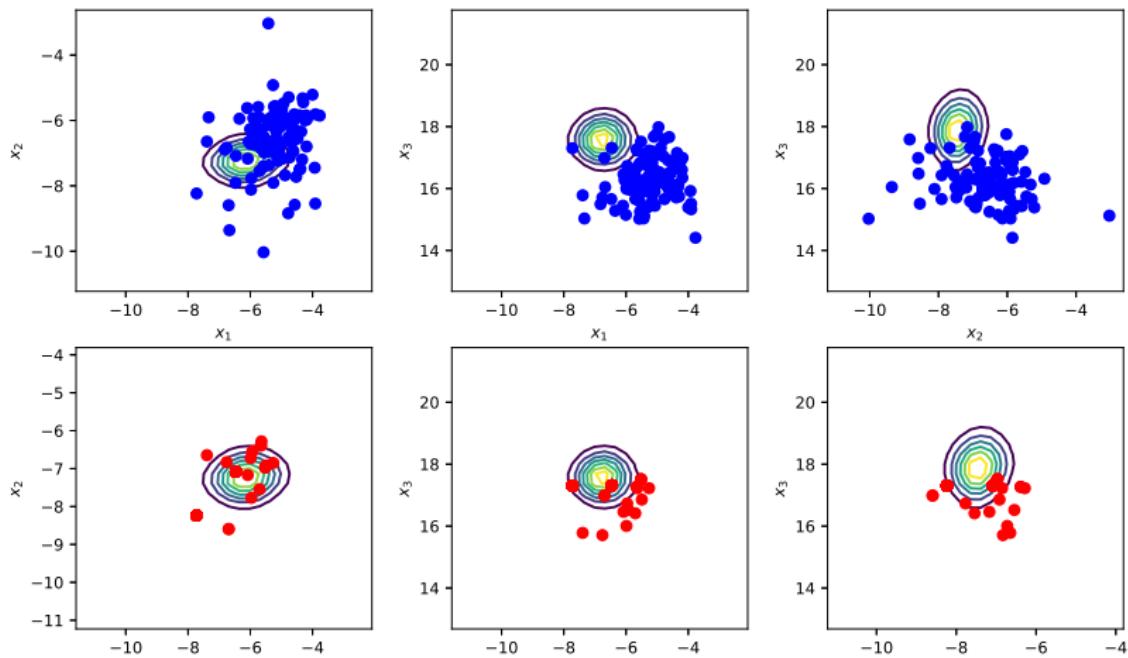


## Filtros de flujos de partículas

Transformando deterministica o estocasticamente las partículas de la densidad prior a la densidad posterior.

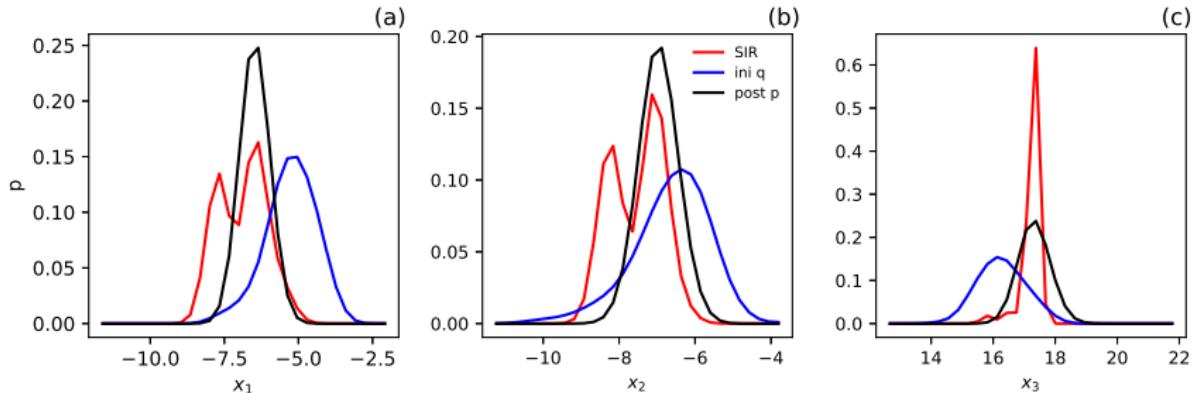
Hacia un muestreo basado en técnicas del aprendizaje automatizado.

## Motivación



Distribución de las partículas en el filtro del SIR para el sistema de Lorenz-63 usando 10 partículas. Puntos azules partículas de la densidad pronostico. Puntos rojos corresponden a las partículas que sobreviven después del resampling.

# Motivación



Distribuciones resultantes (marginalizadas) en cada una de las variables.  
Las curvas rojas son la inferencia que se esta haciendo con el SIR y las negras es lo que deberia dar (target density).

Es decir que debido al resampling terminamos con un empobrecimiento del muestreo y eso repercute en la representación de la posterior.

## Transporte óptimo

Dada la distribución de masa de la proposal  $q(\mathbf{x})$  y la distribución de masas de la target  $p(\mathbf{z})$ , se quiere transportar las unidades de masa de  $q$  a  $p$  usando un mapa  $T : \mathbf{x} \rightarrow \mathbf{z}$  que minimiza el costo del transporte, el cual esta definido por

$$C = \int q(\mathbf{x})c(\mathbf{x}, T(\mathbf{x}))d\mathbf{x} = \mathcal{E}_{x \sim q}[c(\mathbf{x}, T(\mathbf{x}))]$$

Rigurosamente  $T$  es un mapa de transporte difeomorfico  $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$  entre las dos medidas de probabilidad.

El mapa óptimo  $T$  es el que da la mínima  $C$ . Este es el problema clásico de Monge-Kantorovich.

## Distancia Wasserstein o la distancia de la tierra móvil (EMD)

Definimos a la distancia de Wasserstein entre dos densidades por

$$W(q(\mathbf{x}), p(\mathbf{z})) = \inf \{\mathcal{E} [d(\mathbf{X}, \mathbf{Z})]\}$$

donde  $\mathbf{X} \sim q(\mathbf{x})$ ,  $\mathbf{Z} \sim p(\mathbf{z})$ . El infimo de todas las posibles distribuciones conjuntas que tienen marginales  $q$  y  $p$ .

Cuando  $d(\mathbf{X}, \mathbf{Z}) = \|\mathbf{x} - \mathbf{y}\|$  la distancia se suele llamar **Earth mover's distance (EMD)**.

Entonces, el costo de transporte mínimo del problema de Monge-Kantorovich es la distancia de Wasserstein entre  $p$  y  $q$ .

En otras palabras

$$W(q(\mathbf{x}), p(\mathbf{z})) = \inf_{T \in \Gamma(q, p)} \int \int d(\mathbf{x}, \mathbf{z}) JT(\mathbf{x}, \mathbf{z}) d\mathbf{z} d\mathbf{x} = \inf_{T \in \Gamma(q, p)} \int d(\mathbf{x}, \mathbf{z}) dT(\mathbf{x}, \mathbf{z})$$

donde  $\Gamma(p, q)$  es el conjunto de todos los acoplamientos posibles entre  $q$  y  $p$ , mientras  $T$  es el acoplamiento que da la minima  $W$ .

