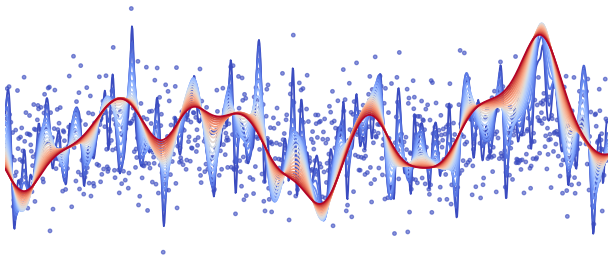
Modelado generativo de series temporales

De los sistemas lineales a los modelos de flujos

Felipe Tobar

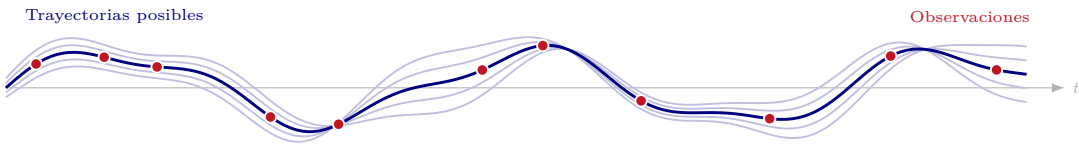
Departamento de Matemáticas
Imperial College London



https://www.ma.ic.ac.uk/~ft410/talks/

¿Por qué modelar series temporales?

Para entender cómo una **variable de interés evoluciona en el tiempo**, normalmente en base a observaciones parciales.



problema inverso: observaciones parciales compatibles con varias realizaciones

Áreas de aplicación:

Ciencias de la vida, clima, audio, finanzas, salud, astronomía, sistemas mecánicos y más.

Tareas comunes:

Generación incondicional, predicción, eliminación de ruido, imputación, compresión, clasificación y más.

Conocer el mecanismo generativo de los datos ayuda a entender sus causas subyacentes.

¿Cuál es el desafío?

Representaciones

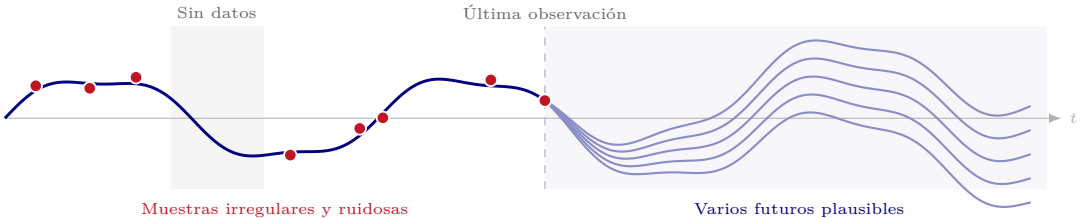
Explosión combinatorial: modelos distribuidos (tiempo) versus localizados (espectro).

Múltiples patrones

Memoria corta/larga, oscilaciones, transientes, cambios de régimen y rugosidad/suavidad.

Observaciones parciales y ruidosas

Grillas irregulares, datos faltantes, ruido y mediciones indirectas.



La extracción de información no es trivial: Las características relevantes del estado suelen surgir de interacciones entre retardos, más que de valores puntuales.

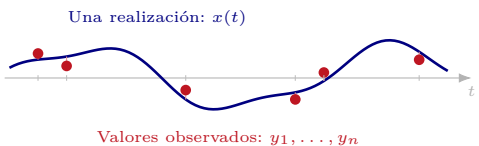
Contexto y notación

i) En general[1], consideraremos que las series temporales son **procesos estocásticos**:

$$X = \{X(t) : t \in \mathcal{T}\}, \quad X(t) \in \mathbb{R}^d, \qquad \mathcal{T} = \mathbb{Z} \text{ o } \mathcal{T} \subseteq \mathbb{R}.$$

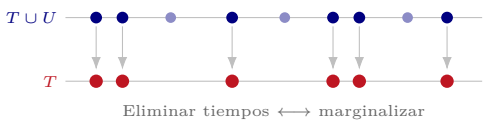
ii) En una grilla finita, observamos parte de una **realización** x.

$$T = \{t_1, \ldots, t_n\}, \qquad Y_i = X(t_i) + V_i.$$



Los observaciones pueden ser asíncronas; V_i describe el ruido de medición.

iii) Las observaciones en distintas grillas deben describir el mismo proceso.



$$p_T(\mathbf{x}_T) = \int p_{T \cup U}(\mathbf{x}_T, \mathbf{x}_U)\, \mathrm{d}\mathbf{x}_U.$$

Reordenar los tiempos solo debe reordenar las variables correspondientes, no cambiar la ley.

Leyes finito-dimensionales compatibles definen la ley del process (ver teorema de extensión de Kolmogórov).

[1] Habrá excepciones interesantes.

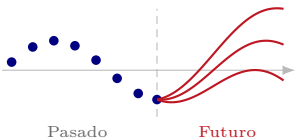
El enfoque generativo

Aprender una distribución a priori permite i) generar nuevas muestras, ii) analizar la naturaleza del proceso, y iii) resolver problemas inversos:

$$\mathbf{Y}_{\text{obs}} = H_{\text{obs}}\mathbf{X}_T + \mathbf{V}, \qquad \underbrace{p(\mathbf{x}_T \mid \mathbf{y}_{\text{obs}})}_{\text{a posteriori}} \propto \underbrace{p(\mathbf{y}_{\text{obs}} \mid \mathbf{x}_T)}_{\text{verosimilitud}} \underbrace{p(\mathbf{x}_T)}_{\text{a priori}},$$

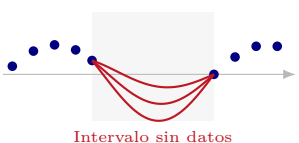
donde H_{obs} selecciona o transforma, y $\mathbf{V}$ modela el ruido de medición. Distintos H_{obs} dan lugar a:

Predicción



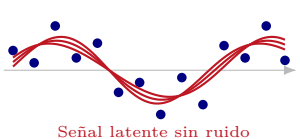
$p(\mathbf{x}_{\text{futuro}} \mid \mathbf{y}_{\text{pasado}})$

Imputación



$p(\mathbf{x}_{\text{faltantes}} \mid \mathbf{y}_{\text{obs}})$

Eliminación de ruido



$p(\mathbf{x}_T \mid \mathbf{y}_{\text{obs}})$

• Observaciones — Trayectorias condicionales

Un marco general de inferencia
(que no siempre es tratable)

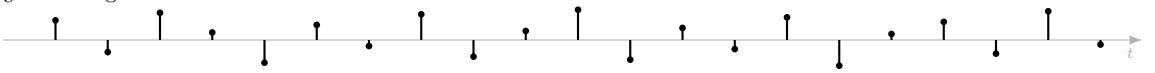
Plan de la charla

Idea recurrente: transformar una fuente aleatoria *simple* en una señal *estructurada*.

I	**Procesos gaussianos**	Pasar de tiempo discretas a continuo Kernels, espectros, muestreo, condicionamiento y aprendizaje
II	**Modelos de flujo**	Aprender transformaciones no lineales Variables latentes, alineación y transporte funcional
III	**Ejemplos**	Formular tareas como problemas inversos Estimación espectral, filtrado pasabanda y deconvolución

Punto de partida: Ruido blanco como entrada

¿Cómo elegir una señal con **máxima variabilidad**?



Definición (Tasa de entropía)

La incertidumbre media por muestra de un proceso estacionario es

$$\bar{h}(Z) = \lim_{t\to\infty} \frac{h(Z_{1:t})}{t} = \lim_{t\to\infty} h(Z_t \mid Z_{1:t-1}),$$

donde $h(Z) = -\mathbb{E}\log p(Z)$. La regla de la cadena y la estacionariedad dan la forma condicional.



$$\bar{h}(Z) = \lim_{t\to\infty} h(Z_t \mid Z_{1:t-1}) \;\le\; h(Z_t) \;\le\; \tfrac{1}{2}\log(2\pi e\sigma^2), \quad \text{donde } \sigma^2 = \mathrm{Var}(Z_t).$$

⋆ **Independencia:** conocer el pasado no reduce la incertidumbre.

⋆ **Gaussianidad:** máxima entropía a varianza fija σ^2.

