Aprendizaje profundo basado en la física

Semana 6: De simulaciones a inferencia

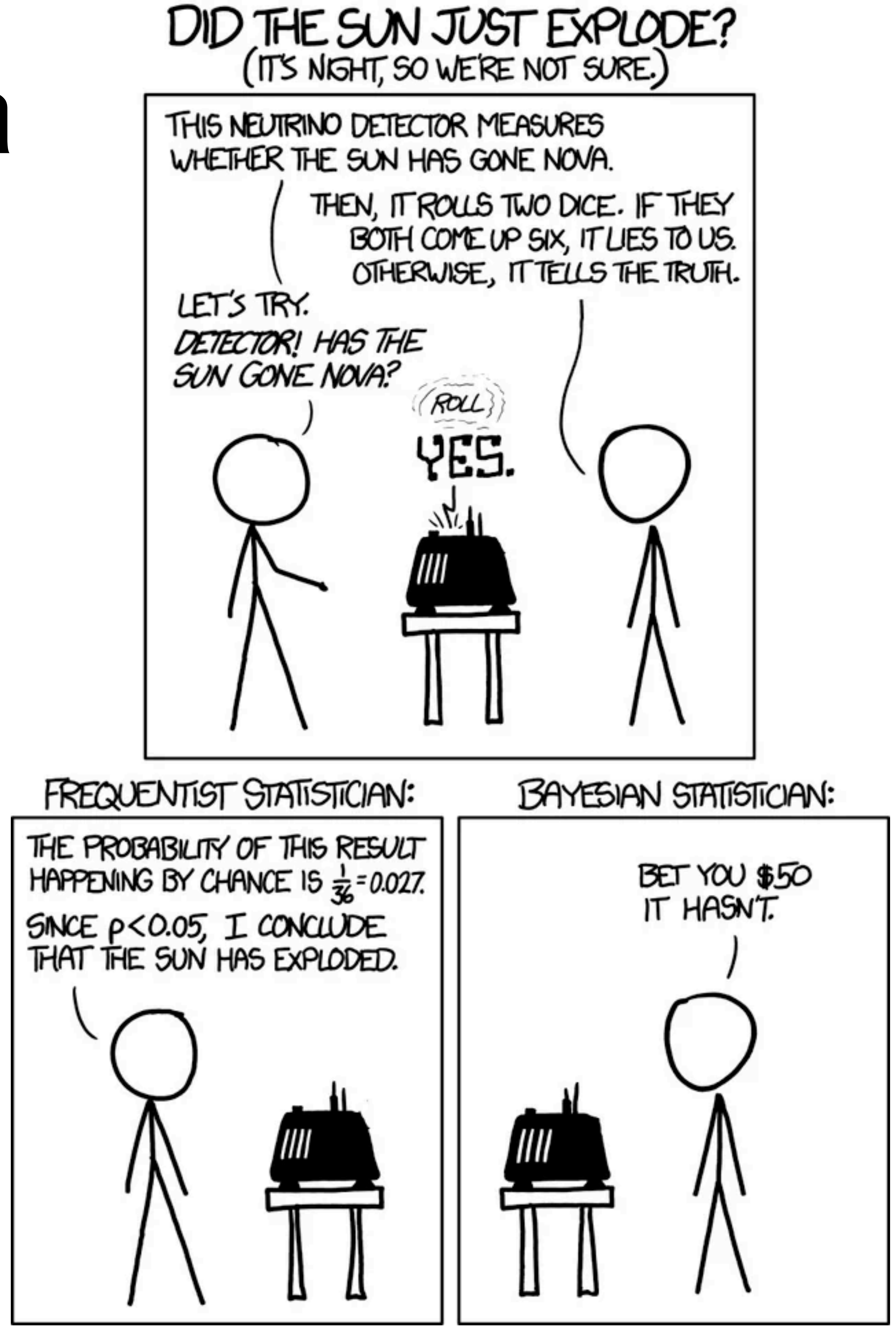
Docente: José I. Robledo - 14/05/2026

De simulaciones a inferencia

Introducción a la estadística Bayesiana

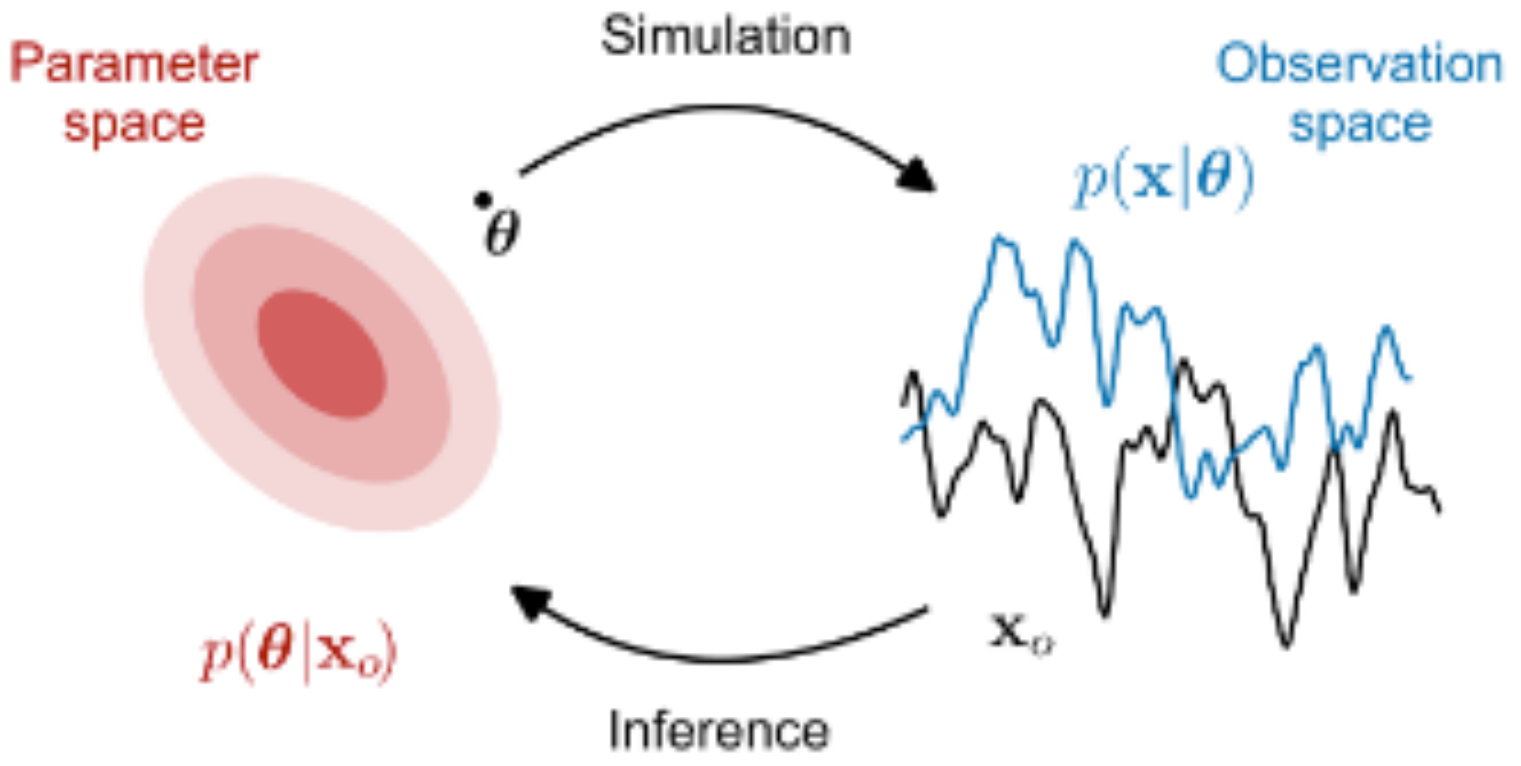
- Estadística frecuentista: estimar parámetros *poblacionales* de distribuciones subyacentes a las mediciones. *Existen* los parámetros poblacionales y nosotros tratamos de buscarlos.

- Estadística Bayesiana: los parámetros del modelo son *variables aleatorias.* Nuestra incertidumbre sobre ellos se describe mediante sus distribuciones de probabilidad.



Estadística Bayesiana

Teorema de Bayes



Buscamos construir una distribución de probabilidad sobe los parámetros que refleje lo que sabemos antes de observar los datos y cómo la información se actualiza luego de observarlos

Verosimilitud *a priori*

a posteriori

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)}$$

evidencia

$P(\text{lie}) = 1/36$

$P(\text{exploded}) = 1/4.38 \times 10^{13}$

$$\begin{aligned} P(\text{exploded}\,|\,\text{yes}) &= \frac{P(\text{yes}\,|\,\text{exploded})P(\text{exploded})}{P(\text{yes})} \\ &= \frac{P(\text{exploded})(1-P(\text{lie}))}{P(\text{exploded})(1-P(\text{lie}))+P(\text{lie})(1-P(\text{exploded}))} \\ &\approx \frac{1}{1.25226 \times 10^{12}} \end{aligned}$$

Actualizar creencias usando observaciones.