## Divergencia de Kullback-Leibler

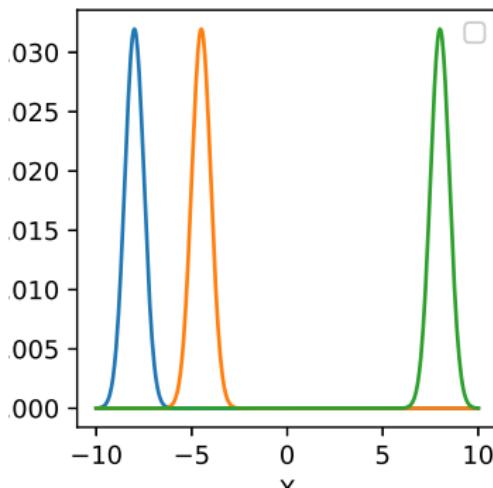
En otras circunstancias para medir la diferencia entre las variables aleatorias  $\mathbf{X}$  y  $\mathbf{Z}$  definimos a la entropía relativa (de la teoría de información de Shannon) entre dos densidades  $q$  y  $p$ ,

$$c(\mathbf{X}, \mathbf{Z}) = \log \left( \frac{q(\mathbf{z})}{p(\mathbf{x})} \right)$$

Esta definición no satisface con las propiedades de medidas de distancia y se la llama **divergencia** de Kullback-Leibler. En muchos contextos es suficiente.

La KLD solo mide diferencias entre dos densidades.

La distancia de Wasserstein mide la distancia entre dos densidades.



## Kullback-Leibler divergence

Considering the coupling  $\mathbf{z} = T(\mathbf{x})$ , the resulting transportation cost is

$$C(p, q) = \int q(T(\mathbf{x})) \log \left( \frac{q(T(\mathbf{x}))}{p(\mathbf{x})} \right) d\mathbf{x}$$

This is the Kullback-Leibler divergence between  $p$  and  $q(T)$ ,

$$C(p, q) = KL(p \| q).$$

We are seeking for the  $T$  that minimizes the differences between  $p$  and  $q$  in the sense of the Kullback-Leibler divergence.

In information terms, we are seeking to maximize the amount of information that is present in  $q$  from  $p$ , or to minimize the uncertainty, i.e. the relative entropy between  $q$  and  $p$ .

## Mapas óptimos en inferencia

Moselhy and Marzouk JCP 2012.

Buscamos un mapa  $T$  que transforme de la prior  $p_0(x)$  a la posterior  $p(x|y)$ .

$$p(f(x)) = \tilde{p}_0(x)|\text{Jac}(f)|^{-1}$$

donde  $\tilde{p}_0(x)$  es una aproximación a  $p_0$ . Entonces si planteamos un problema de optimización

$$D_{KL}(\tilde{p}_0 \| p_0) = \int p_0(x) \log \frac{p_0(x)}{\tilde{p}_0(x)} dx$$

Pensamos en

$$\tilde{p}_0(x) = F^{-1} p(y|f(x)) p_0(f(x)) |\text{Jac}(f)|$$

con  $F$  la constante de normalización.

Notando que  $-\log F$  queda como una constante aditiva de la KLD se puede redefinir la KLD por  $\tilde{D}_{KL}(\tilde{p}_0 \| p_0) = D_{KL}(\tilde{p}_0 \| p_0) + \log F$ . En términos del mapa  $f$  esta constante no tiene impacto.

## Mapas óptimos

Entonces se busca a la función  $f$  (mapa de transporte óptimo) que logre minimizar  $D_{KL}$ .

Para regularizar el problema para que exista un único mapa  $f^*$  utilizan transporte óptimo

$$\min_f \{D_{KL}(\tilde{p}_0 \| p_0) + \lambda \mathcal{E} [\|x - f(x)\|]\}.$$

Estamos buscando una **transformación global**  $f$  que nos lleve de  $p_0(x)$  a  $p(x|y)$ .

$f(x)$  yace en un espacio infinito, en general se puede parametrizar  $f$  a algun espacio de funciones. Moselhy and Marzouk JCP 2012 propone representar el mapa por una expansión caótica de polinomios.

Notar que no se puede utilizar funciones base localizadas porque se realiza una transformación global.

$$f(x) = \sum_i a_i \psi_i(\mathbf{x})$$

donde  $\psi_i$  son n-variate polinomios de la forma:  $\psi_i(\mathbf{x}) = \prod_{j=1}^{N_x} \phi_{ij}(x_j)$ ,  $\mathbf{i}$  son índices múltiples (para cada variable).

## Mapas óptimos

Usando estas expansiones se puede propagar fácilmente las incertezas y calcular momentos (covarianzas) muestrales condicionadas a las observaciones (Rupert and Miller, JCP'07).

Asumimos tenemos una muestra de  $N_o$  partículas de  $p_0(x)$ .

Este algoritmo requiere de optimización.

En Moselhy and Marzouk usan Newton y cuadrados mínimos nolineales.

Cualquier método se podría utilizar, e.g. gradientes conjugados.

Se requiere del adjunto de  $\mathcal{H}$  para poder evaluar los gradientes de la likelihood.

Tienen dificultades en situaciones complejas por la cantidad de coeficientes cuando se trabaja con polinomios de relativamente alto orden.

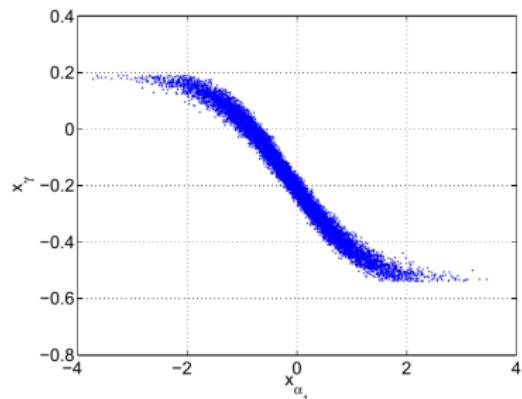
Proponen realizar mapas multiples para disminuir el numero de coeficientes.

## Ejemplo: Genetic togle switch

El modelo tiene 3 variables y 6 parametros.

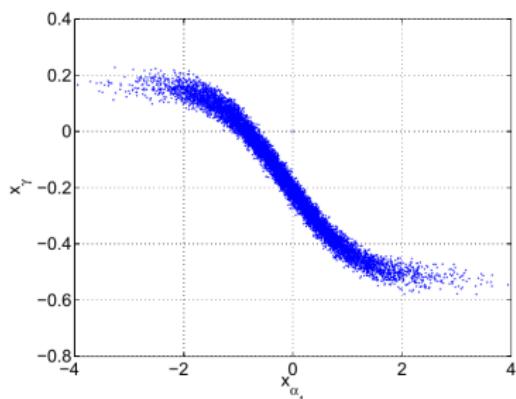
Problema inverso solamente.

Una sola de las variables es medida.



Optimal Map

Gaussian prior.  $10^6$  muestras.



MCMC

## Ensemble transform particle filter (ETPF)

Un trabajo posterior, Reich 2013 propone realizar una **transformación lineal** entre las partículas de la prior density y las de la posterior.

Supongamos que tenemos a las partículas de la prior density  $p_0 \sim x_k^{f(:N_p)}$ . Si hacemos el naive particle filter los pesos resultantes de la posterior usando las partículas de la prior son

$$w_j = \frac{p(\mathbf{y}_k | \mathbf{x}_k^{f(j)})}{\sum_{i=1}^{N_p} p(\mathbf{y}_k | \mathbf{x}_k^{f(i)})}$$

La idea sería transformar las partículas

$$p(\mathbf{x}_k | \mathbf{y}_{1:k}) \sim \left( x_k^{f(j)}, w_k^{(j)} \right)_{j=1}^{N_p}$$

a un conjunto de partículas con pesos iguales:

$$p(\mathbf{x}_k | \mathbf{y}_{1:k}) \sim \left( x_k^{a(j)}, 1/N_p \right)_{j=1}^{N_p}$$

Esto es similar al EnKF.