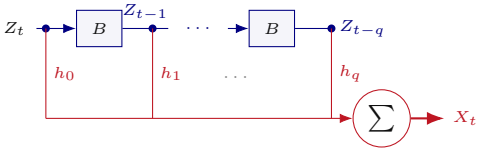
Ruido blanco gaussiano discreto: $Z_t \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$, $t \in \mathbb{Z}$.

Estructura desde el ruido

1) Mezcla lineal (filtro finito).

Z_t: ruido blanco de media cero, $\mathrm{Var}(Z_t) = \sigma^2$.

$$X_t = \sum_{j=0}^{q} h_j Z_{t-j}.$$



B retrasa una muestra: $BZ_t = Z_{t-1}$.

2) La memoria puede ser infinita:

$$X_t = \sum_{j=0}^{\infty} h_j Z_{t-j}, \qquad \sum_{j=0}^{\infty} h_j^2 < \infty.$$

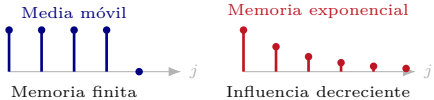
Se requiere varianza finita.

3) El ruido compartido crea correlación.

$$\gamma_X(k) = \mathrm{Cov}(X_t, X_{t-k})$$
$$= \sigma^2 \sum_{j=0}^{\infty} h_j h_{j+|k|}.$$

La dependencia surge del solapamiento entre copias desplazadas de h. Estacionariedad débil.

4) Elegir la memoria mediante h_j.



Media móvil: $h_j = 1/m$ para $0 \le j < m$; cero en otro caso.

Exponencial: $h_j = \phi^j$, $|\phi| < 1$, da

$$X_t = \phi X_{t-1} + Z_t, \qquad \gamma_X(k) = \frac{\sigma^2 \phi^{|k|}}{1 - \phi^2}.$$

I - Procesos gaussianos



Tiempo discreto: X_t, $t \in \mathbb{Z}$

Tiempo continuo: $X(t)$, $t \in \mathbb{R}$

Objetivos

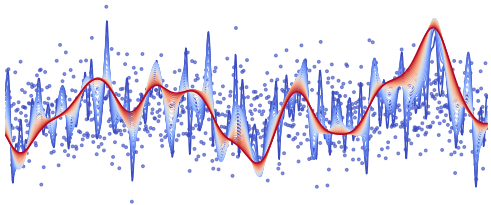
Construir modelos coherentes de trayectorias en tiempo continuo.

Diseñar mediante covarianza o espectro; aprender de los datos.

Extensión a tiempo continuo

Motivación: Observaciones parciales, en una grilla que no controlamos.

Objetivo: Modelar el proceso continuo subyacente, con observaciones en **cualquier conjunto finito de tiempos**.



Tres requisitos para prescindir de la grilla

1. Fuente de ruido

Un análogo continuo del ruido blanco Z_t.

2. Condiciones del filtro

Sustituir la suma ponderada por una integral estocástica bien definida.

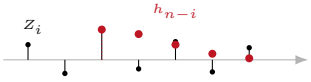
3. Grillas coherentes

Añadir o quitar tiempos preserva la distribución conjunta de los restantes.

Ruido blanco en tiempo continuo

Paso 0: Un filtro desplazado en cada impulso de ruido.

$$X_n = \sum_{i=1}^{N} Z_i h_{n-i}.$$

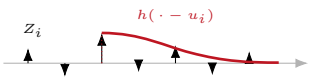


Sumamos copias del filtro con pesos i.i.d. para obtener X_n.

Paso 1: Conservar los impulsos y usar un filtro continuo.

$$Z_{\text{grilla}}(u) = \sum_{i=1}^{N} Z_i \delta(u - u_i),$$

$$X_N(t) = \int h(t-u) Z_{\text{grilla}}(u)\,\mathrm{d}u = \sum_{i=1}^{N} Z_i h(t-u_i).$$

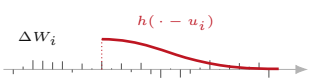


Tiempo continuo, pero solo N coeficientes aleatorios (grados de libertad).

Paso 2: Refinar la grilla y ajustar la escala de los pesos.

$$\Delta W_i = \sqrt{\Delta u_i}\, Z_i, \qquad Z_i \overset{\text{iid}}{\sim} \mathcal{N}(0,1),$$

$$\sum_i h(t-u_i)\Delta W_i \;\longrightarrow\; X(t) = \int h(t-u)\,\mathrm{d}W(u).$$



El browniano W tiene trayectorias continuas no derivables, $W(0) = 0$ e incrementos independientes $\Delta W_i \sim \mathcal{N}(0, \Delta u_i)$.

El proceso gaussiano

Toda colección finita de evaluaciones del proceso de convolución continuo anterior es **conjuntamente gaussiana**.

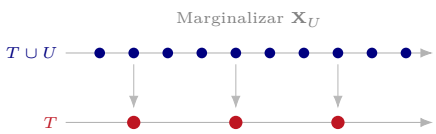
Definición (Proceso gaussiano)

$X \sim \mathcal{GP}(m, k)$ si, para cualquier conjunto finito de tiempos $T = \{t_1, \ldots, t_n\}$,

$$\mathbf{X}_T = \big(X(t_1), \ldots, X(t_n)\big)^\top \sim \mathcal{N}(\mathbf{m}_T, K_T),$$

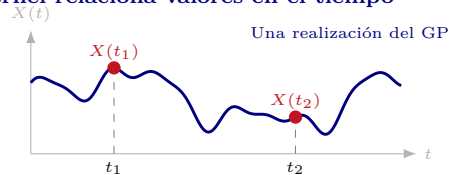
donde $(\mathbf{m}_T)_i = m(t_i)$ y $(K_T)_{ij} = k(t_i, t_j)$.

Coherencia entre grillas



Quitar tiempos preserva la ley conjunta en los restantes. El proceso se define independientemente de la grilla de observación.

El kernel relaciona valores en el tiempo



El kernel $k(t_1, t_2) = \mathrm{Cov}\big(X(t_1), X(t_2)\big)$ controla cómo varían conjuntamente los valores en dos tiempos.

¿Qué determina el kernel?

El filtro determina la covarianza de $X(t) = \int h(t-u)\,\mathrm{d}W(u)$:

$$k(t,s) = \int_{\mathbb{R}} h(t-u)h(s-u)\,\mathrm{d}u, \qquad k(t,t) = \|h\|_2^2 < \infty \quad \text{para } h \in L^2.$$

Elegir directamente un kernel semidefinido positivo k_θ.



Fuerte sesgo inductivo

La familia de kernels elegida especifica correlación, regularidad y oscilaciones en la serie. Sus parámetros se aprenden de las observaciones.

Moore–Aronszajn da una factorización en características $k(t,s) = \langle \Phi(t), \Phi(s) \rangle_{\mathcal{H}}$ para todo kernel semidefinido positivo; no necesariamente es una representación por convolución.

Espectro de potencia: el kernel en frecuencia

Estacionariedad en sentido amplio: media constante, varianza finita y $k(t,s) = \kappa(\tau)$ con $\tau = t - s$.

Definición (Densidad espectral de potencia, DEP)

Para una covarianza estacionaria continua e integrable κ,

$$S_X(\omega) = \int_{\mathbb{R}} \kappa(\tau) e^{-i\omega\tau}\, \mathrm{d}\tau, \qquad \kappa(\tau) = \frac{1}{2\pi} \int_{\mathbb{R}} S_X(\omega) e^{i\omega\tau}\, \mathrm{d}\omega.$$

Covarianza ⇔ espectro de potencia.

