n|\mathbf{y}_{1:n}) = \frac{p(\mathbf{y}_n|\mathbf{x}_n)p(\mathbf{x}_n|\mathbf{y}_{1:n-1})}{p(\mathbf{y}_n|\mathbf{y}_{1:n-1})}$$



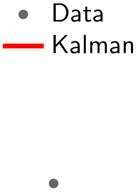
Introduction - Gaussian model

Gaussian emission, transition, and initial probabilities are commonly assumed to simplify state estimation.

- The **Kalman filter** is an efficient exact inference algorithm for state space models with **Gaussian state and data noise**.

✗ However, the Kalman filter's performance degrades when data noise is non-Gaussian. Examples:

- outliers, heavy-tailed noise, glint noise, & noise level changes



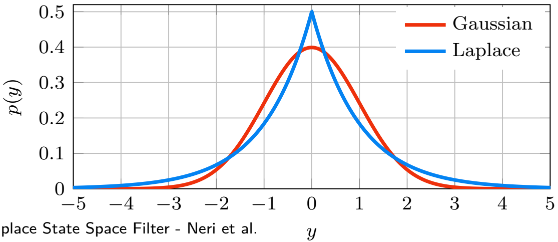
Laplace state space model

The **Laplace state space model** assumes the data is corrupted by Laplace-distributed noise with scale R.

$$p(y_n|\mathbf{x}_n) = \mathcal{L}(y_n|\mathbf{c}\mathbf{x}_n, R) = \frac{1}{2R}\exp\left(-\frac{|y_n - \mathbf{c}\mathbf{x}_n|}{R}\right)$$

- Heavy-tailed distributions represent outliers well.
- Assuming Laplace noise has proven beneficial for a variety of non-Gaussian noises.
- Exact inference is not tractable $\rightarrow$ approximate inference.

✗ Existing filters for the Laplace model (particle filtering and variational inference) are inefficient.



Filtering - Overview

Goal: Compute the marginal posterior of $\mathbf{x}_n$:

$$p(\mathbf{x}_n|y_{1:n}) = \frac{p(y_n|\mathbf{x}_n)p(\mathbf{x}_n|y_{1:n-1})}{p(y_n|y_{1:n-1})}$$

- Predictive distribution:
  $p(\mathbf{x}_n|y_{1:n-1}) = \int p(\mathbf{x}_n|\mathbf{x}_{n-1})p(\mathbf{x}_{n-1}|y_{1:n-1})d\mathbf{x}_n$
- Marginal likelihood:
  $p(y_n|y_{1:n-1}) = \int p(y_n|\mathbf{x}_n)p(\mathbf{x}_n|y_{1:n-1})d\mathbf{x}_n$

We can **analytically derive** the **locally-exact marginal posterior** for the Laplace model when the predictive distribution is Gaussian.

Condition is met by converting locally-exact p into a Gaussian:

$$\tilde{p}(\mathbf{x}_{n-1}|y_{1:n-1}) = \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_n, \mathbf{V}_n)$$

Filter - Predict

The predictive distribution is given by

$$
\begin{aligned}
p(\mathbf{x}_n|y_{1:n-1}) &= \int p(\mathbf{x}_n|\mathbf{x}_{n-1})\tilde{p}(\mathbf{x}_{n-1}|y_{1:n-1})d\mathbf{x}_{n-1} \\
&= \mathcal{N}(\mathbf{x}_n|\mathbf{m}_{n-1}, \mathbf{P}_{n-1})\,,
\end{aligned}
$$

where the predictive mean and covariance matrix are, respectively,

$$
\begin{aligned}
\mathbf{m}_{n-1} &= \mathbf{A}\boldsymbol{\mu}_{n-1}\,, \\
\mathbf{P}_{n-1} &= \mathbf{A}\mathbf{V}_{n-1}\mathbf{A}^{\mathsf{T}} + \mathbf{Q}\,.
\end{aligned}
$$

$\rightarrow$ Identical to Kalman filter prediction step!

Filter - Update

The marginal likelihood has an analytic, closed-form solution.

$$
\begin{aligned}
p(y_n|y_{1:n-1}) &= \int p(y_n|\mathbf{x}_n)p(\mathbf{x}_n|y_{1:n-1})d\mathbf{x}_n \\
&= \int \mathcal{L}(y_n|\mathbf{c}\mathbf{x}_n, R)\mathcal{N}(\mathbf{x}_n|\mathbf{m}_{n-1}, \mathbf{P}_{n-1})d\mathbf{x}_n \\
&= \frac{\Phi_n^{(-)} + \Phi_n^{(+)}}{4R} \exp\left(-\frac{\tilde{y}_n^2}{2S_n}\right),
\end{aligned}
$$

where we define

$$
\begin{aligned}
\widehat{y}_n &= \mathbf{c}\mathbf{m}_{n-1}\,, && \text{(filtered data)} \\
\tilde{y}_n &= y_n - \widehat{y}_n && \text{(residual)} \\
S_n &= \mathbf{c}\mathbf{P}_{n-1}\mathbf{c}^{\mathsf{T}}\,, && \text{(data variance)} \\
\Phi_n^{(\pm)} &= \mathsf{erfcx}\left(\frac{\sqrt{S_n}}{\sqrt{2R^2}} \pm \frac{\tilde{y}_n}{\sqrt{2S_n}}\right).
\end{aligned}
$$

