Common Task Framework: Evaluation Metrics and Challenges for Dynamics [Notes]

J. Nathan Kutz
Applied Mathematics and Electrical and Computer Engineering University of Washington, Seattle, WA

1 Introduction

The *common task framework* (CTF) for dynamic systems aims to evaluate algorithms and methods on a variety of tasks that are common for engineering and science. The goals include forecasting and reconstruction of time-series and spatio-temporal data under the challenges of limited data, noise and parametric dependence.

What will be provided to challengers is an HD5 file which includes 11 matrices

$$\mathbf{X}_j \in \mathbb{R}^{n\times m} \quad j = 1, 2, \cdots, 11 \tag{1}$$

where

$$\begin{aligned} n &= \text{ dimension of dynamical system} \\ m &= \text{ number of time points} \\ j &= \text{ parameter regime for the } j\text{th matrix.} \end{aligned}$$

Thus, the rows represent the dimension of the system under consideration and the columns are the sequential temporal sampling of the dynamics.

Twelve tasks and evaluations are assigned to the challengers, with a matrix produced for each one, put together into a single zipped filed with numpy arrays. Each matrix will be of the following form

$$\hat{\mathbf{X}}_{Jtest} \in \mathbb{R}^{n\times m} \quad J = 1, 2, \cdots, 12 \tag{2}$$

where n (dimension or dynamics) and m (number of time points) will be specified for each of the 12 evaluations. The users approximation to the test set will be $\mathbf{X}_{Jtest}$ so that $\mathbf{X}_{Jtruth}$ and $\mathbf{X}_{Jtest}$ will be compared in the evaluation metrics.

SCORING

Scoring will be on a scale with 100 being a perfect score and a score of zero corresponding to a guess of zeros for $\mathbf{X}_{Jtest}$. Negative scores will show that the model is worse than guessing zeros. In summary, we have

$$\begin{aligned} \text{SCORE} = 100: &\quad \text{Perfect match between test set and truth} \\ \text{SCORE} = 0: &\quad \text{Score if the model simply guesses zeros for the test set} \end{aligned}$$

Thus, the mechanics of *each* challenge, three dynamical systems (Lorenz, Rössler, double-pendulum) and three spatio-temporal systems (Kuramoto-Sivashinsky, Lorenz96, Burgers), will provide the user 14 training data sets, with the requirement of 11 approximations to the test sets returned for evaluation. The 14 training sets and 11 approximations to the test set are detailed in an example using the Kuramoto-Sivashinsky model.

2 Example Spatio-Temporal System: Kuramoto-Sivashinsky

The Kuramoto-Sivashinsky (KS) equation is a fourth order, nonlinear partial differential equation. It is considered a canonical example of spatio-temporal chaos in a one-dimensional PDE and is therefore commonly used as a test problem for data-driven algorithms. The KS equation is a particularly challenging case for fitting algorithms due to its combination of high dimensionality, nonlinearity, and sensitivity to initial conditions (chaotic behavior):

$$u_t + uu_x + u_{xx} + \mu u_{xxxx} = 0. \tag{3}$$

Solutions of the (3) are on a grid $[0, 32\pi]$ with periodic boundary conditions. A numerical integrator with unknown time-step Δt will be used to evolve the solution forward m-steps.

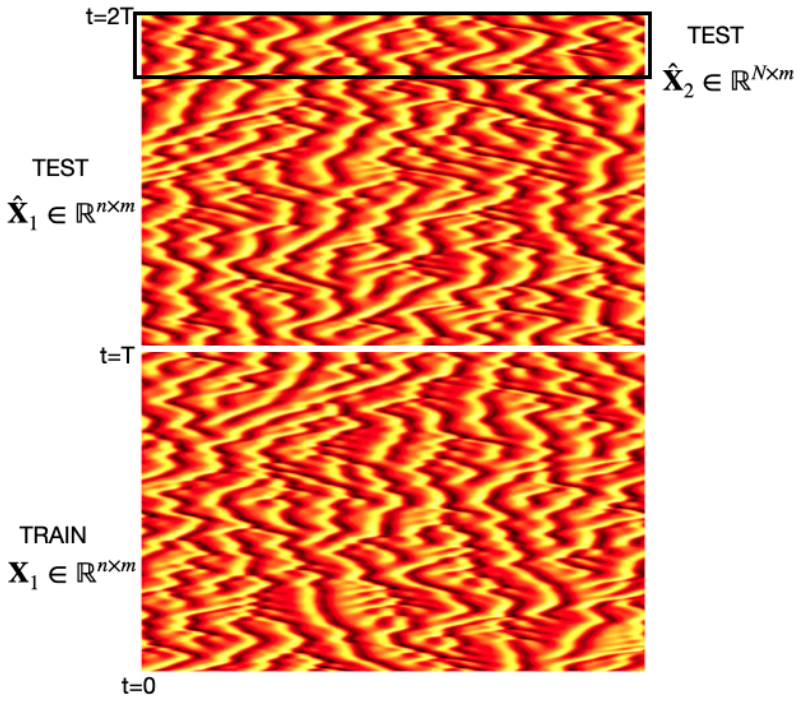


Figure 1: Test 1: The evaluation of forecasting capabilities of a method with numerically accurate data for both short- and long-time forecasting. A single training data set $\mathbf{X}_{1train} \in \mathbb{R}^{n\times m}$ is given and a single forecasting approximation matrix is returned $\mathbf{X}_{1test} \in \mathbb{R}^{n\times m}$. From the forecasting approximation, two error metrics are returned.