$S_X(\omega) \geq 0$ distribuye la potencia de la señan en frecuencias. La potencia total da la varianza:

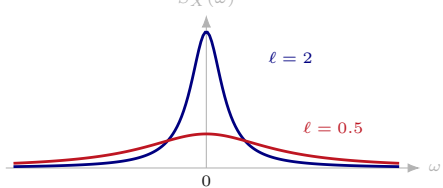
$$\mathrm{Var}[X(t)] = \kappa(0) = \frac{1}{2\pi} \int_{\mathbb{R}} S_X(\omega)\, \mathrm{d}\omega.$$

Desde el filtro: para $X(t) = \int h(t-u)\, \mathrm{d}W(u)$,

$$S_X(\omega) = |H(\omega)|^2, \qquad H = \mathcal{F}[h].$$

Ejemplo: el kernel exponencial.

$$\kappa(\tau) = \sigma^2 e^{-|\tau|/\ell}, \qquad S_X(\omega) = \frac{2\sigma^2 \ell}{1 + \ell^2 \omega^2}.$$

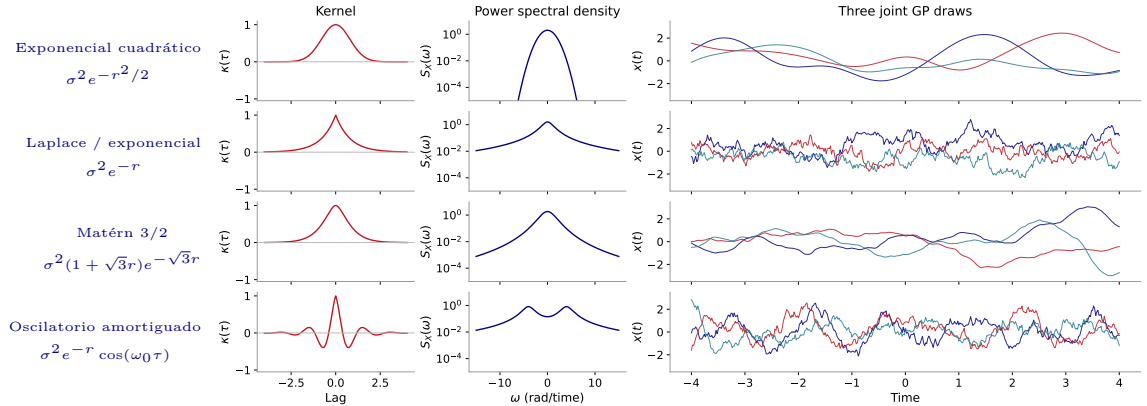


Misma varianza $\sigma^2 = 1$; correlaciones más largas concentran más potencia cerca de frecuencia cero.

Elección del kernel: en tiempo o en frecuencia.

Una biblioteca de kernels

Kernels estacionarios, espectros y realizaciones: $\tau = t - s$, $r = |\tau|/\ell$ y $\kappa(\tau) \overset{\mathcal{F}}{\longleftrightarrow} S_X(\omega)$.



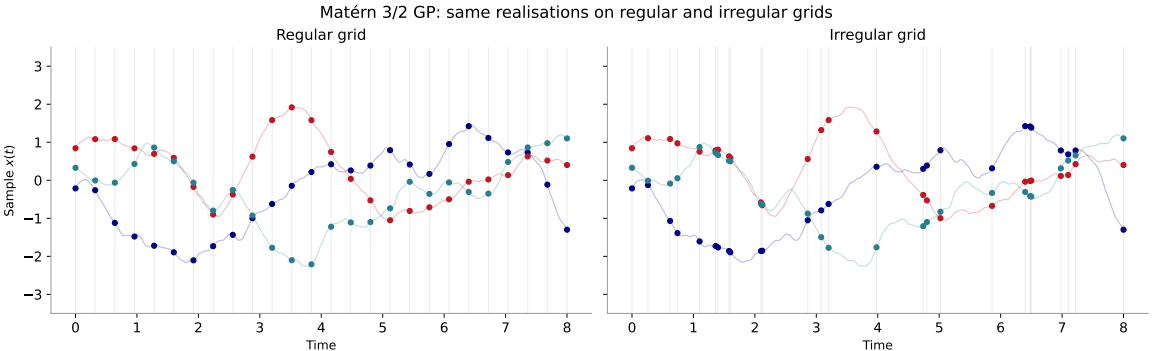
σ^2: varianza; ℓ: escala de longitud; ω_0: frecuencia de oscilación. Aquí $\sigma^2 = 1$.

Construir nuevos kernels: las sumas y los productos puntuales preservan la validez; pesos no negativos controlan sus contribuciones.

Muestreo a priori: grilla, matriz de covarianza y realización

$$T = (t_1, \ldots, t_n) \longrightarrow K_{ij} = k(t_i, t_j) \longrightarrow K_T = LL^\top \longrightarrow \mathbf{X}_T = \mathbf{m}_T + L\mathbf{U}$$

$$\mathbf{U} \sim \mathcal{N}(0, I_n) \quad \Longrightarrow \quad \mathbf{X}_T \sim \mathcal{N}(\mathbf{m}_T, K_T).$$



Muestreo conjunto: en ambos casos se usaron las mismas tres realizaciones. Tiempos compartidos = valores idénticos.

Condicionar: actualización Bayesiana

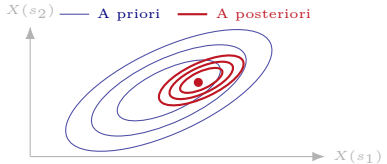
Observaciones: $\mathbf{t} = (t_1, \ldots, t_n)^\top$, $\mathbf{y} = (y_1, \ldots, y_n)^\top$, con $y_i = X(t_i) + v_i$, $v_i \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_n^2)$.

Sean $\mathbf{m} = (m(t_i))_{i=1}^n$, $\mathbf{k}_t = (k(t, t_i))_{i=1}^n$, y $C_{ij} = k(t_i, t_j) + \sigma_n^2 \delta_{ij}$.

$$m_{\text{post}}(t) = m(t) + \mathbf{k}_t^\top C^{-1}(\mathbf{y} - \mathbf{m}),$$

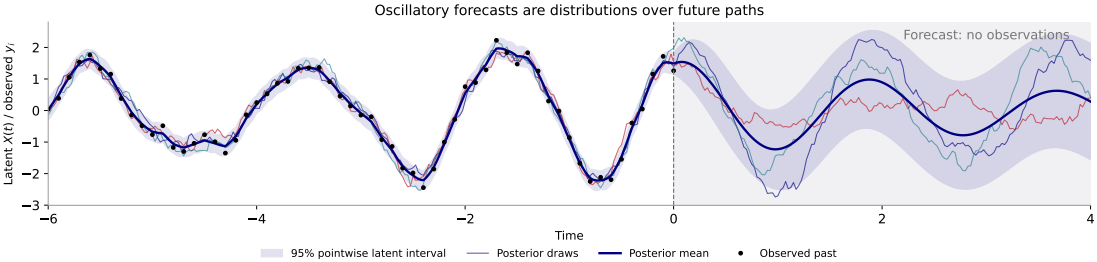
$$k_{\text{post}}(t, s) = k(t, s) - \mathbf{k}_t^\top C^{-1} \mathbf{k}_s.$$

El carácter no paramétrico: los momentos posteriores incorporan un término adicional por cada observación; su representación crece con los datos.



Esquema: una trayectoria da $(X(s_1), X(s_2))$

Ejemplo: kernel oscilatorio amortiguado $k(t, s) = e^{-|t-s|/4} \cos\Big(2\pi(t-s)/1.8\Big)$, $m(t) = 0$, $\sigma_n^2 = 0.04$.



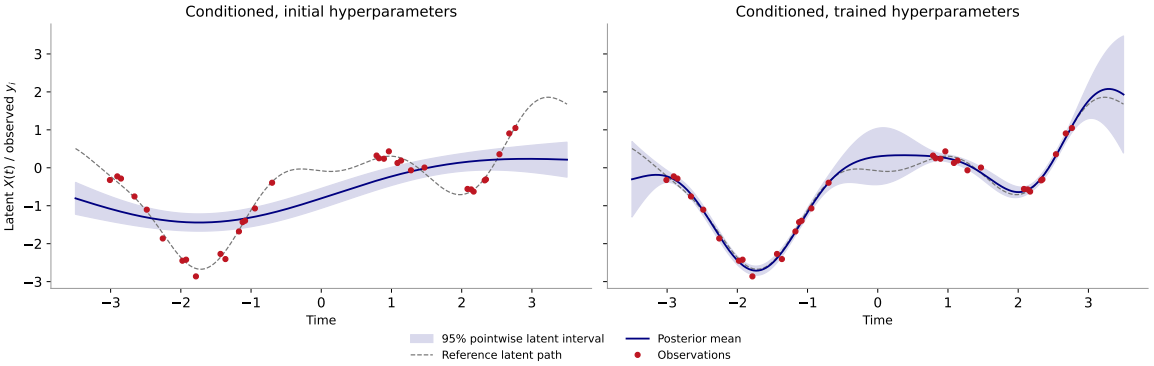
Curvas finas: **realizaciones a posteriori**. Curva gruesa: media a posteriori. Banda: intervalo creíble puntual del 95 %.

