# Time Series Forecasting with deep learning

Jingren XIE

Introduction

This is to write some notes for time series forecasting. The story is to learn to predict and interact with time. The key idea is to learn the temporal pattern and dynamics. There are two forecasting methods including point estimate and probabilistic forecasting. With data flowing into the model, we can find the approximate and partial pattern of the historical data. We will process univariate and multivariate data and image data evolving with time. We will augment data, explore the model architecture, optimize the results using model fusion, incorporate dynamics, and give uncertainty.
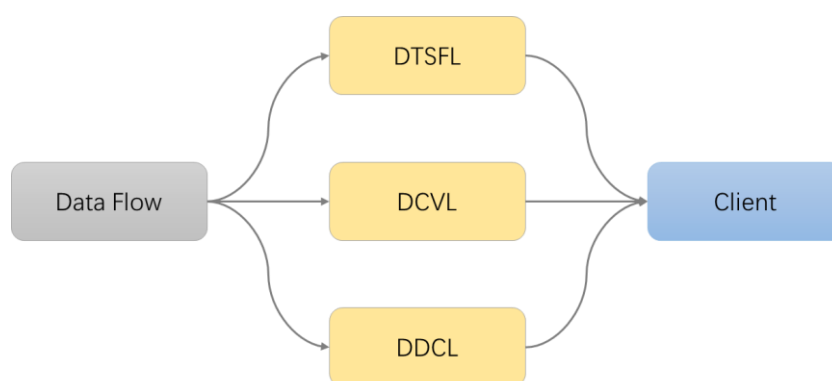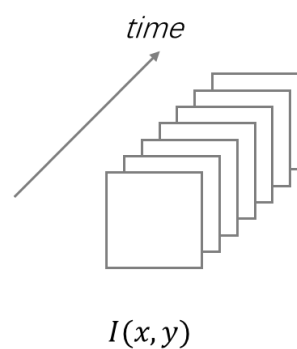

DTSFL = Deep Time Series Forecasting Library
DCVL = Deep Computer Vision Library
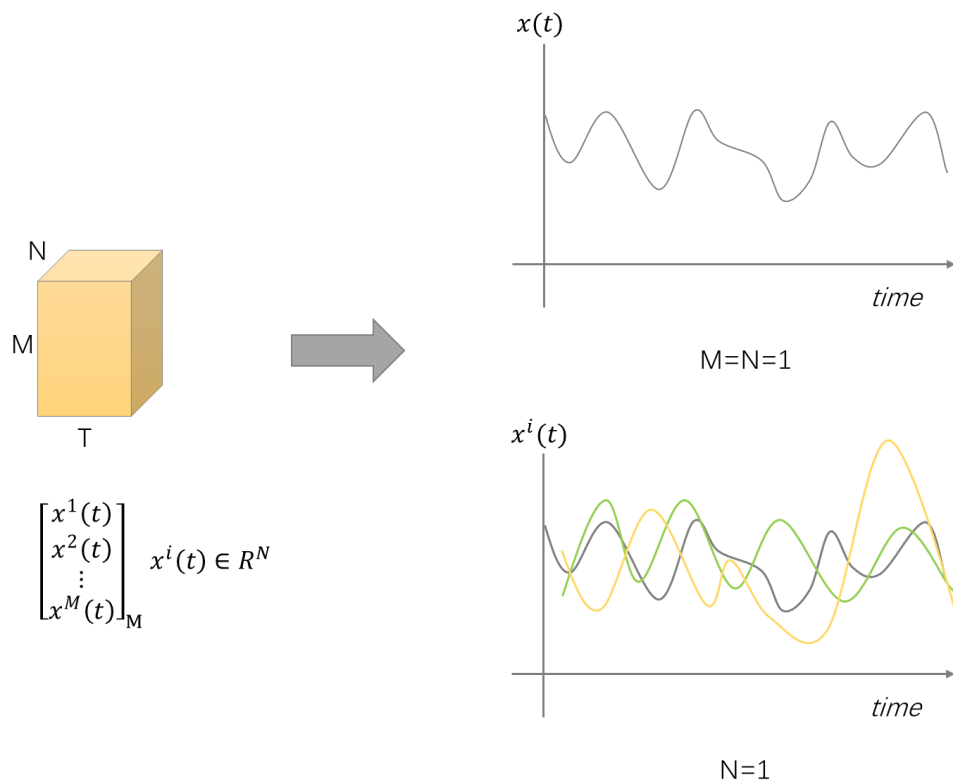DDCL = Deep Dynamics Control Library