2.1 Test 1: Forecasting (2 scores)

The first test of the method, as illustrated in Fig. 1, involves the approximation of the future state of the system. Thus, given a data matrix representing the dynamics from $t \in [0, T]$ ($\mathbf{X}_1 \in \mathbb{R}^{n\times m}$), the forecast requested is from $t \in [T, 2T]$ ($\tilde{\mathbf{X}}_1 \in \mathbb{R}^{n\times m}$). The forecasting score is actually composed of two scores evaluating both the short-time forecast (the "weather forecast") which is computed using root-mean square error fitting between the test set and the users approximation, and the long-term forecast (the "climate forecast"), which is based upon the power spectral density. As such, the following two error scores are computed:

$$E_{\text{ST}} = \frac{\|\hat{\mathbf{X}}_1[:, 1:k] - \tilde{\mathbf{X}}_1[:, 1:k]\|}{\|\hat{\mathbf{X}}_1[:, 1:k]\|} \qquad \text{(weather forecast)} \tag{4}$$

and

$$E_{\text{LT}} = \frac{\|\hat{\mathbf{P}}_1[\mathbf{k}, N-k:N] - \tilde{\mathbf{P}}_1[\mathbf{k}, N-k:N]\|}{\|\hat{\mathbf{P}}_1[\mathbf{k}, N-k:N]\|} \qquad \text{(climate forecast)}. \tag{5}$$

Here, E_{ST} is the short-time error evaluated using least-squares and E_{LT} (See Fig. 2) is the long-time error which is computed by least-squares fitting of the power spectrum $\mathbf{P}_j[:, k] = \ln(|\text{FFT}(\mathbf{X}_j[:, k])|^2)$, where the **fftshift** has been used to model the data in the wavenumber domain and $\mathbf{k} = n/2 - k_m : n/2 + (k_m + 1)$ with $k_m = 100$. This then looks at the first 100 wavenumbers in order to determine the decay rate of the power spectrum. It is clear that there are many ways to evaluate the long-range forecasting capabilities. However, we have chosen a simple metric fully understanding that more nuanced scoring could be used.

The first error metric evaluates the overall forecasting skill for short-time prediction:

$$E_1 = 100(1 - E_{\text{ST}}). \tag{6}$$

Note that as a baseline a solution guess of zeros $\tilde{\mathbf{X}}_1[:, 1:k] = \mathbf{0}$ gives a score of $E_1 = 0$. The second error metric evaluates the long-range capabilities of an algorithm for matching the correct long-term behavior of

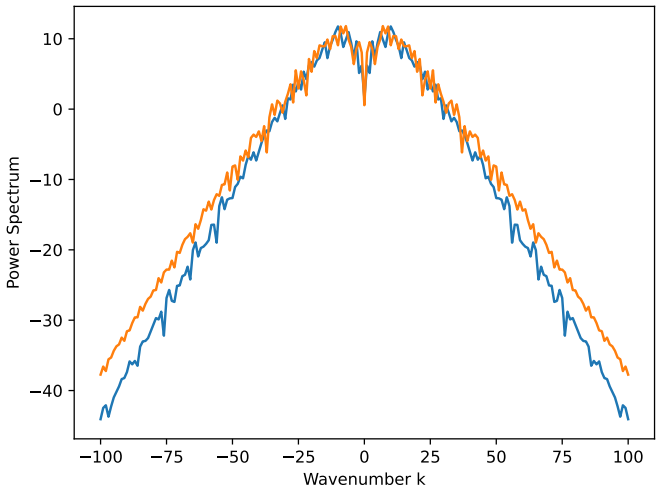


Figure 2: Power spectrum of the true solution versus an approximation. The long-term forecasting is evaluated by looking at the least-square difference between these two power spectra. Only the first k_m modes (here $k_m = 100$) are used to quantify the error.

the system

$$E_2 = 100(1 - E_{\text{LT}}). \tag{7}$$

Note that as a baseline a solution guess of zeros $\tilde{\mathbf{P}}_1[\mathbf{k}, N-k : N] = \mathbf{0}$ gives a score of $E_2 = 0$. For the challenge dynamics of interest, sensitivity of initial conditions is common so that long range forecasting to match the test set is not a reasonable task given fundamental mathematical limitations with Lyapunov times.

Input: $\mathbf{X}_{1train} \in \mathbb{R}^{n\times m}$
Output: $\mathbf{X}_{1test} \in \mathbb{R}^{n\times m}$
Scores: E_1, E_2

2.2 Test 2: Noisy Data (4 scores)

The ability to handle noise is critical in all data-driven applications as sensors and measurement technologies are by default embedded with varying levels of noise. Methods that work with numerically accurate data, for example data points that are 10^{-6} accurate, may be useful for model reduction, but are rarely suitable for discovery and engineering design from real-world data. Both strong and weak noise are considered as these represent realistic challenges to be addressed in practice.

Figures 3 and 4 demonstrate the challenges to be addressed. The challenge is very similar to Test 1, but now with noise added to the data. Specifically, what is given to the challenger is a data matrix $\mathbf{X}_{2train} \in \mathbb{R}^{n\times m}$ and $\mathbf{X}_{3train} \in \mathbb{R}^{n\times m}$ representing the evolution over with medium or high noise. The objective is to first produce a reconstruction of the data itself, i.e. denoise the data to produce an estimate of the true state of the dynamics, and the second objective is to then forecast the future state, matrices $\mathbf{X}_{2test} \in \mathbb{R}^{n\times m}$ and $\mathbf{X}_{3test} \in \mathbb{R}^{n\times m}$ for $\mathbf{X}_{2train}$ and matrices $\mathbf{X}_{4test} \in \mathbb{R}^{n\times m}$ and $\mathbf{X}_{5test} \in \mathbb{R}^{n\times m}$ for $\mathbf{X}_{3train}$. For the first task, a least-square fit is used between the approximation of the denoised data and the truth, which is given by E_1 for all snapshots m. The forecasting score is given by the error metric E_2. Thus, two error scores E_3 (reconstruction) and E_4 (forecast) are produce for medium noise, and two error scores E_5 (reconstruction) and E_6 (forecast) are produce for high noise.