Entrenamiento: elegir el kernel (experto) y sus parámetros (optimización)

Verosimilitud exacta: $Y \sim \mathcal{N}(m_T, C_\theta)$, $C_\theta = K_\theta(T,T) + \sigma_n^2 I$.

$$\mathcal{L}(\theta) = \underbrace{\tfrac{1}{2} r^\top C_\theta^{-1} r}_{\text{ajuste a datos}} + \underbrace{\tfrac{1}{2}\log|C_\theta|}_{\text{volumen de covarianza}} + \tfrac{n}{2}\log(2\pi), \qquad r = y - m_T.$$

Sesgo inductivo del kernel ⇒ pocos parámetros definen un modelo de dimensión infinita. De otro modo, aprender sería inviable.



Ambos modelos usan los mismos datos. El entrenamiento solo cambia los hiperparámetros.

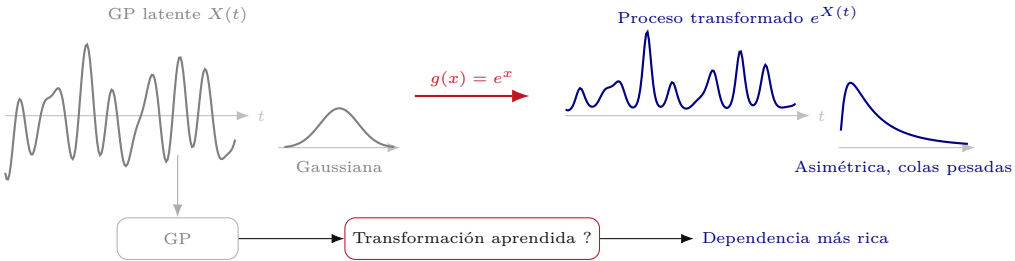
Procesos gaussianos: base para modelos más expresivos

Ventajas

- Coherencia entre grillas y covarianza interpretable.
- Condicionamiento y entrenamiento exactos con ruido gaussiano.

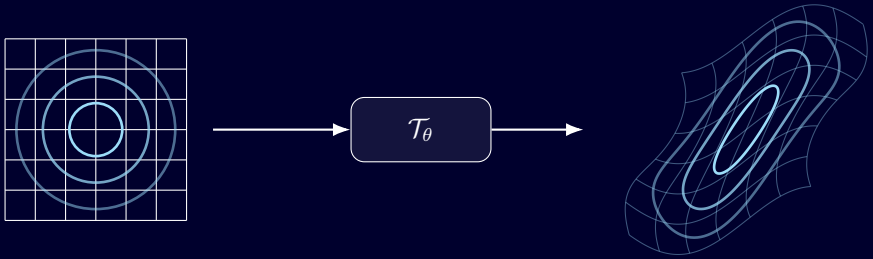
Limitaciones

- Elección manual del kernel y «solo gaussiano».
- Coste computacional cúbico en el número de datos.



A continuación: aprender transformaciones de GP con distintos niveles de complejidad.

II - Modelos de flujo



Distribución base simple

Aprender la transformación

Estructura conjunta más rica

Objetivos de aprendizaje

Generar series temporales mediante transporte no lineal aprendido.

Usar representaciones y física para modelar la estructura de la señal.

GP latentes: verosimilitudes no gaussianas

Inferencia exacta requiere **verosimilitud gaussiana**, sin embargo podemos considerar otras verosimilitudes manteniendo la **independencia condicional**.

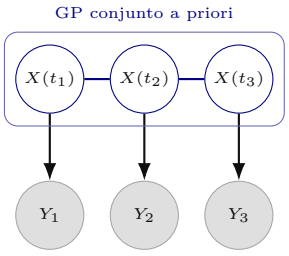
$$X \sim \mathcal{GP}(m, k), \qquad p(\mathbf{y} \mid \mathbf{X}_T) = \prod_i p(y_i \mid X(t_i)).$$

Regresión: $Y_i \mid X(t_i) \sim \mathcal{N}(X(t_i), \sigma_n^2)$,

Clasificación: $Y_i \mid X(t_i) \sim \text{Bernoulli}(s(X(t_i)))$,
$s(a) = (1 + e^{-a})^{-1}$,

Conteos: $Y_i \mid X(t_i) \sim \text{Poisson}\left(X(t_i)^2\right)$,

Valores positivos: $Y_i \mid X(t_i) \sim \text{Gamma}\left(\alpha, \alpha e^{-X(t_i)}\right)$.



- Modelar dependencia temporal (con GPs) tiene aplicaciones **va más allá del problema de regresión.**
- Las verosimilitudes no gaussianas suelen requerir **aproximaciones para entrenamiento e infrencia**:

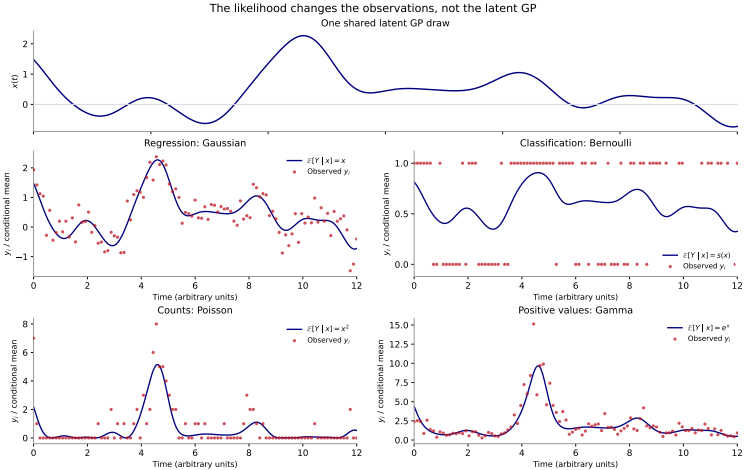
$$p(\mathbf{x}_T \mid \mathbf{y}) = \frac{p(\mathbf{y} \mid \mathbf{x}_T)p(\mathbf{x}_T)}{p(\mathbf{y})}, \qquad \underbrace{p(\mathbf{y}) = \int p(\mathbf{y} \mid \mathbf{u})p(\mathbf{u})\,\mathrm{d}\mathbf{u}}_{\text{generalmente intratable}}.$$

La constante de normalización es intratable, y requiere aproximaciones de Laplace, variacionales o de Monte Carlo.

Rasmussen y Williams (2006), GPML, capítulo 3.

Ejemplo: GP latente con distintas verosimilitudes

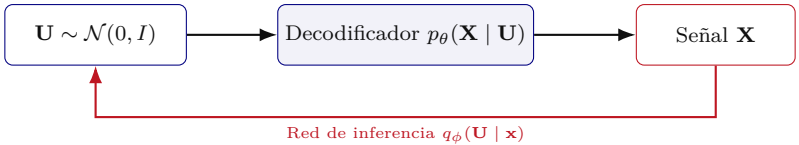
La misma trayectoria latente genera los cuatro casos; la verosimilitud determina el rango y la variabilidad de las observaciones.



Azul: media condicional. **Rojo**:: observaciones simuladas. Una dinámica latente suave puede generar observaciones ruidosas, binarias, de conteo o positivas.

Tranformación mediante Autoencoder Variacional (VAE)

[Vuelta al caso finito] Sustituimos el filtro lineal por un **decodificador no lineal** que recibe un vector gaussiano.



$$\log p_\theta(\mathbf{x}) \;\geq\; \underbrace{\mathbb{E}_{q_\phi(\mathbf{U}|\mathbf{x})}[\log p_\theta(\mathbf{x} \mid \mathbf{U})]}_{\text{reconstrucción}} - \underbrace{\mathrm{KL}\big(q_\phi(\mathbf{U} \mid \mathbf{x}) \| p(\mathbf{U})\big)}_{\text{regularización latente}} = \mathrm{ELBO}.$$

Ejemplo gaussiano: $p(\mathbf{U}) = \mathcal{N}(0, I)$, $q_\phi(\mathbf{U} \mid \mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_\phi, \mathrm{diag}(\boldsymbol{\sigma}_\phi^2))$, $p_\theta(\mathbf{x} \mid \mathbf{U}) = \mathcal{N}(f_\theta(\mathbf{U}), \sigma_{\mathrm{obs}}^2 I)$. Aquí f_θ es una red neuronal, σ_{obs}^2 es fija, y $\boldsymbol{\mu}_\phi, \boldsymbol{\sigma}_\phi$ son las salidas del codificador en $\mathbf{x}$.