Library Framework
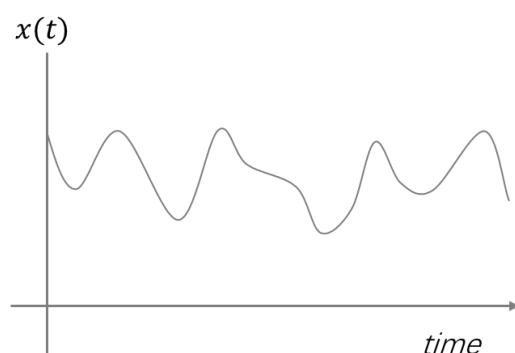
1-D signal   2-D signal

Time Series Example

Date Representation

Given a time series $x_{1:T}^i \in X, x^i(t) \in R^n, n = 1, 2, \cdots, N, i = 1, 2, \cdots, M$, we sample a data sequence and make predictions of the future, $x_{T+1:T+\tau}^i$.

$$x^i(t) \in R^N$$

$$\begin{bmatrix} x^1(t) \\ x^2(t) \\ \vdots \\ x^M(t) \end{bmatrix}_M$$

Data in Matrix Format

$$\begin{bmatrix} x_0 & x_1 & \cdots & x_{T-1} \\ x_1 & x_2 & \cdots & x_T \\ \vdots & \vdots & \ddots & \vdots \\ x_M & x_{M+1} & \cdots & x_{M+T} \end{bmatrix} \rightarrow \begin{bmatrix} x_T & x_{T+1} & \cdots & x_{T+\tau} \\ x_{T+1} & x_{T+2} & \cdots & x_{T+1+\tau} \\ \vdots & \vdots & \ddots & \vdots \\ x_{T+M+1} & x_{T+M+2} & \cdots & x_{M+T+\tau} \end{bmatrix}$$

Data Augmentation

$$\left[x^i(t)\right]_{1\times T} + \left[\varepsilon^i(t)\right]_{1\times T} \rightarrow \left[\hat{x}^i(t)\right]_{R\times T}$$

where $\varepsilon^i(t)$ is a probability distribution such as Gaussian distribution $\varepsilon^i(t) \sim N(0, \sigma^2)$.

