DETERMINISTIC AND STOCHASTIC GAUSSIAN PARTICLE SMOOTHING

Onno Zoeter
Microsoft Research Cambridge
onno@zoeter.nl

Alexander Ypma
GNResound
aypma@gnresound.com

Tom Heskes
Radboud University Nijmegen
tomh@cs.ru.nl

ABSTRACT

In this article we study inference problems in non-linear dynamical systems. In particular we are concerned with assumed density approaches to filtering and smoothing. In models with uncorrelated (but dependent) state and observation, the extended Kalman filter and the unscented Kalman filter break down. We show that the Gaussian particle filter and the one-step unscented Kalman filter make less assumptions and potentially form useful filters for this class of models. We construct a symmetric smoothing pass for both filters that does not require the dynamics to be invertible.

We investigate the characteristics of the methods in an interesting problem from mathematical finance. Among others we find that smoothing helps, in particular for the deterministic one-step unscented Kalman filter.

1. INTRODUCTION

Filtering and estimation problems in linear dynamical systems are well understood. The Kalman filter and Rauch-Tung-Streibel smoother provide efficient algorithms for exact online and off-line estimates. These algorithms are based on two main characteristics of the canonical linear dynamical system. The first is the conditional independence structure, which allows for a fast recursive algorithm. The second characteristic is that all required local operations can be done analytically and efficiently: due to the linear-Gaussian assumptions in the model only Gaussians need to be multiplied and integrated out.

Non-linear dynamical systems share the conditional independencies, but the local operations, in general, have no analytic solutions.

The oldest approximate inference algorithm, the extended Kalman filter (EKF) (see e.g. [10] and references therein), approximates the filtered posteriors at every time step by a Gaussian. Local integrals are approximated using an explicit linearization. The unscented Kalman filter (UKF) [4] improves upon the extended filter. As emphasized in [6] the UKF can be interpreted as an EKF where the linearization is obtained using Gaussian quadrature.

Both methods are examples of assumed density filtering methods: the recursively computed approximations are projected onto a chosen parametric form, the assumed density (a Gaussian in this case).

The measurement update steps in both the EKF and UKF are (implicitly) based on a linearization of the observation model. A simple argument in [12] shows that for models with uncorrelated (but dependent) state and observation, approximations based on such explicit linearizations break down. They result in a Gaussian approximation where state and observation are uncorrelated, but therefore, due to their Gaussianity, also independent. I.e. the approximate filter never updates the state prediction after making an observation.

Throughout this article we will consider the following general class of non-linear models

$$\mathbf{x}_t = f(\mathbf{x}_{t-1}, \boldsymbol{\epsilon}_t), \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(0, Q), \quad \text{for } t = 2:T \quad (1)$$

$$\mathbf{y}_t = g(\mathbf{x}_t, \boldsymbol{\eta}_t), \quad \boldsymbol{\eta}_t \sim \mathcal{N}(0, R), \quad \text{for } t = 2:T \quad (2)$$

with states $\mathbf{x}_{1:T}$ and observations $\mathbf{y}_{1:T}$. All disturbances $\boldsymbol{\epsilon}_t$ and $\boldsymbol{\eta}_t$ are assumed to be independently drawn, and $\mathbf{x}_1 \sim \mathcal{N}(\mathbf{m}_1, V_1)$. In the above, boldface variables denote vectors, and $\mathcal{N}(\mathbf{m}, V)$ denotes the (multivariate) Gaussian probability density with mean $\mathbf{m}$ and covariance V. Occasionally we will write $\mathcal{N}(\mathbf{x}; \mathbf{m}, V)$ to emphasize that $\mathbf{x}$ is normally distributed with mean $\mathbf{m}$ and covariance V.

For practical reasons we will assume that $p(\mathbf{y}_t|\mathbf{x}_t) = \int g(\mathbf{x}_t, \boldsymbol{\eta}_t) p(\boldsymbol{\eta}_t) d\boldsymbol{\eta}_t$ with $p(\boldsymbol{\eta}_t)$ Gaussian, can be computed analytically. We will come back to this in Section 2.

2. DETERMINISTIC AND STOCHASTIC GAUSSIAN PARTICLE FILTERING

The extra Gaussian approximation of $p(\mathbf{x}_t, \mathbf{y}_t|\mathbf{y}_{1:t-1})$ in the measurement update of the traditional unscented filter can be circumvented by approximating $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ directly. We investigate a Monte Carlo and a deterministic approach. Below the measurement updates for univariate problems are presented; extensions to multivariate problems are straightforward.