Filter - Update

The marginal likelihood has an analytic, closed-form solution.

$$
\begin{aligned}
p(y_n|y_{1:n-1}) &= \int p(y_n|\mathbf{x}_n)p(\mathbf{x}_n|y_{1:n-1})d\mathbf{x}_n \\
&= \int \mathcal{L}(y_n|\mathbf{c}\mathbf{x}_n, R)\mathcal{N}(\mathbf{x}_n|\mathbf{m}_{n-1}, \mathbf{P}_{n-1})d\mathbf{x}_n \\
&= \frac{\Phi_n^{(-)} + \Phi_n^{(+)}}{4R} \exp\left(-\frac{\tilde{y}_n^2}{2S_n}\right),
\end{aligned}
$$

where we define

$$
\begin{aligned}
\widehat{y}_n &= \mathbf{c}\mathbf{m}_{n-1}\,, && \text{(filtered data)} \\
\tilde{y}_n &= y_n - \widehat{y}_n && \text{(residual)} \\
S_n &= \mathbf{c}\mathbf{P}_{n-1}\mathbf{c}^{\mathsf{T}}\,, && \text{(data variance)} \\
\Phi_n^{(\pm)} &= \mathsf{erfcx}\left(\frac{\sqrt{S_n}}{\sqrt{2R^2}} \pm \frac{\tilde{y}_n}{\sqrt{2S_n}}\right).
\end{aligned}
$$

Filter - Update

The locally-exact marginal posterior of state $\mathbf{x}_n$

$$p(\mathbf{x}_n | y_{1:n}) = \frac{p(y_n|\mathbf{x}_n)p(\mathbf{x}_n|y_{1:n-1})}{p(y_n|y_{1:n-1})}$$

has the following expected value and covariance

$$\begin{aligned} \mathbb{E}\left[\mathbf{x}_n\right] &= \mathbf{m}_{n-1} + \mathbf{k}_n \delta_n \, , \\ \mathsf{cov}[\mathbf{x}_n, \mathbf{x}_n] &= \mathbf{P}_{n-1} + \mathbf{k}_n \mathbf{k}_n^{\mathsf{T}} \Delta_n \, , \end{aligned}$$

where we define

$$\begin{aligned} \mathbf{k}_n &= \mathbf{P}_{n-1} \mathbf{c}^{\mathsf{T}} R^{-1} \, , \\ \delta_n &= \frac{\Phi_n^{(-)} - \Phi_n^{(+)}}{\Phi_n^{(-)} + \Phi_n^{(+)}} \, , \\ \Delta_n &= \frac{1}{\Phi_n^{(-)} + \Phi_n^{(+)}} \left(\frac{4\Phi_n^{(-)}\Phi_n^{(+)}}{\Phi_n^{(-)} + \Phi_n^{(+)}} - \sqrt{\frac{8R^2}{\pi S_n}} \right) \end{aligned}$$

Filter - Properties

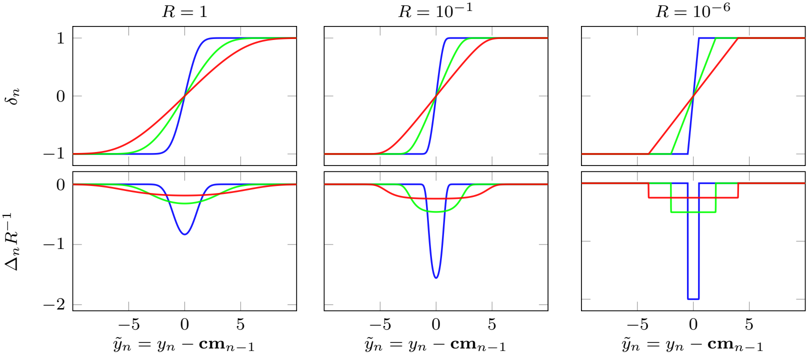
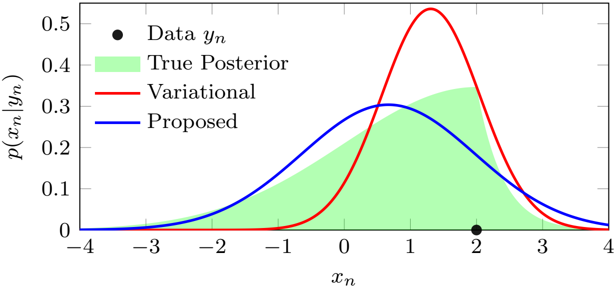


Figure: The blue, green, and red lines correspond to $\mathbf{P}_{n-1} = R/2$, $2R$, and $4R$, respectively, and $\mathbf{c} = 1$.

Filter - Properties

Our moment-matching Gaussian approximation is superior to variational inference with Gaussian scale mixtures.