## La transformación como un problema de transporte óptimo

Queremos encontrar un optimal linear coupling  $\mathbf{T}$  entre  $\mathbf{x}_k^{a(j)}$  y  $\mathbf{x}_k^{f(j)}$  tal que

$$T_{ij} \geq 0$$

$$\sum_{i=1}^{N_p} T_{ij} = 1/N_p, \quad \sum_{j=1}^{N_p} T_{ij} = w_k^{(i)}$$

Se quiere minimizar la distancia:

$$\mathcal{E}_{X^f, X^a}(\mathbf{x}^f, \mathbf{x}^a) = \sum_{i,j=1}^{N_p} T_{i,j} d(\mathbf{x}^{f(i)}, \mathbf{x}^{f(j)})$$

se define con la distancia Euclidea:

$$d(\mathbf{x}^{f(i)}, \mathbf{x}^{f(j)}) = \|\mathbf{x}^{f(i)} - \mathbf{x}^{f(j)}\|^2$$

Este problema es un problema de transporte óptimo y se soluciona encontrando la  $T$  que minimiza la distancia de Wasserstein que en este caso es la Earth mover distance (EMD).

Las partículas de la posterior son la transformación de las de pronóstico:

$$\mathbf{x}^{a(j)} = \sum_{i=1}^M T_{ij} \mathbf{x}^{f(i)}$$

## ETPF

El ETPF es similar entonces al SIR pero se reemplaza el resampling por la transformación.

### PROs

- ▶ Fácil implementación.
- ▶ Fácil para realizar localización debido a la presencia del acoplamiento  $T$  entre variables.

### CONS

- ▶ Una transformación lineal no puede generar densidades posteriores complejas a partir de densidades prior simples.
- ▶ El ETPF es ciego a las observaciones (solo transforma acorde a los pesos del prior).
- ▶ De todas maneras se requiere de un resampling estocástico.

## Homotopía

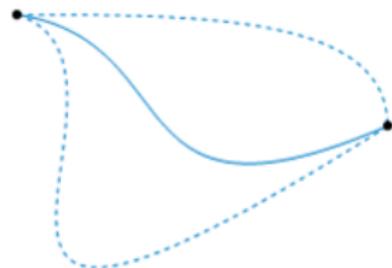
**Homotopía:** Dadas dos funciones cada una en un espacio topológico arbitrario, son las deformaciones continuas que llevan de una función a la otra.

**Example:**

$$q(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}, \quad p(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}$$

Then,

$$T(\mathbf{x}, \lambda) : \mathbb{R}^n \times [0, 1] \rightarrow \mathbb{R} \quad T(\mathbf{x}, \lambda) = p(\mathbf{x})^\lambda q(\mathbf{x})^{1-\lambda}$$



Las deformaciones son continuas e invertibles.

**Podemos usar el concepto de homotopía para muestreo de importancia (importance sampling)?**

## Regla de Bayes homotópica

Dadas dos funciones de probabilidad,  $p$  y  $q$ , relacionamos a estas a través de una homotopía por ejemplo

$$q_\lambda(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{x})^\lambda q(\mathbf{x})}{C_\lambda}$$

Esto define una transformación continua desde el prior a la posterior. El parámetro  $\lambda$  puede ser interpretado como un pseudo-tiempo que varía entre 0 (prior) a 1 (posterior) en un tiempo real fijo.

Este caso particular de homotopía es conocido como ‘tempering’ (templado).

## Tempering + jittering

Notar que estamos introduciendo la función verosimilitud de forma suave. De esta forma introducimos los efectos de las observaciones en muchos pasos.

En general la prior es dispersa mientras la verosimilitud es un “pico”, al realizar el tempering lo que estamos haciendo es transformar suavemente de una densidad ancha a una concentrada (o eventualmente multimodal con varios picos concentrados).

**Pero que hacemos con las partículas?** Si solo cambiamos los pesos en cada  $\delta\lambda$  no sirve de nada.

“Jittering” resampleo estocástico y adición de model error en cada pseudo-time (e.g. trabajos de Dan Crisan y colaboradores).

## Flujos homotópicos

Los trabajos de Daum and Huang, 2008, 2009, etc fueron los primeros en proponer flujos homotópicos.

Si tenemos una densidad de probabilidad que esta inmersa en un flujo, los cambios de esta vienen dados por la ecuación de Fokker-Planck

$$\partial_\lambda q_\lambda = -\text{Tr} [\partial_{\mathbf{x}_\lambda} (\mathbf{v}_\lambda(\mathbf{x}_\lambda) q_\lambda(\mathbf{x}_\lambda))] + \sum_{i,j} \partial_{\mathbf{x}_i \mathbf{x}_j}^2 (D_{i,j}(\mathbf{x}_\lambda) q_\lambda(\mathbf{x}_\lambda))$$

donde  $D_{i,j}(\mathbf{x}_\lambda) = \frac{1}{2} \boldsymbol{\eta}_\lambda(\mathbf{x}_\lambda) \boldsymbol{\eta}_\lambda(\mathbf{x}_\lambda)^\top$  es el término difusivo y  $\mathbf{v}_\lambda$  es la velocidad del flujo.

Si no tenemos difusión/dispersión  $\boldsymbol{\eta}_\lambda(\mathbf{x}_\lambda) = 0$  y entonces  $D_{i,j} = 0$ , la ecuación de Fokker-Planck se simplifica en la ecuación de Liouville .  $\lambda$  pseudo-time.

Alternativamente puede interpretarse como la ecuación de continuidad/conservación de la masa donde  $q$  hace las veces de la densidad del fluido.

## Flujos de partículas homotópicos templados

Las densidades son interpretadas a través de las partículas que conforman la muestra  $\{x^{(j)}\}_{j=1}^{N_p} \sim q(\mathbf{x})$ .

Las deformaciones continuas son pensadas como partículas que se mueven en un flujo gobernadas por **ecuaciones diferenciales estocásticas**

$$\frac{d\mathbf{x}_\lambda}{d\lambda} = \mathbf{v}_\lambda(\mathbf{x}_\lambda) + \boldsymbol{\eta}_\lambda(\mathbf{x}_\lambda)$$

donde  $\mathbf{v}_\lambda$  es la velocidad, y  $\boldsymbol{\eta}_\lambda$  es un término aleatorio que representa los procesos difusivos.

## Flujos de partículas homotópicos templados

Los movimientos de las partículas es la evolución de la densidad desde la prior a la posterior. Daum and Huang propone la fórmula de tempering alrededor de  $\lambda$  se obtiene:

$$q_{\lambda+\delta\lambda}(\mathbf{x}) \approx q_\lambda(\mathbf{x})[1 - \delta\lambda(\log p(\mathbf{y}|\mathbf{x}) - \log p_\lambda(\mathbf{y}))]$$

Asumiendo que  $D_{i,j} = 0$ , la restricción del flujo resultante de la ecuación de Liouville es

$$\nabla \cdot (q_\lambda \mathbf{v}_\lambda) = q_\lambda(\mathbf{x})[1 - \delta\lambda(\log p(\mathbf{y}|\mathbf{x}) - \log p_\lambda(\mathbf{y}))]$$

La solución de esta restricción para  $\mathbf{v}_\lambda$  no es única. Nuevamente deberíamos recurrir a transporte óptimo o alguna otra restricción para determinar  $\mathbf{v}_\lambda$ .

