

Difusão: um relato honesto do que acho que sei

Thiago Rodrigo Ramos

28/12/2025



Objetivo



Objetivo



A imagem abaixo foi gerada por um modelo de difusão.
Nosso objetivo é entender como.

Equações Diferenciais Estocásticas

- Uma EDE descreve a evolução de um sistema com ruído.
- O ruído é modelado pelo movimento browniano W_t .
- Forma geral:

$$dX_t = b(X_t, t) dt + \sigma(X_t, t) dW_t.$$

- Chamamos b de *drift* e σ de *coeficiente de difusão*.

Equações Diferenciais Estocásticas

- Uma EDE descreve a evolução de um sistema com ruído.
- O ruído é modelado pelo movimento browniano W_t .
- Forma geral:

$$dX_t = b(X_t, t) dt + \sigma(X_t, t) dW_t.$$

- Chamamos b de *drift* e σ de *coeficiente de difusão*.

Calma!!! Já vamos entender isso.

Passo 1: Movimento Browniano

$$dX_t = b(X_t, t) dt + \sigma(X_t, t) dW_t.$$

- ❶ Processo estocástico contínuo $\{W_t\}_{t \geq 0}$ com $W_0 = 0$.
- ❷ Incrementos independentes: $W_t - W_s$ indep. de \mathcal{F}_s .
- ❸ Incrementos gaussianos:

$$W_t - W_s \sim \mathcal{N}(0, t - s).$$

- ❹ Trajetórias quase certamente contínuas, mas não diferenciáveis.

O mais importante pra gente são as propriedades 2 e 3.

Passo 2: Discretizando

$$dX_t = b(X_t, t) dt + \sigma(X_t, t) dW_t.$$

- O símbolo d indica uma pequena variação.
- dt é um pequeno passo no tempo: Δ .
- dX_t é a pequena variação de X_t :

$$dX_t \approx X_{t+\Delta} - X_t.$$

$$dX_t = b(X_t, t) dt + \sigma(X_t, t) dW_t.$$

- Um pequeno incremento do browniano é:

$$dW_t = W_{t+\Delta} - W_t.$$

- O browniano tem incrementos independentes e gaussianos:

$$W_{t+\Delta} - W_t \sim \mathcal{N}(0, \Delta).$$

- Logo:

$$dW_t \approx \sqrt{\Delta} Z_t,$$

onde $Z_t \sim \mathcal{N}(0, 1)$.

Passo 3: Euler–Maruyama

$$dX_t = b(X_t, t) dt + \sigma(X_t, t) dW_t.$$

- Substituímos:

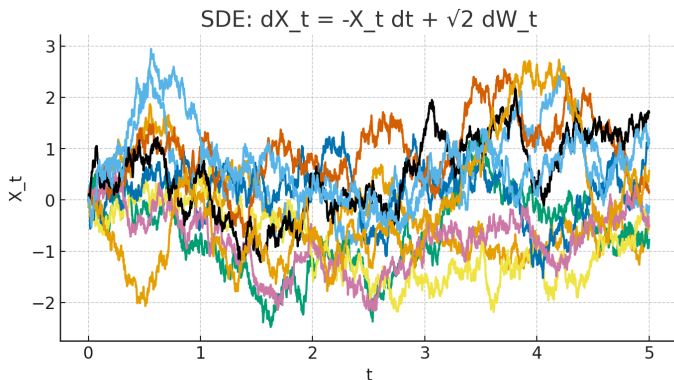
$$dX_t \approx X_{t+\Delta} - X_t, \quad dt = \Delta, \quad dW_t \approx \sqrt{\Delta} Z_t.$$

- Isso leva ao esquema de Euler–Maruyama:

$$X_{t+\Delta} = X_t + b(X_t, t) \Delta + \sigma(X_t, t) \sqrt{\Delta} Z_t.$$

- Na prática, escolhemos $\Delta = T/N$ onde T é o tempo total que queremos evoluir o sistema e $1/N$ é o tamanho do passo. Quanto menor $1/N$, melhor.