- Variational inference is mode-seeking and inclusive, while moment matching is mean-seeking and exclusive.



Filter - Nonlinear dynamics

An elegant extension to models with nonlinear dynamics.
Linearize $h(.)$ around the previous state estimate $\boldsymbol{\mu}_{n-1}$,

$$\mathbf{m}_{n-1} = h(\boldsymbol{\mu}_{n-1})$$
$$\mathbf{A}_n = \frac{dh}{d\mathbf{x}_{n-1}}\Big|_{\boldsymbol{\mu}_{n-1}}$$

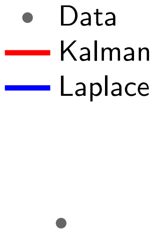
Linearize $g(.)$ around the predicted mean $\mathbf{m}_{n-1}$,

$$\widehat{y}_n = g(\mathbf{m}_{n-1})$$
$$\mathbf{c}_n = \frac{dg}{d\mathbf{x}_n}\Big|_{\mathbf{m}_{n-1}}$$

Also possible to use the unscented transform.

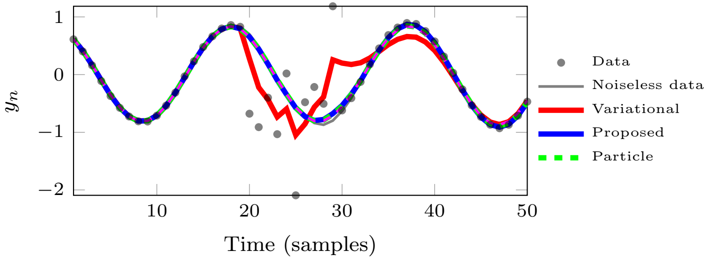
Results - Linear

The Laplace state space filter is robust to outliers, periods of increased noise, and other heavy-tailed noises.



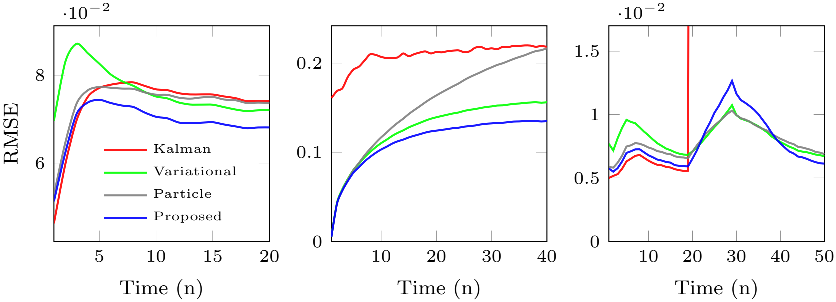
Results - Linear

It's faster and provides better approximations to the true posterior of the Laplace model than existing methods like variational inference and particle filtering.



Results - Linear

Performance compared with the Kalman filter, a variational-inference based Laplace filter, and a bootstrap Particle filter with multinomial resampling.

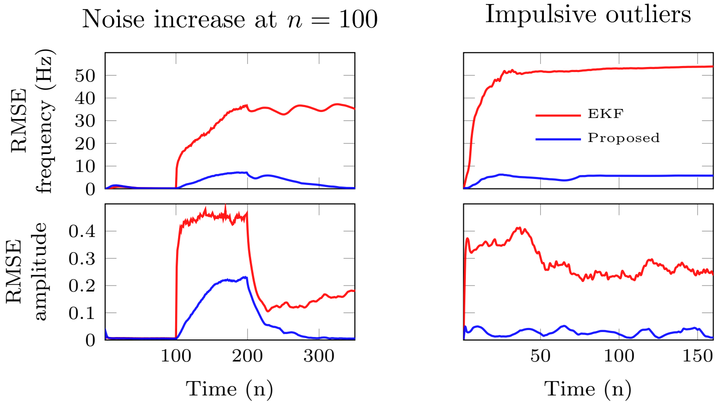


The proposed method outperforms all the others, and with comparable speed to the Kalman filter.

Results - Nonlinear

Audio frequency estimation from a noisy sinusoidal oscillation.

$$\mathbf{x}_n = \begin{bmatrix} \phi_n \\ f_n \\ a_n \end{bmatrix}, \quad h(\mathbf{x}_{n-1}) = \begin{bmatrix} \phi_{n-1} + 2\pi T f_{n-1} \\ f_{n-1} \\ a_{n-1} \end{bmatrix}, \quad g(\mathbf{x}_n) = a_n \sin(\phi_n).$$



Conclusion & Future Work

We presented a **new Bayesian filter** for state space models with **univariate Laplace data noise** that

- ✓ computes locally-exact posterior moments, propagates them forward with Gaussian density,
- ✓ is robust to outliers and a variety of non-Gaussian noises,
- ✓ is simple to implement and fast like the Kalman filter,
- ✓ and provides better state estimation than variational Bayes and particle filtering methods.

Future work:

- Infer from multivariate Laplace-distributed observations.
- Devise smoothing algorithm and automatic learning of parameters for the Laplace model.