Model Based Learning of Sigma Points in Unscented Kalman Filtering

Ryan Turner

Kittilä, Finland
September 1, 2010

joint work with Carl Rasmussen
Motivation

- Estimating (latent) states and predicting future observations from noisy measurements

(a) State estimation for control

(b) Time series prediction
Setup

\[ x_t = f(x_{t-1}) + w, \quad w \sim \mathcal{N}(0, Q) \]
\[ z_t = g(x_t) + v, \quad v \sim \mathcal{N}(0, R) \]

- \( x \in \mathbb{R}^D \): latent state, \( z \in \mathbb{R}^M \): measurement
Filtering

1) predict next hidden state
   \[ p(x_{t-1}|z_{1:t-1}) \]
   \[ p(x_t|z_{1:t-1}) \]
   time update

2) predict measurement
   \[ p(x_t|z_{1:t-1}) \]
   \[ p(z_t|z_{1:t-1}) \]
   measurement update

3) hidden state posterior
   \[ p(x_t|z_{1:t}) \]
   measure \( z_t \)

1) predict next hidden state
2) predict measurement
3) hidden state posterior
Approximate predictions

- Extended Kalman filter (EKF): local linearizations of the function; propagating Gaussians exact through linearized function [5]
Approximate predictions

- Extended Kalman filter (EKF): local linearizations of the function; propagating Gaussians exact through linearized function [5]

- Unscented Kalman filter (UKF): approximation of the density by a number of sigma points [3]
The UKF

- We will focus on the UKF
- The UKF uses the whole distribution on $x_t$ not just the mean (like the EKF)
- Filtering and prediction uses unscented transform (UT)
The UKF

1. We will focus on the UKF.
2. The UKF uses the whole distribution on $x_t$ not just the mean (like the EKF).
3. Filtering and prediction uses **unscented transform** (UT).
4. In 1D approximates distribution by mean and $\alpha$-standard deviation points.
5. Loosely interpret as approximating distribution by $2D + 1$ point masses.

\[ h(X) \sim \sum_{i=0}^{2D+1} w_i \phi_i \]

\[ h^\prime(X^\prime) \sim \sum_{i=0}^{2D+1} w_i \phi_i(X^\prime) \]

Turner (Engineering, Cambridge)
The UKF

1. We will focus on the UKF.
2. The UKF uses the whole distribution on $x_t$ not just the mean (like the EKF).
3. Filtering and prediction uses unscented transform (UT).
4. In 1D approximates distribution by mean and $\alpha$-standard deviation points.
5. Loosely interpret as approximating distribution by $2D + 1$ point masses.

$\beta$ affects weight of center point; $\alpha, \kappa$ affect the spread of points.

Sample mean and covariance of sigma points match original distribution.

Turner (Engineering, Cambridge) Learning of Sigma Points in Unscented Kalman Filtering
Remarks

1. Reconstructs the mean and covariance on $x_{t+1}$ had the dynamics been linear.

2. No guarantee of matching the true moments of the non-Gaussian distribution.

3. Must fix parameters $\theta := \{\alpha, \beta, \kappa\}$ before seeing data.
Remarks

1. Reconstructs the mean and covariance on $x_{t+1}$ had the dynamics been linear.
2. No guarantee of matching the true moments of the non-Gaussian distribution.
3. Must fix parameters $\theta := \{\alpha, \beta, \kappa\}$ before seeing data.
4. Some heuristics for setting them e.g. $\beta = 2$ optimal for Gaussian state distribution [9, 2].
5. Common default $\alpha = 1$, $\beta = 0$, and $\kappa = 2$. 