**Sin embargo ya restringimos al templado a la evolución de la densidad!**

## Aproximaciones: flujos de partículas Gaussianos

La mayoría de los trabajos basados en los flujos de partículas templados (Daum and Huang 08,11,13; **Bunch and Gosill JASA 2016**, Li and Coates 2017) se basan en distintos sabores de la aproximación Gaussiana. En ese caso el término de la velocidad viene dado por

$$\mathbf{v}(\mathbf{x}_\lambda^{(j)}, \lambda) = \mathbf{A}(\lambda)\mathbf{x}_\lambda^{(j)} + \mathbf{b}(\lambda)$$

where

$$\mathbf{A}(\lambda) = -\frac{1}{2}\mathbf{P}\mathbf{H}^\top(\lambda\mathbf{H}\mathbf{P}\mathbf{H}^\top + \mathbf{R})^{-1}\mathbf{H}$$

$$\mathbf{b}(\lambda) = (\mathbf{I} + 2\lambda\mathbf{A})[(\mathbf{I} + \lambda\mathbf{A})\mathbf{P}\mathbf{H}^\top\mathbf{R}^{-1}\mathbf{y} + \mathbf{A}\bar{\mathbf{x}}_0]$$

donde  $\mathcal{H}$  se evalua en el estado medio o en las partículas dependiendo de las diferentes aproximaciones.

**Esta aproximación es diferente de aproximar la densidad proposal con el resultado del filtro de Kalman (e.g. Robert and Kunsch).**

Aquí las partículas se mueven suavemente y las covariances evolucionan junto con el campo de velocidad.

Aun así notar que la velocidad es un campo lineal.

## Filtro de partículas por mapeo variacional

Pulido and vanLeeuwen, JCP 2019.

Idea: realizar una **optimización** para encontrar la muestra de partículas  $\mathbf{x}_k^{(1:N_p)}$  que **mejor** representa a la densidad posterior.

Para esto deberíamos optimizar:

$$\mathcal{D}_{KL}(q(\mathbf{x}_k) | p(\mathbf{x}_k)) = \int q(\mathbf{x}_k) \log \left[ \frac{q(\mathbf{x}_k)}{p(\mathbf{x}_k)} \right] d\mathbf{x}$$

donde  $q$  es dado por una muestra de partículas.

Dada la transformación  $\mathbf{z} = T(\mathbf{x})$ , la densidad transformada resultante es

$$q_T(\mathbf{z}) = [J_{\mathbf{z}}(\mathbf{x})]^{-1} q(\mathbf{x})$$

y la nueva KLD:

$$\mathcal{D}_{KL}(\mathbf{q}_T | p) = \int q_T(\mathbf{z}) \log \left[ \frac{q_T(\mathbf{z})}{p(\mathbf{z})} \right] d\mathbf{z}$$

Se espera que:  $\mathcal{D}_{KL}(\mathbf{q}_T | p) < \mathcal{D}_{KL}(\mathbf{q}_0 | p)$

## Transformación local. Perturbación infinitesimal a la identidad

En lugar de realizar una transformación global como en Moselhy and Marzouk hacemos una transformación local. Buscamos transformar infinitesimalmente (suavemente) a la densidad  $q$ .

Entonces usamos una transformación homotópica con un pseudo-tiempo. La transformación global será la composición de un conjunto de mapas locales:  $T(\mathbf{x}) = T_I \circ \dots \circ T_2 \circ T_1(\mathbf{x})$ .

Supongamos que realizamos una transformación suave a lo largo de la dirección  $\phi(\mathbf{x})$

$$\mathbf{z} = T(\mathbf{x}) = \mathbf{x} + \epsilon \phi(\mathbf{x})$$

asumiendo  $\epsilon$  pequeño, esta puede ser vista como una pequeña perturbación a la identidad.

## Derivada de la divergencia de KL

La derivada de Gateaux de  $\mathcal{D}_{KL}$  (pensado como un funcional) es dada por

$$D_\phi \mathcal{D}_{KL} = - \int q(\mathbf{x}) \, d_\epsilon \log p_{T^{-1}}(\mathbf{x})|_{\epsilon=0} \, d\mathbf{x}.$$

Entonces, la derivada direccional es

$$D_\phi \mathcal{D}_{KL} = - \int q(\mathbf{x}) \left[ (\nabla_x \log p(\mathbf{x}))^\top \phi(\mathbf{x}) + \text{Tr}(\nabla_x \phi) \right] \, d\mathbf{x}.$$

Esto nos da como cambia  $\mathcal{D}_{KL}$  en la dirección  $\phi$  para cada  $\mathbf{x}$ .

Pero lo que necesitamos encontrar es la dirección de mayor descenso para cada  $\mathbf{x}$ .

## Incrustando el mapa en un espacio de Hilbert (RKHS)

Un avance significativo para la optimización con partículas fue obtenido por Liu and Wang NIPS 2016. Ellos propusieron que el mapa este en un reproducing kernel Hilbert space (RKHS)  $\mathcal{F}$ .

Entonces cualquier función en  $\mathcal{F}$  puede representarse por la propiedad de reproducción

$$\phi(\mathbf{x}) = \langle K(\cdot, \mathbf{x}), \phi(\cdot) \rangle_{\mathcal{F}}$$

Reemplazando en la derivada direccional

$$D_{\phi} \mathcal{D}_{KL} = \left\langle - \int q(\mathbf{x}') [K(\mathbf{x}', \cdot) \nabla_x \log p(\mathbf{x}') + \nabla_x K(\mathbf{x}', \cdot)] \, d\mathbf{x}', \, \phi(\cdot) \right\rangle_{\mathcal{F}}.$$

La definición del campo de gradientes es  $D_{\phi} f(\mathbf{x}) = \langle \nabla f(\mathbf{x}), \phi \rangle$  para  $|\phi| = 1$ , entonces hemos obtenido el gradiente de la KLD:

$$\nabla \mathcal{D}_{KL} = - \mathcal{E}_{x' \sim q} [K(\mathbf{x}', \mathbf{x}) \nabla_x \log p(\mathbf{x}') + \nabla_x K(\mathbf{x}', \mathbf{x})].$$

Para un dado  $\mathbf{x}$ , el gradiente nos dice la dirección de máximo ascenso y el cambio de la KLD si realizamos una transformación local alrededor de ese punto.

## Propiedades de convergencia

Usando la propiedad de reproducción encontramos que el

$$\mathbf{v}_i = -\nabla \mathcal{D}_{KL} = q(\mathbf{x}_i) [\nabla_x \log p(\mathbf{x}_i) - \nabla_x q(\mathbf{x}_i)].$$

Asumiendo transformaciones continuas y un flujo de velocidad, la ecuación de Liouville resultante es:

$$\partial_\lambda q = -\nabla q(\mathbf{x}) [\nabla_x \log p(\mathbf{x}) - \nabla_x q(\mathbf{x})].$$

Ecuación nolineal para medios porosos. La función  $q$  converge a la target  $p$  para  $\lambda \rightarrow \infty$ . Esto ha sido demostrado por el pionero trabajo de Tabak y Vanden-Eijnden (2010).