Input: $\mathbf{X}_{2train} \in \mathbb{R}^{n\times m}$, $\mathbf{X}_{3train} \in \mathbb{R}^{n\times m}$
Output: $\mathbf{X}_{2test} \in \mathbb{R}^{n\times m}$, $\mathbf{X}_{3test} \in \mathbb{R}^{n\times m}$, $\mathbf{X}_{4test} \in \mathbb{R}^{n\times m}$, $\mathbf{X}_{5test} \in \mathbb{R}^{n\times m}$
Scores: E_3, E_4, E_5, E_6

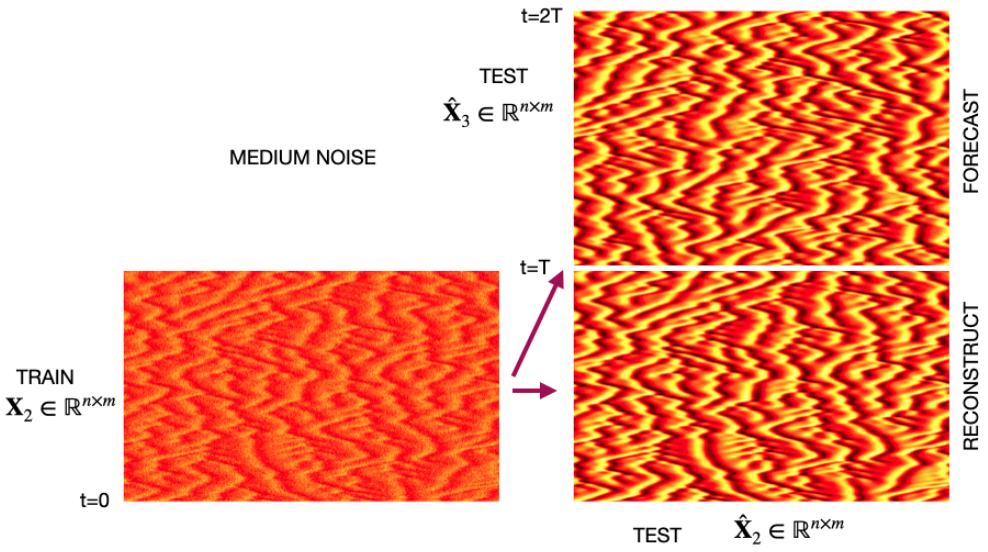


Figure 3: Test 2 (medium noise): The evaluation of the forecasting and reconstruction capabilities of a method with data with medium noise. A single training data set $\mathbf{X}_{2train} \in \mathbb{R}^{n\times m}$ is given and a reconstruction matrix $\mathbf{X}_{2test} \in \mathbb{R}^{n\times m}$ is returned along with a forecasting approximation matrix $\mathbf{X}_{3test} \in \mathbb{R}^{n\times m}$. From these two matrices, the error metrics are computed for the reconstruction using error E_1 and for the forecast using error E_2, these are errors E_3 and E_4 respectively.

2.3 Test 3: Limited Data (2 scores)

Data limitations are also present in many systems, which often change which data-driven architectures are most successful. The low-data limit is critically important in many applications in engineering and science, thus requiring the evaluation of methods under these conditions.

Figure 5 demonstrates the nature of the test. In this case only a limited number of snapshots M on numerically accurate data are given $\mathbf{X}_{4test} \in \mathbb{R}^{n\times M}$. From this limited data, a forecast must be made which is evaluated with the error metrics both E_1 and E_2 on the approximated future $\mathbf{X}_{6test} \in \mathbb{R}^{n\times m}$. The experiment is repeated with noise on the measurements using the training matrix $\mathbf{X}_{5train} \in \mathbb{R}^{n\times M}$ for which a forecasting prediction matrix is produced $\mathbf{X}_{7test} \in \mathbb{R}^{n\times m}$. Two error scores ($E_1$ and E_2) are produced for the noise-free and noisy limited data. These scores are E_7 (short) and E_8 (long) for the noise free case and E_9 (short) and E_10 (long) for the noisy case.

Input: $\mathbf{X}_{4train} \in \mathbb{R}^{n\times M}$, $\mathbf{X}_{5train} \in \mathbb{R}^{n\times M}$
Output: $\mathbf{X}_{6test} \in \mathbb{R}^{n\times m}$, $\mathbf{X}_{7test} \in \mathbb{R}^{n\times m}$
Scores: E_7, E_8, E_9, E_{10}

2.4 Test 4: Parametric Generalization (4 scores)

Finally, the ability of a model to generalize to different parameter values is evaluated. For this case, the model's ability to interpolate and extrapolate to new parameter regimes is considered with noise-free data and noisy data as well. The interpolation and extrapolation are each their own score. This gives a total of four scores that evaluate parametric dependence.

Figure 6 shows the basic architecture of the test. For the noise-free case, three training data sets are provided with three different parameter values $\mathbf{X}_{6train} \in \mathbb{R}^{n\times m}$, $\mathbf{X}_{7train} \in \mathbb{R}^{n\times m}$ and $\mathbf{X}_{8train} \in \mathbb{R}^{n\times m}$. Construction of the dynamics in parametric regimes that are interpolatory $\mathbf{X}_{8test} \in \mathbb{R}^{n\times m}$ and extrapolatory $\mathbf{X}_{9test} \in \mathbb{R}^{n\times m}$ are required. for both of the test regimes, a burn in matrix is given . The error metric E_1 is used to evaluate the reconstructions of the interpolatory and extrapolatory regimes $\mathbf{X}_{9train} \in \mathbb{R}^{n\times M}$ and $\mathbf{X}_{10train} \in \mathbb{R}^{n\times M}$ respectively.