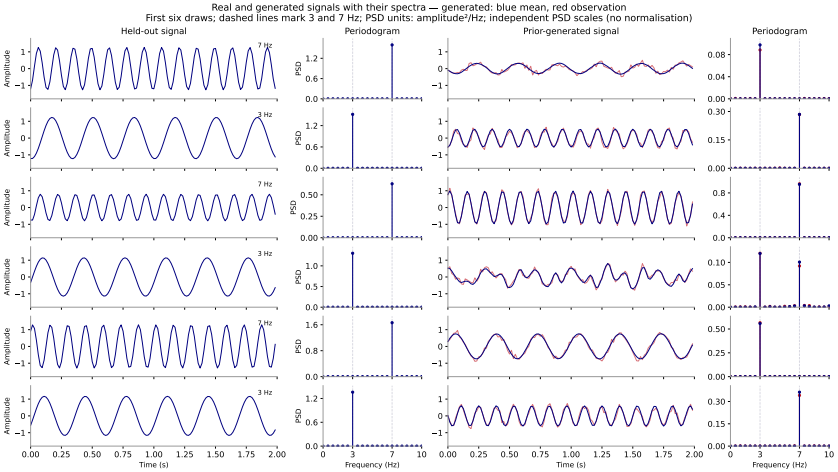
$$-\mathrm{ELBO} = \underbrace{\frac{1}{2\sigma_{\mathrm{obs}}^2}\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,I)}\,\|\mathbf{x} - f_\theta(\boldsymbol{\mu}_\phi + \boldsymbol{\sigma}_\phi \odot \boldsymbol{\epsilon})\|^2}_{\text{error de reconstrucción}} + \underbrace{\frac{1}{2}\sum_{j=1}^{d}(\mu_{\phi,j}^2 + \sigma_{\phi,j}^2 - 1 - \log\sigma_{\phi,j}^2)}_{\text{KL gaussiana analítica}} + \frac{n}{2}\log(2\pi\sigma_{\mathrm{obs}}^2).$$

d: dimensión del código; n: dimensión de la señal.

La reconstrucción no garantiza leyes físicas ni dinámica temporal. Kingma y Welling (2013), Auto-Encoding Variational Bayes.

Experimento VAE: ¿se preserva la dinámica de las series?

Ajuste con 3000 sinusoides de 3 o 7 Hz, con amplitud y fase aleatorias (128 muestras, 2 segundos). VAE convolucional: código 3D, codificador y decodificador residuales.

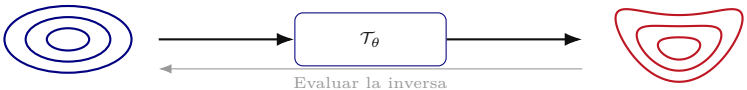


Datos de entrenamiento (izquierda) versus realizaciones sintéticas a priori (derecha)
La mayoría de las señales recupera una frecuencia, **existen errores de amplitud y mezcla ocasional de modos**.

Flujos normalizantes (NF): difeomorfismos y verosimilitud exacta

El VAE parece demasiado ambicioso. Consideremos una biyección diferenciable entre objetos de la **misma dimensión**:

$$\mathbf{X} = \mathcal{T}_\theta(\mathbf{X}_0), \qquad \mathbf{X}_0 \sim p_0, \qquad \det D\mathcal{T}_\theta \neq 0.$$



$$p_\theta(\mathbf{x}) = p_0\left(\mathcal{T}_\theta^{-1}(\mathbf{x})\right)\left|\det D\mathcal{T}_\theta^{-1}(\mathbf{x})\right|.$$

Para $\mathcal{T} = \mathcal{T}_L \circ \cdots \circ \mathcal{T}_1$, los logaritmos de los determinantes se suman:

$$\log p_\theta(\mathbf{x}_L) = \log p_0(\mathbf{x}_0) - \sum_{\ell=1}^{L} \log\left|\det D\mathcal{T}_\ell(\mathbf{x}_{\ell-1})\right|.$$

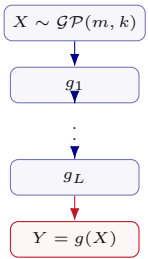
Si la inversa y el determinante son tratables: **ajuste por máxima verosimilitud exacta** (sin integrar variables latentes).

Papamakarios et al. (2021), Normalizing Flows for Probabilistic Modeling and Inference.

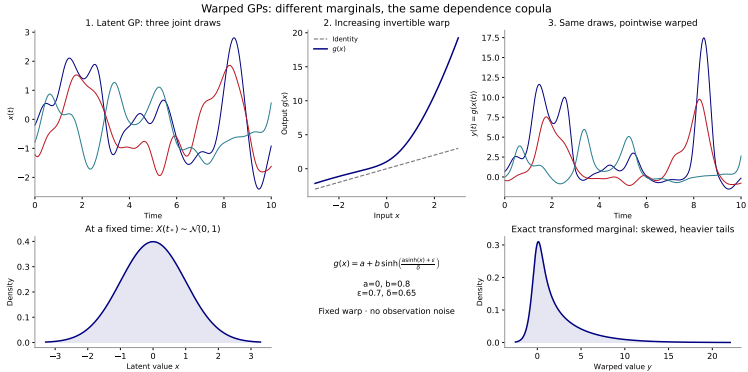
Ejemplo: GP con deformaciones compuestas

La idea de los NF se aplica a los GP mediante **transformaciones por coordenadas**. Consideremos $X \sim \mathcal{GP}(m, k)$, $Y(t) = g(X(t))$, con una función puntual invertible $g = g_L \circ \cdots \circ g_1$. En una grilla con K_T no singular:

$$p_Y(\mathbf{y}) = \mathcal{N}\left(g^{-1}(\mathbf{y}); \mathbf{m}_T, K_T\right) \prod_{i=1}^{n} \left|(g^{-1})'(y_i)\right| .$$



Funciones escalares compartidas en cada instante t



Marginales no gaussianas con densidad tratable. Las deformaciones crecientes preservan la cópula gaussiana. Este ejemplo sin ruido usa una deformación sinh–arcsinh.

Rios y Tobar (2019), Compositionally-Warped Gaussian Processes.

Flujos normalizantes continuos

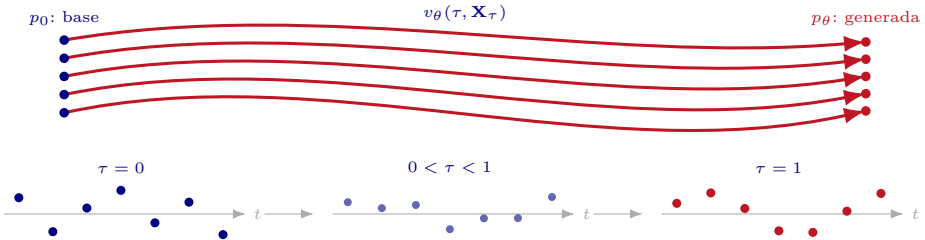
[Vuelta al caso finito] Consideremos una sequencia $\mathbf{X}_0, \mathbf{X}_1, \mathbf{X}_2, \ldots \in \mathbb{R}^n$ tal que



Flujo discreto: componer funciones

Cada capa transforma $\mathbf{X}_k \mapsto \mathbf{X}_{k+1}$. Escribiendo $\mathbf{X}_{k+1} - \mathbf{X}_k = \Delta\tau\, v_\theta(\tau_k, \mathbf{X}_k)$, el límite $\Delta\tau \to 0$ da una **EDO neuronal**:

$$\frac{\mathrm{d}\mathbf{X}_\tau}{\mathrm{d}\tau} = v_\theta(\tau, \mathbf{X}_\tau), \qquad \mathbf{X}_0 \sim p_0, \qquad \mathbf{X}_1 = \mathcal{T}_\theta(\mathbf{X}_0).$$



Dos relojes: t indexa las muestras; τ transporta la señal (no describe su dinámica física).

Flujo biyectivo: v_θ es Lipschitz en el estado, con soluciones en todo $[0, 1]$; la propia red no necesita ser invertible.