La verosimilitud muchas veces no la podemos escribir analíticamente

$p(x|\theta) \rightarrow$ simulador muestrea la verosimilitud

https://www.explainxkcd.com/wiki/index.php/1132:_Frequentists_vs._Bayesians

Estadística Bayesiana

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)}$$

Problema

$$p(x) = \int p(x|\theta)p(\theta)d\theta$$

Solución

Cuando no se puede resolver analíticamente $p(x)$ pero se conoce $p(x|\theta)$, entonces se recurre a métodos de muestreo como:

- Metropolis-Hasting
- Hamiltonian Monte-Carlo
- Gibbs sampling

Con los que se evita calcular esta integral usando cadenas de Markov (MCMC) para estimar la distribución a posteriori!

Estadística Bayesiana

Metropolis Hasting

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)}$$

Muy inteligentemente:

- Empezamos de un valor inicial θ_t y proponemos θ' usando una distribución de propuesta (típicamente aleatoria), i.e. $\theta' \sim q(\theta'|\theta_t)$

- Comparamos que tan probable es el nuevo estado θ' respecto a θ_t usando el cociente de las posteriores respectivas. Si la probabilidad posterior es mayor, entonces nos quedamos con la propuesta. Tasa de aceptación:

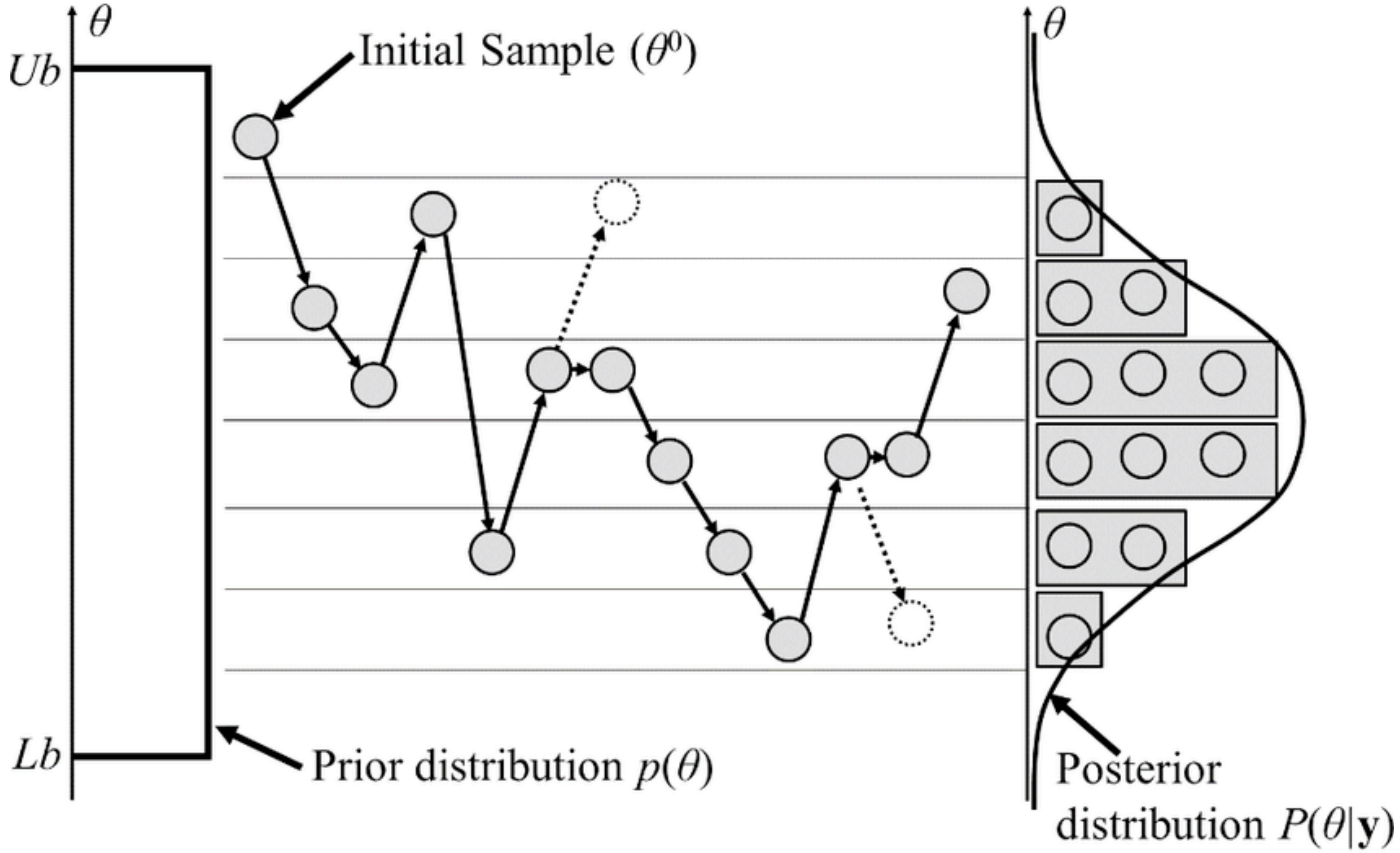
$$\alpha = \min\left(1, \frac{p(\theta'|x)}{p(\theta_t|x)}\right) = \min\left(1, \frac{p(x|\theta')p(\theta')}{p(x|\theta_t)p(\theta_t)}\right)$$

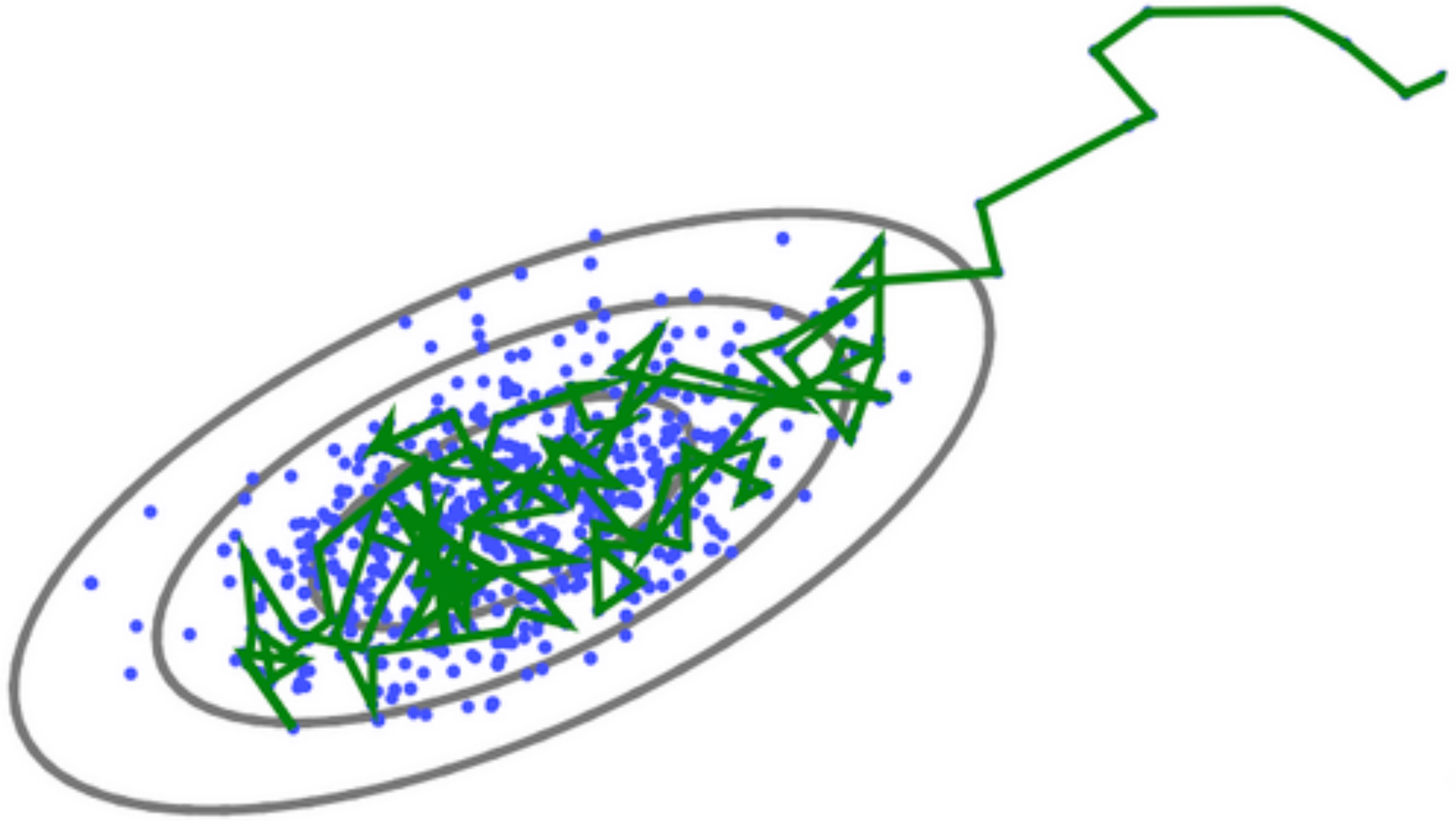
Si el nuevo punto explica mejor los datos, nos lo quedamos. Si es peor, todavía hay una probabilidad que lo aceptemos y es proporcional a cuánto menor es la posterior respecto al estado anterior.