## Optimización por descensos de gradientes?

Una vez que hemos obtenido el gradiente de la función de costo estariamos en condiciones de aplicar un método de optimización por ejemplo un por descensos de gradientes

$$\mathbf{x}_{\lambda_{i+1}} = \mathbf{x}_{\lambda_i} - \alpha \nabla \mathcal{D}_{KL}$$

Sin embargo notar que lo que estamos tratando de optimizar es un funcional.

Requerimos de una nueva  $\rho_{i+1}(\mathbf{x})$  y esa es representada a través de un conjunto de partículas IID (la muestra).

Entonces no es una optimización de una condición inicial simple sino de un conjunto de partículas:

$$\mathbf{x}_{\lambda_{i+1}}^{(j)} = \mathbf{x}_{\lambda_i}^{(j)} - \alpha \nabla \mathcal{D}_{KL}(\rho_{i+1}(\mathbf{x}) | \mathbf{x}_{\lambda_i}^{(1:N_p)})$$

El gradiente de la función de costo depende de las partículas anteriores. Es decir las partículas son interactivas y dan lugar a la función de costo en cada paso.

## Integración de Monte Carlo

Si asumimos que la densidad intermedia  $q$  solo la conocemos a través de la muestra (tanto inicialmente como en el mapeo  $i$ -esimo)

$$q_i(\mathbf{x}) = \sum_{j=1}^{N_p} \delta(\mathbf{x} - \mathbf{x}_i^{(j)})$$

Entonces el gradiente de la KLD se puede evaluar a través de una integración de Monte-Carlo

$$\nabla \mathcal{D}_{KL}(\mathbf{x}_{k,i}) = -\frac{1}{N_p} \sum_{l=1}^{N_p} \left[ K(\mathbf{x}_{k,i}^{(l)}, \mathbf{x}) \nabla_{\mathbf{x}} \log p(\mathbf{x}_{k,i}^{(l)}) + \nabla_{\mathbf{x}} K(\mathbf{x}_{k,i}^{(l)}, \mathbf{x}) \right].$$

Entonces las posiciones de las partículas en una dada iteración (mapeo) nos definen al gradiente. Estas forman la  $q_i$  y por ende en el gradiente de la KLD.

El primer término actúa de **campo central** atrayendo las partículas al potencial  $\log p(\mathbf{x})$ . Para esto se realizan una promoción localizada (de las partículas cercanas).

Mientras  $\nabla_{\mathbf{x}} K(\mathbf{x}', \mathbf{x})$  es una **fuerza repulsiva** entre las partículas.

## Fuerza repulsiva en un kernel RBF

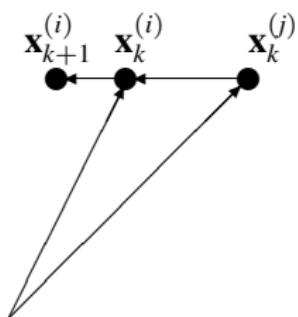
Suppose  $k(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{1}{2h} \|\mathbf{x} - \mathbf{x}'\|^2\right)$

$$\nabla k(\mathbf{x}, \mathbf{x}') = -\frac{1}{h}(\mathbf{x} - \mathbf{x}')k(\mathbf{x}, \mathbf{x}')$$

Partículas que están lejanas no contribuyen ya que  $\|\mathbf{x}_i - \mathbf{x}_j\|^2 \gg h$ . Sin embargo las partículas que están dentro de la escala del kernel sentirán una fuerza repulsiva dada por:

$$\mathbf{x}_{k+1}^{(i)} = \mathbf{x}_k^{(i)} + \gamma(\mathbf{x}_k^{(i)} - \mathbf{x}_k^{(j)})$$

where  $\gamma = \frac{k(\mathbf{x}_k^{(i)}, \mathbf{x}_k^{(j)})}{h} > 0$ .



## Algoritmo del filtro de partículas de mapeo variacional

---

**Input:** Given  $\{\mathbf{x}_{k-1}^{(j)}\}_{j=1}^{N_p}$ ,  $\mathbf{y}_k$ ,  $\mathcal{M}(\cdot)$ ,  $p(\mathbf{y}|\mathbf{x})$  and  $p(\boldsymbol{\eta})$

---

```
for  $j = 1, N_p$  do
     $\mathbf{x}_{k,0}^{(j)} \leftarrow \mathcal{M}(\mathbf{x}_{k-1}^{(j)}) + \boldsymbol{\eta}_k$                                 ▷ Forecast stage
end for

while  $|\phi_{i+1}| - |\phi_i| > \delta$  do
    for  $j = 1, N_e$  do
         $\phi(\mathbf{x}_{k,i-1}^{(j)}) \leftarrow$ 
         $\frac{1}{N_p} \sum_{l=1}^{N_p} \left[ K(\mathbf{x}_{k,i-1}^{(l)}, \mathbf{x}_{k,i-1}^{(j)}) \nabla_x \log p(\mathbf{x}_{k,i-1}^{(l)}) + \nabla_x K(\mathbf{x}_{k,i-1}^{(l)}, \mathbf{x}_{k,i-1}^{(j)}) \right]$ 
         $\mathbf{x}_{k,i}^{(j)} \leftarrow \mathbf{x}_{k,i-1}^{(j)} + \epsilon \phi(\mathbf{x}_{k,i-1}^{(j)})$       ▷  $\epsilon$  from ADAM to account part of the
                                                    Hessian
    end for
end while
```

---

**Output:**  $\{\mathbf{x}_k^{(j)}\}_{j=1}^{N_e}$

---

Los cálculos de los Jacobianos de la transformación no se realizan.

## Densidad apriori

Aun cuando el algoritmo variacional no requiere de una densidad apriori conocida, requiere de una target density conocida para el calculo de su gradiente:

$$\nabla \log p(\mathbf{x}|\mathbf{y}) = \nabla p(\mathbf{y}|\mathbf{x}) + \nabla p_0(\mathbf{x})$$

Si asumimos que la prior es el resultado de pronosticos con error de modelo aditivo y Gaussiano en cada una de las particulas se puede asumir que

$$p_0(\mathbf{x}) = \sum_{j=1}^{N_p} \mathcal{N}(\mathbf{x}^{f(j)}, \mathbf{Q})$$

es decir asumimos la densidad prior es una mezcla de Gaussianas.

Si el numero de particulas es lo suficientemente grande esta no es una aproximacion problemática, sin embargo la covarianza del model error  $\mathbf{Q}$  pasa a tener un rol esencial.

Los filtros de partículas que funcionan con el gradiente de la posterior requieren de la covarianza del error de modelo. Actualmente un área de activa investigación es la estimación de  $\mathbf{Q}$ .

Existen técnicas basadas en el algoritmo de Expectation-Maximization

## Kernels

En Pulido and vanLeeuwen (2019) asumimos que los kernels son funciones base radiales de la forma

$$K(\mathbf{x}, \mathbf{x}') = \exp[-1/2\|\mathbf{x} - \mathbf{x}'\|_{\mathbf{A}}] = \exp[-1/2(\mathbf{x} - \mathbf{x}')^\top \mathbf{A}^{-1}(\mathbf{x} - \mathbf{x}')]$$

En alta dimensionalidad una cuestión esencial es la covarianza del kernel **A**.

Opciones posibles:

1. Proporcional a la covarianza del model error  $\mathbf{A} = \alpha \mathbf{Q}$
2. Proporcional a la covarianza del análisis aislado  
$$\mathbf{A} = \alpha(\mathbf{Q}^{-1} + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H})$$
3. Proporcional a la covarianza muestral en cada iteración:

$$\mathbf{A} = 1/(N_p - 1) \sum_{j=1}^{N_p} (\mathbf{x}^j - \bar{\mathbf{x}})^\top (\mathbf{x}^j - \bar{\mathbf{x}})$$

## Optimizacion por enjambres de partículas

En asimilacion variacional minimizamos a una funcion de costo en función de una sola muestra. En swarm optimization utilizamos un conjunto de partículas interactuantes.