Chen et al. (2018), Neural Ordinary Differential Equations.

Ajuste de flujos condicional: regresión sobre el campo de velocidades

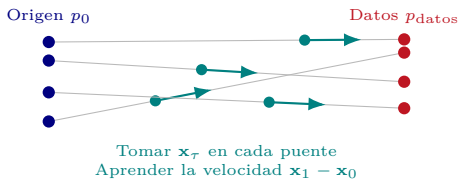
Tomamos extremos independientes $\mathbf{X}_0 \sim p_0$ y $\mathbf{X}_1 \sim p_{\text{datos}}$, y $\tau \sim \mathcal{U}(0,1)$. El puente lineal condicionado en los extremos tiene velocidad conocida:

$$\mathbf{X}_\tau = (1-\tau)\mathbf{X}_0 + \tau \mathbf{X}_1, \qquad \dot{\mathbf{X}}_\tau = \mathbf{X}_1 - \mathbf{X}_0.$$

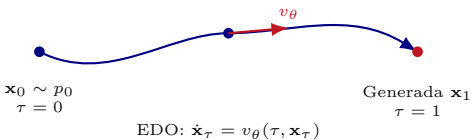
$$\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}\left[\left\| v_\theta(\tau, \mathbf{X}_\tau) - (\mathbf{X}_1 - \mathbf{X}_0) \right\|^2 \right].$$

En el óptimo poblacional, $v^*(\tau, \mathbf{x}) = \mathbb{E}[\mathbf{X}_1 - \mathbf{X}_0 \mid \mathbf{X}_\tau = \mathbf{x}]$: el promedio de las velocidades de los puentes transporta sus distribuciones marginales.

Ajuste: regresión supervisada de velocidad



Generación: integrar el campo aprendido



Partimos de $\mathbf{x}_0 \sim p_0$ e integramos de 0 a 1.

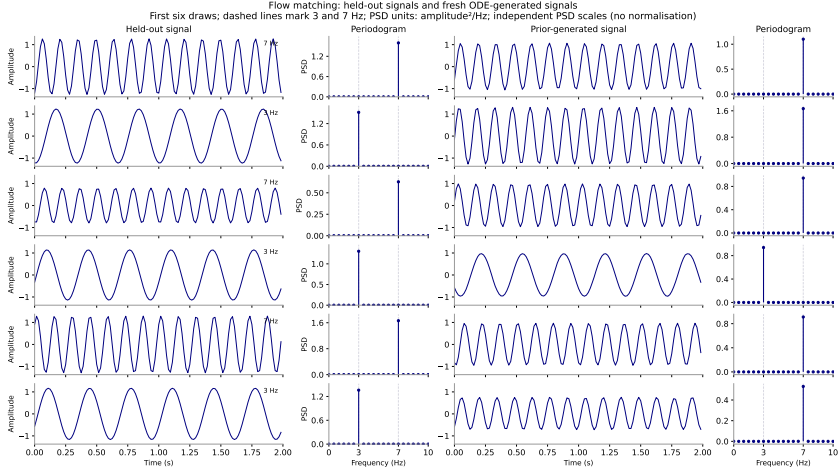
No se requiere resolver EDO durante entrenamiento.

¿Es este el transporte natural para series de tiempo?.

Lipman et al. (2023), Flow Matching; Tong et al. (2024), generalised CFM.

Ajuste de flujos: generar las mismas señales de dos frecuencias

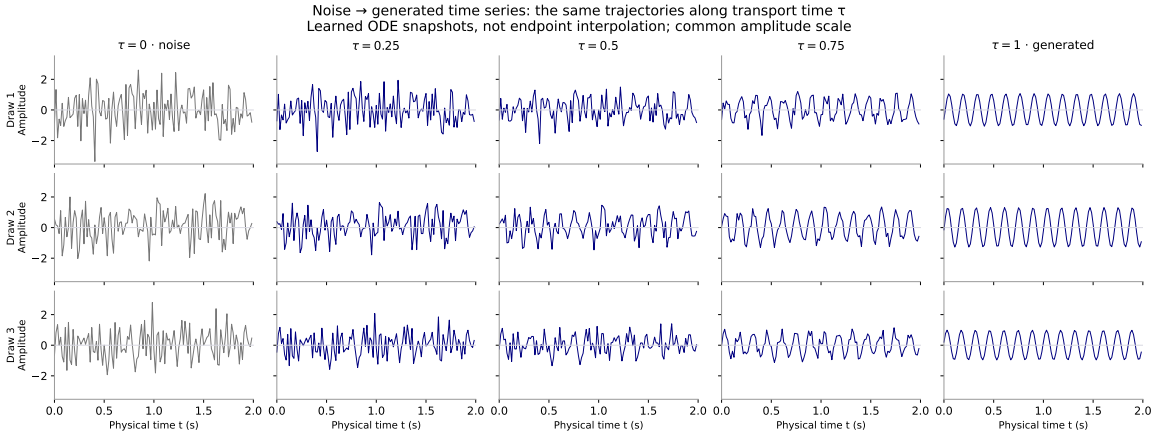
Mismos datos que en la diapositiva 21; una U-Net 1D residual condicionada en el tiempo aprende la velocidad. Generamos desde ruido gaussiano nuevo resolviendo la EDO.



99.6 % de la energía generada está en 3 o 7 Hz; espectros con escalas verticales independientes.
El **balance de frecuencias y las amplitudes aún son imperfectos**.

Ajuste de flujos: del ruido a una señal oscilatoria

Cada fila sigue la misma realización de ruido por la EDO aprendida.



Las columnas avanzan en el tiempo de transporte τ; cada eje horizontal es el tiempo físico t. Escala común de amplitud.

Desafío: ¿qué propiedades queremos preservar, o *alinear*?

Un generador científicamente útil debe reproducir las propiedades relevantes para la aplicación: espectros, dependencia temporal, restricciones o leyes físicas.

Por construcción
Restricciones de soporte, simetrías, decodificadores estructurados

Solo garantiza lo que impone la construcción

En el objetivo
Discrepancias espectrales, residuos físicos

Las penalizaciones favorecen, pero no garantizan

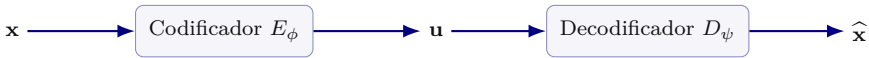
El **problema de la alineación** también surge en modelos de lenguaje: ¿el sistema aprendido se comporta como se espera?

En IA para la ciencia, las propiedades físicas y dinámicas requieren definiciones y evaluaciones cuantitativas.

Flujos normalizantes latentes: comprimir, transportar, decodificar

Representamos cada señal $\mathbf{x} \in \mathbb{R}^n$ mediante un código de menor dimensión $\mathbf{u} = E_\phi(\mathbf{x}) \in \mathbb{R}^r$, con $r \ll n$.

1. Aprender una representación: codificar y reconstruir



2. Generar: transportar ruido a los códigos y decodificar



Ajustamos el flujo con códigos $E_\phi(\mathbf{x})$ del autocodificador.

Menor costo computacional

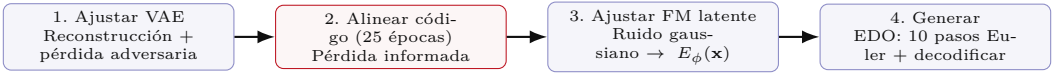
Aprender e integrar en r dimensiones, en vez de n.

Controlar lo que preserva el código

Alinear el VAE con las propiedades de la señal, como su estructura espectral, antes de ajustar el flujo.

Ejemplo: Ajuste de flujos latentes con alineación espectral

Problema: la compresión latente puede introducir artefactos espectrales.
Idea: alinear el VAE con las propiedades de la señal **antes de** ajustar el flujo.



Asignar capacidad de reconstrucción al detalle. Con $\widehat{\mathbf{x}} = D_\psi(E_\phi(\mathbf{x}))$, añadimos

$$\mathcal{L}_T = \sum_k w_k^2 \left\| T_k(\mathbf{x}) - T_k(\widehat{\mathbf{x}}) \right\|^2.$$

Opciones para T: **Fourier**, **Ondículas** y **Signaturas**.

Los pesos controlan el papel de las frecuencias objetivo.

Evidencia: menos artefactos espectrales.