Equações Diferenciais Estocásticas



Na figura, simulamos a EDE com $T = 5$ e $N = 3000$

$$dX_t = -X_t dt + \sqrt{2} dW_t.$$

Conversa e caminhada do bêbado: 1

Sem o termo de difusão, a EDE vira apenas:

$$X_{t+\Delta} = X_t + b(X_t, t) \Delta.$$

Isso é equivalente a:

$$\frac{X_{t+\Delta} - X_t}{\Delta} = b(X_t, t).$$

Ou seja:

- o lado esquerdo é “distância / tempo” em um intervalo pequeno;
- então $b(X_t, t)$ representa a **velocidade instantânea**;
- exatamente como em uma equação diferencial comum (sem ruído).

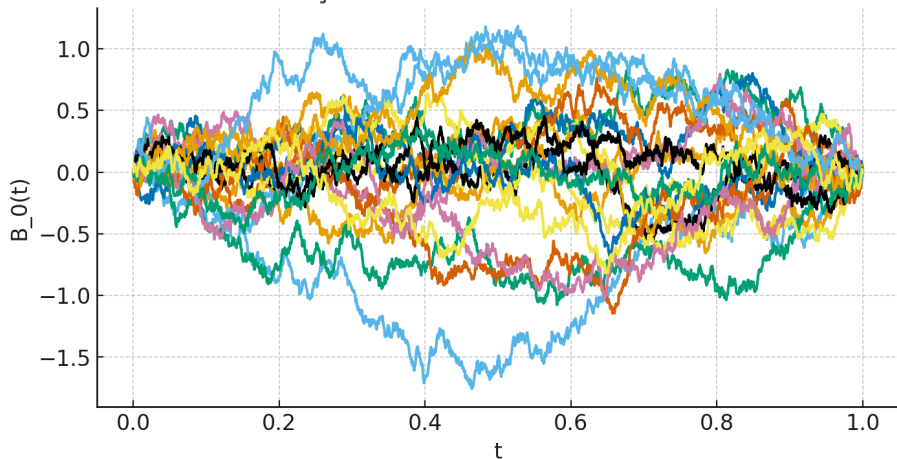
Conversa e caminhada do bêbado: 2

- A *ponte browniana padrão* é o processo que parte de 0 em $t = 0$ e é “forçado” a voltar para 0 em $t = 1$.
- Ela pode ser descrita como a solução da EDE:

$$dB_0(t) = -\frac{B_0(t)}{1-t} dt + dW_t, \quad B_0(0) = 0, \quad t \in [0, 1).$$

Conversa e caminhada do bêbado: 2

Simulação da Ponte Browniana via EDE



Conversa e caminhada do bêbado: 3

Existe um resultado clássico — o **Teorema de Donsker** — que descreve o comportamento assintótico das flutuações da distribuição empírica.

Ele considera o processo

$$G_n(x) = \sqrt{n} (\hat{F}_n(x) - F(x)),$$

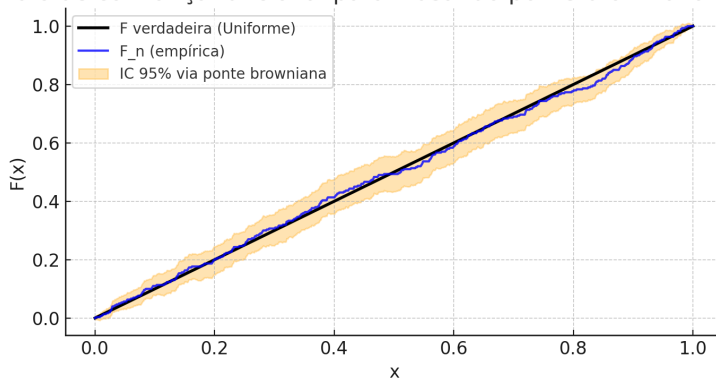
e garante que

$$G_n \Rightarrow G_F,$$

Esse processo limite G_F é exatamente a **ponte browniana associada a F** .

