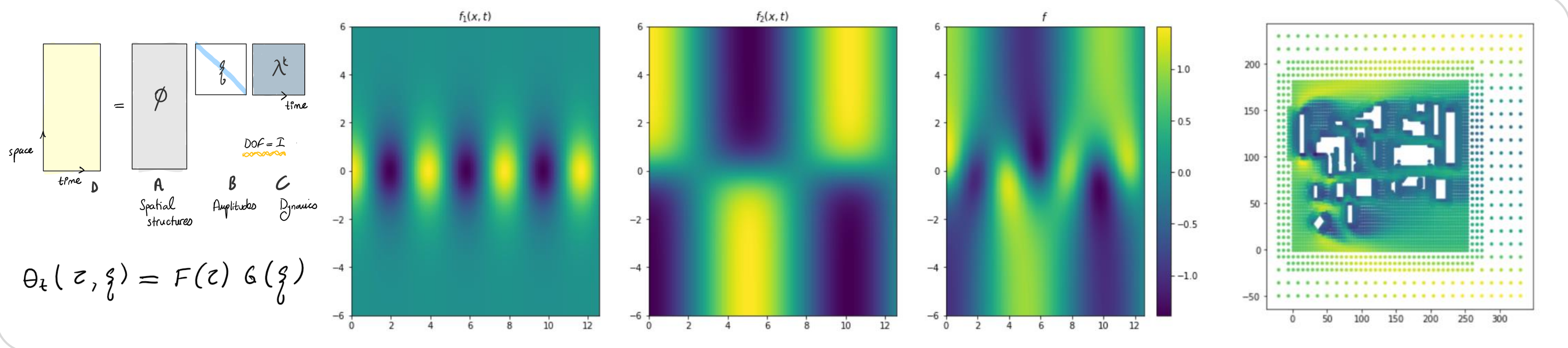
Data Driven Engineering II: Advaced Topics

Dynamic Mode Decomposition

Institute of Thermal Turbomachinery
Prof. Dr.-Ing. Hans-Jörg Bauer



Term Projects

Welcome to DDE II projects!

If you are interested in the group projects for fun or planning to take the final exam for credits, you need to register to a topic before 14.05.021.Note that each topic has a number of maximum participants. You may find the details in Lecture 1.

Particle Image Density Analysis in PIV Recordings

An object detection study for PIV analysis

Free places: 1

Physical interpretation of LCSs

Data driven model discovery in air blast atomizers

Free places: 5

Time resolved flow field analysis in film cooling

PIV data will be used for flow analysis.

Free places: 5

Others

for HPC access

Period of Event: Today - 14. May 2021

Today's Agenda

- Dynamical system analysis
- Nonlinear ⇒ linear mapping
- Koopman analysis
- DMD: how it works?
- Applications → alternatives
- PyDMD

Non-linear dynamical systems:

$$\frac{d}{dt}x = f(x, t, u, \mu)$$

state (vector)
– information –

control

parameters → g, T, P ...

$f :=$ Describes the system dynamics

- NS equations
- Maxwell eqn.
- Schrödinger eqn.
- "Rule of Thumbs"

f assumed

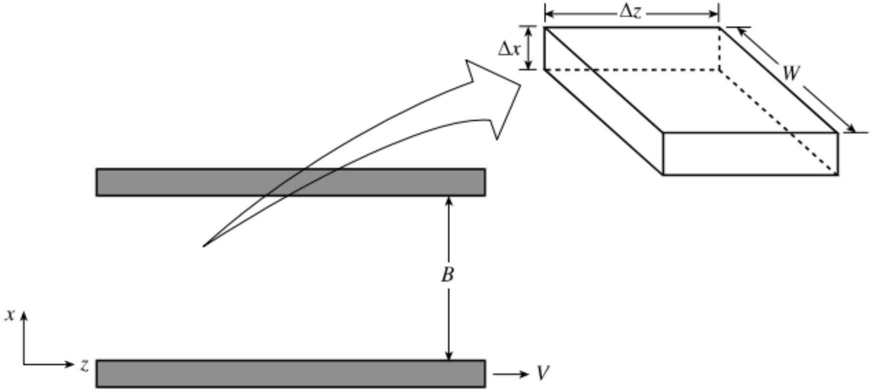
{Solution methods}

Transfer learning

f unknown

?