Estadística Bayesiana

Metropolis Hasting

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)}$$





$$\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(N)} \sim p(\theta|x) \implies \mathbb{E}[f(\theta)] = \int f(\theta)p(\theta|x)d\theta \approx \frac{1}{N}\sum_{i=1}^{N} f(\theta^{(i)})$$

MCMC

Notebook Bayesiano Parte 1

Inferencia Basada en Simulaciones

Motivación

Muchos sistemas físicos:

- Tienen simuladores complejos
- Poseen ruido no gaussiano
- Tienen dinámicas altamente no lineales
- ***Resulta imposible escribir la verosimilitud de forma explicita.***

Cosmología

Dinámica Molecular

Inferencia climática

Sistemas cuánticos

Física de partículas

Inferencia Basada en Simulaciones

Primera idea: Aproximate Bayesian Computation

- El método de **Cómputo Bayesiano Aproximado** :
  - Muestrea θ de nuestra creencia a priori $p(\theta)$.
  - Simula datos x_{sim} para los parámetros θ, i.e. $x_{sim} \sim p(x \mid \theta)$
  - Compara simulaciones con x_{obs} para ver si decidimos aceptar o no según algún criterio de distancia, como por ej.

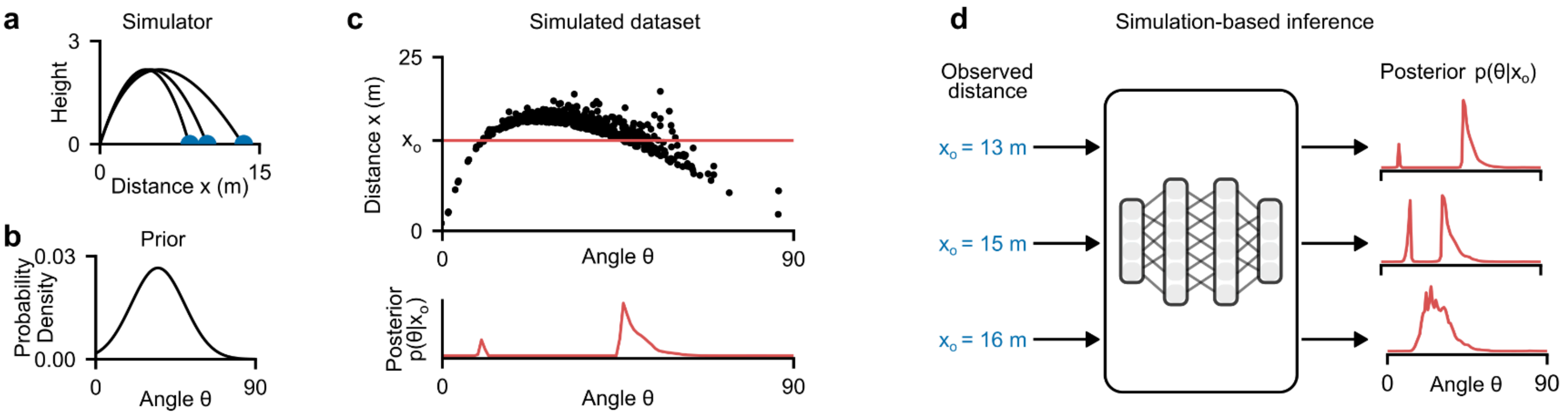
$$d(x_{sim}, x_{obs}) < \epsilon$$

Si se cumple, se acepta, sino se descarta. No nos hace falta la verosimilitud explícita!

Notebook Bayesiano Parte 2

Inferencia Basada en Simulaciones

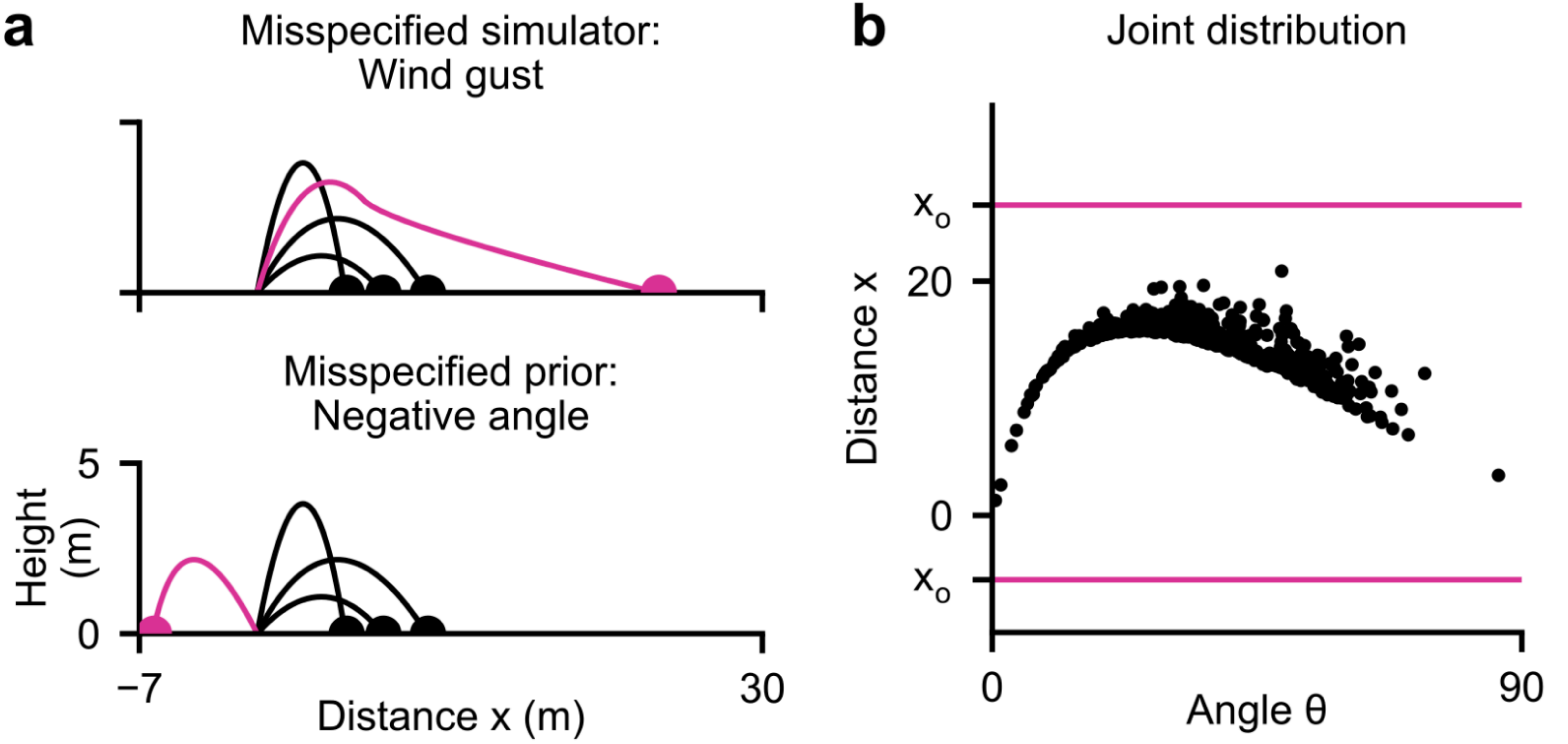
Segunda idea: Metamos redes neuronales! SBI



https://arxiv.org/abs/2508.12939

Inferencia Basada en Simulaciones

Modelo mal especificado: las NNs no hacen magia!