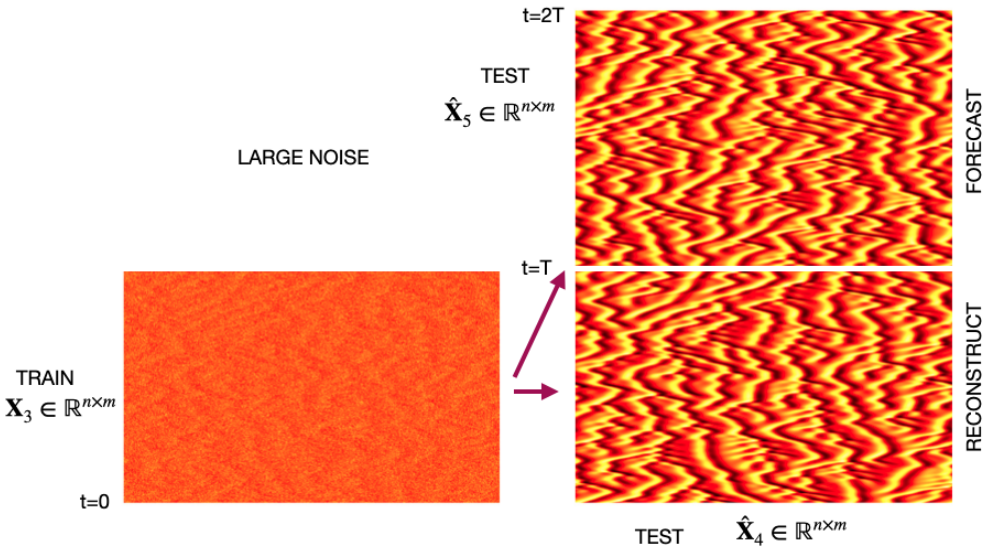


Figure 4: Test 2 (high noise): The evaluation of the forecasting and reconstruction capabilities of a method with data with high noise. A single training data set $\mathbf{X}_{3train} \in \mathbb{R}^{n\times m}$ is given and a reconstruction matrix $\mathbf{X}_{4test} \in \mathbb{R}^{n\times m}$ is returned along with a forecasting approximation matrix $\mathbf{X}_{5test} \in \mathbb{R}^{n\times m}$. From these two matrices, the error metrics are computed for the reconstruction using error E_1 and for the forecast using error E_2, these are errors E_5 and E_6 respectively.

Input: $\mathbf{X}_{6train} \in \mathbb{R}^{n\times m}$, $\mathbf{X}_{7train} \in \mathbb{R}^{n\times m}$, $\mathbf{X}_{8train} \in \mathbb{R}^{n\times m}$, $\mathbf{X}_{9train} \in \mathbb{R}^{n\times M}$, $\mathbf{X}_{10train} \in \mathbb{R}^{n\times M}$
Output: $\mathbf{X}_{8test} \in \mathbb{R}^{n\times m}$, $\mathbf{X}_{9test} \in \mathbb{R}^{n\times m}$
Scores: E_{11}, E_{12}

2.5 Summary Evaluation

To evaluate the overall performance of a method, a radar plot is developed highlighting the various scores associated with the challenge. Figure 7 shows how each method will look overall. It is a profile of the method rather than a single score. Of course, the average of all scores can be computed in order to provide a composite score. But ultimately, different tasks will excel in different areas. Some will do well with noise, others will not. Others might excel in the limited data regime, while being boor under parametric generalization. Profiles are important in order to provide a more comprehensive and well-rounded metric of performance.

3 Example Dynamical System: Lorenz

One of the most influential dynamical systems in history, the Lorenz dynamical system is given by

$$\begin{aligned}\frac{dx}{dt} &= \sigma(y-x)\\ \frac{dy}{dt} &= rx - xz - y\\ \frac{dz}{dt} &= xy - bz\end{aligned}$$

where the parameters $b = 8/3$ and $\sigma = 10$ are typically fixed at these values and r is explored as a bifurcation parameter. With r sufficiently large, chaotic behavior ensues on a strange attractor shown in Fig. 9(a). In 1963, Edward Lorenz, with the help of Ellen Fetter who was responsible for the numerical simulations and figures, and Margaret Hamilton who helped in the initial, numerical computations leading up to the findings of the Lorenz model, developed a simplified mathematical model for atmospheric convection in terms of