Turner (Engineering, Cambridge)
The Achilles’ Heel of the UKF
The Achilles’ Heel of the UKF
The Achilles’ Heel of the UKF
The Achilles’ Heel of the UKF
The Achilles’ Heel of the UKF

Typical UKF failure mode: **sigma point collapse.** \( \alpha = 1 \) gives delta spike while \( \alpha = 0.68 \) gives optimal moment matches solution. ➡️ Can we find empirically which \( \theta \) are most likely to give good solutions?
Alternative View of the UKF

1. EKF and UKF approximate nonlinear system as nonstationary linear system
2. The UKF defines its own generative process of the time series
3. Can sample from the UKF via predict-sample-correct
4. \( \{\alpha, \beta, \kappa\} \) are generative parameters
5. We can learn the parameters \( \theta \) in a principled way!
Model Based Learning

1. Learn the parameters $\theta$ in a principled way
2. In model based view, maximize marginal likelihood:

$$\ell(\theta) := \log p(z_{1:T}|\theta) = \sum_{t=1}^{T} \log p(z_t|z_{1:t-1}, \theta). \quad (1)$$

3. With learning: UKF-L. Using Default $\theta$: UKF-D
4. One-step-ahead predictions $p(z_t|z_{1:t-1})$
Likelihood Illustrations

(c) $\alpha$

(d) $\beta$

(e) $\kappa$

Not much hope of gradient based optimization based on these cross-sections :(

Turner (Engineering, Cambridge)
Gaussian Process Optimizers

1. Do derivative free optimization [6]
2. Treat optimization as a sequential decision problem: reward $r$ for right input $\theta$ to get a large $\ell(\theta)$
Gaussian Process Optimizers

1. Do derivative free optimization [6]
2. Treat optimization as a sequential decision problem: reward $r$ for right input $\theta$ to get a large $\ell(\theta)$
3. Must place model on $\ell(\theta)$ to compute $\mathbb{E}[\ell(\theta)]$ and $\text{Var}[\ell(\theta)]$
4. Gaussian processes (GPs) are priors on functions
5. Used for integration in [7]
Gaussian Process Optimizers

1. **greedy** strategy will go where $\mathbb{E}[\ell(\theta)]$ is maximized
2. **explorative** strategy will go where $\text{Var}[\ell(\theta)]$ is maximized
Gaussian Process Optimizers

1. **Greedy** strategy will go where $\mathbb{E}[\ell(\theta)]$ is maximized
2. **Explorative** strategy will go where $\text{Var}[\ell(\theta)]$ is maximized
3. $J(\theta)$ trades-off exploration with exploitation using $K$

$$J(\theta) := \mathbb{E}[\ell(\theta)] + K \sqrt{\text{Var}[\ell(\theta)]}$$ (2)
GPO Demo

Learning of Sigma Points in Unscented Kalman Filtering
GPO Demo

Log Likelihood

θ
GPO Demo

The GEnerative Process and Learning

Experiments and Results

Conclusions and Future Work

Log Likelihood

θ

Learning of Sigma Points in Unscented Kalman Filtering

Turner (Engineering, Cambridge)
GPO Demo

![Graph showing log likelihood over the range of θ from 0 to 10. The graph includes a smooth black line and a red line with fluctuations, indicating learning of sigma points in unscented Kalman filtering. The vertical axis represents log likelihood values ranging from -2.5 to 2.5.]
GPO Demo

Log Likelihood

$\theta$

Turner (Engineering, Cambridge) Learning of Sigma Points in Unscented Kalman Filtering 22
GPO Demo

Log Likelihood

\[ \theta \]

Turner (Engineering, Cambridge)
GPO Demo

![Graph showing log likelihood over a range of θ values. The graph includes a red line with data points and a shaded area indicating the standard deviation.](image-url)
GPO Demo

The UKF

The Generative Process and Learning

Experiments and Results

Conclusions and Future Work

Learning of Sigma Points in Unscented Kalman Filtering

Turner (Engineering, Cambridge)
Now We Can Learn

1. We can use derivative free optimization to learn $\ell(\theta)$
2. We can find the best $\alpha$, $\beta$, and $\kappa$
Experimental Setup

1. Three dynamical systems: sinusoidal dynamics [8], Kitagawa dynamics [1, 4], and pendulum dynamics [1]

2. Compare UKF-D, EKF, the GP-UKF, and GP-ADF, and the time independent model (TIM)
Experimental Setup

1. Three dynamical systems: sinusoidal dynamics [8], Kitagawa dynamics [1, 4], and pendulum dynamics [1]
2. Compare UKF-D, EKF, the GP-UKF, and GP-ADF, and the time independent model (TIM)
3. UKF-D used standard parameters $\alpha = 1, \beta = 0, \kappa = 2$
4. Method evaluated on negative log-predictive likelihood (NLL) and the mean squared error (MSE)
The Dynamical Systems