https://arxiv.org/abs/2508.12939

Inferencia Basada en Simulaciones

Variantes

Podríamos estar interesados en dos problemas distintos

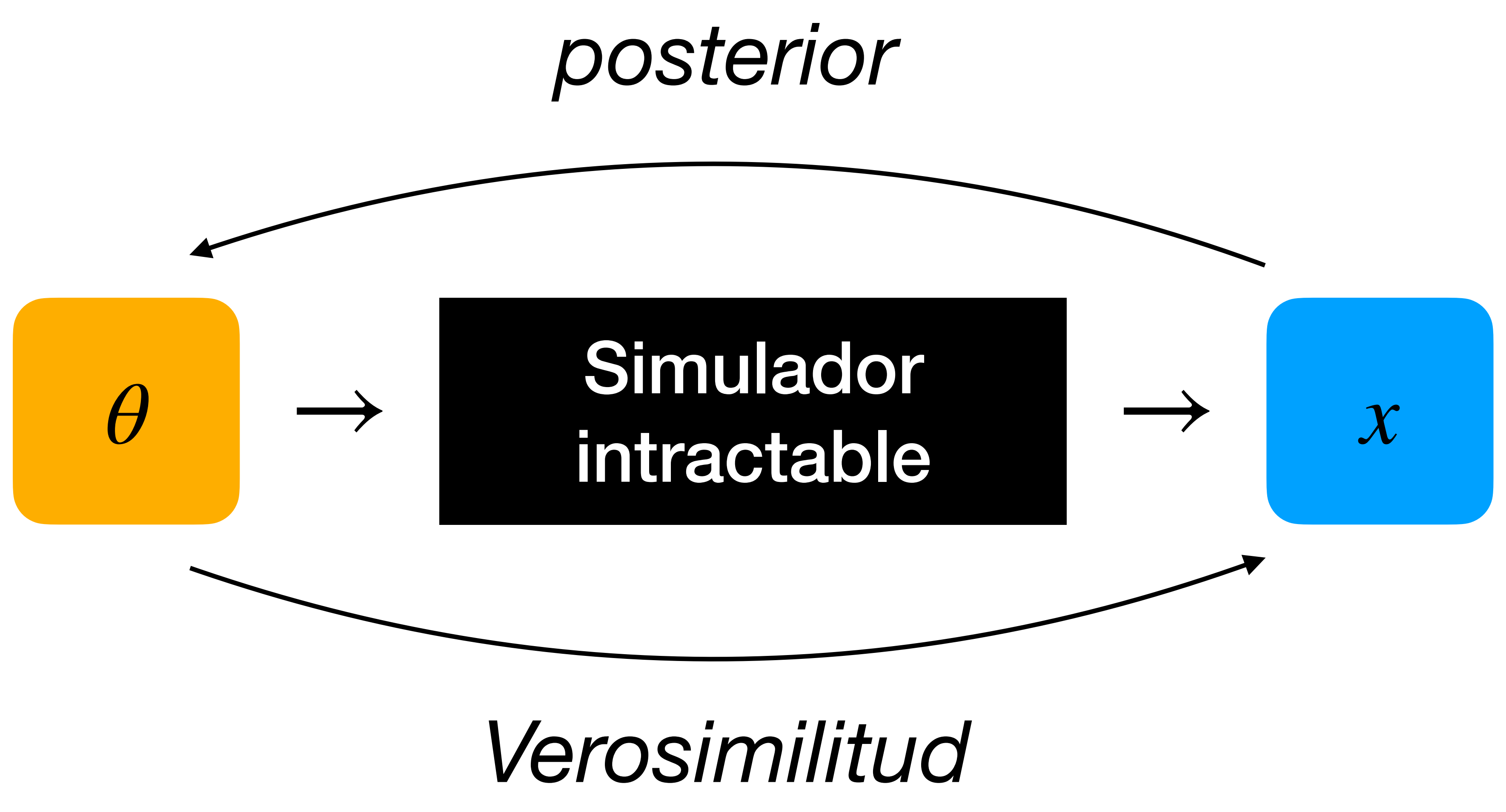
Forward

$\theta \rightarrow x$

Los parámetros físicos producen observaciones

$p(x \mid \theta)$

Verosimilitud



$(\theta, x) \rightarrow$ NN

Inverse

$x \rightarrow \theta$

Los datos nos permiten estimar los parámetros físicos

$p(\theta \mid x)$

posterior

Usamos al simulador como un modelo generativo que genera pares (θ, x)

Inferencia Basada en Simulaciones

Neural Posterior Estimation (NPE)

La red neuronal aprende la distribución posterior $p(\theta | x)$ a partir de los datos generados mediante simulaciones $(\theta, x) = \{(\theta_i, x_i)\}_{i=1,...,N}$ utilizando θ o alguna función de $\theta, g(\theta)$ como características de entrada y x o alguna función de $x, f(x)$ como target.

Los θ fueron generados maestreando de alguna distribución a priori poco informativa (si no se cuenta con información previa)

Queremos aprender $p(\theta | x)$ a partir de (θ, x)…

Los **Normalizing Flows** aparecen como solución inmediata!

Los MDN también son excelentes opciones

Inferencia Basada en Simulaciones

Modelo probabilístico de fondo

De los modelos probabilísticos que vimos, por qué los NFs?

- No sólo queremos generar muestras plausibles de la posterior, queremos representar la probabilidad condicional y evaluarla numéricamente. En NFs aprendíamos la distribución directamente, sin método variaciones.

- Simples de entrenar

- Permite muestras, evaluar densidad, entrenar por MLE, estimar posterior normalizada.

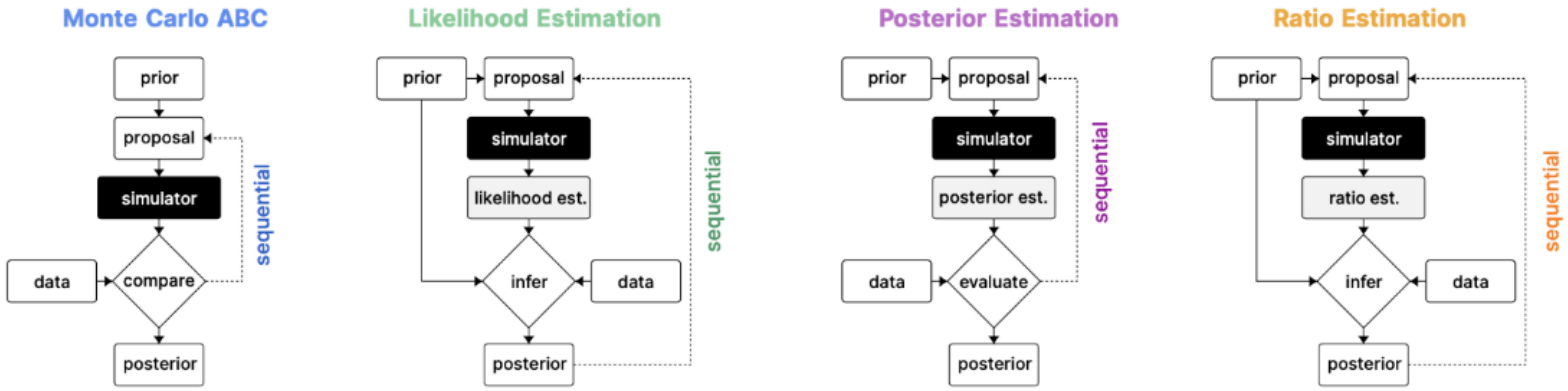
- Expresivos

Inferencia Basada en Simulaciones

NPE, NLE, NRE?

Podríamos tener distintos objetivos:

- Estimar la distribución posterior (Neural Posterior Estimation, NPE) $p(\theta|x)$
- Estimar la verosimilitud (Neural Likelihood Estimation, NLE) $p(x|\theta)$
- Estimar un cociente entre verosimilitudes (Neural Ratio Estimation, NRE) $\dfrac{p(x|\theta)}{p(x)}$



Inferencia Basada en Simulaciones

NPE - Estimación neuronal de la posterior

objetivo $p(\theta|x)$

Amortizar es compartir parámetros entre distintos modelos para producir distintas predicciones

$$q_\phi(\theta|x) \approx p(\theta|x)$$

Generamos simulaciones $(\theta_i, x_i) \sim p(\theta)p(x|\theta) = p(\theta, x)$

Entrenamos $\max_\phi \sum_i \log q_\phi(\theta_i|x_i)$ $\mathscr{L}_{NPE}(\phi) = \mathbb{E}_{p(\theta,x)}\left[\log q_\phi(\theta|x)\right]$

Equivale a minimizar $KL(p(\theta|x)||q_\phi(\theta|x))$ promediado sobre $p(x)$

https://github.com/mlcolab/sbi-workshop/blob/main/slides/2_2_snpe.ipynb

Inferencia Basada en Simulaciones

NPE - inferencia

Una vez entrenado

$$q^*_\phi(\theta|x) \approx p(\theta|x)$$

Por lo tanto $q^*_\phi(\theta|x_o)$ es directamente el posterior aproximado.

