Mth 610: Final Project (optional)

Extended Kalman filter for the Lorenz '63 model

Due by noon on June 10

This project is to implement the extended Kalman filter (EKF) for the Lorenz '63 model [1]

$$
\begin{aligned}
x'(t) &= \sigma(y-x) \qquad (1)\\
y'(t) &= \rho x - y - xz \qquad (2)\\
z'(t) &= xy - \beta z \qquad (3)
\end{aligned}
$$

At any given time t, the model state is the three-dimensional ($n=3$) vector $\mathbf{x}(t) = [x(t)\, y(t)\, z(t)]^{\mathrm{T}}$ and we write the system (1)-(3) in a compact form as [2]

$$\mathbf{x}'(t) = \mathbf{f}(\mathbf{x}, t, \mathbf{p}) \qquad (4)$$

where $\mathbf{p} = [\sigma,\ \rho,\ \beta]$ denotes the vector of model parameters. The parameter values are specified as

$$\sigma = 10, \rho = 28, \beta = 8/3$$

and the initial state is taken as

$$\mathbf{x}(0) = [1 \quad 1 \quad 1]^{\mathrm{T}} \qquad (5)$$

A discrete solution $\mathbf{x}_i, i = 0, 1, \ldots$ to the initial value problem (4-5) is obtained using numerical integration with the 4^{th} order Runge-Kutta method ("the Runge-Kutta method") to advance the state in time from t_i to $t_{i+1} = t_i + h$,

$$
\begin{aligned}
\mathbf{k}_1 &= h\mathbf{f}(\mathbf{x}_i, t_i, \mathbf{p}) \qquad (6)\\
\mathbf{k}_2 &= h\mathbf{f}\left(\mathbf{x}_i + 0.5\mathbf{k}_1, t_i + 0.5h, \mathbf{p}\right) \qquad (7)\\
\mathbf{k}_3 &= h\mathbf{f}\left(\mathbf{x}_i + 0.5\mathbf{k}_2, t_i + 0.5h, \mathbf{p}\right) \qquad (8)\\
\mathbf{k}_4 &= h\mathbf{f}\left(\mathbf{x}_i + \mathbf{k}_3, t_i + h, \mathbf{p}\right) \qquad (9)\\
\mathbf{x}_{i+1} &= \mathbf{x}_i + \frac{1}{6}\left(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4\right) \qquad (10)
\end{aligned}
$$

Integration with a time step $\bar{h} = 0.01$ provides the true state trajectory in the time interval $0 \le t \le t_{end}$. We take $t_{end} = 100$, thus $ndt = 10000$ time steps are taken. *Henceforth, we refer to this state trajectory as the "true state" of the dynamical system.* A plot of the 3D state trajectory is shown in Fig. 1.

A model to the true state evolution (henceforth, "the model") is obtained by numerical integration with a time step $h = 10\bar{h} = 0.1$

$$\hat{\mathbf{x}}_{i+1} = \mathcal{M}_i(\hat{\mathbf{x}}_i, \mathbf{p}, h), \quad i = 0, 1, \ldots \qquad (11)$$

where $\mathcal{M}_i(\hat{\mathbf{x}}_i)$ is defined by the sequence of computational steps (stages) (6-10). Notice that $N = 1000$ time steps are taken in the time interval $0 \le t \le t_{end}$ such that the model trajectory approximates the true state on a coarse time scale (the model step h corresponds to 10 steps $\bar{h}$ of the true state integration.)

The evolution of the true state from t_i to $t_{i+1} = t_i + h$ is represented as

$$\mathbf{x}_{i+1} = \mathcal{M}_i(\mathbf{x}_i, \mathbf{p}, h) + \mathbf{w}_i$$

where $\mathbf{w}_i \in \mathbb{R}^n$ is an unknown model error induced by the increased time step $h = 10\bar{h}$ in the model integration.

---

[1] Edward N. Lorenz (1963). Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, Vol. 20, p130–141.

[2] Notice that the system is autonomous, $\mathbf{f}(\mathbf{x}, t, \mathbf{p}) = \mathbf{f}(\mathbf{x}, \mathbf{p})$