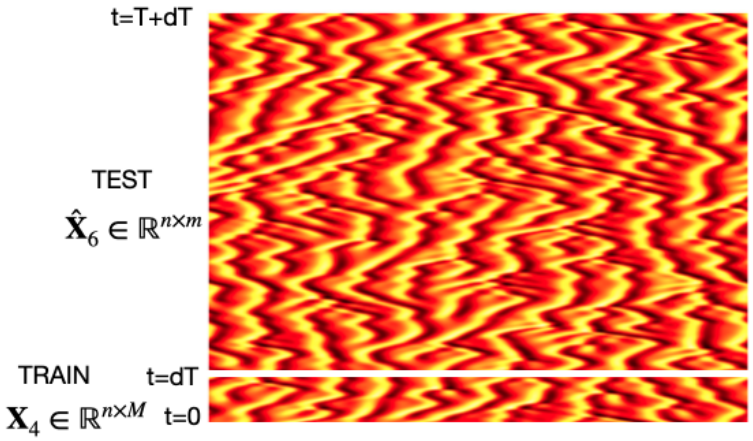


Figure 5: Test 3: The evaluation of forecasting capabilities of a method with limited data for both short- and long-time forecasting. A single training data set $\mathbf{X}_{4train} \in \mathbb{R}^{n\times M}$ where $M \ll m$ is given and a single forecasting approximation matrix is returned $\mathbf{X}_{6test} \in \mathbb{R}^{n\times m}$. This test is done first with noise free data $\mathbf{X}_{4train} \in \mathbb{R}^{n\times M}$, then with noisy data $\mathbf{X}_{5train} \in \mathbb{R}^{n\times M}$ which produces the test approximation $\mathbf{X}_{7test} \in \mathbb{R}^{n\times m}$

the three variables of the Lorenz equations. The equations relate the properties of a two-dimensional fluid layer uniformly warmed from below and cooled from above. In particular, the equations describe the rate of change of three quantities with respect to time: x is proportional to the rate of convection, y to the horizontal temperature variation, and z to the vertical temperature variation. It was this equation that led to the first observations of initial sensitivity to initial conditions, or chaos. Thus, it has a historical significance as the ability to forecast a given system was finally understood to be subject to sensitivity in the initialization.

The training and testing is identical as for the spatio-temporal systems aside from the long range (climate) forecast score. Data matrices for testing and training are now of the form:

$$\mathbf{X}_{Jtrain} \in \mathbb{R}^{n\times m} \quad J = 1, 2, \cdots, 10$$
$$\mathbf{X}_{Jtest} \in \mathbb{R}^{n\times m} \quad J = 1, 2, \cdots, 9$$

where now $n = 3$ is the dimension of the dynamical system. It is then inappropriate to use the power spectral density of the differential equation since it is no longer a spatial system. Instead, the long-time forecasting considers the histogram of the evolution of the variables over the last k time steps (e.g. $k = 1000$). The histogram for a time series is computed using the histogram command with the bins specified. For the x variable in Lorenz, this would give

```
M = np.arange(-20, 21, 1)
yhistxt, xhistx = np.histogram(yt[0, :], bins=M)
```

The difference of the histogram between the truth (x, y and z) and prediction ($\tilde{x}, \tilde{y}$ and $\tilde{z}$) and for each variable is given by

$$E_{\text{LT X}} = \frac{\|Hist_x - Hist_{\tilde{x}}\|}{\|Hist_x\|} \tag{8}$$

This gives the long-time error score:

$$E_{\text{LT}} = (E_{\text{LT X}} + E_{\text{LT Y}} + E_{\text{LT Z}})/3 \qquad \text{(climate forecast)}. \tag{9}$$

As with the spatio-temporal system and the power spectral density, this gives a simple measure of the accuracy of the prediction from a statistical viewpoint since long-time prediction is well beyond the Lyapunov time which would not allow for a least-square match between trajectories of the truth and prediction.

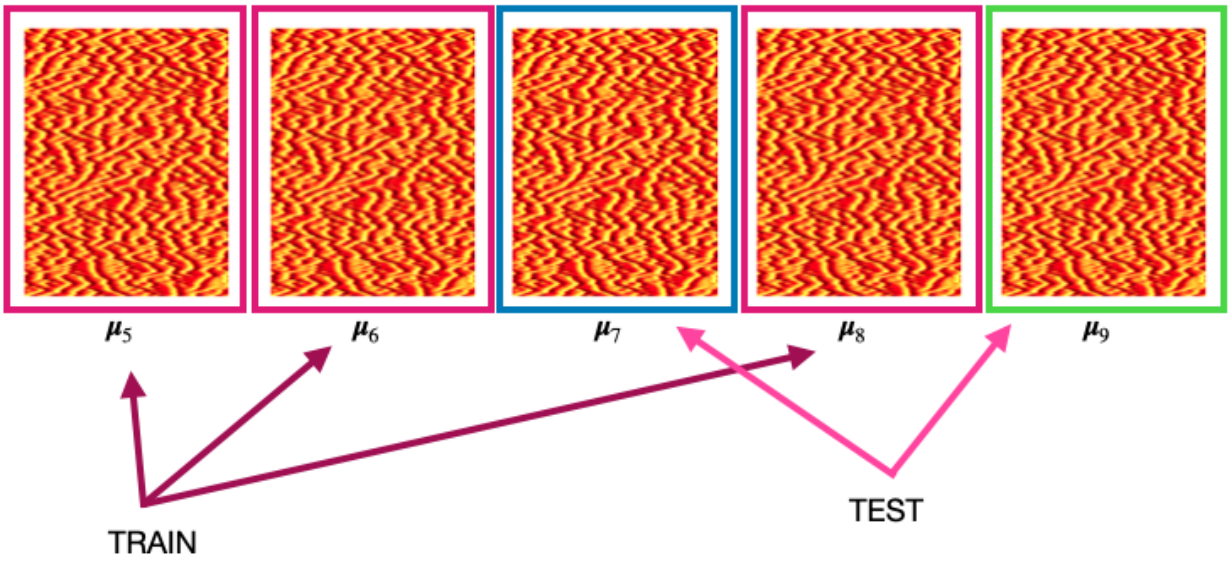


Figure 6: Test 4: The evaluation of the reconstruction capabilities of a method with parametrically dependent data with is evaluated in a noise-free and noisy data environment. Three training data sets are provided with three different parameter values $\mathbf{X}_{6train} \in \mathbb{R}^{n\times m}$, $\mathbf{X}_{7train} \in \mathbb{R}^{n\times m}$ and $\mathbf{X}_{8train} \in \mathbb{R}^{n\times m}$. Construction of the dynamics in parametric regimes that are interpolatory $\tilde{\mathbf{X}}_{6test} \in \mathbb{R}^{n\times m}$ and extrapolatory $\tilde{\mathbf{X}}_{7test} \in \mathbb{R}^{n\times m}$ are required, where an initial time stamp is given $\mathbf{X}_{9train} \in \mathbb{R}^{n\times M}$ and $\mathbf{X}_{10train} \in \mathbb{R}^{n\times M}$.