Ventajas:

- Inferencia rápida
- Muestreo directo

Desventajas:

- Difícil en dimensiones altas de θ
- Condicionado en x

Resuelve problema **inverso**

$$x \rightarrow \theta$$

Notebook SBI

Inferencia Basada en Simulaciones

NLE - Estimation neuronal de la verosimilitud

objetivo $p(\theta | x)$

En NLE aproximamos la verosimilitud primero

$$q_\phi(x | \theta) \approx p(x | \theta)$$

Generamos simulaciones $(\theta_i, x_i) \sim p(\theta)p(x | \theta) = p(\theta, x)$

Entrenamos $\max_\phi \sum_i \log q_\phi(x_i | \theta_i)$ $\mathscr{L}_{NLE}(\phi) = \mathbb{E}_{p(\theta,x)}\left[\log q_\phi(x | \theta)\right]$

Equivale a minimizar $KL(p(x | \theta) || q_\phi(x | \theta))$

Inferencia Basada en Simulaciones

NLE - inferencia

Una vez entrenado, debemos usar Bayes

$$\hat{p}(\theta|x_o) \propto q_\phi^*(x_o|\theta)p(\theta)$$

Es decir que debemos utilizar MCMC, HMC, o algún método de sampleo para estimar la posterior

Ventajas:

- Reutilizable para múltiples priors
- Más estable
- Likelihood explícita

Desventajas:

- Require MCMC posterior
- Costoso en inferencia
- Difícil para alta dimension de x

Resuelve problema **forward**

$$\theta \to x$$

Notebook SBI

Inferencia Basada en Simulaciones

NRE - Estimation neuronal de la razón

objetivo $p(\theta | x)$

En NRE aproximamos la razón

$$r(x, \theta) = \frac{p(x | \theta)}{p(x)}$$

Observemos que

$$p(\theta | x) = \frac{p(x | \theta) p(\theta)}{p(x)} = r(x, \theta) p(\theta)$$

Inferencia Basada en Simulaciones

$$r(x,\theta) = \frac{p(x|\theta)}{p(x)}$$

NRE - Estimation neuronal de la razón

En NRE transformamos la inferencia en clasificación binaria! Para esto construimos dos distribuciones

Distribución conjunta $p_1(x,\theta) = p(x,\theta)$ *Verdadera distribución*

Producto marginal $p_0(x,\theta) = p(x)p(\theta)$ *Supuesto independencia*

Generamos (x,θ) de ambas distribuciones y entrenamos un clasificador

$$y = \begin{cases} 1 & (x,\theta) \sim p(x,\theta) \\ 0 & (x,\theta) \sim p(x)p(\theta) \end{cases} \qquad \begin{array}{l} (\theta_i, x_i) \\ (\theta_i, x_j),\ i \neq j \end{array}$$

Entrenamos red

$$d_\phi(x,\theta) \approx p(y=1 \mid x,\theta)$$

$$\mathcal{L}_{\text{NRE}}(\phi) = \mathbb{E}_{p(x,\theta)}[\log d_\phi(x,\theta)] + \mathbb{E}_{p(x)p(\theta)}[\log(1 - d_\phi(x,\theta))]$$

Inferencia Basada en Simulaciones

NRE - Estimation neuronal de la razón

$$d_\phi(x,\theta) \approx p(y=1 \mid x,\theta)$$

Usando el teorema de Bayes, el clasificador optimo satisface

$$d^*(x,\theta) = \frac{p(x,\theta)}{p(x,\theta)+p(x)p(\theta)}$$

$$= \frac{p(x\,|\,\theta)p(\theta)}{p(x,\theta)+p(x)p(\theta)} = \frac{p(x\,|\,\theta)}{p(x\,|\,\theta)+p(x)}$$

como $1 - d^* = \dfrac{p(x)}{p(x\,|\,\theta)+p(x)} \implies \boxed{\dfrac{d^*}{1-d^*} = \dfrac{p(x\,|\,\theta)}{p(x)} = r(x,\theta)}$

Chance (Odds)

Aprendemos qué tan compatible es θ con x

Inferencia Basada en Simulaciones

NRE - inferencia

Una vez entrenado, debemos usar

$$\hat{p}(\theta|x_o) \propto r^*_\phi(x_o, \theta)p(\theta)$$

Nuevamente necesitamos MCMC, HMC, o algún método de sampleo para estimar la posterior

Ventajas:

- Util cuando la verosimilitud es compleja
- Clasificación es más estable
- Dimensionalidad alta

Desventajas:

- Require MCMC posterior
- Depende de los falsos
- Calibración puede ser difícil

Notebook SBI

Inferencia Basada en Simulaciones

Estadísticas de resumen (Summary Statistics)

A veces conviene aprender una transformación o función de los datos

$$x \to s(x)$$

Donde $s(x) \in \mathbb{R}^k$ es na representación comprimida de la observación original.

La idea es reemplazar datos complejos x por características informativas más fáciles de inferir

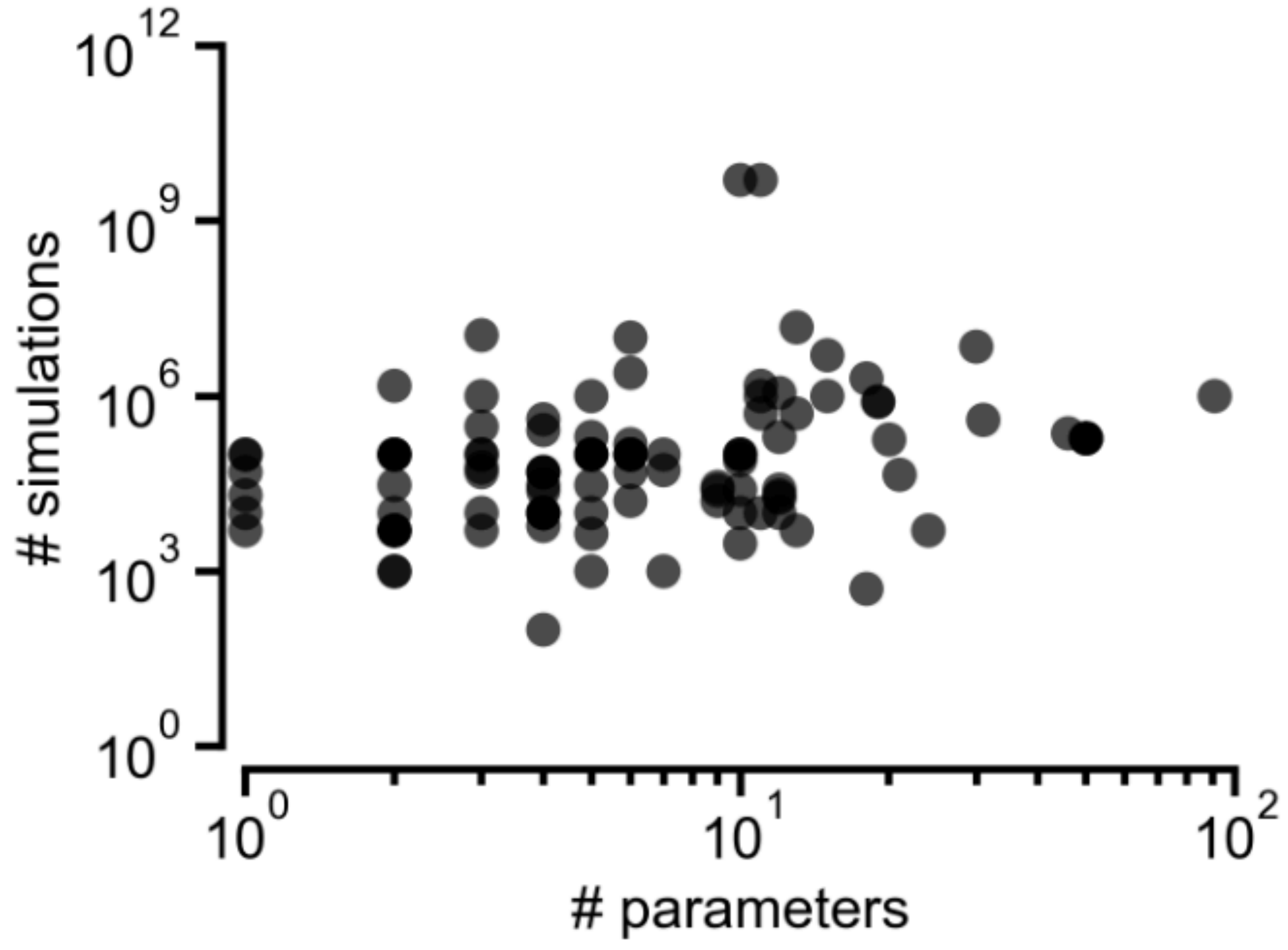
En vez de aprender $p(\theta \mid x)$ aprendemos $p(\theta \mid s(x))$