Los sistemas de partículas interactuantes fueron inicialmente propuestos por Del Moral para la inferencia Bayesiana, Del Moral, 2004. Feynman-Kac formulae.,

Particle swarm optimization (Kennedy and Elberhart, 1995).

- ▶ Interacciones son usualmente locales y con el entorno.
- ▶ El comportamiento colectivo es descentralizado pero es auto-organizado → lleva a la aparición de comportamiento global inteligente.

Ejemplos: bird flocking, colonias de hormigas. Esto es diferente de enjambres de abejas (EnVar?).

## Assimilación en un ciclo por el enjambre

Dado el enjambre de partículas, la muestra, en el tiempo  $k - 1$ ,  $\{\mathbf{x}_{k-1}^{1:N_e}\}$  y una observacion en  $k$ ,  $\mathbf{y}_k$ .

Como se mueve a las partículas al tiempo  $k$ ?

El enjambre se mueve colectivamente acorde a la nueva informacion  $\mathbf{y}_k$  pero cada particula solo usa la informacion local (modelo dinamico y vecinos).

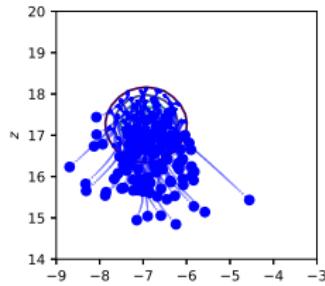
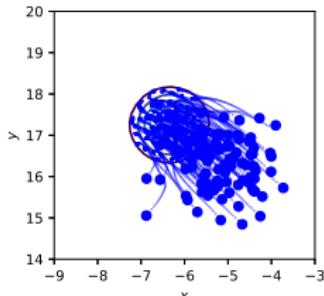
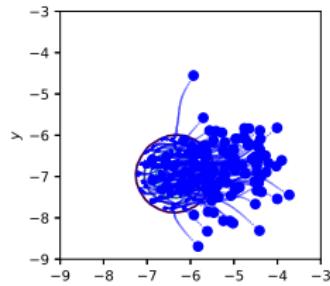
Starlings responde a un numero fijo de vecinos cercanos (Young et al. Plos CB, 2013)

Solo 7!!!!

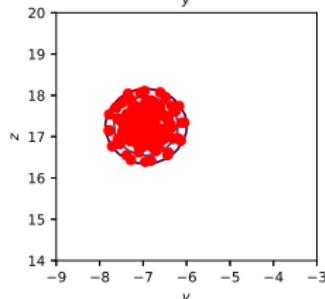
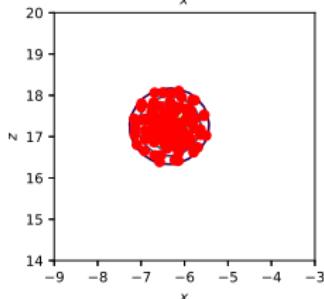
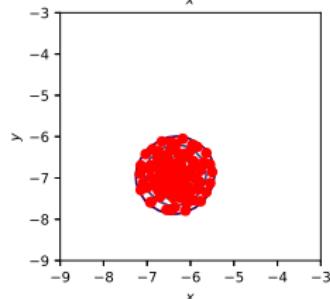
Localidad (kNNs) + Entorno



# Experiment with Lorenz-63



Distribution of particles  
of the prediction density

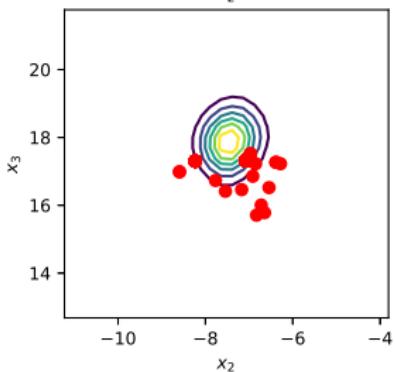
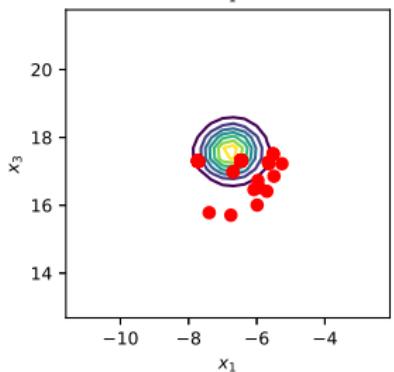
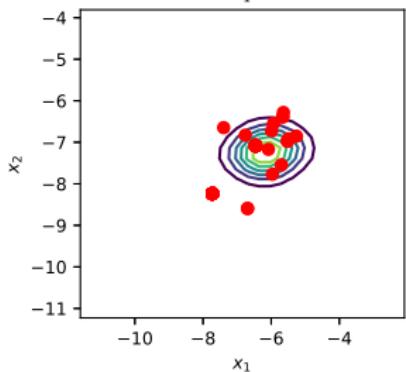
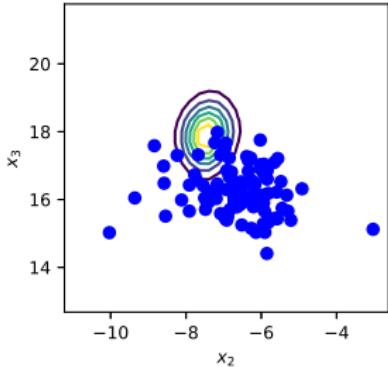
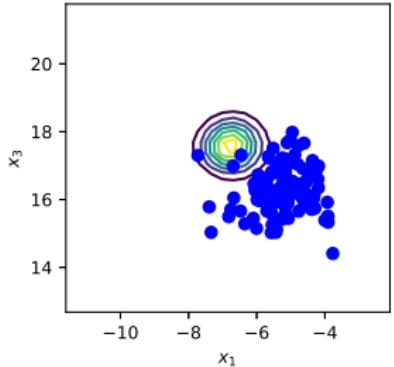
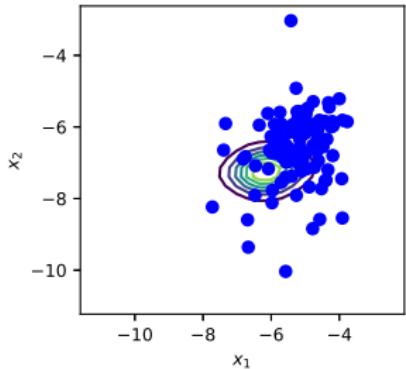