Measurement update $\tilde{p}(x_t|y_{1:t}) = \mathcal{N}(x_t; m_{t|t}, v_{t|t})$, with

$$\begin{aligned} m_{t|t} &= \int x_t \frac{p(y_t|x_t)\tilde{p}(x_t|y_{1:t-1})}{Z_t} dx_t \\ Z_t &= \int p(y_t|x_t)\tilde{p}(x_t|y_{1:t-1}) dx_t = \tilde{p}(y_t|y_{1:t-1}) \\ v_{t|t} &= \int (x_t - m_{t|t})^2 \frac{p(y_t|x_t)\tilde{p}(x_t|y_{1:t-1})}{Z_t} dx_t \ . \end{aligned}$$

These integrals can be approximated using importance sampling, which leads to the (stochastic) Gaussian particle filter [5]. Often the prior can be used as a proposal distribution, but more elaborate proposal distributions can be used as well.

More in spirit with the original UKF, the above integrals can also be approximated using points determined by Gaussian quadrature [1, 12]. Monomial points and weights $\{\mathcal{X}_i, W_i\}$ are determined for the state $x_t \sim \mathcal{N}(m_{t|t-1}, v_{t|t-1})$, and the mean and variance are updated as

$$Z_t = \sum_i W_i p(y_t|\mathcal{X}_i) \quad (3)$$

$$m_{t|t} = \sum_i W_i \mathcal{X}_i \frac{p(y_t|\mathcal{X}_i)}{Z_t} \quad (4)$$

$$v_{t|t} = \left[\sum_i W_i \mathcal{X}_i^2 \frac{p(y_t|\mathcal{X}_i)}{Z_t}\right] - \left(m_{t|t}\right)^2 \ . \quad (5)$$

In our implementation we have used the McNamee-Stenger quadrature rules [7].

There is a strong similarity between the two approaches. The importance sampling based approach would have the same updates (3)–(5) but would *draw*, instead of deterministically place, $\mathcal{X}_i$ from the prior $\tilde{p}(x_t|y_{1:t-1})$ and always associate $W_i = 1$.

For any finite number of samples, the importance sampling approximation of the local integrals is biased. In the limit of infinite samples they converge to the exact expectations. The deterministic approach is also biased, since $p(y_t|x_t)$, the likelihood of observing observation y_t, is not a polynomial in x_t. Both methods fail when the prior is ill-matched to the posterior. In such cases more elaborate proposal distributions may permit useful approximations.

In the measurement update we have assumed that the integral

$$p(y_t|x_t) = \int p(y_t|x_t, \eta_t) p(\eta_t) d\eta_t,$$

with $p(y_t|x_t, \eta_t) \equiv \delta_{y_t = g(x_t, \eta_t)}$ a Kronecker delta function, can be done analytically. This holds if g is linear in η_t. So the important class of a non-linear mapping with additive Gaussian noise can be treated in this way. More complex observation models leading to χ^2, t, or F distributions are on the boundary of what can be handled by a one-step filter.

The product of all the normalization constants

$$\prod_{t=1}^{T} Z_t = \prod_{t=1}^{T} \tilde{p}(y_t|y_{1:t-1}) \approx p(y_{1:t}) \ ,$$

gives an approximation of the likelihood.

3. ITERATIVE GAUSSIAN PARTICLE SMOOTHING

Using the expectation propagation framework [8] we can formulate a symmetric smoothing pass for the general class of models under consideration, without inverting the latent state dynamics. This is in contrast to previous approaches where the inverse of the latent state dynamics was determined analytically (when it existed) or approximated using e.g. a multilayer perceptron [9]. The filtering and smoothing pass can be iterated, in an attempt to make the greedy local approximations consistent on a global level. In [11] a related smoothing pass is introduced for the traditional two-step UKF. Due to the two-step nature of the underlying filter this smoother does not allow iterative refinement.

We will give a brief introduction to expectation propagation and introduce some notation, but refer the interested reader to [8] and [2] for more details. As in Section 2 we will introduce the material in a univariate setting; extensions to the multivariate case are again straightforward.