Conversa e caminhada do bêbado: 3

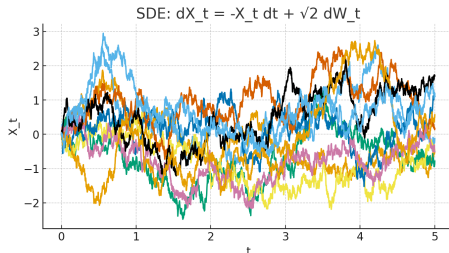
Intervalo de confiança funcional para F usando ponte browniana (Uniforme)



Voltando ao assunto

Conseguimos simular trajetórias **aleatórias** usando Euler–Maruyama:

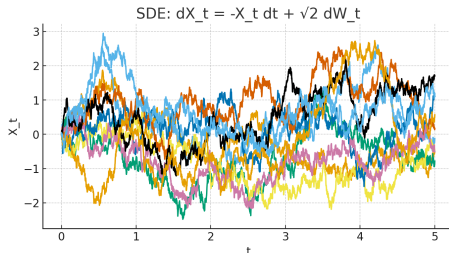
$$X_{t+\Delta} = X_t + b(X_t, t) \Delta + \sigma(X_t, t) \sqrt{\Delta} Z_t.$$



Voltando ao assunto

Conseguimos simular trajetórias **aleatórias** usando Euler–Maruyama:

$$X_{t+\Delta} = X_t + b(X_t, t) \Delta + \sigma(X_t, t) \sqrt{\Delta} Z_t.$$



Mas afinal, qual é a distribuição dessas trajetórias?

Equação de Fokker–Planck

Considere a difusão

$$dX_t = b(X_t, t) dt + \sigma(X_t, t) dW_t,$$

onde b é o *drift* e σ é o coeficiente de *difusão*.

A densidade $p(x, t)$ de X_t satisfaz a **equação de Fokker–Planck**:

$$\frac{\partial p}{\partial t} = -\nabla \cdot (b(x, t) p(x, t)) + \frac{1}{2} \nabla \cdot \left(\nabla \cdot (\sigma(x, t) \sigma(x, t)^\top p(x, t)) \right).$$

Equação de Fokker–Planck

Considere a difusão

$$dX_t = b(X_t, t) dt + \sigma(X_t, t) dW_t,$$

onde b é o *drift* e σ é o coeficiente de *difusão*.

A densidade $p(x, t)$ de X_t satisfaz a **equação de Fokker–Planck**:

$$\frac{\partial p}{\partial t} = -\nabla \cdot (b(x, t) p(x, t)) + \frac{1}{2} \nabla \cdot \left(\nabla \cdot (\sigma(x, t) \sigma(x, t)^\top p(x, t)) \right).$$

CALMA!!!

Equação de continuidade

Fokker–Planck também pode ser escrita como uma **equação de continuidade**.

$$\frac{\partial p}{\partial t} = -\nabla \cdot J,$$

onde o *fluxo de probabilidade* é

$$J(x, t) = b(x) p(x, t) - D \nabla p(x, t).$$

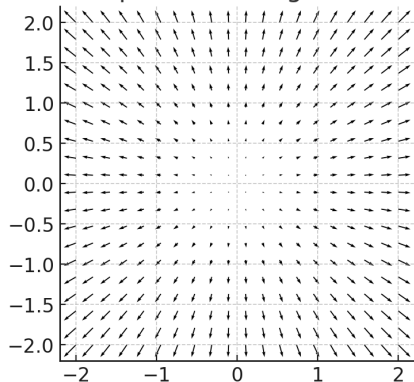
Conversa e caminhada do bêbado: 4

Um resumo rápido de física:

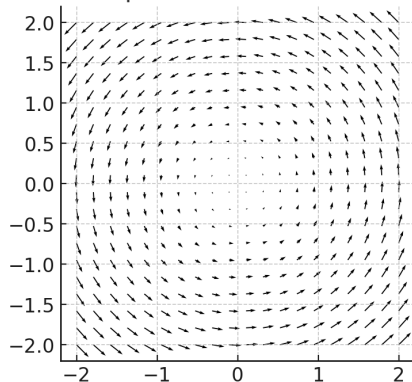
- O **divergente** $\nabla \cdot v$ mede quanto um campo “espalha” ou “concentra” fluxo num ponto.
- O **rotacional** $\nabla \times v$ mede quanto o campo “gira” ou “forma vórtices” ao redor de um ponto.
- $\nabla \cdot v > 0$ empurra densidade para fora. $\nabla \cdot v < 0$ puxa densidade para dentro.