Simple Example: 1-D, transient flow



$$\Rightarrow \quad \rho \frac{\partial v_z}{\partial t} = \mu \frac{\partial^2 v_z}{\partial x^2}$$

- $t=0 \qquad v_z = 0$
- $x=0 \qquad v_z = V$
- $x=B \qquad v_z = 0$

Simple Example: 1-D, transient flow

Step 1 make it simple!

$$\frac{\text{viscous forces}}{\text{rt. of momentum accum.}} = \frac{\mu V / B^2}{\rho V / t} = \frac{\nu t}{B^2} \Bigg\} \text{Fourier number}$$

- $\theta = v_z / V \; ; \; \xi = x/B \; ; \; \tau = \nu t / B^2$

- $$\frac{\partial \theta}{\partial \tau} = \frac{\partial^2 \theta}{\partial \xi^2}$$

- $\tau = 0 \; ; \; \theta = 0$
  $\xi = 0 \; ; \; \theta = 1$ * not homogeneous
  $\xi = 1 \; ; \; \theta = 0$

Simple Example: 1-D, transient flow

Step II Divide & Conquer

(i) $\Theta(\tau, \xi) = \underbrace{\Theta_\infty(\xi)}_{\text{steady}} - \underbrace{\Theta_t(\tau, \xi)}_{\text{transient}}$

(ii) $\Theta_\infty = 1 - \xi$ } st. st. solution

(iii) $\Theta = 1 - \xi - \Theta_t$;

$$\frac{\partial \Theta_t}{\partial \tau} = \frac{\partial^2 \Theta_t}{\partial \xi^2} \quad \left\{ \begin{array}{ll} \tau = 0 ; & \Theta_t = 1 - \xi \\ \xi = 0 ; & \Theta_t = 0 \\ \xi = 1 ; & \Theta_t = 0 \end{array} \right.$$

! PDE is separable;

$$\Theta_t(\tau, \xi) = F(\tau)\, G(\xi)$$

Simple Example: 1-D, transient flow

(iv) $$\Theta(\xi,\tau) = 1 - \xi - \frac{2}{\pi} \sum_{n=1}^{\infty} \frac{1}{n} \exp\left(-n^2\pi^2\tau\right) \sin\left(n\pi\xi\right)$$

Eng. Practice:

(1) Split the spatial & temporal dynamics

(2) Change coord. & use eigen values / functions

(3) $\infty \rightarrow N$ $\Rightarrow$ Truncate the solution

Non-linear dynamical systems :

nonlinear dynamical system $\longrightarrow$ Coordinate Transformation (ϕ) $\longrightarrow$ Equivalent linear dynamical system

$\dot{x} = f(x)$ $\qquad$ $z = \phi(x)$ $\qquad$ $\dot{z} = Lz$

Obj : $X = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix} \xrightarrow{\text{'math'}} f$

* high Dim.
* multiscale
* chaotic
* Noise
* uncertainity

Data-driven algorithm

↳ general

↳ linear

Non-linear dynamical systems:

Typically;

(i) $X_{n+1} = F(X_n)$

(ii) $g(X)$ → observable vector

→ function of state vector X

(iii) $t \to n+1$;

$$g(X_{n+1}) = g(F(X_n)) = K\,g(X_n)$$

linear operator

$n \longrightarrow n+1$

Koopman operator

Non-linear dynamical systems:

(iv)

$$\cdots \rightarrow x_n \xrightarrow{F} x_{n+1} \xrightarrow{F} x_{n+2} \rightarrow \cdots$$

$$\downarrow \qquad \downarrow \qquad \downarrow$$

Observable $\Big[\cdots \rightarrow g_n \xrightarrow{K} g_{n+1} \xrightarrow{K} g_{n+2} \rightarrow \cdots$

? What is happening here $\Rightarrow$ (finite-dim. nonlinear system) $\rightarrow$ (infinite-dim. linear system) any function you need it to be!