Sinusoidal dynamics:

\begin{align}
  x_{t+1} &= 3 \sin(x_t) + w, \quad w \sim \mathcal{N}(0, 0.1^2), \\
  z_t &= \sigma(x_t/3) + v, \quad v \sim \mathcal{N}(0, 0.1^2).
\end{align}
The Dynamical Systems

Sinusoidal dynamics:

\[ x_{t+1} = 3 \sin(x_t) + w, \quad w \sim \mathcal{N}(0, 0.1^2), \quad (3) \]
\[ z_t = \sigma(x_t/3) + v, \quad v \sim \mathcal{N}(0, 0.1^2). \quad (4) \]

The Kitagawa model:

\[ x_{t+1} = 0.5x_t + \frac{25x_t}{1 + x_t^2} + w, \quad w \sim \mathcal{N}(0, 0.2^2), \quad (5) \]
\[ z_t = 5 \sin(2x_t) + v, \quad v \sim \mathcal{N}(0, 0.01^2). \quad (6) \]
Pendulum

1. Track pendulum (fairly linear system)
2. Nonlinear measurements
3. Partially observed (no measurements of angular velocity)
4. Time series of 80 s
## Quantitative Results

### Results for accuracy of $p(z_{t+1}|z_{1:t-1})$:

<table>
<thead>
<tr>
<th>Method</th>
<th>NLL</th>
<th>p-value</th>
<th>MSE</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sinusoid ($T = 500$ and $R = 10$)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UKF-D</td>
<td>$10^{-1} \times -4.58 \pm 0.168$</td>
<td>$&lt;0.0001$</td>
<td>$10^{-2} \times 2.32 \pm 0.0901$</td>
<td>$&lt;0.0001$</td>
</tr>
<tr>
<td>UKF-L *</td>
<td>$-5.53 \pm 0.243$</td>
<td>N/A</td>
<td>$1.92 \pm 0.0799$</td>
<td>N/A</td>
</tr>
<tr>
<td>EKF</td>
<td>$-1.94 \pm 0.355$</td>
<td>$&lt;0.0001$</td>
<td>$3.03 \pm 0.127$</td>
<td>$&lt;0.0001$</td>
</tr>
<tr>
<td>GP-ADF</td>
<td>$-4.13 \pm 0.154$</td>
<td>$&lt;0.0001$</td>
<td>$2.57 \pm 0.0940$</td>
<td>$&lt;0.0001$</td>
</tr>
<tr>
<td>GP-UKF</td>
<td>$-3.84 \pm 0.175$</td>
<td>$&lt;0.0001$</td>
<td>$2.65 \pm 0.0985$</td>
<td>$&lt;0.0001$</td>
</tr>
<tr>
<td>TIM</td>
<td>$-0.779 \pm 0.238$</td>
<td>$&lt;0.0001$</td>
<td>$4.52 \pm 0.141$</td>
<td>$&lt;0.0001$</td>
</tr>
</tbody>
</table>

| Kitagawa ($T = 10$ and $R = 200$) | | | | |
| UKF-D      | $10^0 \times 3.78 \pm 0.662$ | $<0.0001$ | $10^0 \times 5.42 \pm 0.607$ | $<0.0001$ |
| UKF-L *    | $2.24 \pm 0.369$ | N/A          | $3.60 \pm 0.477$ | N/A         |
| EKF        | $617 \pm 554$ | 0.0149       | $9.69 \pm 0.977$ | $<0.0001$   |
| GP-ADF     | $2.93 \pm 0.0143$ | 0.0001       | $18.2 \pm 0.332$ | $<0.0001$   |
| GP-UKF     | $2.93 \pm 0.0142$ | 0.0001       | $18.1 \pm 0.330$ | $<0.0001$   |
| TIM        | $48.8 \pm 2.25$ | $<0.0001$   | $37.2 \pm 1.73$ | $<0.0001$   |