Conversa e caminhada do bêbado: 4

Campo com divergente $\neq 0$



Campo com rotacional $\neq 0$



Equação de continuidade

A Fokker–Planck nos diz como a densidade de X_t evolui.

$$\frac{\partial p}{\partial t} = -\nabla \cdot J,$$

- O lado esquerdo

$$\frac{\partial p}{\partial t}$$

mede **quanto a densidade muda no tempo** em um intervalo muito pequeno.

- O lado direito

$$-\nabla \cdot J$$

diz **quanto de densidade entra ou sai** de cada região do espaço.

- A igualdade significa:

mudança temporal da densidade = fluxo que entra – fluxo que sai.

Pausa para respirar



Fokker–Planck e distribuição estacionária

Vamos assumir que, para tempos grandes, a distribuição de X_t **estabiliza**. Ou seja, existe uma densidade estacionária $p(x)$ tal que a Fokker–Planck zera:

$$0 = -\nabla \cdot (b(x)p(x) - \nabla p(x)) \quad (\text{caso isotrópico: } \sigma = \sqrt{2} I).$$

- Queremos que $p(x)$ seja a distribuição estacionária.
- Então precisamos escolher um **drift** $b(x)$ que satisfaça essa igualdade.
- Em outras palavras: o **fluxo total** deve ter divergente zero.
- Agora vamos pensar: **que forma de b faz isso valer para todo x ?**

Da condição de estacionariedade vimos que o drift ideal deve satisfazer

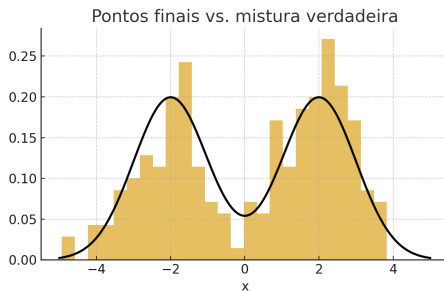
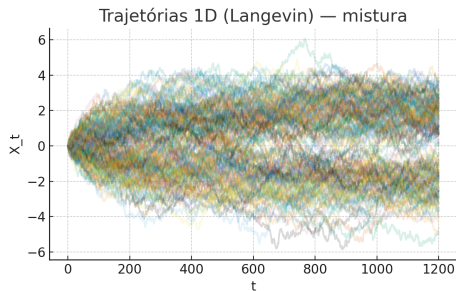
$$b(x) p(x) = \nabla p(x) \implies b(x) = \nabla \log p(x) = \frac{\nabla p(x)}{p(x)}.$$

Isso produz a **dinâmica de Langevin**:

$$dX_t = \nabla \log p(X_t) dt + \sqrt{2} dW_t.$$

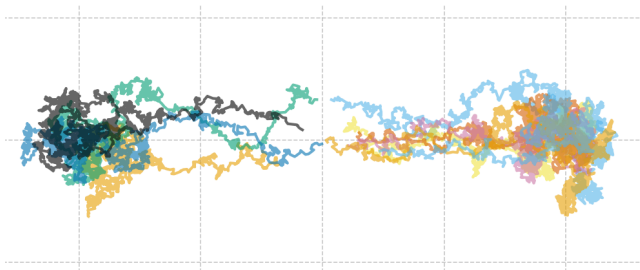
Dinâmica de Langevin

Abaixo geramos trajetórias usando o score de uma mistura de normais.



Dinâmica de Langevin

Abaixo geramos trajetórias usando o score de uma mistura de normais bivariadas.