Koopman := "Trade nonlinearity with dimensionality"

~~Eg.~~ How infinite is the "infinite"? $\infty \rightarrow 3$

Karlsruhe Institute of Technology

① $\dot{x}_1 = a x_1$

$\dot{x}_2 = b(x_2 - x_1^2)$ } finite dim. nonlinear

② $x_1 \downarrow y_1$, $x_2 \downarrow y_2$

$\{y_1, y_2, \cancel{y_1^2}\ y_3\}$ "Koopman embedding"

"how my equations look like in y space..."

③
$$\frac{d}{dt}\begin{pmatrix} y_1 \\ y_2 \\ \cancel{y_1^2}\ y_3 \end{pmatrix} = \begin{pmatrix} a & \emptyset & \emptyset \\ \emptyset & b & -b \\ \emptyset & \emptyset & 2a \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \cancel{y_1^2}\ y_3 \end{pmatrix}$$
} linear system

Eq

$$x_{n+1} = \beta x_n (1 - x_n) \rightarrow \text{logistic map}$$

$$x_{n+1} = \beta x_n - \beta x_n^2 \longrightarrow \{ y_n , y_n^2 \}$$

$$\begin{pmatrix} y \\ y^2 \\ \vdots \end{pmatrix}_{n+1} = \begin{pmatrix} \beta & -\beta & \cdots \\ & \ddots & \\ & & \ddots \end{pmatrix} \begin{pmatrix} y \\ y^2 \\ \vdots \end{pmatrix}_n \quad \infty \rightarrow \infty$$

Powering idea did not work

"exp" would work.

* How do we reach closure?

↘ in a single recipe!

* we need invariant subspace

Non-linear dynamical systems:

* we need invariant subspace ⇨ use eigen functions !

$$g(x) = \phi\xi \longrightarrow g(x_k) = K^k \phi\xi = \phi\xi\lambda^k$$

} how do we get those ?