Distribution of particles  
of the posterior density  
after the VMPPF

Model error  $\mathcal{N}(\mathbf{0}, \mathbf{Q})$

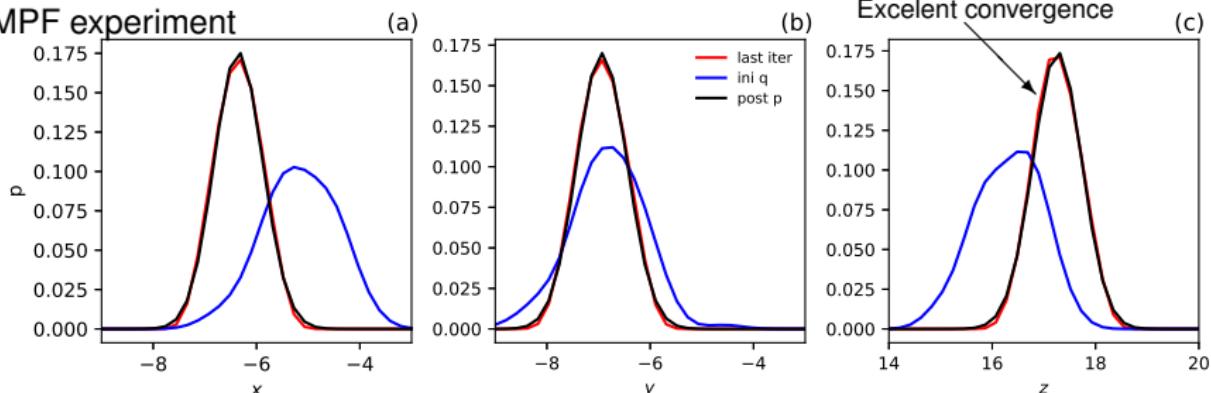
Observational error  $\mathcal{N}(\mathbf{0}, \mathbf{R})$

# SIR

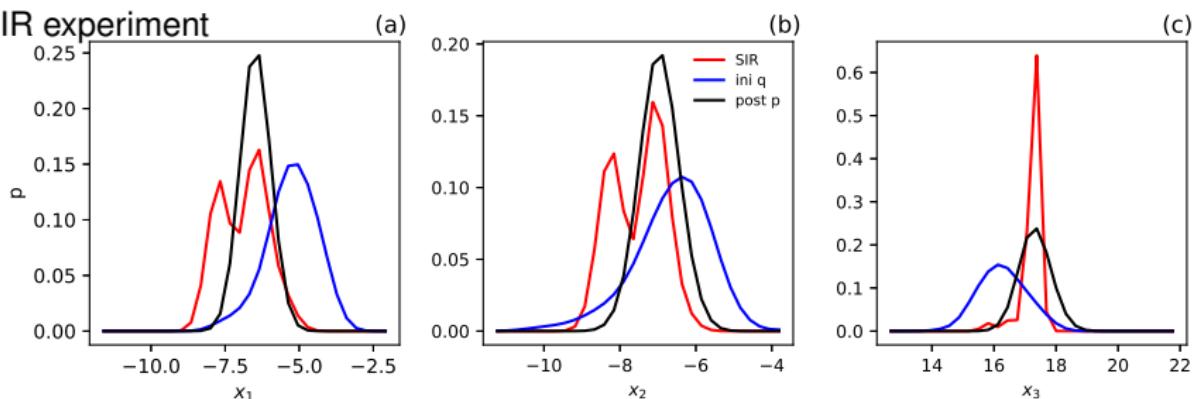


# Marginal distributions

MPF experiment

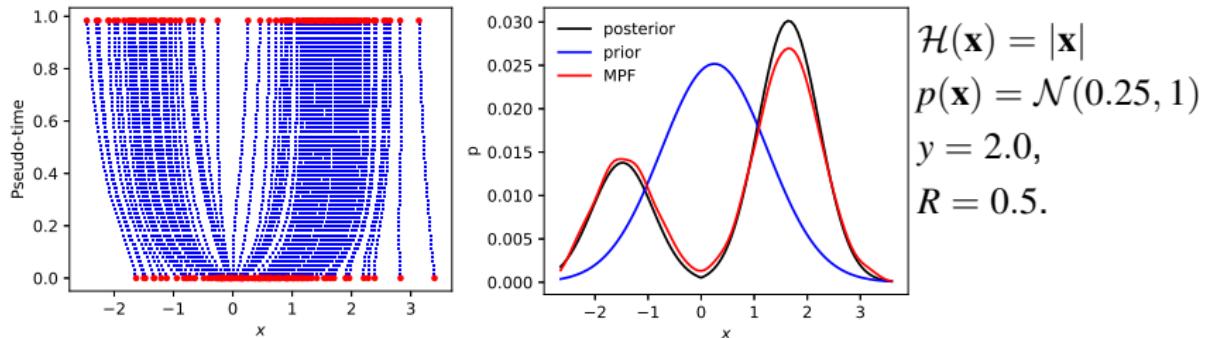


SIR experiment



RMSE for 100 particles is 0.482, for 5 particles is 0.489!

# Nonlinear observational operator



Inferencia Bayesiana en un problema inverso con posterior bimodal.

## La localidad que introducen los kernels en la posterior

Para calcular los gradientes de la verosimilitud para mapeos observacionales nolineales podemos usar la granularidad que esta introduciendo el VMPF. En ambas el gradiente de la posterior y de la intermedia.

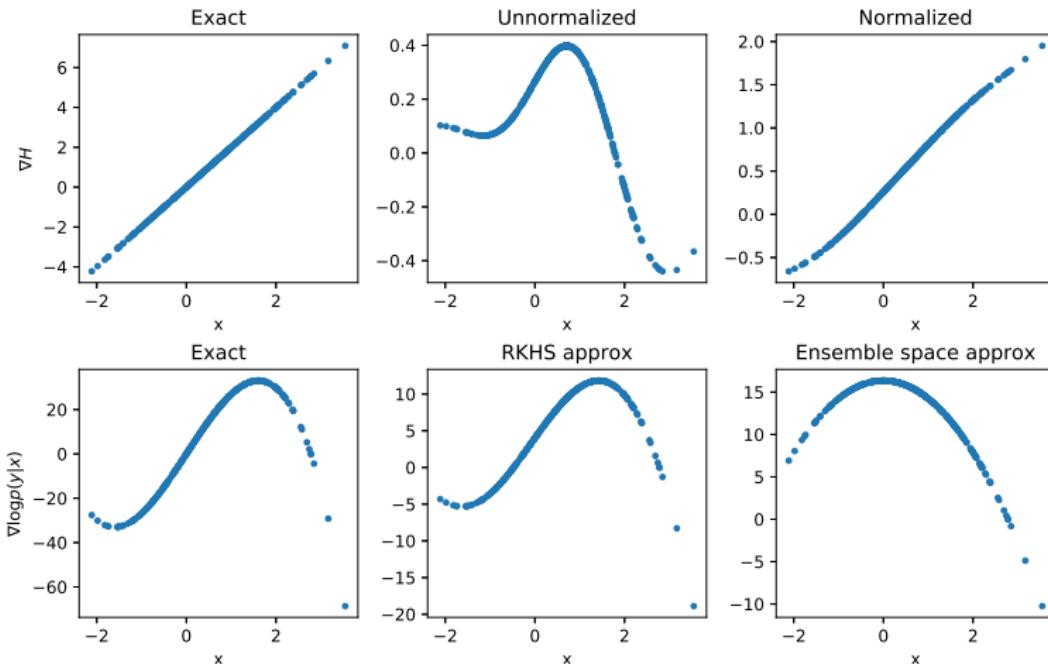
La propiedad de reproducción del RKHS aplicada al mapa observacional  $\mathcal{H}$ :

$$\nabla \mathcal{H}(\mathbf{x}) \approx \frac{1}{N_p} \sum_{j=1}^{N_p} \mathcal{H}(\mathbf{x}^j) \nabla K(\mathbf{x}, \mathbf{x}^j).$$

Normalizando el peso de los kernels:

