Final Project

Statistical Contraction Theory for Learning-based Control of Nonlinear Systems

- You are welcome to collaborate, but you must write up your own solutions.
- You can use any programming language, but you must add comments to each section of the codes to clearly explain your intent.
- If you use any resources other than your brain and lecture notes to solve the problems, you must cite them.
- In the following, $\|\cdot\|$ denotes the Euclidean norm for vectors and the induced 2-norm for matrices, i.e., $\|\cdot\| = \|\cdot\|_2$, unless otherwise noted.

Option 1: A Dynamical System of Your Choice

To support the course objective of developing meaningful future research projects and papers, you are encouraged to work on a challenging dynamical system relevant to your interests/ongoing projects, which can be expressed in the following form:

$$\dot{x} = \underbrace{f(x,u,t)}_{\text{known part of dynamics}} + \underbrace{d(x,u,t)}_{\text{unknown part of dynamics}} \tag{1}$$

where t denotes time, x denotes the state, and u denotes the control input. Stochastic and discrete-time/hybrid dynamics are also welcome if necessary. The following questions give some guidelines for your project.

1. Define your dynamics as in (1).
2. Design a contraction condition for the known part of (1).
3. Use conformal inference to quantify the unknown part of (1).
4. Provide a theoretical argument that the system is contracting, or can be made contracting, and validate it via numerical simulations.
5. Discuss potential novelty in your approach and challenges encountered in the process.

Option 2: State Estimation around Unknown Astronomical Body

1 Implementation

This problem is optional. Consider the following dynamical system of a spacecraft orbiting around an unknown astronomical body:

$$\dot{\text{æ}}_c = f(\text{æ}_c) + \zeta(\text{æ}_c) = \begin{bmatrix} v \\ -\frac{\mu}{r^2} + \frac{h^2}{r^3} \\ 0 \\ 0 \\ 0 \\ \frac{h}{r^2} \end{bmatrix} + \begin{bmatrix} 0 \\ \zeta_v(\text{æ}_c) \\ \zeta_h(\text{æ}_c) \\ \zeta_\Omega(\text{æ}_c) \\ \zeta_i(\text{æ}_c) \\ \zeta_\theta(\text{æ}_c) \end{bmatrix} \tag{2}$$

$$y = H\text{æ}_c = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \text{æ}_c = \begin{bmatrix} r \\ \Omega \\ i \\ \theta \end{bmatrix}$$

where $\mu = 1$, $æ_c = [r, v, h, \Omega, i, \theta]^\top$ is the spacecraft state, y is the measurement, $\zeta(æ_c) = [\zeta_v(æ_c), \zeta_h(æ_c), \zeta_\Omega(æ_c), \zeta_i(æ_c), \zeta_\theta(æ_c)]^\top$ represents the uncertain part of the dynamics, and the definitions of all the symbols are given in Table 1.

For all the numerical integration in this problem, always use the discretization time interval $\Delta t = 0.001$ and the initial state $[r(0), v(0), h(0), \Omega(0), i(0), \theta(0)] = \left[0.2, 0, \sqrt{\mu r(0)}, \frac{\pi}{3}, \frac{\pi}{3}, 0\right]$. This system is already implemented in `unknown_planet.py`.

1. By running `unknown_planet.py`, you can numerically integrate the following learned dynamics:

$$\dot{æ}_\ell = f(æ_\ell) + \begin{bmatrix} 0 \\ \zeta_{nn}(æ_\ell) \end{bmatrix}$$

   where f is given by (2) and ζ_{nn} denotes the trained neural network. The residual $\begin{bmatrix} 0 \\ \zeta_{nn}(æ_\ell) \end{bmatrix}$ can be computed by the function `res_dyn` of `unknown_planet.py`. Assuming that f is Lipschitz and $\|\zeta(æ_c) - \zeta_{nn}(æ_\ell)\|$ is upper-bonded, provide a <u>brief</u> theoretical justification for the numerical behavior of $æ_\ell$ in the simulation. Detailed derivation of the estimation error is not required.