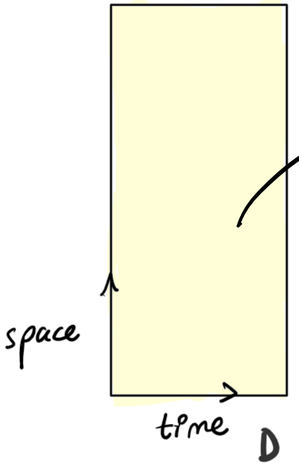
eig. vectors

eig. values

spatial structures

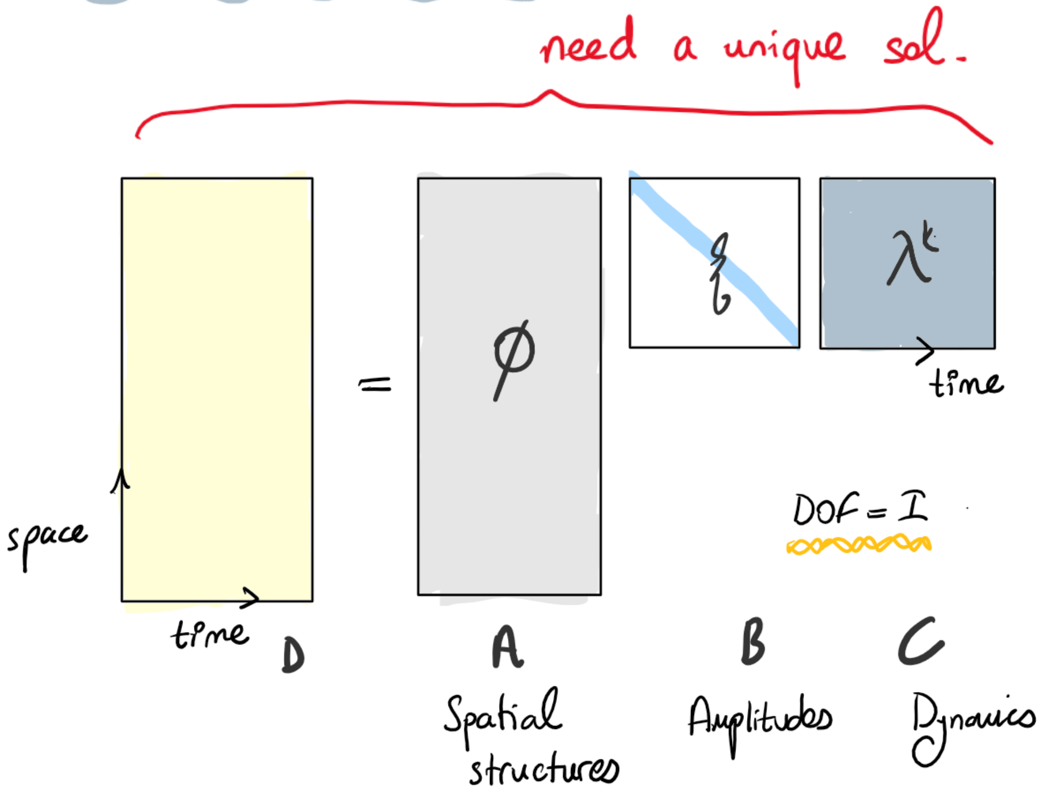
amplitudes

temporal ~dynamics



Data (ξ, z)

Data Decomposition:



- * A := orthogonal columns ⇒ "POD"
- * C := statistically independent ⇒ "ICA"
- * B → block diagonal, C → maximally sparse ⇒ Blind Compressive Sensing

In DMD : Shape of 'C' matrix

$$C = \begin{bmatrix} 1 & \lambda_1 & \lambda_1^2 & \cdots \\ 1 & \lambda_2 & \lambda_2^2 & \cdots \\ 1 & \lambda_3 & \lambda_3^2 & \cdots \\ 1 & \vdots & \vdots & \\ \vdots & \vdots & \vdots & \\ 1 & \lambda_n & \lambda_n^2 & \cdots \end{bmatrix} \Rightarrow$$

n independent relations

$$\Rightarrow K = \begin{bmatrix} 0 & & & a_1 \\ 1 & & & a_2 \\ & \ddots & \cdots & a_3 \\ & & \ddots & \vdots \\ & & 1 & a_n \end{bmatrix}$$

– Component matrix –

In DMD : Shape of 'C' matrix

$$K = \begin{bmatrix} 0 & & & a_1 \\ 1 & & \cdots & a_2 \\ & \ddots & & a_3 \\ & & \ddots & \vdots \\ & & 1 & a_n \end{bmatrix}$$

– Component matrix –

* $K = C^{-1} B C$;
* $D = A \underbrace{B C}$
* $D = A \underbrace{C K}$
* $D = D' K$

✓ ✗ ✗

! $K \Rightarrow$ shifts the columns ;

↪ $D = D' K$ "

↪ back-shifted data matrix "

↪ D' & K can be found together

↪ by looking II conseq. snapshots

In DMD: Shape of 'C' matrix

Regression ⇒ least sq. fitting

$$X = \begin{bmatrix} | & | & | & & | \\ x_1 & x_2 & x_3 & \cdots & x_{m-1} \\ | & | & | & & | \end{bmatrix} \; ; \; X' = \begin{bmatrix} | & | & | & & | \\ x_2 & x_3 & x_4 & \cdots & x_m \\ | & | & | & & | \end{bmatrix}$$

$\circ \longrightarrow$ time

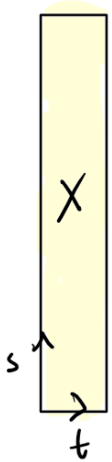
(i) $X' = KX$

(ii) $K = X'X^{+}$ → pseudo-inverse

Can we practically do it?