The required joint posterior over all latent states can be represented as a product over factors Ψ_t defined as

$$\begin{aligned} \Psi_1(x_1) &\equiv p(y_1|x_1)p(x_1) \\ \Psi_t(x_{t-1,t}) &\equiv p(y_t|x_t)p(x_t|x_{t-1}) \ , \end{aligned}$$

such that

$$p(x_{1:T}|y_{1:T}) \propto \prod_{t=1}^{T} \Psi_t(x_{t-1,t}) \ . \quad (6)$$

Any required marginal can be computed from this joint by integration. However, computing the product in (6) explicitly is computationally too intensive and the required integrals cannot be done analytically.

To get an approximation, a tractable family $q(x_{1:T})$ is introduced. For the one-step unscented smoother we choose

$$q(x_{1:T}) = \prod_t q(x_t), \quad q(x_t) \equiv \mathcal{N}(x_t; m_{t|T}, v_{t|T}) \ , \quad (7)$$

a fully factorized Gaussian distribution. The algorithm is initialized with arbitrary approximations $\tilde{\Psi}$ of the factors Ψ_t such that their product is a member of $q(x_{1:T})$. Since $q(x_{1:T})$ factors, the approximation of $\Psi_t(x_{t-1,t})$ factors into a contribution to $q(x_{t-1})$ and a contribution to $q(x_t)$. So we can use without loss of generalization the notation

$$\Psi_t(x_{t-1,t}) \approx \tilde{\Psi}_t(x_{t-1,t}) \equiv c_t \beta_{t-1}(x_{t-1}) \alpha_t(x_t) \ . \quad (8)$$

With this choice $q(x_t) \propto \alpha_t(x_t)\beta_t(x_t)$. In the above c_t emphasizes that the product of α_t and β_t need not be normalized. The α_t and β_t are often referred to as messages, and are Gaussian potentials. Readers familiar with the HMM forward-backward algorithm can keep in mind that the choice of notation in (8) implies that the α_t and β_t messages have a similar interpretation here as they do in the HMM algorithms.

The expectation propagation based approximation then proceeds by iteratively updating the approximating factors as follows.

Update of the approximation of Ψ_t

1. Remove $\beta_{t-1}(x_{t-1})\alpha_t(x_t)$, the old approximation of $\Psi_t(x_{t,t-1})$, by division

$$\begin{aligned}
&\frac{q(x_{t-1,t})}{\beta_{t-1}(x_{t-1})\alpha_t(x_t)} \\
&= \frac{\alpha_{t-1}(x_{t-1})\beta_{t-1}(x_{t-1})\alpha_t(x_t)\beta_t(x_t)}{\beta_{t-1}(x_{t-1})\alpha_t(x_t)} \\
&= \alpha_{t-1}(x_{t-1})\beta_t(x_t) \ .
\end{aligned}$$

2. Put in the exact factor $\Psi_t(x_{t,t-1})$

$$r(x_{t-1,t}) = \frac{\alpha_{t-1}(x_{t-1})\Psi_t(x_{t-1,t})\beta_t(x_t)}{Z_{t|T}} \ ,$$

with $Z_{t|T} \equiv \int \alpha_{t-1}(x_{t-1})\Psi_t(x_{t-1,t})\beta_t(x_t)dx_{t-1,t}$.

3. Since $r(x_{t-1,t})$ is not in the chosen family, approximate it by $q(x_{t-1})q(x_t)$ closest in Kullback-Leibler sense. For $q(x_{t-1})$ this becomes

$$\begin{aligned}
q^{\text{new}}(x_{t-1}) &= \mathcal{N}(x_{t-1}; m^{\text{new}}_{t-1|T}, V^{\text{new}}_{t-1|T}), \text{with} \\
m^{\text{new}}_{t-1|T} &= \int x_{t-1} r(x_{t-1,t}) dx_{t-1,t} \ , \\
Z^{\text{new}}_{t|T} &= \int \alpha_{t-1}(x_{t-1})\Psi_t(x_{t,t-1})\beta_t(x_t)dx_{t-1,t} \ , \\
v^{\text{new}}_{t-1|T} &= \int (x_{t-1} - m_{t-1|T})^2 r(x_{t-1,t}) dx_{t-1,t} \ ,
\end{aligned}$$

and analogously for $q(x_t)$.

4. Infer the contribution of Ψ_t by division

$$\begin{aligned}
\beta^{\text{new}}_{t-1}(x_{t-1}) &= \frac{q^{\text{new}}(x_{t-1})}{\alpha_{t-1}(x_{t-1})} \\
\alpha^{\text{new}}_t(x_t) &= \frac{q^{\text{new}}(x_t)}{\beta_t(x_t)} \ .
\end{aligned}$$

In principle the updates can be done in any order, but an iteration of forward-backward passes seems most logical.