4 Models: Dynamical Systems

The above example with Kuramoto-Sivashinsky details how methods will be evaluated on a diversity of criteria. A method is also evaluated on a diversity of models: three classical dynamical systems and three classical spatio-temporal systems. The choice of these specific canonical models is due to the fact that they routinely show up in research papers that demonstrate a method's efficacy. As such, the CTF enables challengers a rapid assessment of their method on a diversity of metrics which can then be used in a table for a paper for comparative purposes. Specifically, a permanent leaderboard will be kept for these important example systems in order to help promote and evaluate new methods.

The following subsections highlight the models to be considered for the permanent CTF collection.

4.1 Rössler

The Rössler attractor was intended to behave similarly to the Lorenz attractor, but also be easier to analyze qualitatively. An orbit within the attractor follows an outward spiral close to the $x-y$ plane around an unstable fixed point. Once the graph spirals out enough, a second fixed point influences the trajectory, causing a rise and twist in the z-dimension. In the time domain, it becomes apparent that although each variable is oscillating within a fixed range of values, the oscillations are chaotic. This attractor has some similarities to the Lorenz attractor, but is simpler and has only one manifold. Otto Rössler designed the model in 1976, but the originally theoretical equations were later found to be useful in modeling equilibrium in chemical reactions. The Rössler equations are given by

$$\begin{aligned}\frac{dx}{dt} &= -y - z\\ \frac{dy}{dt} &= x + ay\\ \frac{dz}{dt} &= b + z(x - c)\end{aligned}$$

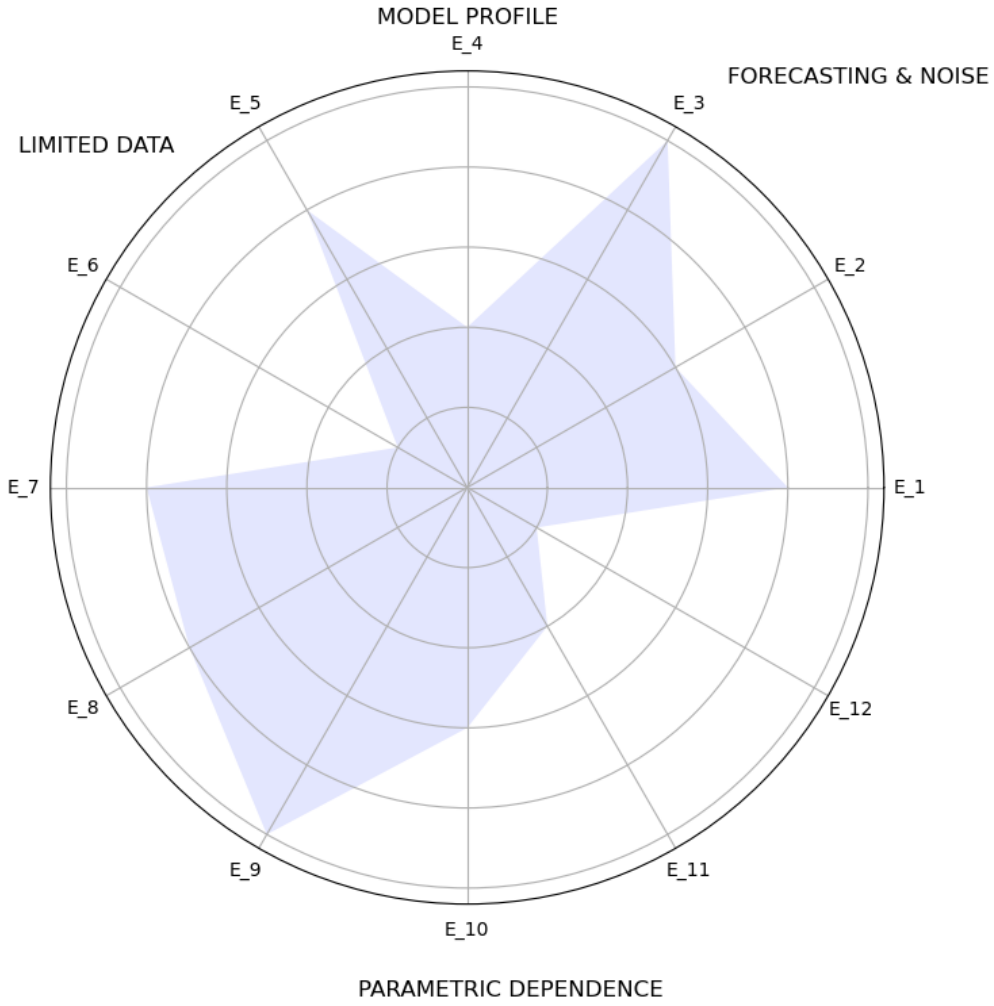


Figure 7: Evaluation of a method's capabilities across the twelve evaluation metrics. The radar plot profiles how well a method does on the various tasks associated with forecasting and reconstruction with noise, limited data and parametric dependency.

and its dynamics are shown in Fig. 9(b). Specifically, a number of trajectories are shown displaying the canonical excursion into the z-direction.