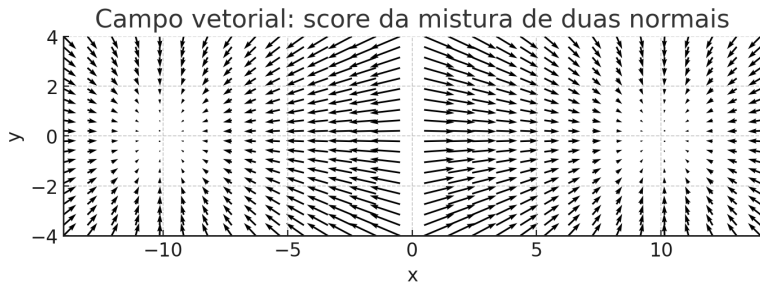
Conversa e caminhada do bêbado: 5

Sem o termo de difusão, a dinâmica de Langevin vira:

$$X_{t+\Delta} = X_t + \nabla \log p(X_t) \Delta.$$

- Isso nada mais é que uma **descida de gradiente** na função $\log p(x)$.
- O sistema caminha sempre na direção onde a densidade $p(x)$ aumenta mais rápido.
- Ou seja: o bêbado sobe a ladeira da probabilidade.

Conversa e caminhada do bêbado: 5



Quando colocamos o termo de difusão, a trajetória fica orbitando em torno da região onde a densidade é maior.

- Se tivermos acesso ao **score** $\nabla \log p(x)$, conseguimos simular novos dados usando a dinâmica:

$$dX_t = \nabla \log p(X_t) dt + \sqrt{2} dW_t.$$

- Mas e se *não* tivermos acesso ao score?
- Em aprendizado de máquina, só temos **exemplos** de dados — não conhecemos a densidade $p(x)$.
- Como exemplo: queremos gerar **novas imagens de gatos de chapéu**, mas não sabemos qual é a distribuição $p(\text{imagem})$.

Aprendendo o Score via DSM

A ideia do DSM é transformar o problema de estimar $\nabla \log p(x)$ em uma tarefa de **regressão supervisionada**.

- Temos amostras reais $x \sim p(x)$.
- Adicionamos ruído gaussiano:

$$\tilde{x} = x + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2 I).$$

- O modelo recebe \tilde{x} e tenta aprender o ruído usado para distorcer x .

O alvo de regressão no DSM

O DSM define como alvo:

$$t = \frac{x - \tilde{x}}{\sigma^2}.$$

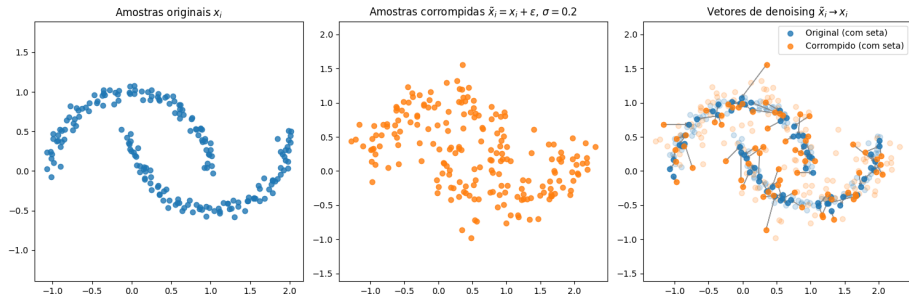
- O modelo recebe como **entrada** o ponto ruidoso $\tilde{x} = x + \varepsilon$.
- O modelo deve prever como **saída** o vetor t , que aponta da versão ruidosa \tilde{x} de volta para o dado original x .
- Ou seja, aprendemos uma função

$$s_{\theta}(\tilde{x}, \sigma) \approx t,$$

que é o score da densidade ruidosa p_{σ} .

- Dessa forma, o problema vira uma **regressão supervisionada** comum.