Point Estimate vs Probabilistic Forecasting

Point Estimate
Mean or median value

Probabilistic Forecasting
Probabilistic Parametric Estimation

$$L_G = -logl(\mu, \sigma^2|y) = -\log\left[(2\pi\sigma^2)^{-\frac{1}{2}} exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right)\right]$$
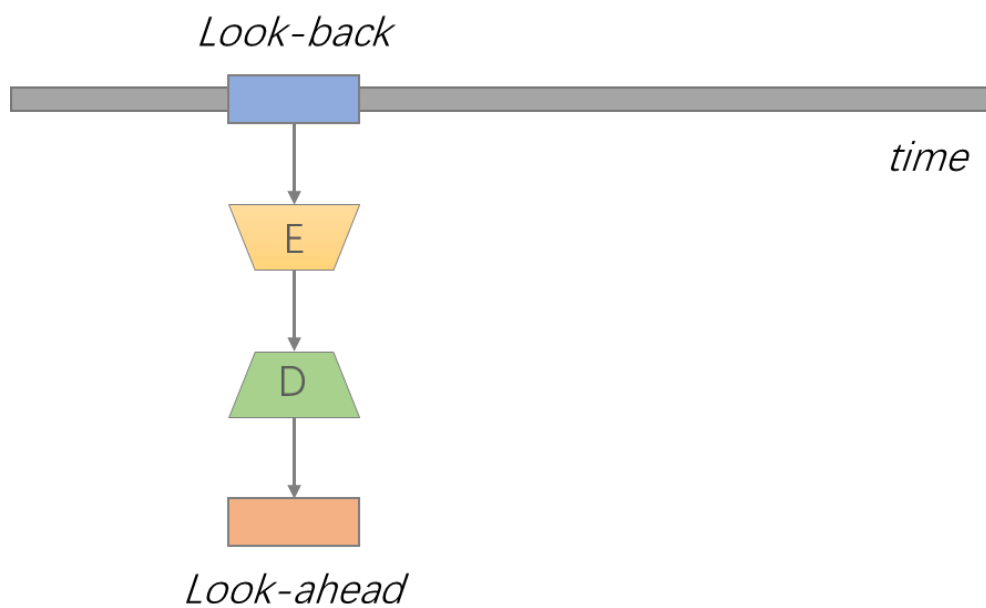
$$= \frac{1}{2}\log(2\pi) + \log(\sigma) + \frac{(y-\mu)^2}{2\sigma^2}$$

Quantile Regression

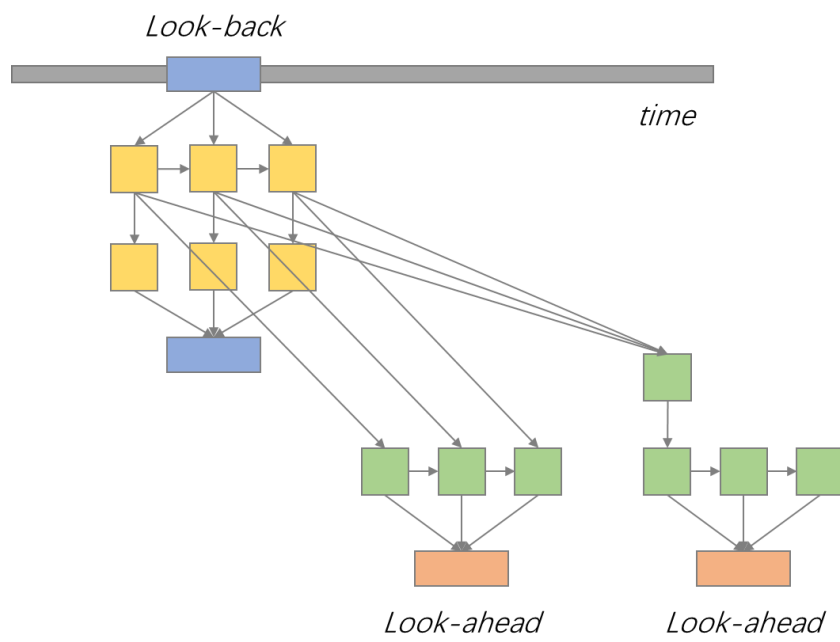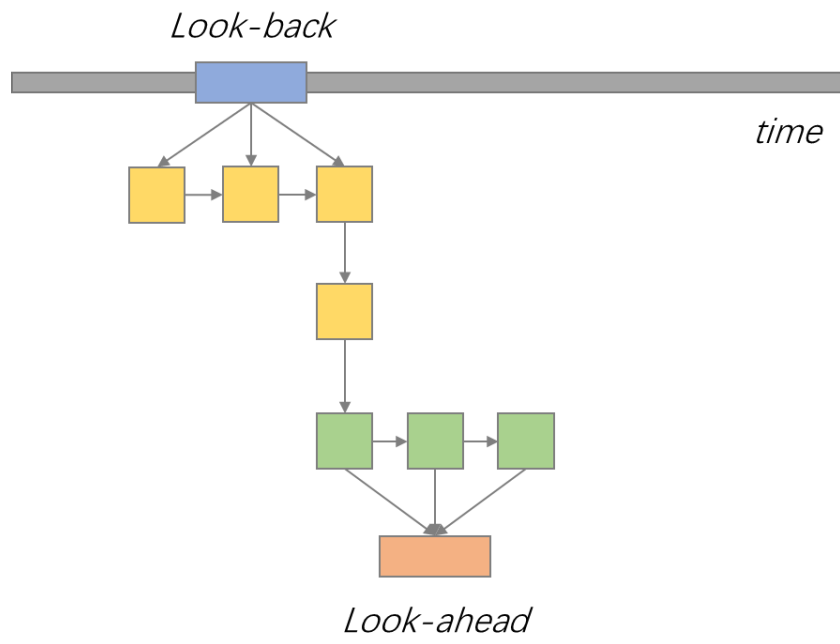$$L_q(y, \hat{y}) = q(y - \hat{y}^q)^+ + (1-q)(\hat{y}^q - y)^+$$