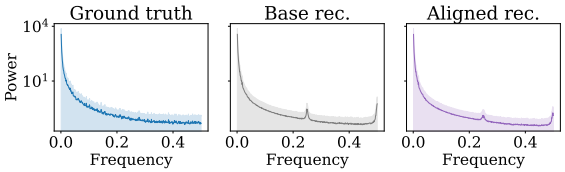


Fig. 1 del artículo: tipos de cambio, *reconstrucciones*; espectros de potencia, media ±1 DE (escala log). La alineación atenúa el pico espurio cerca de 0.25.

Resultados de generación: la generación espectralmente coherente mejora el rendimiento en tareas posteriores.

Idea clave: dar forma a la representación para preservar la estructura de la señal, y después aprender el transporte.

Carvajal Reyes y Tobar (2026), *Time series generation with spectrally aligned latent flow matching.* Fig. 1, tabla 1 y secciones 3–6.

Desafío: La generación condicional es intratable

Condicionamos una señal completa $\mathbf{x}$ en observaciones $\mathbf{y}$, con tiempos, canales y máscara conocidos.



Con un modelo de observación $\mathbf{y} = H\mathbf{x} + \boldsymbol{\varepsilon}$:

$$p_\theta(\mathbf{x} \mid \mathbf{y}) = \frac{p_\theta(\mathbf{x}, \mathbf{y})}{p_\theta(\mathbf{y})} \propto \underbrace{p(\mathbf{y} \mid \mathbf{x})}_{\substack{\text{verosimilitud} \\ \text{de observación}}} \quad \underbrace{p_\theta(\mathbf{x})}_{\substack{\text{distribución a priori} \\ \text{FM aprendida}}} .$$

Dos vías: aprender un generador condicional o inferir con la distribución a priori fija.

Aprender flujo como función del contexto.
Aprender $v_\phi(\tau, \mathbf{x}_\tau, \mathbf{y})$ para aproximar $p_\theta(\mathbf{x} \mid \mathbf{y})$. Construir pares de ajuste enmascarando o dividiendo series (y añadiendo ruido de medición cuando corresponda). Generar con $\mathbf{y}$ fijo.

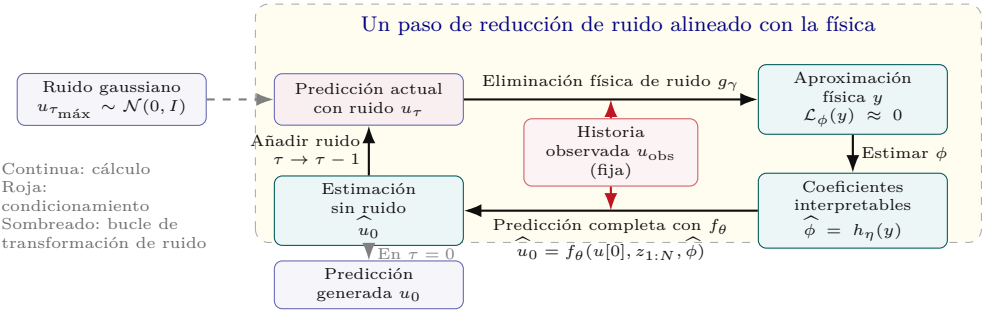
MCMC con propuesta a priori: sin reajuste.
Proponer $\mathbf{x}' \sim p_\theta$ con el modelo FM ajustado; aceptar con probabilidad

$$\text{mín}\left(1, \frac{p(\mathbf{y} \mid \mathbf{x}')}{p(\mathbf{y} \mid \mathbf{x})}\right).$$

Solo se evalúa la verosimilitud.

Ejemplo: difusión interpretable alineada con la física (PAID)

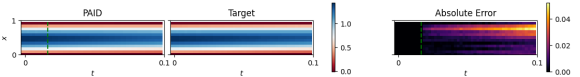
Objetivo: generar predicciones de $\mathcal{F}_\vartheta(u) + \mathcal{L}_\phi(u) = 0$: $\mathcal{F}_\vartheta$ desconocido, $\mathcal{L}_\phi$ conocido, con coeficientes ϕ por inferir.



Ejemplo: difusión–advección.

$$\underbrace{u_t - \beta u_{xx}}_{\text{forma difusiva conocida}} + \underbrace{\alpha u_x}_{\text{advección no modelada}} = 0.$$

Inferir β y aprender correcciones para la dinámica no modelada. Convención habitual: $\beta > 0$.



Ejemplo: predicción PAID, objetivo y error absoluto. Verde discontinuo: fin de la historia observada.

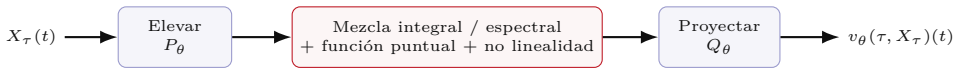
Purchase & Tobar, *Physics-Aligned Interpretable Diffusion*, en preparación.

Idea al cierre: flujos normalizantes sobre funciones

Transportes sobre funciones, no solo vectores en una grilla fija. El ajuste de flujos funcionales sigue la idea de regresión de velocidad:

$$X_\tau = (1-\tau)X_0 + \tau X_1, \qquad \mathcal{L}(\theta) = \mathbb{E}\left\| v_\theta(\tau, X_\tau) - (X_1 - X_0) \right\|_{\mathcal{H}}^2.$$

Un **operador neuronal** representa la velocidad funcional $v_\theta(\tau, \cdot) : \mathcal{H} \to \mathcal{H}$:



Repetir capas de mezcla; incluir t y τ como entradas.

$$\partial_\tau X_\tau = v_\theta(\tau, X_\tau), \qquad X_0 \sim \text{una base GP adecuada.}$$

Objetivo: generar una señal subyacente y evaluarla en distintas grillas, incluidos nuevos tiempos de consulta y resoluciones.

Nota: Esto requiere más tiempo del que nos queda, lo podemos revisar en el próximo tutorial :)

Kerrigan, Migliorini y Smyth (2024), Functional Flow Matching. Kovachki et al. (2023), Neural Operator.

Qué aportan las transformaciones y cuáles son sus límites

Construcción	Objeto aleatorio	Aprendizaje / inferencia	Estructura y límites
GP + verosimilitud	Función latente; observaciones	Verosimilitud marginal; distribución a posteriori a menudo aproximada	GP a priori coherente; la verosimilitud cambia las marginales observadas, no la gaussianidad latente.
VAE	Código latente y señal decodificada	Cota variacional	Decodificador flexible; reconstruir no garantiza dinámica ni coherencia entre grillas.
Flujo finito	Vectores de igual dimensión	Verosimilitud por cambio de variables	Función invertible y densidad tratable por diseño; sin invariantes científicos automáticos.
CNF / CFM	Transporte de vectores	campo de velocidades	Flujo de EDO bajo regularidad; el tiempo físico difiere del tiempo de transporte.
Flujo latente	Código y luego señal decodificada	VAE + transporte latente	El decodificador puede imponer restricciones; la compresión puede perder estructura relevante.
Flujo funcional	Funciones aleatorias	Regresión de velocidad funcional	Formulación en espacios funcionales; la regularidad y la convergencia numérica requieren cuidado.

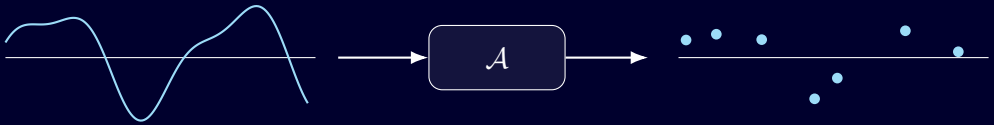
Lo que no vimos

Modelos multicanal
Correlaciones cruzadas y estructura latente compartida.

Distancias entre series temporales
El problem de interés determinan la geometria apropiada.

Modelos fundacionales
Preentrenamiento y adaptación a nuevos dominios, contextos y tareas.

III - Ejemplos: problemas inversos



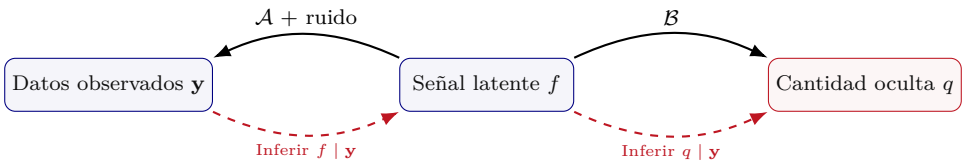
Objetivos de aprendizaje

Resolver problemas inversos al condicionar un modelo generativo de señales.