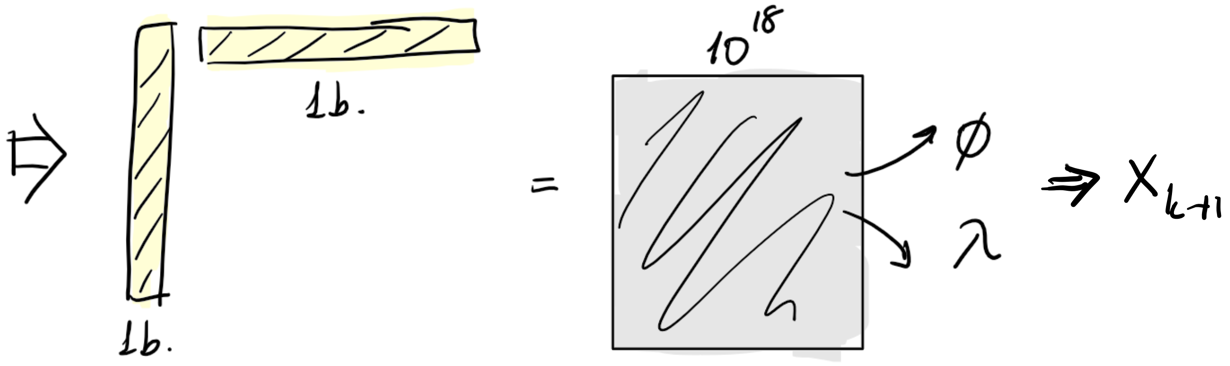
Can we do it? practically?

* Transport Ph. → exp. → PIV → Images
* Transport Ph. → nu. → DNS → 1b. grids [, . . .]



$t \rightarrow 10^4$ steps

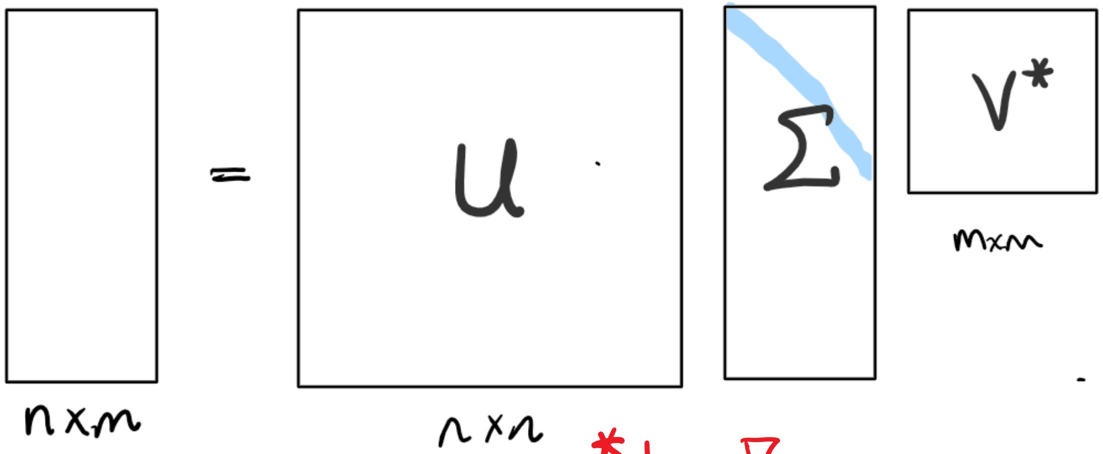
$s \rightarrow \sim 1$ billion.



Too many dim. $\Rightarrow$ Dim. Reduction

* SVD := Singular Value Decomp. (✓)

①



* huge !