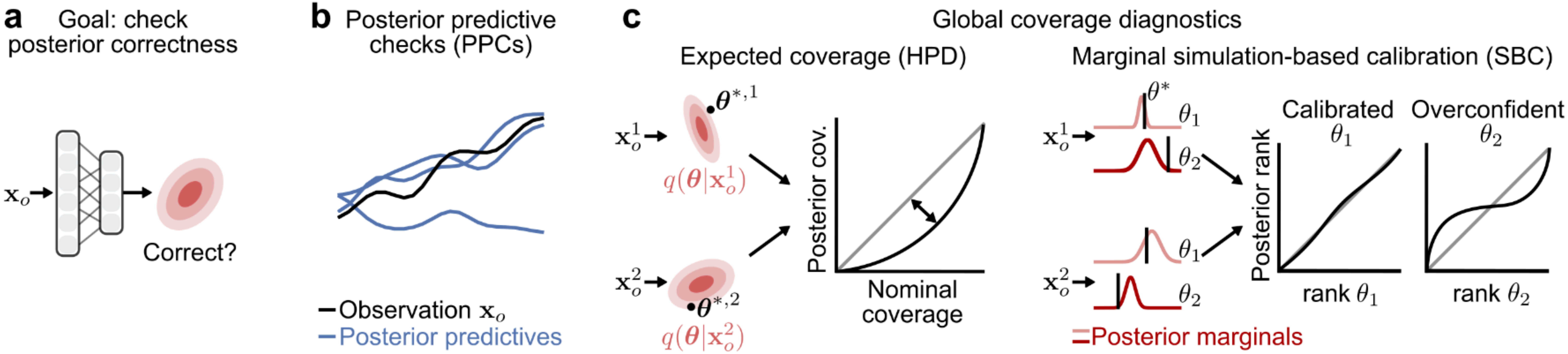
Interesante: Podemos aprender las estadísticas de resumen!

Notebook SBI

SBI

Consideraciones





https://sbi-applications-explorer.streamlit.app/

Inferencia Basada en Simulaciones

Posterior Predictive Checks (PPC)

Si sampleamos parámetros desde la posterior inferida y volvemos a correr el simulador con ellos, obtenemos datos parecidos a los observados?

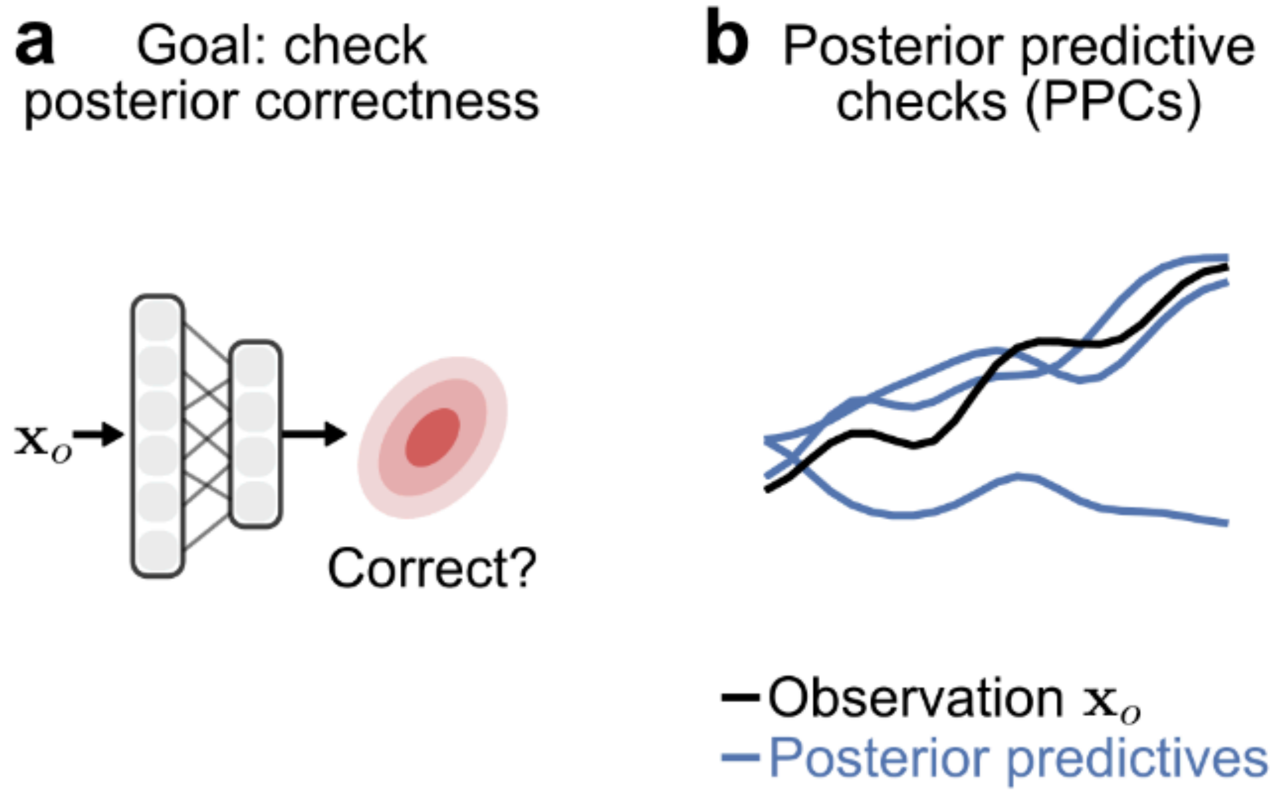
$$x^{rep} \sim p(x^{rep} \mid x_{obs}) \approx \int p(x^{rep} \mid \theta) q_\phi(\theta \mid x_{obs}) d\theta$$

Distribución predictiva posterior

Muestreamos distribución aprendida $\theta^{(s)} \sim q_\phi(\theta \mid x_o)$

Usamos simulador $x^{rep,(s)} \sim p(x \mid \theta^{(s)})$

Comparamos $x^{rep,(s)}$ con x_0



Inferencia Basada en Simulaciones

Expected coverage

Las regiones de credibilidad de la posterior… ¿contienen al parámetro verdadero con la frecuencia correcta?

Prior: $\theta \sim p(\theta)$

Simulador: $x \sim p(x \mid \theta)$

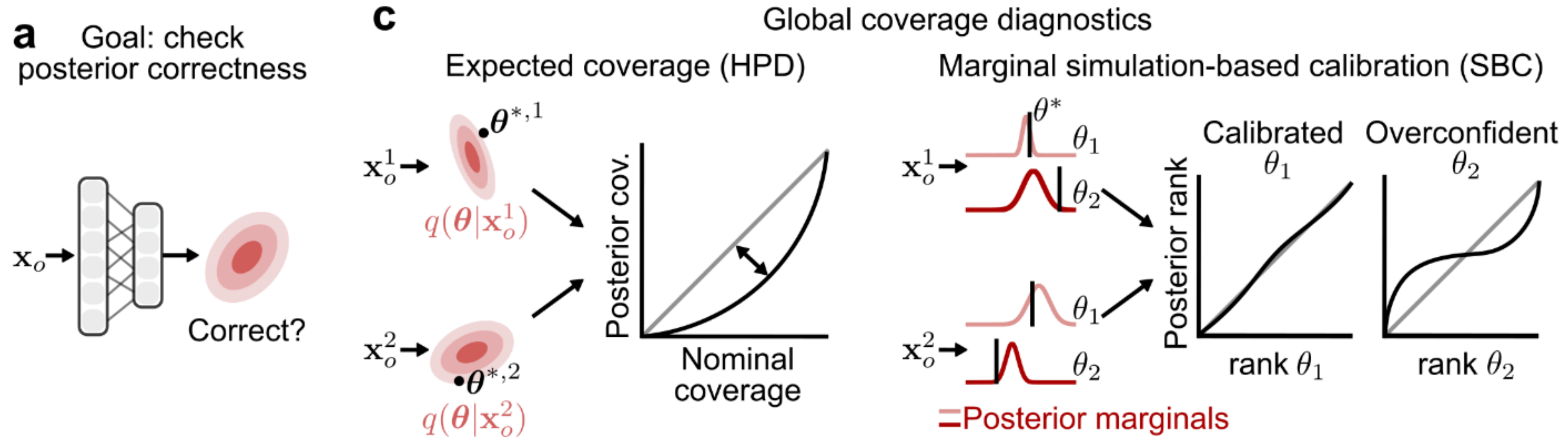
Posterior aproximada: $q_\phi(\theta \mid x)$