2. Numerically integrate the following nominal estimator dynamics without ζ:

$$\dot{\hat{æ}} = f(\hat{æ}) + L(\hat{æ})(y - H\hat{æ}) \tag{3}$$

   where f, y, and H are given in (2) and $L(\hat{æ})$ is computed by the following EKF-like formulaiton:

$$\begin{aligned} &L(\hat{æ}) = \Sigma(\hat{æ})H^\top R^{-1} \\ &A(\hat{æ})\Sigma(\hat{æ}) + \Sigma(\hat{æ})A(\hat{æ})^\top - \Sigma(\hat{æ})H^\top R^{-1} H\Sigma(\hat{æ}) + Q = 0 \end{aligned} \tag{4}$$

   with $A(\hat{æ})$ being the partial derivative of f evaluated at $\hat{æ}$, $\Sigma(\hat{æ}) \succ 0$, $Q = \mathbb{I}$, and $R = \mathbb{I}$. Note that $\mathbb{I}$ are the identity matrices of appropriate dimensions.

   Also, plot the time evolution of the spacecraft's horizontal and vertical positions, $(p_x, p_y) = (r\cos\theta, r\sin\theta)$, of the estimated state $\hat{æ}$ of (3) and of the true state $æ_c$ given in `Xtrue.npy` in a single figure. (Hint: $\Sigma(\hat{æ})$ in (4) can be computed by `control.lqr` of the Python Control Systems Library using the duality of estimation.)

3. Let $q(\eta, t)$ be a smooth path parameterized by $\eta \in [0, 1]$, which connects $æ_c(t)$ and $\hat{æ}(t)$ to have $q(0, t) = æ_c(t)$ and $q(1, t) = \hat{æ}(t)$. Roughly speaking, locally, the dynamics of the estimation error of 2 can be described by the following virtual dynamical system of q:

$$\begin{aligned} \dot{q} &= f(q) + \Sigma(\hat{æ})H^\top R^{-1}(y - Hq) + (1 - \eta)\begin{bmatrix} 0 \\ \zeta(æ_c) \end{bmatrix} \\ \dot{\Sigma}(\hat{æ}) &= \frac{\partial f(q)}{\partial q}\Sigma(\hat{æ}) + \Sigma(\hat{æ})\frac{\partial f(q)}{\partial q}^\top - \Sigma(\hat{æ})H^\top R^{-1} H\Sigma(\hat{æ}) + Q. \end{aligned} \tag{5}$$

   Assuming that $\|\zeta(æ_c)\|$ and $\|\Sigma(\hat{æ})\|$ are upper-bonded and $\Sigma(\hat{æ}) \succ 0$, use (5) to provide a <u>brief</u> theoretical justification for the result of 2. Detailed derivation of the estimation error is not required.

4. Numerically integrate the following estimator dynamics now with the neural net approximation ζ_{nn}:

$$\dot{\hat{æ}}_\ell = f(\hat{æ}_\ell) + L(\hat{æ}_\ell)(y - H\hat{æ}_\ell) + \begin{bmatrix} 0 \\ \zeta_{nn}(\hat{æ}_\ell) \end{bmatrix} \tag{6}$$

   where f, y, and H are given in (2) and $L(\hat{æ}_\ell)$ is computed by the following EKF-like formulaiton:

$$\begin{aligned} &L(\hat{æ}_\ell) = \Sigma(\hat{æ}_\ell)H^\top R^{-1} \\ &A(\hat{æ}_\ell)\Sigma(\hat{æ}_\ell) + \Sigma(\hat{æ}_\ell)A(\hat{æ}_\ell)^\top - \Sigma(\hat{æ}_\ell)H^\top R^{-1} H\Sigma(\hat{æ}_\ell) + Q = 0 \end{aligned}$$

   with $A(\hat{æ}_\ell)$ being the partial derivative of f evaluated at $\hat{æ}_\ell$, $\Sigma(\hat{æ}_\ell) \succ 0$, $Q = \mathbb{I}$, and $R = \mathbb{I}$. Note that $\mathbb{I}$ are the identity matrices of appropriate dimensions.