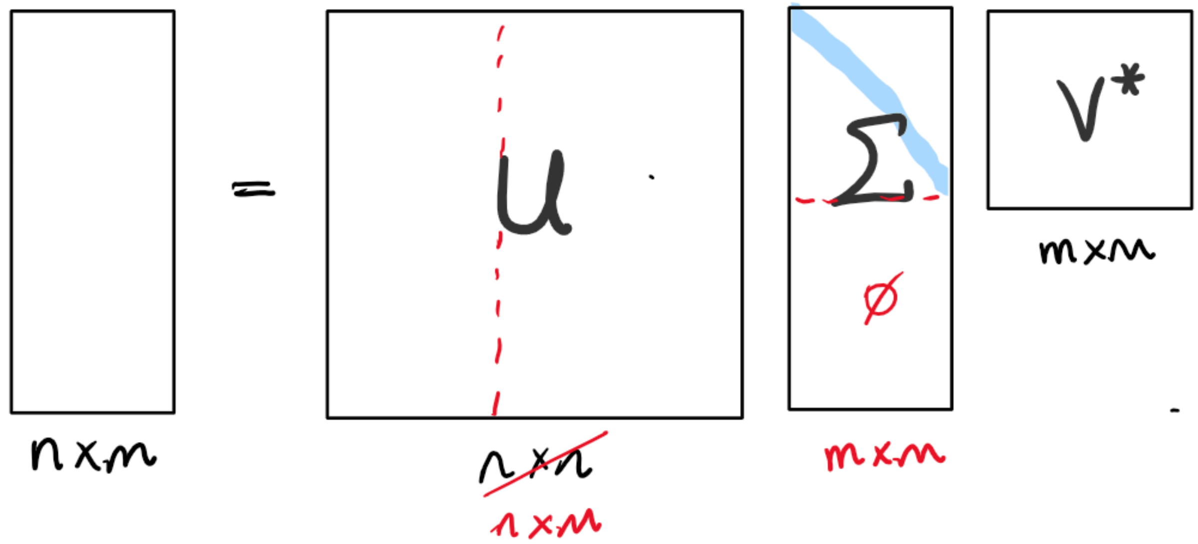
- $U, V \rightarrow$ unitary

  $U^*U = UU^* = I$

- $\Sigma \rightarrow$ diagonal, decreasing, non-negative

Too many dim. ⇒ Dim. Reduction

* SVD := Singular Value Decomp. (✓)

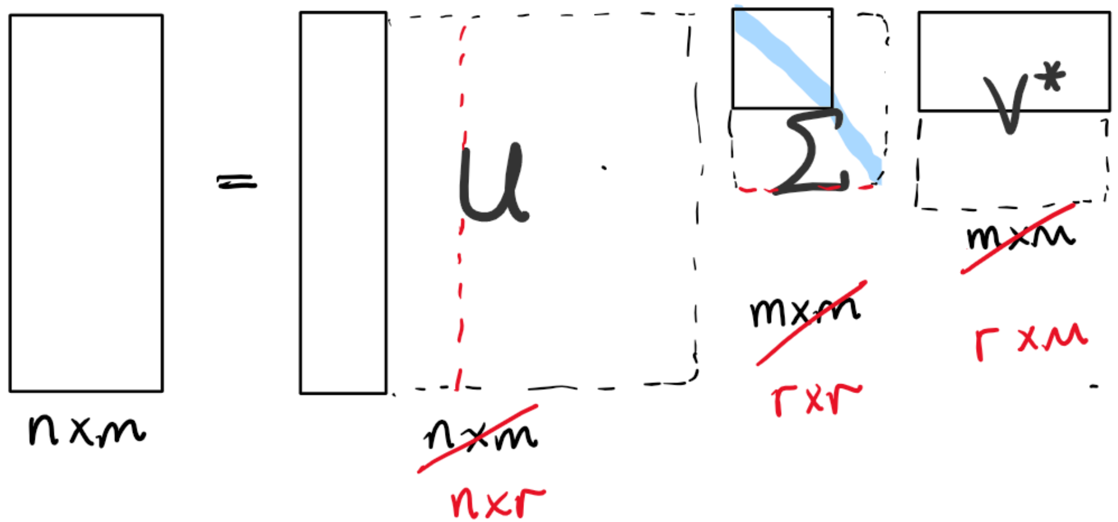
② 

[Economy SVD]

Too many dim. ⇒ Dim. Reduction

* SVD := Singular Value Decomp. (✓)

3



Truncated SVD

→ r

$X = U \Sigma V^*$

↓

$X \approx U_r \Sigma_r V_r^*$

DMD Algorithm :

Step #1 Dimensionality reduction with SVD.

