Companion Matrices for Differential Equations

We briefly describe companion matrices for turning $n$-order linear SDEs into first-order by representing the system as a linear operator on an augmented state variable.

Consider the 2-order system

$$a_0 x(t) + a_1 \frac{d}{dt} x(t) + a_2 \frac{d^2}{dt^2} x(t) = w(t)$$

Define a new variable, $z = dx/dt$, and substitute into the above equation:

$$a_0 x(t) + a_1 z(t) + a_2 \frac{d}{dt} z(t) = w(t)$$

This is now the 1-order system:

$$\frac{d}{dt} x(t) = z(t)$$
$$\frac{d}{dt} z(t) = -\tilde{a}_0 x(t) - \tilde{a}_1 z(t) + a_2^{-1} w(t),$$

where $\tilde{a}_0$ and $\tilde{a}_1$ are $a_0/a_2$ and $a_1/a_2$ respectively.

We can write this using a joint state, $x(t) = [x(t) \ z(t)]^\top$:

$$\frac{d}{dt} x(t) = \begin{bmatrix} 0 & 1 \\ -\tilde{a}_0 & -\tilde{a}_1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ a_2^{-1} \end{bmatrix} w(t)$$

The companion matrix for linear $n$-order systems scales in a similar manner, with the final matrix consisting of a final row of scalars and off-diagonal band with value 1. The extension to non-linear systems is a straightforward extension here, simply replacing the matrix with a vector-valued function.

Unscented Transform

The unscented transform is a means for propagating a random variable, $x$ through a non-linear functional, $f$, by optimally sampling about the mean and propagating each sample through $f$ and combining the results as a weighted sum (Julier and Uhlmann, 1997). These so-called sigma points are defined as:

$$\chi_i = \begin{cases} 
E[x] & i = 0 \\
E[x] + \sqrt{(n + \eta) \text{cov}[x]} & i = 1, \ldots, n \\
E[x] - \sqrt{(n + \eta) \text{cov}[x]} & i = n + 1, \ldots, 2n
\end{cases}$$

Note that $[\cdot]_i$ indicates the $i^{th}$ column of a matrix. $n$ describes the dimension of the random variable $x$ and $\eta$ is a scaling parameter, defined such that $\eta = \alpha^2(n + \kappa) - n$.

The unscented transform consists of transforming each sigma point, $\gamma_i = f(\chi_i)$ and constructing a weighted sum. The approximation of $y = f(x)$ is given by $y \sim \mathcal{N}(\mu, \Sigma)$, where

$$\mu = \sum_{i=0}^{2n} \omega_i^{(m)} \gamma_i,$$
$$\Sigma = \sum_{i=0}^{2n} \omega_i^{(c)} (\mu - \gamma_i)(\mu - \gamma_i)^\top.$$

The weights are defined by

$$\omega_0^{(m)} = \eta(n + \eta)^{-1},$$
$$\omega_0^{(c)} = \eta(n + \eta + 1 - \alpha^2 + \beta),$$
$$\omega_i^{(m)} = \omega_i^{(c)} = (2n + 2\eta)^{-1},$$

where $\alpha$, $\beta$, and $\kappa$ are hyperparameters controlling the spread of sigma points. There are a number of reported settings for values of these hyperparameters, one such is $\alpha = 1$, $\beta = 0$, and $\kappa = n$ (Julier and Uhlmann, 1997). These are the values used in this paper.

Implementation

The implementation was written in TensorFlow, using TensorFlow probability. All experiments were optimised using Adam with a learning rate of $5e^{-3}$. For additional numerical stability, gradient clipping was performed based on the global norm.

The unscented filtering updates were implemented using the sigma-point dynamics as described in Sarkka (2007).

The number of flows for each experiment was set to $2d$, where $d$ is the latent state dimension.