Intervalo de credibilidad

Dado nivel $\alpha \in (0,1)$

Definimos una *región creíble* $C_\alpha(x) \in \Theta$, tal que

$$\int_{C_\alpha(x)} q_\phi(\theta \mid x) d\theta = \alpha$$



Inferencia Basada en Simulaciones

Expected coverage

Imaginemos repetir el experimento muchas veces. Generamos

$$\theta_i \sim p(\theta); x_i \sim p(x \mid \theta_i)$$

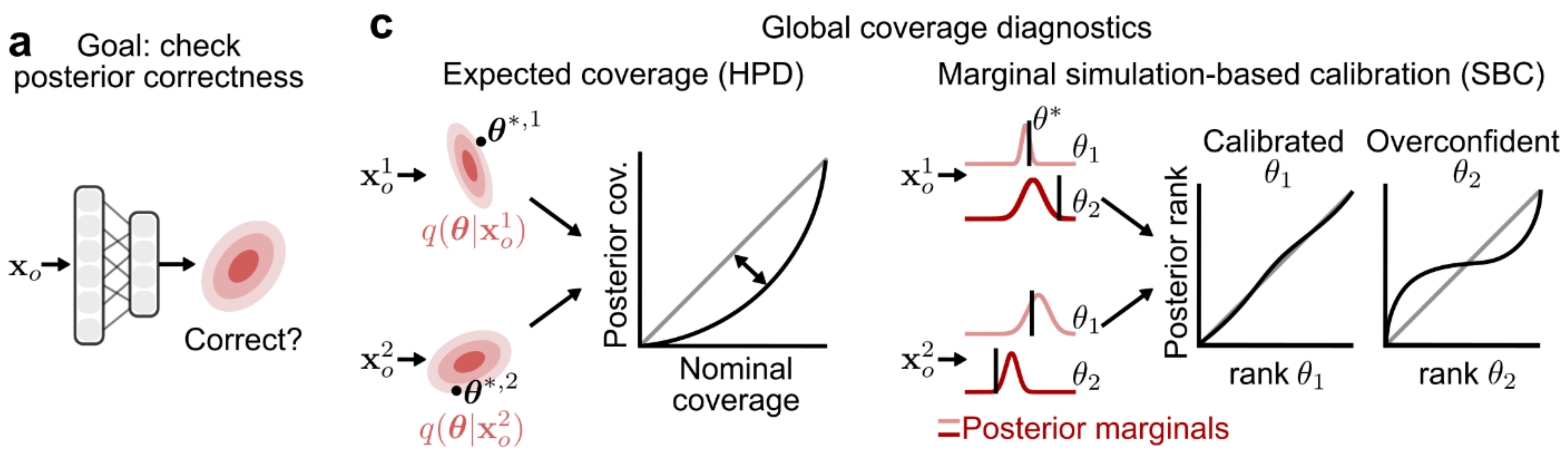
Y luego computamos $C_\alpha(x_i)$.

El ***expected coverage*** a nivel α es:

$$P(\theta \in C_\alpha(x)), \qquad \text{donde } (\theta, x) \sim p(\theta)p(x \mid \theta).$$

Una posterior calibrada perfectamente satisface

$$P(\theta \in C_\alpha(x)) = \alpha, \ \forall \alpha \in [0,1]$$



Inferencia Basada en Simulaciones

Expected Coverage - Highest Posterior Density (HPD)

Si hacemos marginales, podemos calcular $C_\alpha(x_i)$ calculando los cuantiles correspondientes (más fácil)

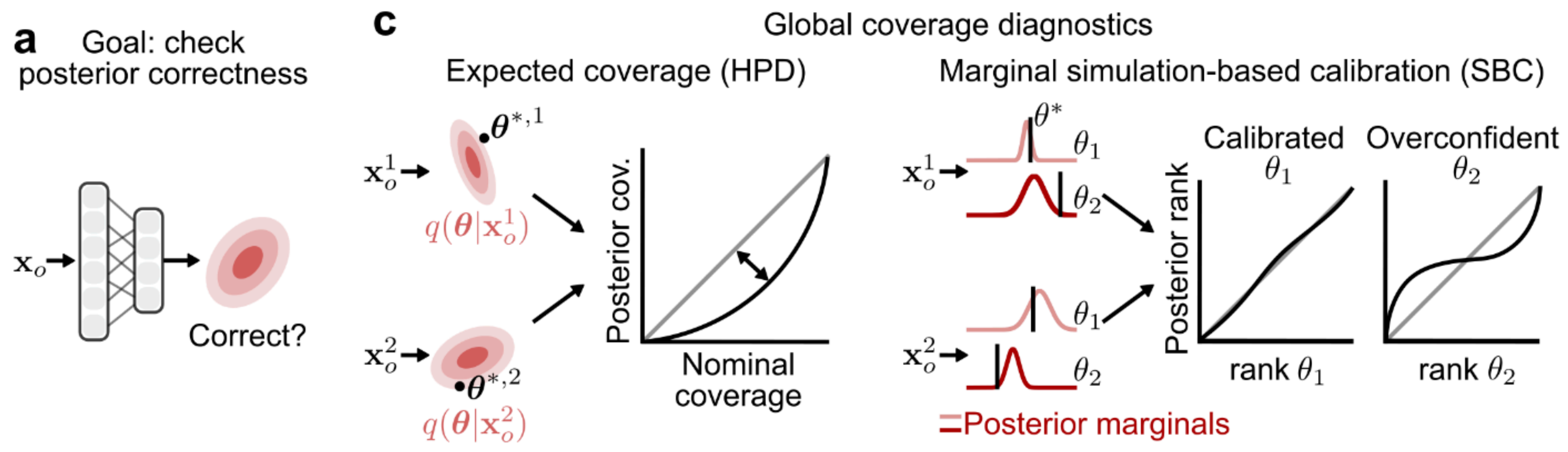
En alta dimension $\theta \in \mathbb{R}^d$ los cuantiles marginales no definen correctamente una región conjunta. Entonces usamos *High Posterior Density (HPD):*

$$C_\alpha(x) = \{\theta : q_\phi(\theta \,|\, x) > t_\alpha\},$$

Donde t_α es elegida tal que $\displaystyle\int_{q_\phi(\theta|x)>t_\alpha} q_\phi(\theta \,|\, x) d\theta = \alpha$

Evaluación MC:

1. $\theta^{(i)} \sim q_\phi(\theta \,|\, x)$
2. $s_i = q_\phi(\theta^{(i)} \,|\, x)$
3. $s_{(1)} \geq s_{(2)} \geq \dots$
4. $t_\alpha = quantile_{1-\alpha}(\{s_i\})$



Inferencia Basada en Simulaciones

Expected Coverage - highest Posterior Density (HPD)

Ahora queremos saber si $\theta_{true} \in C_\alpha(x)$:

Definimos $r = P_{q_\phi(\theta|x)}(q_\phi(\theta \,|\, x) \geq q_\phi(\theta_{true} \,|\, x))$

HPD Rank

Cuanta masa tiene densidad mayor que θ_{true}?

Entonces $\theta_{true} \in C_\alpha(x)$ si $r < \alpha$

El HPD rank indica qué tan "central" es θ_{true} dentro de la posterior: valores pequeños están cerca del modo, valores grandes en las colas.

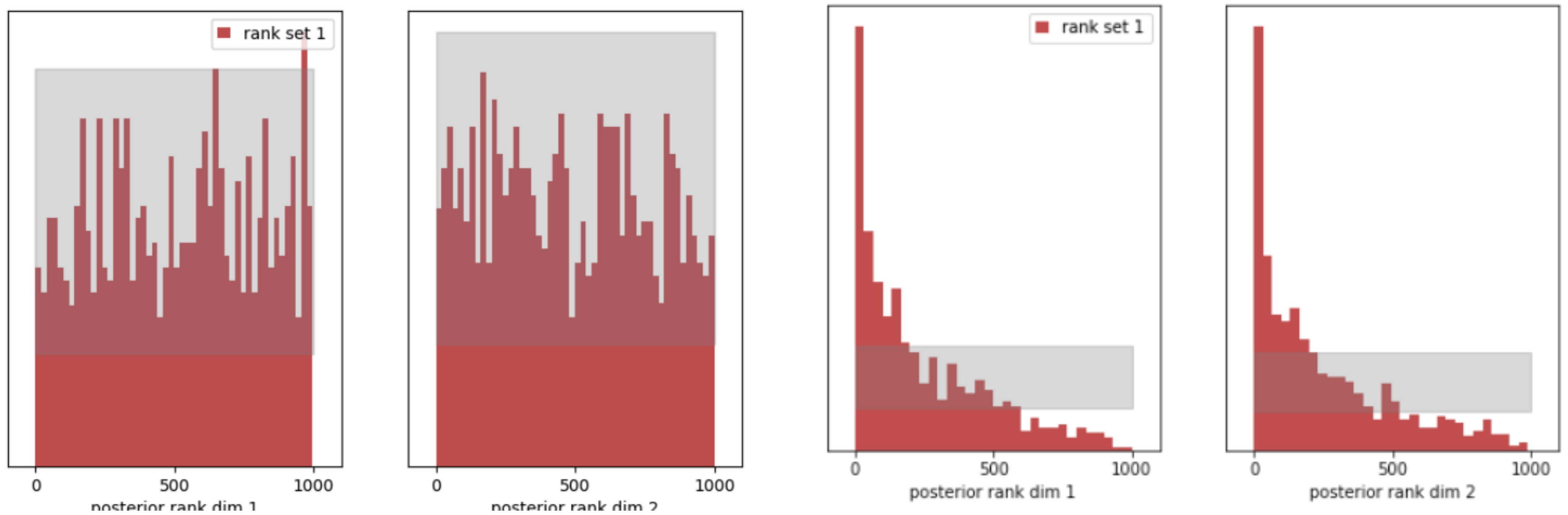
https://sbi-dev.github.io/sbi/0.22/tutorial/13_diagnostics_simulation_based_calibration/