- $X \cong \underbrace{U_r \Sigma_r V_r}^{*}$ (r rank)

  Subspace we will work at.

Step #2 Regression

- $K = X' X^{+}$; in the reduced subspace;

  $K \to S = U_r^* X' V_r \Sigma_r^{-1}$

- we do not need to find high dim. K;

  $X'(t+1) \approx S \, X(t)$

DMD Algorithm:

Step 3: Spectral decomposition $\Rightarrow$ ϕ, λ (coord. tr.)

- $S\phi = \phi\Lambda$ (instead of K)

  ϕ $\to$ eig. vectors (columns)

  Λ $\to$ Diagonal $\Rightarrow$ $\lambda_1, \lambda_2 \ldots$

- $\phi, \Lambda \leftarrow \text{eig}(S)$

DMD Algorithm :

Step 4 : Reconstruct the DM :

- $DM = X' V_R \Sigma_R^{-1} \phi$] columns are eig. vectors of K

  ✓ ✓ ✓ ✓

↳ eig. values
eig. vectors
of K $\Rightarrow$

(i) $\omega_k = \ln(\lambda_k)/\Delta t$ } cont'd rep.

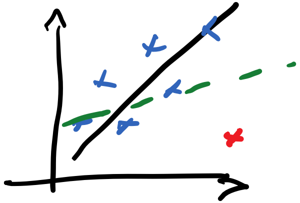
(ii) $x(t) \cong \sum_{k=1}^{r} dm_k \exp(\omega_k t)\, b_k$

colab

Things to remember

* DMD → Regression ⇒ optimization
  ⇒ Error Min } 'tiny errors,
  unit circle ⇒ $\underbrace{e^{\cdots}}_{\text{solution diverges / converges}}$

* Regression



} outliers ⁈
— least sq. fitting — ⇒ Robust DMD

* Nonlinear ⇔ Linear

[Many sol.] - -> ['a' solution] ⇒ other sol. are infinitely away from us.

Why linearize?

Obj: non-linear $\Rightarrow$ linear

* $\left.\frac{dx}{dt} = Ax\right\}$ linear; $x(t_0+t) = e^{At}x(t_0)$

* Linear $\Rightarrow$ easy to change coordinates
  $\rightarrow$ invariant op.

- $x = Tz$

$$\dot{z} = T^{-1}\dot{x} = T^{-1}Ax = T^{-1}ATz = \Lambda z$$

$$\dot{z} = \Lambda z \rightarrow \text{diagonal} \begin{bmatrix} \lambda_1 & 0 & & \\ 0 & \lambda_2 & & 0 \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix} \Rightarrow \text{decoupled ?}$$

Why linearize?

* $\dot{z} = \underbrace{T^{-1}\dot{x}}$

after transformation; dynamics are reflected in z coordinates.

* Note that $\underbrace{AT = T\Lambda}_{\text{Eigen value eq.}}$; $T^{-1}ATz = \Lambda z$