   Also, plot the time evolution of the spacecraft's horizontal and vertical positions, $(p_x, p_y) = (r\cos\theta, r\sin\theta)$, of the estimated state of $\hat{æ}_\ell$ (6) and of the true state $æ_c$ given in `Xtrue.npy` in a single figure.

2 Theoretical Justification

This problem aims to provide a theoretical justification for the dynamical behavior of the estimator in (6). Let $q(\eta, t)$ be a smooth path parameterized by $\eta \in [0,1]$, which connects $\text{æ}_c(t)$ and $\hat{\text{æ}}_\ell(t)$ to have $q(0,t) = \text{æ}_c(t)$ and $q(1,t) = \hat{\text{æ}}_\ell(t)$. Roughly speaking, locally, the dynamics of the estimation error with (6) can be described by the following virtual dynamical system of q:

$$
\begin{aligned}
\dot{q} &= f(q) + \Sigma(\hat{\text{æ}}_\ell)H^\top R^{-1}(y - Hq) + (1-\eta)\begin{bmatrix} 0 \\ \zeta(\text{æ}_c) \end{bmatrix} + \eta \begin{bmatrix} 0 \\ \zeta_{nn}(\hat{\text{æ}}_\ell) \end{bmatrix} \\
\dot{\Sigma}(\hat{\text{æ}}_\ell) &= \frac{\partial f(q)}{\partial q}\Sigma(\hat{\text{æ}}_\ell) + \Sigma(\hat{\text{æ}}_\ell)\frac{\partial f(q)}{\partial q}^\top - \Sigma(\hat{\text{æ}}_\ell)H^\top R^{-1}H\Sigma(\hat{\text{æ}}_\ell) + 2\alpha\Sigma(\hat{\text{æ}}_\ell)
\end{aligned}
\tag{7}
$$

with $\alpha > 0$. Throughout this problem, we will assume that (7) holds globally for simplicity. Note that its differential dynamical is given by

$$
\frac{d}{dt}\left(\frac{\partial q}{\partial \eta}\right) = \frac{\partial f}{\partial q}\frac{\partial q}{\partial \eta} - \Sigma(\hat{\text{æ}}_\ell)H^\top R^{-1}H\frac{\partial q}{\partial \eta} - \begin{bmatrix} 0 \\ \zeta(\text{æ}_c) \end{bmatrix} + \begin{bmatrix} 0 \\ \zeta_{nn}(\hat{\text{æ}}_\ell) \end{bmatrix}.
$$

1. Assuming that $\left\|\Theta(\hat{\text{æ}}_\ell)\frac{\partial q}{\partial \eta}\right\| \neq 0$ for $\Sigma(\hat{\text{æ}}_\ell)^{-1} = \Theta(\hat{\text{æ}}_\ell)^\top\Theta(\hat{\text{æ}}_\ell) \succ 0$ with $\Theta(\hat{\text{æ}}_\ell) \succ 0$, derive an upper bound of $\frac{d}{dt}\left(\int_0^1 \left\|\Theta(\hat{\text{æ}}_\ell)\frac{\partial q}{\partial \eta}\right\| d\eta\right)$. The final upper bound should be written in terms of α, $\|\Theta(\hat{\text{æ}}_\ell)\|$, $\|\zeta_{nn}(\hat{\text{æ}}_\ell) - \zeta(\text{æ}_c)\|$, and $\int_0^1 \left\|\Theta(\hat{\text{æ}}_\ell)\frac{\partial q}{\partial \eta}\right\| d\eta$. (Hint: Use $V = \frac{\partial q}{\partial \eta}^\top \Sigma^{-1}\frac{\partial q}{\partial \eta}$ and compute $\dot{V}$.)