Combining the above steps 1–3 we get

$$m^{\text{new}}_{t-1|T} = \int x_{t-1} \frac{q(x_{t-1})q(x_t)\Psi_t(x_{t-1,t})}{\beta_{t-1}(x_{t-1})\alpha_t(x_t)Z_{t|T}} dx_{t-1,t} \ , \quad (9)$$

and similarly for $Z^{\text{new}}_{t|T}$, $v^{\text{new}}_{t-1|T}$, $m^{\text{new}}_{t|T}$ and $v^{\text{new}}_{t|T}$. In (9) we can identify an integral form $y = \int h(x_{t-1,t})K(x_{t-1,t})dx_{t-1,t}$ with

$$\begin{aligned}
h(x_{t-1,t}) &= \frac{\Psi_t(x_{t-1,t})}{\beta_{t-1}(x_{t-1})\alpha_t(x_t)Z_{t|T}} \\
K(x_{t-1,t}) &= q(x_{t-1})q(x_t) \ .
\end{aligned}$$

So the required local approximations can be done using Gaussian quadrature.

In an analogous way the importance sampling based Gaussian particle filter can be extended to a Gaussian particle smoother. Instead of deterministically placing the $\mathcal{X}_i$, they are drawn from $K(x_{t-1,t})$ and get associated weight $W_i = 1$.

At this point it is perhaps useful to emphasize that the form in (7) does not imply as coarse an approximation as it may at first appear. Due to step 2 in the update, the independent $q(x_t)$'s are updated such that they are as close as possible to the true (dependent) posteriors. The assumed density filters from Section 2 are special cases of the iterative algorithm, so it is good to realize that the iterated approximations are expected to be better instead of coarser than these approximate filters, although starting the derivation with (7) perhaps makes it appear otherwise for readers unfamiliar with EP.

One can verify that the filtering algorithm from the previous section is a first forward pass of the algorithm described above with a suitable choice for the messages α_t and β_t, that is to say, with $\beta_t = 1$ and α_t the prediction as computed in the filtering algorithm.

4. EXPERIMENTS

We will evaluate the introduced methods with a discrete time variant of the stochastic volatility model for stock prices from [3]. In this model y_t represents the *daily log return*, which can be interpreted as a stochastic interest $y_t = \log \frac{S_t}{S_{t-1}}$ that is earned on stock S between days t and $t-1$. This log return is drawn from a normal distribution with fixed mean μ and standard deviation (*volatility* in financial parlance) e^{x_t+l}. The unobserved log volatility (minus a base level l) x_t follows an AR(1) process. With

$$\begin{aligned}
f(x_{t-1}, \epsilon_t) &= ax_{t-1} + \epsilon_t, \quad \text{and,} \\
g(x_t, \eta_t) &= \mathrm{e}^{x_t+l}\eta_t + \mu \ ,
\end{aligned}$$

the stochastic volatility model is a special case of the class of models (1)–(2). It has a non-linear observation model, and uncorrelated (but dependent) state-observation pairs.

Figure 1 presents a typical result from an experiment with artificially generated data. The top plot shows $S_{1:T}$, the daily closing values of an artificial stock. The second plot shows $y_{1:T}$, the corresponding log returns as they were drawn from the model (solid). Also shown are one-day ahead predictions based on the one-step unscented filter and 2 standard

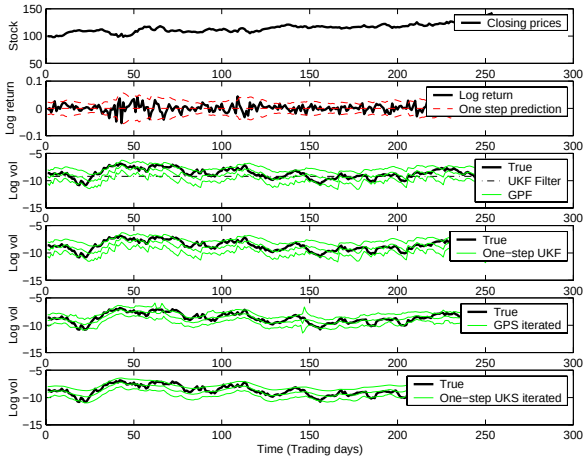

Fig. 1. Results from inference on artificial stock data, see text for details.

deviations errors bars (dashed). The bottom four plots show the estimated log volatilities based on the different methods discussed in this article. The true underlying log volatilities that were drawn from the model are represented as thick solid curves (identical copies in the bottom four plots for clarity).