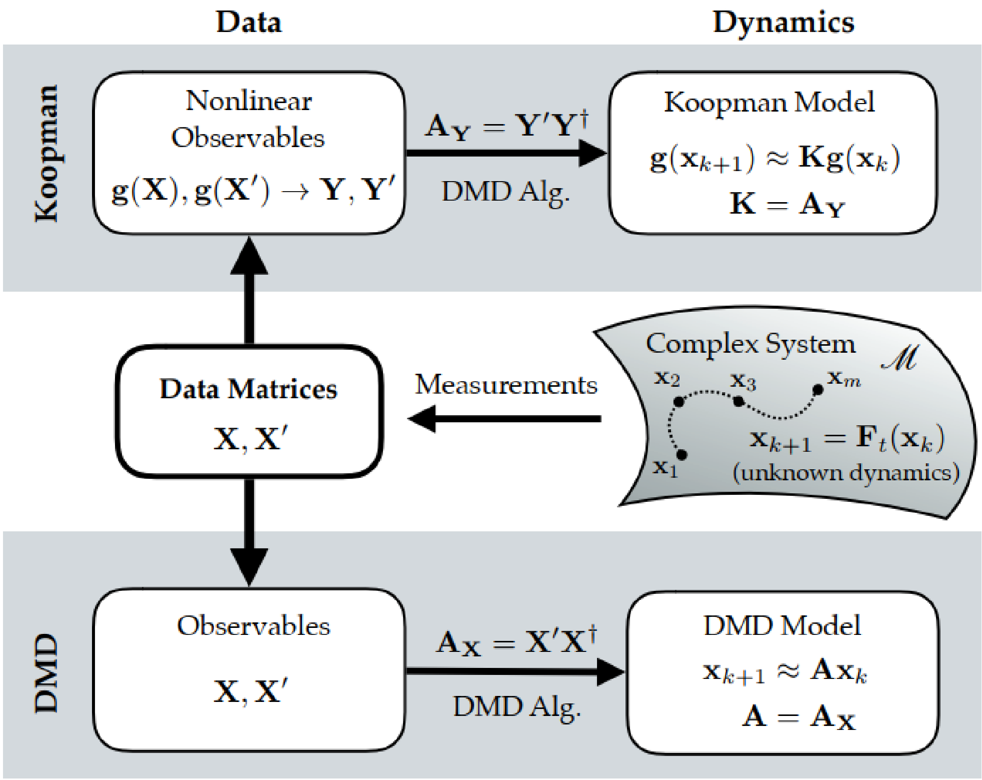
Eigen value eq.

! eig. values, eig. vectors } Turn a linear system into a diagonal system.

Why linearize?

$$x(t) = T\,z(t)$$

$$= \underbrace{T\,\overbrace{\underbrace{e^{\Lambda t}\,\underbrace{T^{-1}\,x(0)}_{z_0 :=\ \text{initial cond. mapped into eigen coord.}}}_{z_t :=\ \text{propagating in time in eig. coordinates.}}}^{z(t)}}_{\text{back-transfer to } x \text{ coordinates}}$$



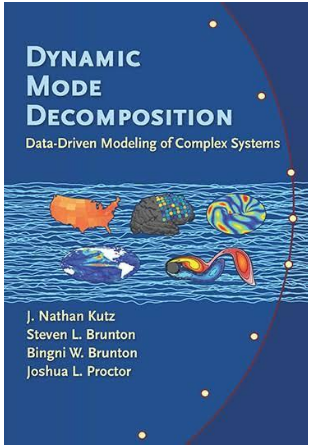


Figure 3.2. *Schematic of how to use data to generate dynamical systems models of an unknown complex system in the DMD/Koopman framework. In standard DMD, we take measurements of the states of the system and construct a model that maps* $\mathbf{X}$ *to* $\mathbf{X}'$*. Koopman spectral analysis enriches the measurements with nonlinear observations* $\mathbf{y} = \mathbf{g}(\mathbf{x})$ *to provide a better mapping from* $\mathbf{Y}$ *to* $\mathbf{Y}'$ *that approximates the infinite-dimensional Koopman mapping. The prediction of the observables in the future from the Koopman model may be used to recover the future state* $\mathbf{x}_{m+1}$*, provided that the observation function* $\mathbf{g}$ *is injective. Both the DMD and Koopman approaches are equation-free, in that they do not rely on knowing* $\mathbf{F}_t$*.*