$$(y)^+ = \max(0, y), q \in [0, 1]$$

Model Architecture

Look-back

E

D

Look-ahead

RNN (State Space + RK + Probability)

Look-back

time

Look-ahead

Look-back

time

Look-ahead   *Look-ahead*
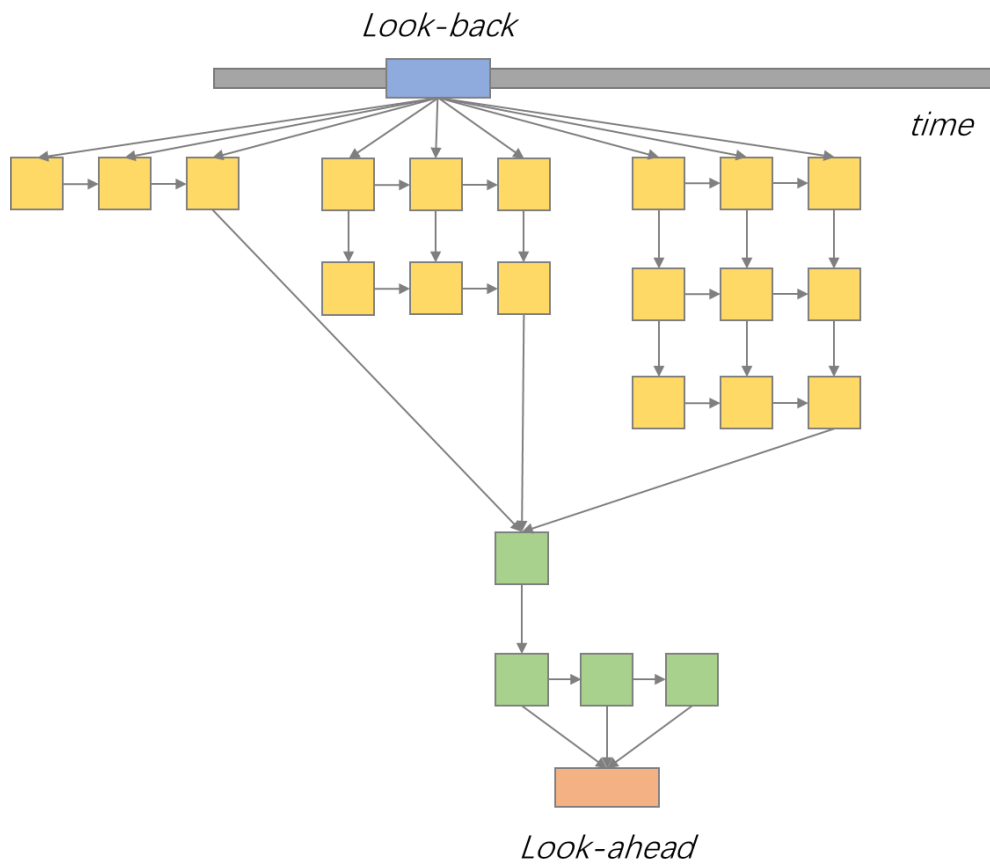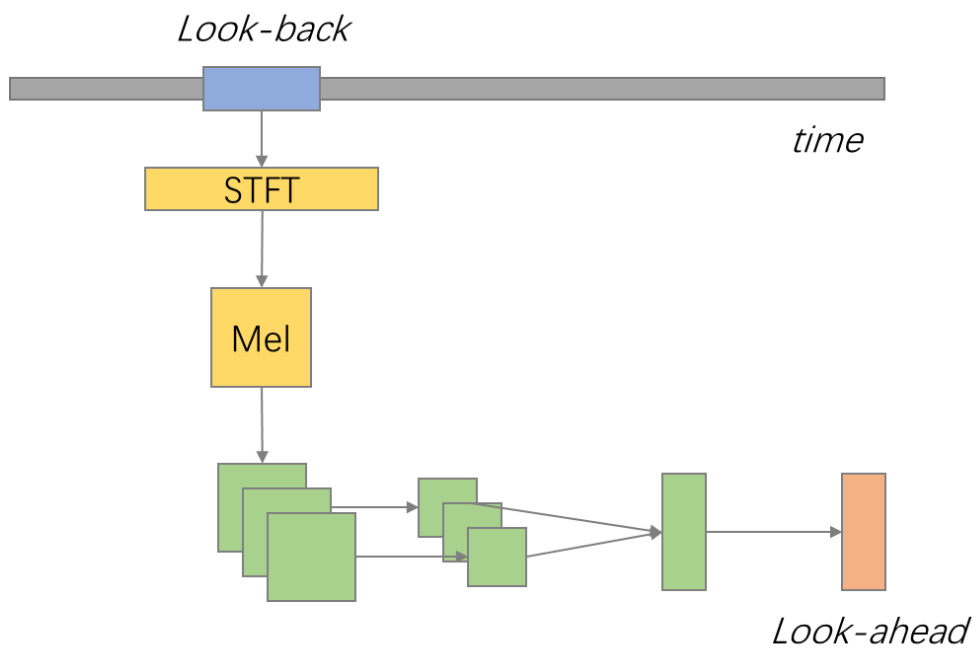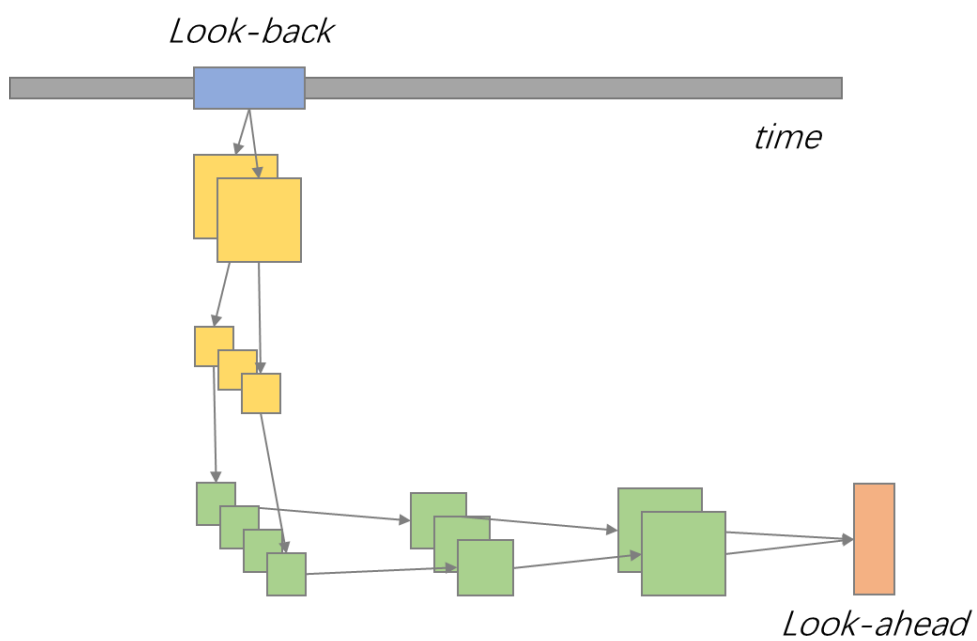