The estimate based on the traditional unscented Kalman filter (dashed, third plot from the top), quickly converges to a straight line and shows that the traditional UKF fails to track the log-volatility. The stochastic Gaussian particle filter and the one-step unscented Kalman filter can track the log volatility reasonably well. Smoothed results give slightly better estimates of the underlying log volatilities. When filtering there is hardly a distinction between the importance sampling based approach (with 1000 samples) and the Gaussian quadrature based approach (with 5 points). When smoothing however, in particular when iteratively improved, we see that the improvement over the filtered estimate is less for the sampling based approach than for the deterministic approach. Whereas with a deterministic scheme we can hope that there is a fixed point of our iterative scheme, due to the stochasticity in the sampler we are certain that for the stochastic Gaussian particle smoother there is none. Due to the iteration certain anomalies can be enlarged (e.g. the dip around the 147-th day).

Figure 2 summarizes the performance of the various approaches on ten drawn problems. The left plot shows the mean squared error between the generated log-volatilities and the inferred posterior means. The bars represent averages over 10 runs, error bars show one standard deviation. Experiments were done for an unscented Kalman filter (1), a stochastic Gaussian particle filter with 1000 samples (2), 100 samples (3), and in (4) with 5 samples for the filter, and 25 samples for the smoother (i.e. with computational complexity identical to the deterministic one-step filter and smoother). Bar (5) represents a deterministic one-step filter with 5 points per dimension. The middle plot gives a *paired* analysis. The bars

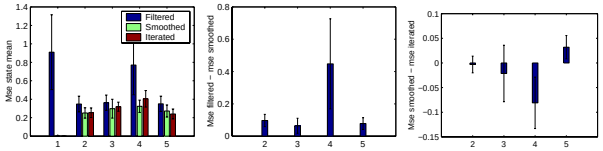

Fig. 2. The above bar plots summarize the performance of the various approaches. See text for details.

show, per algorithm, the difference between the MSE of the filter and the MSE of the smoother, averaged over the ten runs. The error bars show the standard deviation scaled with $t_9(.99)\frac{1}{\sqrt{10}}$, such that they correspond to a one-sided t-test at the 99% level. The right plot is analogous, but shows the difference between one smoothing pass, and ten iterations of EP. We see that smoothing results in a significant improvement for all algorithms, iterating only for the deterministic one-step smoother.

5. REFERENCES

[1] Erik Bølviken and Geir Storvik. Deterministic and stochastic particle filters in state space models. In A. Doucet, N. de Freitas, and N. Gordon, editors, *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 2001.

[2] Tom Heskes and Onno Zoeter. Generalized belief propagation for approximate inference in hybrid Bayesian networks. In *AISTATS*, 2003.

[3] John Hull and Alan White. The pricing of options on assets with stochastic volatilities. *The Journal of Finance*, 42(2):281–300, June 1987.

[4] S.J. Julier and J. K. Uhlmann. A new extension of the kalman filter to nonlinear systems. In *Proceedings of AeroSense*, 1997.

[5] Jayesh Kotecha and Petar Djurić. Gaussian particle filtering. *IEEE Transactions on Signal Processing*, 51(10):2592–2601, 2003.

[6] Uri Lerner. *Hybrid Bayesian Networks for Reasoning about Complex Systems*. PhD thesis, Stanford University, 2002.

[7] J. McNamee and F. Stenger. Construction of fully symmetric numerical integration formulas. *Numerische Mathematik*, 10:327–344, 1967.

[8] T. Minka. Expectation propagation for approximate Bayesian inference. In *Proceedings UAI*, 2001.

[9] Eric Wan and Rudolph van der Merwe. The unscented Kalman filter. In S. Haykin, editor, *Kalman Filtering and Neural Networks*. Wiley, 2001.

[10] M. West and J. Harrison. *Bayesian Forecasting and Dynamic Models*. Springer, 2nd edition, 1997.

[11] Alexander Ypma and Tom Heskes. Novel approximations for inference and learning in nonlinear dynamical systems. In *ESANN*, 2004.

[12] Onno Zoeter, Alexander Ypma, and Tom Heskes. Improved unscented Kalman smoothing for stock volatility estimation. In *Proceedings MLSP*, 2004.