4.2 Double Pendulum

The double pendulum is a classic model of physics as it displays chaos and dynamics in a system that can be easily built and implemented in practice. It is a standard of many physical model demonstrations about chaos and the role of the sensitivity of initial conditions for future state predictions. Figure 9(c)-(e) show the dynamics of the four variables θ_1, $\dot{\theta}_1$, θ_2 and $\dot{\theta}_2$. The equations of motion can be derived from classical

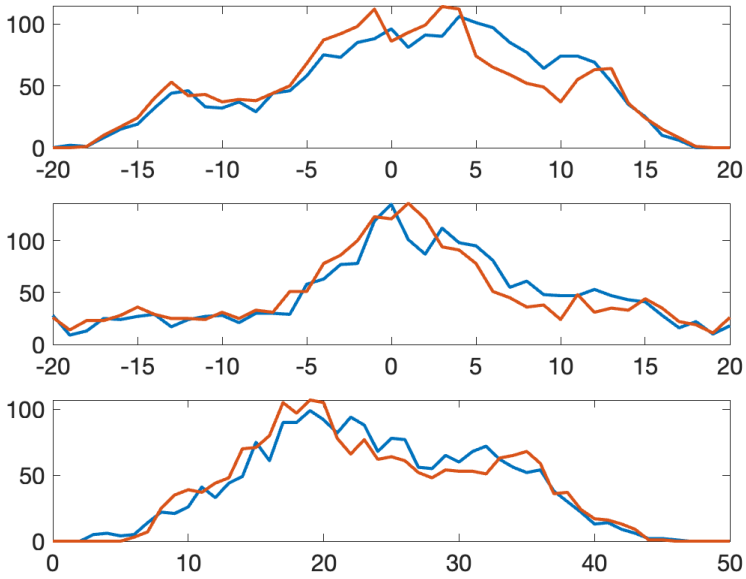


Figure 8: Long-Range Forecast Test: For dynamical systems composed of differential equations, the histogram of the variables, in this case $x(t), y(t)$ and $z(t)$ (top, middle and bottom respectively) are computed from the true solution (blue) and compared in a least-square sense to the predicted solution (red).

mechanics to give

$$\begin{aligned}
\frac{d\theta_1}{dt} &= \omega_1 \\
\frac{d\theta_2}{dt} &= \omega_2 \\
\frac{d\omega_1}{dt} &= \frac{-g(2m_1+m_2)\sin\theta_1 - m_2 g\sin(\theta_1-2\theta_2) - 2\sin(\theta_1-\theta_2)m_2(\omega_2^2 L_2 + \omega_1^2 L_1\cos(\theta_1-\theta_2))}{L_1(2m_1+m_2-m_2\cos(2\theta_1-2\theta_2))} \\
\frac{d\omega_2}{dt} &= \frac{2\sin(\theta_1-\theta_2)(\omega_1^2 L_1(m_1+m_2)+g(m_1+m_2)\cos\theta_1+\omega_2^2 L_2 m_2\cos(\theta_1-\theta_2))}{L_2(2m_1+m_2-m_2\cos(2\theta_1-2\theta_2))}
\end{aligned}$$

where m_1 and m_2 are the masses of the first and second leg of the pendulum which are of length L_1 and L_2 respectively.

Several variants of the double pendulum may be considered; the two limbs may be of equal or unequal lengths and masses, they may be simple pendulums or compound pendulums (also called complex pendulums) and the motion may be in three dimensions or restricted to the vertical plane. Here, the double pendulum is restricted to the vertical plane. As with Rössler and Lorenz, the pendulum exhibits chaotic motion due to the sensitivity to initial conditions.

5 Models: Spatio-Temporal

The first set of toy examples are classical dynamical systems models which are low-dimensional. Lorenz and Rössler are three-dimensional while the double pendulum is four dimensional. The second class of models we consider are spatial temporal so that the dimension is determined by the spatial discretization. These are much higher dimensional than the dynamical systems and often call for some sort of dimensionality reduction to find low-dimensional manifolds on which the dynamics occurs. The Kuramoto-Sivashinsky model has already been used as a demonstration. Two other models are also considered since they can be easily simulated and they are exceptional prototypes for complex spatio-temporal systems.