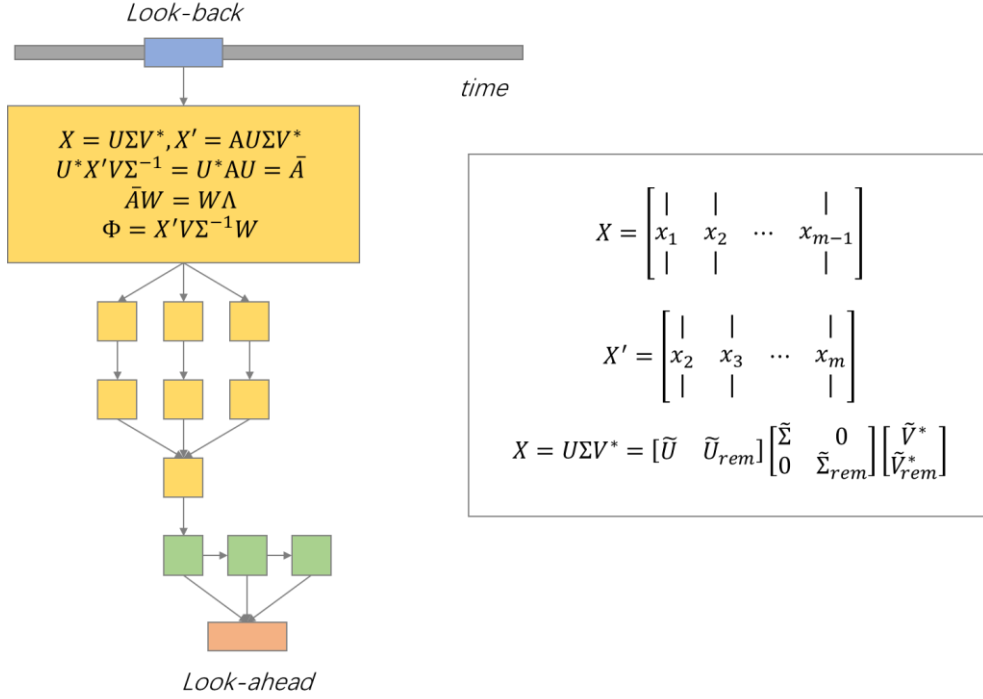
CNN (Sparse Representation)
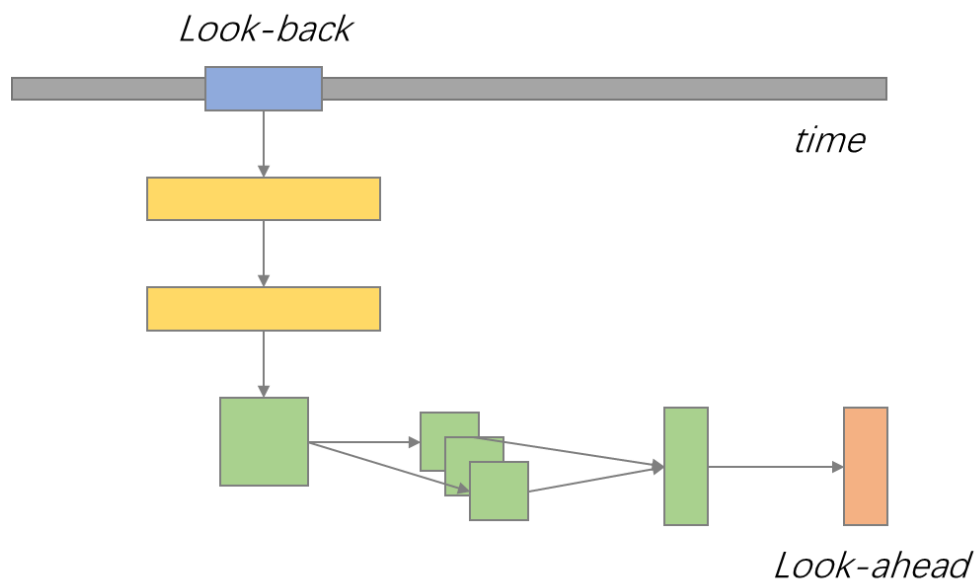
STFT (Nonstationary Signal with Sparse Representation)