Modelos generativos: inversión mediante actualizaciones bayesianas

El operador directo $\mathcal{A}$ filtra, difumina, mezcla o integra; luego muestrea u oculta valores. Invertimos su efecto para inferir la señal latente o una transformación suya:

$$f \sim \mathcal{GP}(0, k), \qquad \underbrace{\mathbf{y} = \mathcal{A}f + \boldsymbol{\varepsilon}}_{\text{observaciones}}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(0, R), \qquad \underbrace{q = \mathcal{B}f}_{\text{cantidad por recuperar}} \quad .$$



Nos centramos en: Los operadores lineales $\mathcal{A}$ preservan la gaussianidad conjunta. Sus covarianzas cruzadas dan la distribución a posteriori:

$$q \mid \mathbf{y} \sim \mathcal{N}\left(K_{qy}C_y^{-1}\mathbf{y},\ K_{qq} - K_{qy}C_y^{-1}K_{yq}\right), \qquad C_y = K_{\mathcal{A}f,\mathcal{A}f} + R.$$

Tarea	Mediciones $\mathcal{A}f$	Objetivo $\mathcal{B}f$
Estimación espectral	Muestras puntuales	Coeficientes de Fourier
Filtrado pasabanda	Muestras puntuales	Componente de la banda elegida
Deconvolución	Muestras de una convolución	Señal fuente original

Aplicación 1: Estimación espectral bayesiana no paramétrica

Objetivo: Recuperar un espectro a partir de muestras temporales incompletas y ruidosas.

Modelamos la señal con un GP estacionario e inferimos su representación de Fourier:

$$f \sim \mathcal{GP}(0, k), \qquad y_i = f(t_i) + \varepsilon_i, \qquad F_\alpha(\xi) = \int_{\mathbb{R}} e^{-\alpha t^2} f(t) e^{-2\pi i \xi t} \, \mathrm{d}t, \quad \alpha > 0.$$

La transformada es lineal.

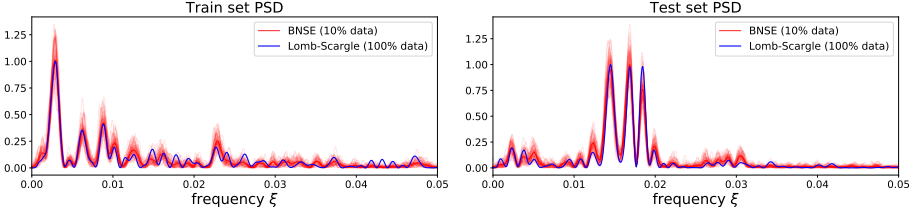
Las observaciones y los coeficientes de Fourier son conjuntamente gaussianos. El sampleo se puede hacer de forma exacta.

Propagar incertidumbre a la potencia espectral.

Para $S_\alpha(\xi) = |F_\alpha(\xi)|^2$,

$$\mathbb{E}[S_\alpha(\xi) \mid \mathbf{y}] = |m_F(\xi)|^2 + v_F(\xi).$$

m_F: media a posteriori; v_F: varianza compleja a posteriori.



Ejemplo: espectros de frecuencia cardíaca. BNSE usa el 10 % de los datos irregulares y ruidosos; Lomb–Scargle usa la serie completa sin ruido. El sombreado a posteriori refleja la incertidumbre por las observaciones limitadas.

Felipe Tobar (2018), *Bayesian Nonparametric Spectral Estimation*, NeurIPS.

Aplicación 2: Filtrado pasabanda con el kernel sinc

Objetivo: Recuperar una componente armónica a partir de muestras irregulares y ruidosas.

Descomponemos la señal en GP independientes dentro y fuera de una banda B:

$$f = f_B + f_{\bar{B}}, \qquad y_i = f(t_i) + \varepsilon_i, \qquad k_B = \mathcal{F}^{-1}\left[S_f(\xi)\,\mathbf{1}_B(\xi)\right].$$

Diseñar la distribución a priori en frecuencia.

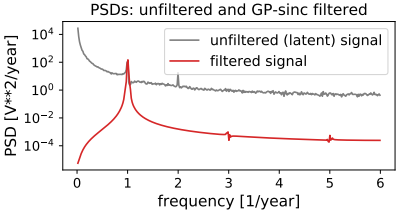
Un par de bandas espectrales rectangulares da

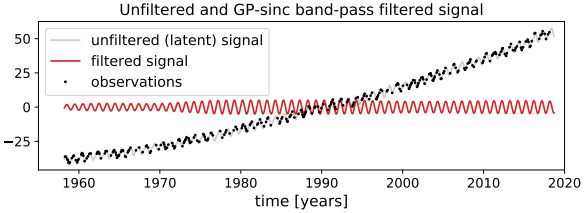
$$k_{\text{sinc}}(\ell) = \sigma^2 \,\text{sinc}(\Delta\ell)\cos(2\pi\xi_0\ell).$$

Ancho de banda Δ, centro ξ_0; $\text{sinc}(z) = \sin(\pi z)/(\pi z)$.

Filtrar equivale a inferir a posteriori.

$\text{Cov}(f_B, f) = k_B$ da la covarianza cruzada para $p(f_B \mid \mathbf{y})$. La salida es una distribución de señales de banda limitada en tiempo continuo.





Ejemplo: CO_2 de Mauna Loa. Componente anual recuperada con 200 de 727 observaciones; $\xi_0 = 1$ año^{-1}, $\Delta = 0.1$ año^{-1}.

Felipe Tobar (2019), *Band-Limited Gaussian Processes: The Sinc Kernel*, NeurIPS.

Aplicación 3: Deconvolución con procesos gaussianos

Objectivo: Recuperar una fuente latente pese al desenfoque, el ruido y las observaciones faltantes.

$$x \sim \mathcal{GP}(0, k_x), \qquad y_i = (h * x)(t_i) + \varepsilon_i = \int h(t_i - s)x(s)\,\mathrm{d}s + \varepsilon_i.$$

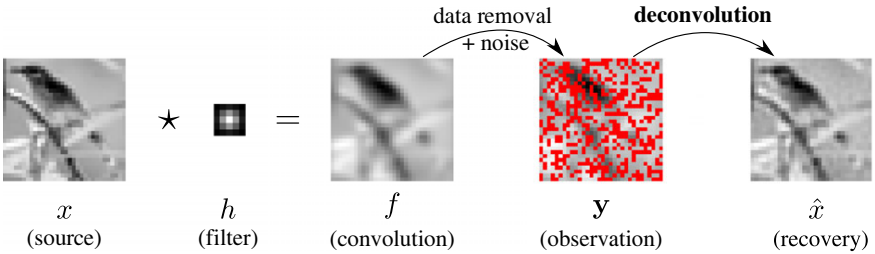


Fig. 1 del artículo: ejemplo en imágenes; la misma construcción se aplica a series temporales.

Aplicar el operador directo a la covarianza.

$$k_{xy}(t, t_i) = \int h(t_i - s)k_x(t, s)\,\mathrm{d}s.$$

Condicionamos el GP conjunto para recuperar $x \mid \mathbf{y}$.

El filtro determina qué se puede recuperar.

- ⋆ Las frecuencias eliminadas por h no son recuperables
- ⋆ Filtros conocidos ⇒ posterior exacta
- ⋆ Filtros desconocidos ⇒ aprendizaje y pueden no ser identificables.

Tobar, Robert y Silva (2023), *Gaussian process deconvolution*, Proc. R. Soc. A 479:20220648.

Fin

Modelado generativo de series temporales

Declaración de uso de IA

Se utilizó OpenAI Codex como apoyo para el formato en LaTeX, en particular las figuras en TikZ; los experimentos en Python; y la traducción de algunas partes del inglés al español.

Recursos adicionales

Apuntes de clase en español

- **Aprendizaje de máquinas**: notas de clase (PDF).
- **Estadística**: notas de clase (PDF).

Más material docente: cursos, ejemplos y videos.

Curso de modelos generativos probabilísticos (en inglés)

- **Probabilistic Generative Models**: curso y notas de clase.

Esta presentación está disponible en:

`https://www.ma.ic.ac.uk/~ft410/talks/`