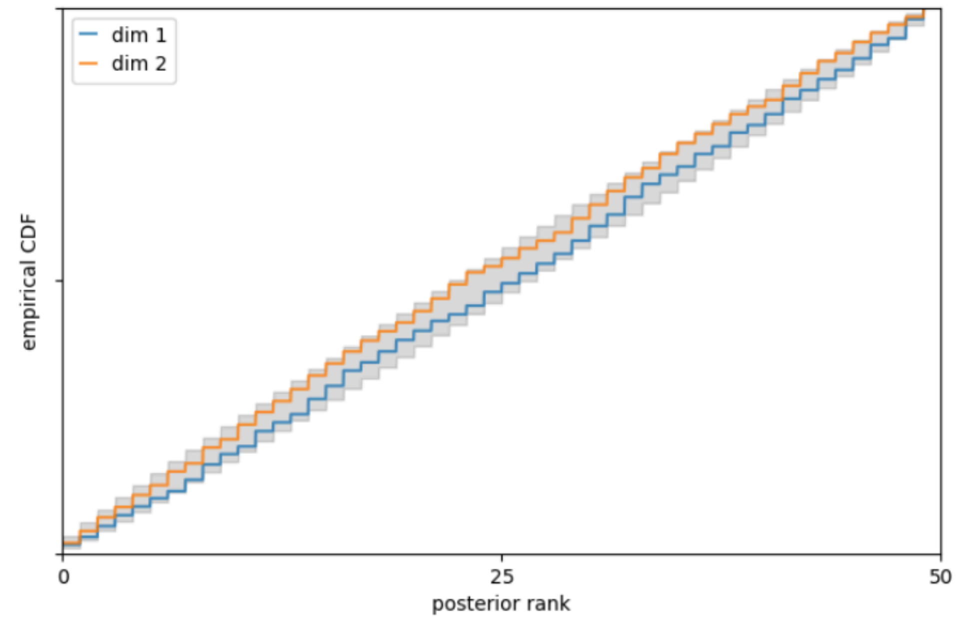
Inferencia Basada en Simulaciones

Simulation Based Calibration (SBC)

Queremos verificar si $q_\phi(\theta \mid x)$ está **calibrado**

1. Para $i = 1,\ldots,N$ muestra $\theta_i \sim p(\theta)$
2. Simular observación $x_i \sim p(x \mid \theta_i)$
3. Inferir posterior $q_\phi(\theta \mid x)$
4. Muestrear posterior $\theta_i^{(1)}, \ldots, \theta_i^{(M)} \sim q_\phi(\theta \mid x_i)$
5. Calcular rank $r_i = \sum_{j=1}^{M} \mathbf{1}[\theta_i^{(j)} < \theta_i]$





SBC nos puede decir si no estamos equivocados. Pero no nos dice si estamos en lo correcto. Checkea una condición necesaria pero no suficiente.

Si la posterior está correctamente calibrada, $r_i \sim U[0,M]$

https://sbi-dev.github.io/sbi/0.22/tutorial/13_diagnostics_simulation_based_calibration/

Inferencia Basada en Simulaciones

Qué hacemos finalmente con la posterior aprendida?

Existen varias maneras de resumirla:

- media o mediana

```
mean = samples.mean(0)
std = samples.std(0)
```

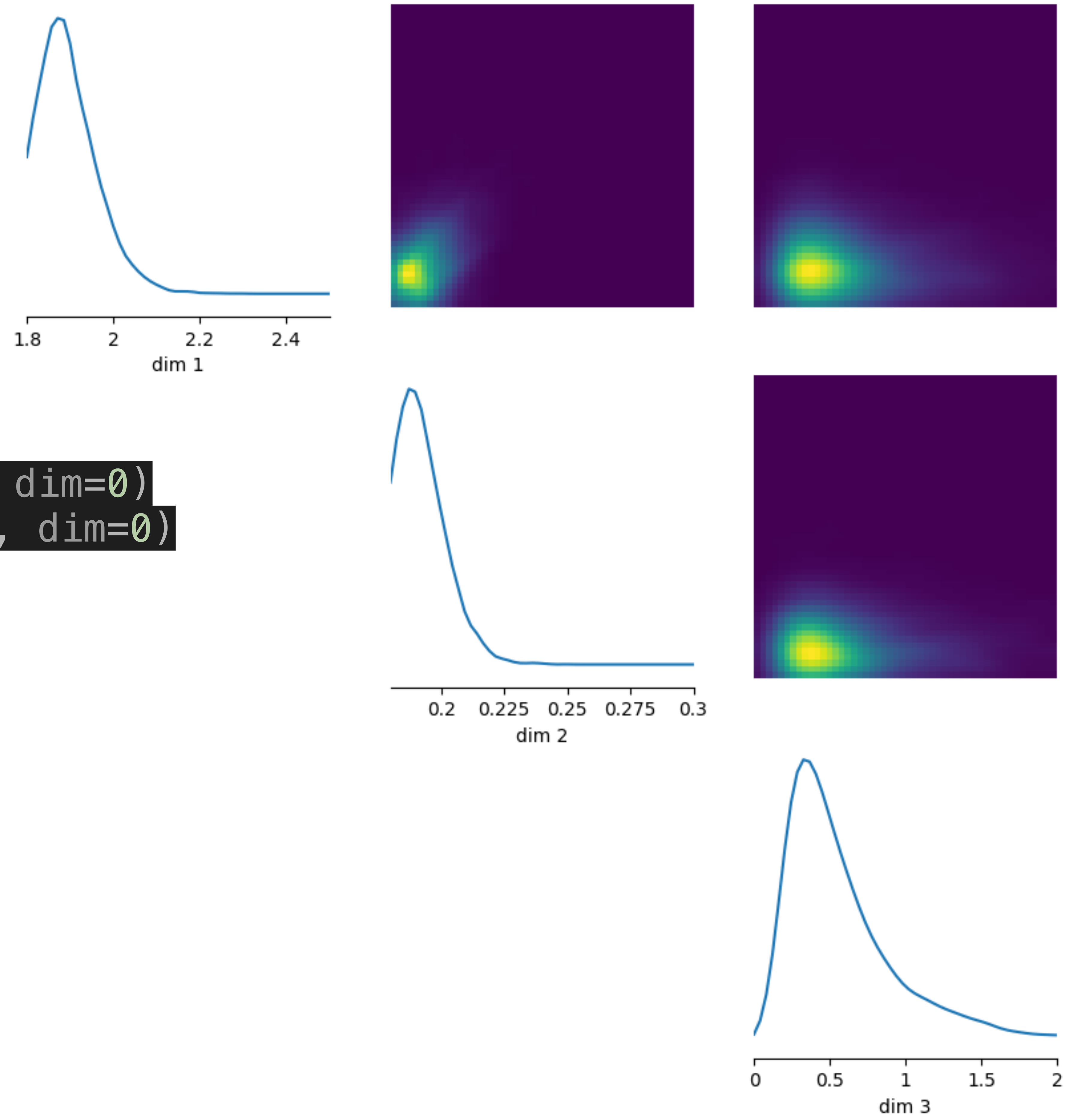
- Intervalos de credibilidad

```
low = samples.quantile(0.025, dim=0)
high = samples.quantile(0.975, dim=0)
```

- Correlaciones
- Posteriores marginales
- MAP: $\theta_{MAP} = \arg\max p(\theta \,|\, x)$

```
samples = posterior.sample((100000,), x=x_o)
log_probs = posterior.log_prob(samples, x=x_o)
map_estimate = samples[log_probs.argmax()]
```

```
from sbi.analysis import pairplot
pairplot(samples)
```



SBI no devuelve un único parámetro óptimo, sino una distribución posterior completa que cuantifica incertidumbre, degeneraciones y multimodalidad.

Notebook SBI