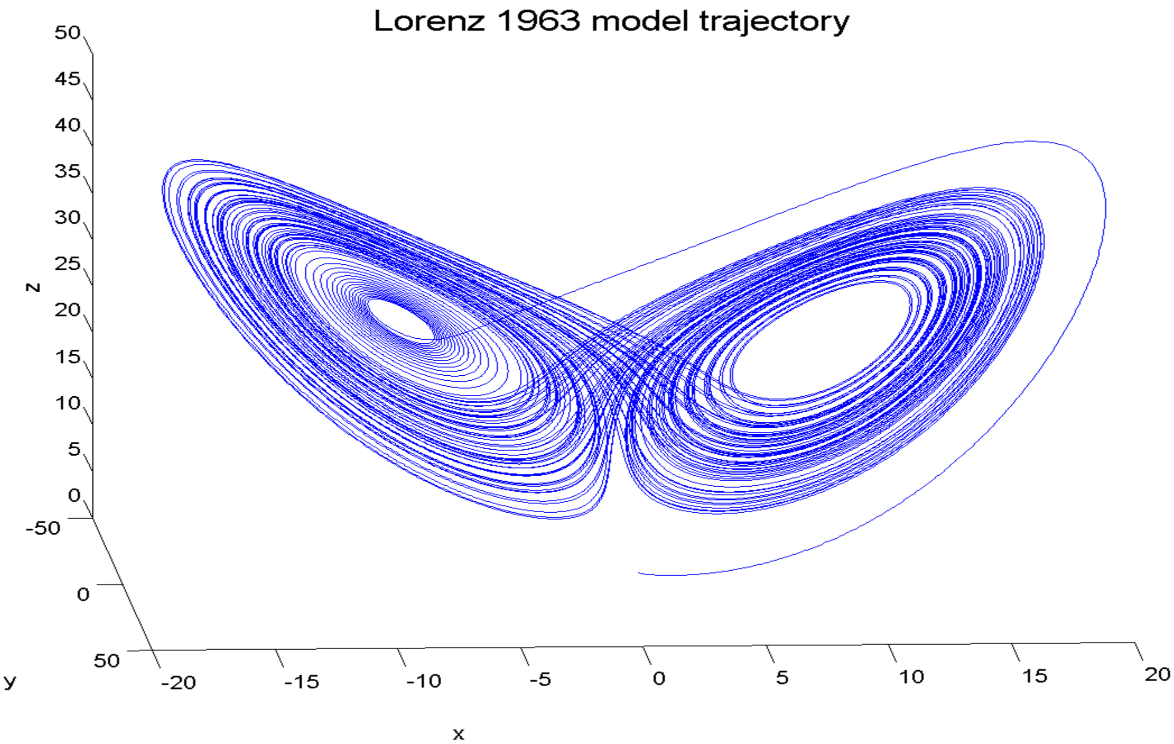

Figure 1: *Illustration of the true state trajectory for the the Lorenz '63 model.* $\sigma = 10, \rho = 28, \beta = 8/3$, $\mathbf{x}(0) = [1,\ 1,\ 1]$

Information on the true state $\mathbf{x}_i$ is obtained through state measurements. Observations are taken as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \epsilon_o \in \mathbb{R}^m, \quad (m = 1\, or\, m = 2\, or\, m = 3) \tag{12}$$

where $\mathbf{H} \in \mathbb{R}^{m\times n}$ denotes the observation matrix and $\epsilon_o \in \mathbb{R}^m$ is a random vector of observational errors.

Our goal is to implement the EKF to estimate and predict the true state $\mathbf{x}$ based on the model (11) and observations (12). The setup is as follows.

- A prior guess to the true initial state at $t = 0$ is prescribed as

  $$\hat{\mathbf{x}}_0 = \mathbf{x}(0) + \epsilon_b$$

  where the initial condition error in each component is $\epsilon_b \sim N(0, \sigma_b^2)$ with $\sigma_b = 1$.
- The observational error is specified as $\epsilon_o \sim N(0, \sigma_o^2)$ where $\sigma_o = 0.1$.
- Each component of the model error is specified as $w_i \sim N(0, \sigma_q^2)$ where $\sigma_q = 0.1$
- The time interval for the analysis it $0 \le t \le 100$, that is $i = 0 : 999$ in (11).

The EKF algorithm is formulated as follows:

$x_0^a = x(0) + \epsilon_b$ % true initial state plus random error taken from $N(0,1)$
$\mathbf{P}^a = \mathbf{I}$ % initial analysis error covariance matrix is set to identity

for $i = 0 : ndt - 1$

% forecast step

$x_{i+1}^f = \mathcal{M}_i(x_i^a, p, h)$

$\mathbf{P}^f = \mathbf{M}_i(x_i^a, p, h)\mathbf{P}^a\mathbf{M}_i^{\mathrm{T}}(x_i^a, p, h) + \mathbf{Q}$ % forecast error covariance

% use a diagonal matrix for the model error covariance $\mathbf{Q} = \sigma_q^2\mathbf{I}$

% generate observation(s) at t_{i+1}

$y = \mathbf{H}x(t_{i+1}) + \epsilon_o$ % generate observation(s) at t_{i+1} of true state plus random errors taken from $N(0, \sigma_o^2)$

% analysis step
$x_{i+1}^a = x_{i+1}^f + \mathbf{K}(y - \mathbf{H}x_{i+1}^f)$ % where $\mathbf{K}$ is the Kalman gain matrix

$\mathbf{P}^a = (\mathbf{I} - \mathbf{KH})\mathbf{P}^f$ % analysis error covariance at t_{i+1}
end % end for loop

Remark 1: Notice that to properly generate observations you will need to properly extract the corresponding true state, the easiest way is by re-defining $\mathbf{x} = \mathbf{x}(1 : 10 : 10000)$.

Remark 2: $\mathbf{M}_i(x_i^a, p, h) \in \mathbb{R}^{3\times3}$ denotes the Jacobian matrix of the model (11) i.e., $\dfrac{\partial \mathbf{x}_{i+1}}{\partial \mathbf{x}_i}$ derived from the computational steps (6-10).

Your job:

- Investigate the performance of the discrete model in pure forecast mode.
- Implement the EKF algorithm, as described above. Provide a qualitative study of the analysis and forecast errors in the time interval $0 \le t \le 100$, that is for $N = 1000$ time steps. Consider 3 cases with observations provided as follows:
  - Case 1: all states observed: x, y, z (m=3)
  - Case 2: observed states x, y only (m=2)
  - Case 3: observed state x only (m=1)

  Include plots for the analysis errors $\mathbf{x}^a - \mathbf{x}$ and forecast errors $\mathbf{x}^f - \mathbf{x}$ in each state component and for the fit to data: observed-minus-analysis (oma) $\mathbf{y} - \mathbf{H}\mathbf{x}^a$ and observed-minus-forecast (omf) $\mathbf{y} - \mathbf{H}\mathbf{x}^f$. Use a 100-step moving average to display results, log scale, etc ...