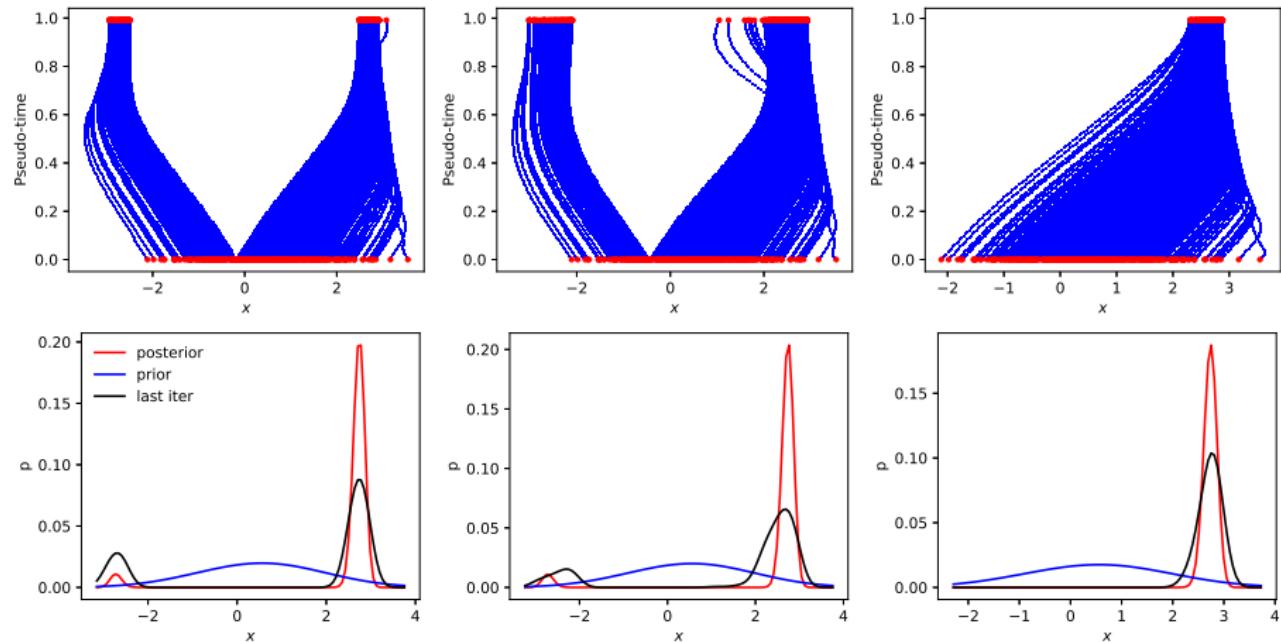
$$\mathcal{H}(\mathbf{x}) \approx \frac{\sum_{j=1}^{N_p} \mathcal{H}(\mathbf{x}^j) K(\mathbf{x}, \mathbf{x}^j)}{\sum_{l=1}^{N_p} K(\mathbf{x}, \mathbf{x}^l)}$$

# Gradiente de $H$



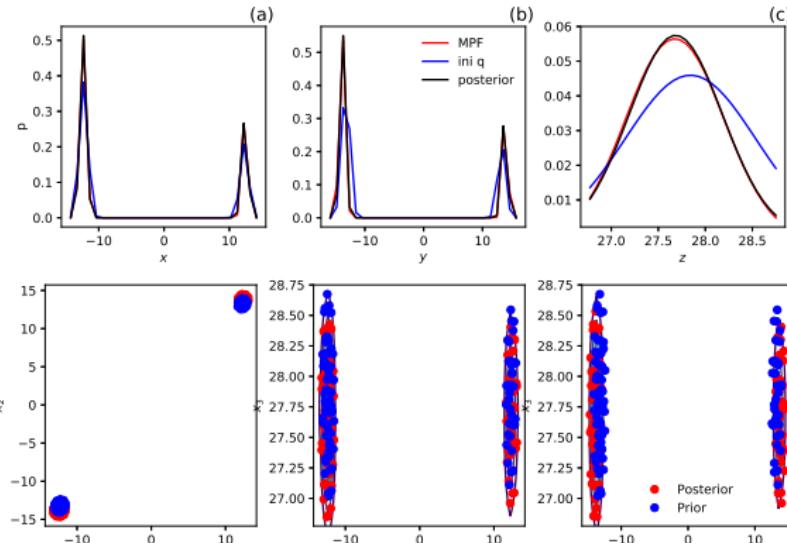
Comparacion de los gradientes para el caso  $\mathcal{H}(x) = x^2$ . Aproximacion con peso y sin pesos. Abajo: Comparacion del gradiente de la funcion verosimilitud para la aproximacion RKHS y para la del ensamble.

# Modelo inverso con $H$ no lineal



Caso  $\mathcal{H}(x) = x^2$  y prior Gaussiano. Exacto, aproximaciones de RKHS y ensamble.

# L63 con H nolineal



$$L63 - \mathcal{H}(\mathbf{x}) = |\mathbf{x}|$$

The adjoint of  $\mathcal{H}$  is avoided expressing  $\mathcal{H}$  in the RKHS (Pulido et al ICCS 2019)

## Flujos de Langevin

La evolucion de las particulas independientes es dada por

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \delta t \nabla \log p(\mathbf{x}_k) + 2\sqrt{\delta t} \zeta, \quad \zeta \sim (0, 1)$$

donde la evolucion tiene una componente estocastica.

La ecuacion que gobierna la densidad de la variable  $\mathbf{x}$  es dada por

$$\partial_t q + \nabla \cdot (q \nabla \log p - \nabla q) = 0$$

esta es la ecuación de Fokker-Planck lineal.

La razon de convergencia del flujo de Langevin de  $q$  a la target density  $p$  es dada por

$$\frac{d}{dt} \mathcal{D}_{KL} = -\mathbb{F}(q, p)$$

donde  $\mathbb{F}(q, p) = \|\nabla \log(q/p)\|_{L_q^2}^2$  es la divergencia de Fisher.

Este flujo es un flujo de gradiente en la métrica de Wasserstein.

## Aplicación de flujo Langevin a SMC

Como en el MPF se mueven las partículas de la prior a la posterior pero usando un flujo de Langevin (Liu et al NIPS 2018).

En este caso en lugar de representar al mapa, incrustamos directamente a la densidad intermedia en el RKHS,

$$q(\mathbf{x}) = \int q(\mathbf{x}') K(\mathbf{x}', \mathbf{x}) d\mathbf{x}' = \frac{1}{N_p} \sum_{j=1}^{N_p} K(\mathbf{x}^j, \mathbf{x})$$

En este caso es una representación por kernel density estimation (KDE) de  $q$ . El flujo determinístico resultante es

$$\mathbf{v} = -\nabla \mathcal{D}_{KL} = \nabla \log p(\mathbf{x}) - \frac{1}{N_p} \sum_{j=1}^{N_p} \nabla K(\mathbf{x}^j, \mathbf{x})$$

Tanto VMPF (y los SGVD en los que esta basado) como los flujos Langevin son técnicas con potencial para la aplicación en muy alta dimensionalidad dado que estan basadas en optimización.

## Desafios de los filtros con flujos de particulas en alta dimensionalidad

- ▶ Los criterios de distancias Eulerianas entre particulas pierden distinguibilidad (Aggarwal et al LNCS 2001).
- ▶ Aprendizaje de la granularidad del espacio. Matriz de covarianza del kernel  $A$ .
- ▶ Covarianza del model error. Su rol en las Gaussianas mezcladas.