AutoEncoder (Generative + Sparse Representation)

Dynamics Incorporation (dynamics and high dimensional data)

Look-back

$$X = U\Sigma V^*, X' = AU\Sigma V^*$$
$$U^* X' V \Sigma^{-1} = U^* AU = \bar{A}$$
$$\bar{A} W = W\Lambda$$
$$\Phi = X' V \Sigma^{-1} W$$

time

$$X = \begin{bmatrix} | & | & & | \\ x_1 & x_2 & \cdots & x_{m-1} \\ | & | & & | \end{bmatrix}$$

$$X' = \begin{bmatrix} | & | & & | \\ x_2 & x_3 & \cdots & x_m \\ | & | & & | \end{bmatrix}$$

$$X = U\Sigma V^* = [\tilde{U} \quad \tilde{U}_{rem}] \begin{bmatrix} \tilde{\Sigma} & 0 \\ 0 & \tilde{\Sigma}_{rem} \end{bmatrix} \begin{bmatrix} \tilde{V}^* \\ \tilde{V}^*_{rem} \end{bmatrix}$$

Look-ahead

Dynamic Mode Decomposition (DMD) is a powerful data-driven approach to analyzing complex systems. DMD analyses the relationship between pairs of measurements from a dynamical system. The measurements, x_k and x_{k+1}, where k indicates the temporal iteration. The dynamical system can be approximated by linearizing the equation. We also think of this a model reduction technique.

$$x_{k+1} \approx A x_k$$

where $x \in R^n$ and $A \in R^{n\times n}$.

In order to approximate matrix A, we first collect data and transform this problem into a regression problem.

$$X = \begin{bmatrix} | & | & & | \\ x_1 & x_2 & \cdots & x_{m-1} \\ | & | & & | \end{bmatrix}$$

$$X' = \begin{bmatrix} | & | & & | \\ x_2 & x_3 & \cdots & x_m \\ | & | & & | \end{bmatrix}$$

Hence, we get

$$X' \approx AX$$
$$A \approx X' X^{pseudo}$$

A is determined by minimizing the Frobenius norm of $|X' - AX|_F$.

$$X = U\Sigma V^* = [\tilde{U} \quad \tilde{U}_{rem}] \begin{bmatrix} \tilde{\Sigma} & 0 \\ 0 & \tilde{\Sigma}_{rem} \end{bmatrix} \begin{bmatrix} \tilde{V}^* \\ \tilde{V}^*_{rem} \end{bmatrix}$$

where $U \in R^{n\times n}$, $\Sigma \in R^{n\times m-1}$, $V^* \in R^{m-1\times m-1}$, $\tilde{U} \in R^{n\times r}$, $\tilde{\Sigma} \in R^{r\times r}$, $\tilde{V}^* \in R^{r\times m-1}$.

$$A \approx X' \tilde{V}^* \tilde{\Sigma}^{-1} \tilde{U}$$

Here, we will summarize the DMD computation procedure.

svd	$X = U\Sigma V^*, X' = AU\Sigma V^*$
*	$U^* X' V \Sigma^{-1} = U^* A U = \bar{A}$
eig	$\bar{A} W = W \Lambda$
*	$\Phi = X' V \Sigma^{-1} W$
*	$\hat{X}(k\Delta t) = \Phi \Lambda^{\mathrm{t}} b_0$

Model Fusion

Uncertainty

$X = U\Sigma V^*, X' = AU\Sigma V^*$

$U^* X' V \Sigma^{-1} = U^* A U = \bar{A}$

$\bar{A} W = W \Lambda$

$\Phi = X' V \Sigma^{-1} W$

$\hat{X}(k\Delta t) = \Phi \Lambda^{\mathrm{t}} b_0$