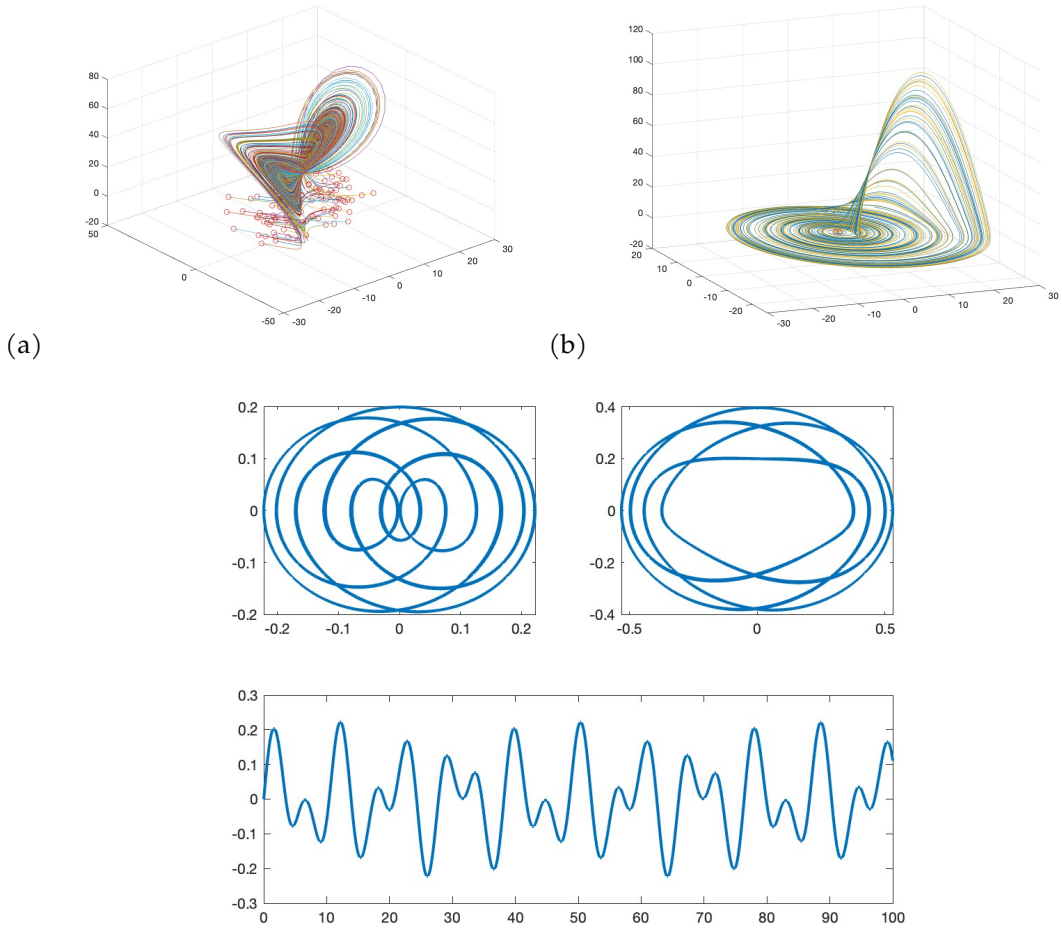


Figure 9: Dynamical systems models to be considered for the CTF. (a) The Lorenz attractor and their myriad of trajectories collapsing to the double-lobed attractor. (b) The Rössler oscillator and its trajectories which include occasional (random) excursions in the z-direction. (c) The double pendulum and its four variables (top panels) along with θ_1 time series. This is the canonical classroom example of chaotic motion and sensitivity to initial conditions.

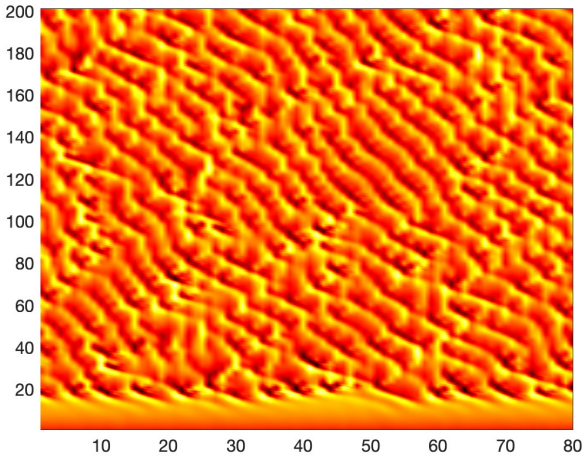
5.1 Lorenz96

The Lorenz 96 system was introduced by E. Lorenz as a follow-up to Lorenz 63 (or what is typically just called the Lorenz equations from the last subsection). It is a test model for studying spatio-temporal predictability in atmospheric models and has since been shown to exhibit chaotic behavior. it is given by,

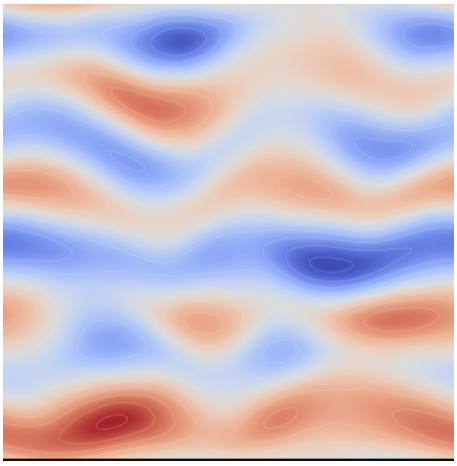
$$\dot{x}_i = (x_{i+1} - x_{i-2})x_{i-1} - x_i + F, \tag{10}$$

where F is a forcing parameter and $i = 1, \cdots, 40$ with period boundary conditions. The degree of chaotic behavior is determined by F, with $F = 8$ resulting in highly chaotic and $F = 16$ resulting in turbulent behavior.

An example of the dynamics of Lorenz96 is given in Figure 10(a). As with other models considered here, forecasting and reconstruction with limited data, noise or parametric variability is exceptionally challenging for this model.



(a)



(b)

Figure 10: Spatio-temporal models for the CTF. (a) Spatio-temporal evolution of the Lorenz96 model which displays spatio-temporal chaos. (b) Spatio-temporal evolution of 2D Kolmogorov flow which also displays spatio-temporal chaos.

5.2 2D Kolmogorov Flow

Kolmogorov flow in two dimensions – the two-dimensional (2D) Navier–Stokes equations with a sinusoidal body force – is considered over extended periodic domains. The dynamics show the kind of spatio-temporal complexity which make it ideal as a challenge data set. At certain parameter regimes of interest, the co-existence of attractors is responsible for the spatio-temporal chaos exhibited. The governing equations are given by

$$\begin{aligned} \nabla \dot{u} &= 0 \\ u_t + u\dot{\nabla}u &= -\nabla p + \nu \Delta u + f \end{aligned}$$

Figure 10(b) shows a snapshot of the flow physics.