$T$ is the length of the test sequences and $R$ is the number of restarts averaged over.
Quantitative Results Continued

Results for accuracy of \( p(z_{t+1} | z_{1:t-1}) \):

<table>
<thead>
<tr>
<th>Method</th>
<th>NLL</th>
<th>p-value</th>
<th>MSE</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pendulum (( T = 200 = 80 \text{s} ) and ( R = 100 ))</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UKF-D</td>
<td>( 10^0 \times 3.17 \pm 0.0808 )</td>
<td>(&lt;0.0001)</td>
<td>( 10^{-1} \times 5.74 \pm 0.0815 )</td>
<td>(&lt;0.0001)</td>
</tr>
<tr>
<td>UKF-L *</td>
<td>( 0.392 \pm 0.0277 )</td>
<td>N/A</td>
<td>( 1.93 \pm 0.0378 )</td>
<td>N/A</td>
</tr>
<tr>
<td>EKF</td>
<td>( 0.660 \pm 0.0429 )</td>
<td>(&lt;0.0001)</td>
<td>( 1.98 \pm 0.0429 )</td>
<td>0.0401</td>
</tr>
<tr>
<td>GP-ADF</td>
<td>( 1.18 \pm 0.00681 )</td>
<td>(&lt;0.0001)</td>
<td>( 4.34 \pm 0.0449 )</td>
<td>(&lt;0.0001)</td>
</tr>
<tr>
<td>GP-UKF</td>
<td>( 1.77 \pm 0.0313 )</td>
<td>(&lt;0.0001)</td>
<td>( 5.67 \pm 0.0714 )</td>
<td>(&lt;0.0001)</td>
</tr>
<tr>
<td>TIM</td>
<td>( 0.896 \pm 0.0115 )</td>
<td>(&lt;0.0001)</td>
<td>( 4.13 \pm 0.0426 )</td>
<td>(&lt;0.0001)</td>
</tr>
</tbody>
</table>

\( T \) is the length of the test sequences and \( R \) is the number of restarts averaged over.

Learned \( \theta \):

1. Sinusoid: \( \theta = \{\alpha = 2.0216, \beta = 0.2434, \kappa = 0.4871\} \)
2. Kitagawa: \( \theta = \{\alpha = 0.3846, \beta = 1.2766, \kappa = 2.5830\} \)
3. Pendulum: \( \theta = \{\alpha = 0.5933, \beta = 0.1630, \kappa = 0.6391\} \)
Qualitative Results

UKF-D vs UKF-L for one-step-ahead prediction for dimension 1 of $z_t$ in the Pendulum model. Red line is the truth, black line and shaded area are prediction.
Conclusions

1. Automatic and model based learning of UKF parameters \( \{\alpha, \beta, \kappa\} \)
2. The UKF can be reinterpreted as a generative process
3. Learning makes sigma point collapse less likely
4. UKF-L significantly better than UKF-D for all error measures and data sets

References

Marc P. Deisenroth, Marco F. Huber, and Uwe D. Hanebeck.
Analytic moment-based Gaussian process filtering.

S. J. Julier and J. K. Uhlmann.
Unscented filtering and nonlinear estimation.

Simon J. Julier and Jeffrey K. Uhlmann.
A new extension of the Kalman filter to nonlinear systems.

Genshiro Kitagawa.
Monte Carlo filter and smoother for non-Gaussian nonlinear state space models.

Peter S. Maybeck.

Gaussian processes for global optimization.
In *3rd International Conference on Learning and Intelligent Optimization (LION3)*, Trento, Italy, January 2009.

Carl E. Rasmussen and Zoubin Ghahramani.
Bayesian Monte Carlo.

Ryan Turner, Marc Peter Deisenroth, and Carl Edward Rasmussen.
State-space inference and learning with Gaussian processes.
In *the 13th International Conference on Artificial Intelligence and Statistics*, volume 9, Sardinia, Italy, 2010.

Eric A. Wan and Rudolph van der Merwe.
Learning of Sigma Points in Unscented Kalman Filtering.