

Practical Course on Molecular Dynamics and Trajectory Analysis

Episode 7: Models and spectra with deeptime

Jordi Villà i Freixa

Universitat de Vic - Universitat Central de Catalunya
Facultat de Ciències, Tecnologia i Enginyeries (FCTE)

jordi.villa@uvic.cat

MD Course and Trajectory Analysis
Concepcion, January 2026

Contents

1 Episodio 7: Modelos y espectros con Deeptime

- Operadores de transferencia
- Espectro y modos
- Bases y proyecciones
- tICA y VAMP
- Modelos avanzados
- Validación y scores
- Pipeline con Deeptime
- Resumen

Definición

$$(\mathcal{T}_\tau f)(x) = \mathbb{E}[f(X_{t+\tau}) \mid X_t = x].$$

- Extiende MSM al espacio continuo de observables.
- PyEMMA y Deeptime comparten la base de datos generada por `simulateAmber.py`.

Koopman vs Perron-Frobenius

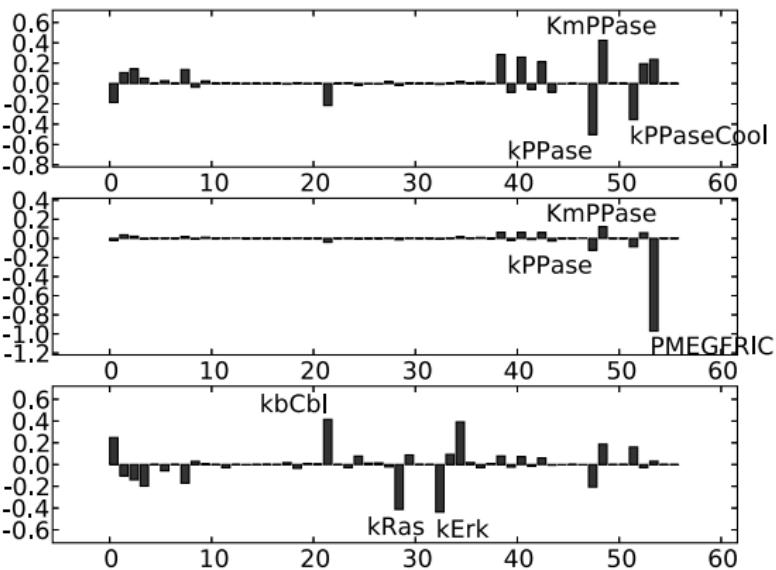
- Koopman actúa sobre observables (funciones).
- Perron-Frobenius actúa sobre densidades.
- Permiten construir 'TransferOperator' en Deeptime.

Problema espectral

$$\mathcal{T}_\tau \psi_i = \lambda_i \psi_i.$$

- Los autovalores λ_i definen tiempos implícitos $t_i = -\tau / \ln \lambda_i$.
- $\lambda_1 = 1$ para equilibrio y valores próximos a 1 indican procesos lentos.

Modos espectrales

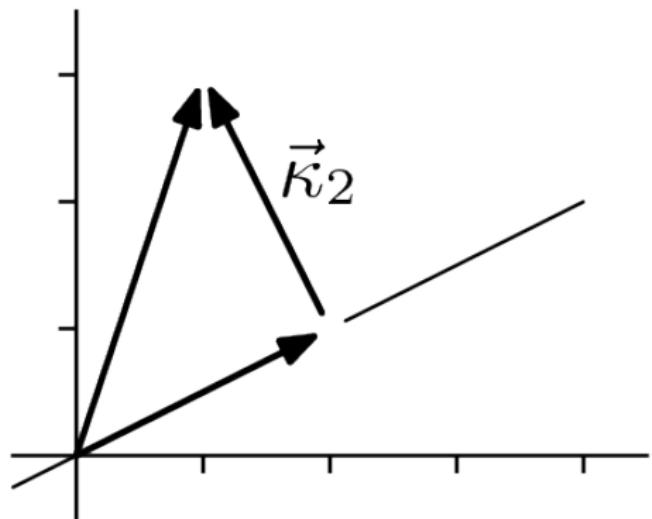


Expansión en bases

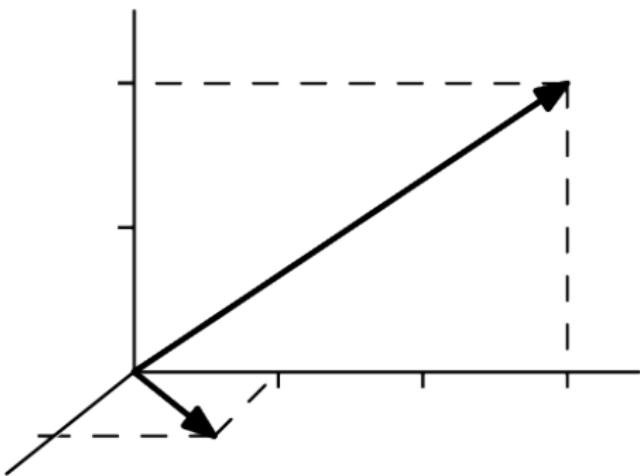
$$f(x) = \sum_i c_i \phi_i(x).$$

- Deeptime permite bases lineales/no lineales: kernels y redes.

Bases y proyecciones



Fuente: referencia pendiente. [2]



Error y regularización

- Bases pobres generan sesgo espectral.
- Regularizar con L_2 y validación cruzada.

tICA en Deeptime

$$C_\tau \mathbf{w} = \lambda C_0 \mathbf{w}.$$

- Maximiza correlación a largo plazo.

VAMP score

$$\mathcal{R}_2 = \sum_i \sigma_i^2.$$

- Evalúa calidad espectral para escoger features.

Aproximación de Koopman

- Expandir observables en bases lineales/no lineales.
- Obtenemos representaciones reducidas y más interpretables.

Modelos kernel y neuronales

- Kernels capturan no linealidades; redes requieren validación estricta.
- Regularizar con dropout o L_2 para prevenir sobreajuste.

Validación y robustez

- Dividir en bloques temporales evita fuga de información.
- Comparar scores espectrales entre conjuntos.
- $T(n\tau) \approx T(\tau)^n$ valida consistencia temporal.
- Variar τ y clustering permite detectar sensibilidad.

Flujo básico

- Features → tICA/VAMP → modelo espectral.
- Usar `simulatePdb.py` para generar trayectorias y ‘OpenMMTools’ para preparar ‘REST’.

Visualización

- Proyecciones 2D con FES y comparación con MSM.

Resumen del episodio

- Deeptime extiende MSM con operadores espectrales y validación cruzada.
- Combinamos features, VAMP y scores para seleccionar modelos robustos.

References I

- [1] OpenMM Cookbook. *Eigenvectors espectrales en Replica Exchange Solute Tempering.* CC BY-SA 4.0. URL:
https://openmm.github.io/openmm-cookbook/latest/notebooks/tutorials/Running_a_REST_simulation.html
(visited on 01/12/2026).
- [2] Autor desconocido. *Reference pending.* Fuente no localizada. n.d.