O alvo de regressão no DSM



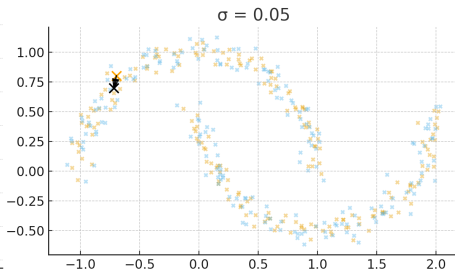
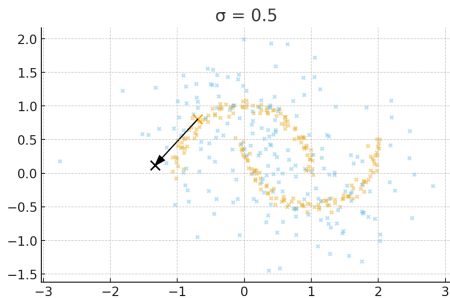
Estimando o score na prática

- Para um único valor de σ , o DSM aprende o score da densidade suavizada q_σ .
- Mas queremos capturar $p(x)$ em **múltiplas escalas**.
- Usamos então vários valores de ruído

$$\sigma_k = \sigma_{\min} \left(\frac{\sigma_{\max}}{\sigma_{\min}} \right)^{k/(K-1)}, \quad k = 0, \dots, K-1.$$

- Isso cobre ruídos fortes (formas globais) até ruídos fracos (detalhes finos).

Estimando o score na prática



Etapa de Inferência via Langevin

Depois de treinar o score $s_\theta(\tilde{x}, \sigma)$, podemos gerar **novas amostras**.

- Usamos o campo aprendido como aproximação de $\nabla \log p(x)$.
- Fazemos uma simulação de **Langevin Anelado** (Annealed Langevin Dynamics – ALD).
- Começamos com muito ruído e vamos “refinando”.

Etapa de Inferência via Langevin

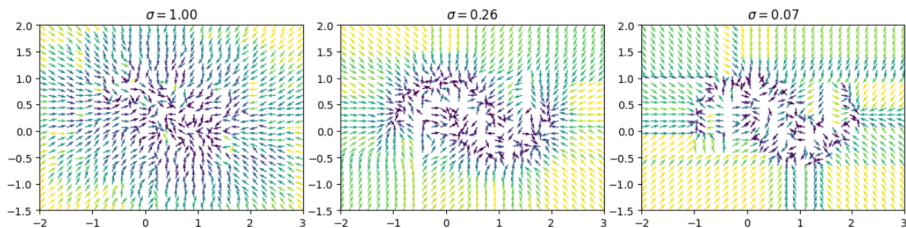
A atualização é:

$$x_t = x_{t-1} + \frac{\alpha_i}{2} s_\theta(x_{t-1}, \sigma_i) + \sqrt{\alpha_i} z_t, \quad z_t \sim \mathcal{N}(0, I), \quad x_0 \sim \mathcal{N}(0, I).$$

- σ_i controla a **escala do ruído**.
- α_i é o passo: normalmente proporcional a σ_i^2 .
- Sigmas vão do maior para o menor.

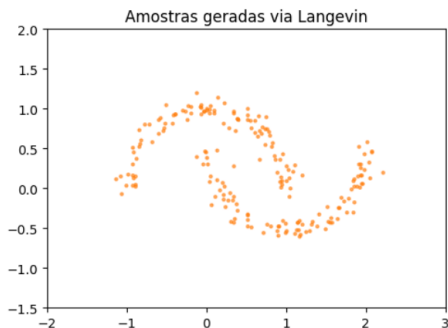
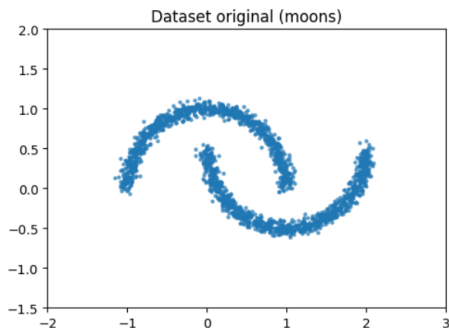
Etapa de Inferência via Langevin

- Em σ_{\max} : passos grandes, exploração global.
- Em sigmas intermediários: ajuste da forma grossa.
- Em σ_{\min} : refinamento fino e detalhes.



Resultado final

Abaixo os dados gerados usando a ideia anterior e aprendendo o campo usando uma floresta aleatória.



Resultado final

Podemos usar a mesma ideia para gerar amostras condicionais.