2. The prediction error term can be bounded as

$$
\|\zeta_{nn}(\hat{\text{æ}}_\ell) - \zeta(\text{æ}_c)\| \leq \|\zeta_{nn}(\text{æ}_c) - \zeta(\text{æ}_c)\| + \|\zeta_{nn}(\hat{\text{æ}}_\ell) - \zeta_{nn}(\text{æ}_c)\|
$$

   Suppose that our neural network ζ_{nn} is Lipschitz, i.e., $\exists L_{nn} > 0$ s.t.

$$
\|\zeta_{nn}(\hat{\text{æ}}_\ell) - \zeta_{nn}(\text{æ}_c)\| \leq L_{nn}\int_0^1 \left\|\frac{\partial q}{\partial \eta}\right\| d\eta. \tag{8}
$$

   Suppose also that we have trajectory data samples $\{(\text{æ}_{c,i}(t), \zeta(\text{æ}_{c,i}(t))\}_{i=1}^N,\ t \in [0,T]$, which are exchangeable with each other and with unseen real-world trajectories, where $T > 0$. Let the prediction nonconformity scores be defined as

$$
S_i = \sup_{t\in[0,T]} \|\zeta_{nn}(\text{æ}_{c,i}(t)) - \zeta(\text{æ}_{c,i}(t))\|,\ i = 1,\cdots,N.
$$

   Using conformal inference and (8), derive a probabilistic upper bound of $\sup_{t\in[0,T]}\|\zeta_{nn}(\hat{\text{æ}}_\ell(t)) - \zeta(\text{æ}_c(t))\|$, i.e., find C of the following inequality:

$$
\mathbb{P}\left[\sup_{t\in[0,T]}\|\zeta_{nn}(\hat{\text{æ}}_\ell(t)) - \zeta(\text{æ}_c(t))\| \leq C\right] \geq 1 - \delta.
$$

   where $\delta \in (0,1)$. The bound C should be written in terms of L_{nn}, $\int_0^1 \left\|\frac{\partial q}{\partial \eta}\right\| d\eta$, δ, N, and $S_i,\ i = 1,\cdots,N$

3. Let $\underline{m} > 0$ and $\overline{m} > 0$ be the uniform lower and upper bounds of $\Sigma(\hat{\text{æ}}_\ell)^{-1}$, i.e., $\underline{m}\mathbb{I} \preceq \Sigma(\hat{\text{æ}}_\ell)^{-1} \preceq \overline{m}\mathbb{I},\ \forall\hat{\text{æ}}$. Assuming that $\alpha - L_{nn}\sqrt{\frac{\overline{m}}{\underline{m}}} > 0$, derive a probabilistic upper bound of $\int_0^1 \left\|\Theta(\hat{\text{æ}}_\ell)\frac{\partial q}{\partial \eta}\right\| d\eta$, i.e., find C of the following inequality:

$$
\mathbb{P}\left[\int_0^1 \left\|\Theta(\hat{\text{æ}}_\ell(t))\frac{\partial q(\eta,t)}{\partial \eta}\right\| d\eta \leq C,\ \forall t \in [0,T]\right] \geq 1 - \delta. \tag{9}
$$

   The bound C should be written in terms of α, L_{nn} and $\int_0^1 \left\|\Theta(\hat{\text{æ}}_\ell(0))\frac{\partial q(\eta,0)}{\partial \eta}\right\| d\eta$, $\underline{m}$, $\overline{m}$, δ, N, and $S_i,\ i = 1,\cdots,N$.

4. Using the result of (9), provide a theoretical justification for the numerical behavior of $\text{æ}_\ell(t)$ in the simulation.

Table 1: Summary of the symbols in (2)

Symbol	Description
$œ_c$	State of spacecraft, $œ_c = [r, v, h, \Omega, i, \theta]^\top$
r	Radial distance from center of target body
v	Radial velocity
h	Specific angular momentum
Ω	Right ascension of ascending node
i	Inclination of orbit
θ	True anomaly
μ	Gravitational constant of target body