Clasificación por KDE con Distancia de Fermat en variedades desconocidas: una aproximación empírica.

Lic. Gonzalo Barrera Borla (IC), Dr. Pablo Groisman (DM) (Facultad de Ciencias Exactas y Naturales, UBA)

I. Síntesis

Siguiendo a Loubes & Pelletier [1], programamos y evaluamos un algoritmo de clasificación basado en *Kernel Density Estimation* («KDC») para v.a. soportadas en variedades de Riemann, «KDC». Luego, reemplazamos la distancia euclídea por la *Distancia (muestral) de Fermat* investigada por Groisman et al. [2], e implementamos el clasificador resultante, Fermat KDC (FKDC")". Finalmente, evaluamos la exactitud («accuracy») de ambos clasificadores contra otros algoritmos estándares: SVC, regresión logística, kNN y Naive Bayes. Resultados preliminares muestran que tanto KDC como FKDC performan consistentemente como los mejores algoritmos en cada tarea, pero la performance de FKDC nunca supera la de su par euclídeo, técnicamente es un caso particular de FKDC. Concluimos con algunas hipótesis sobre el comportamiento observado.

II. Contexto

i. KDE en variedades de Riemann

Sea $(\mathcal{M}, g)$ resp. una variedad Riemanniana $\mathcal{M}$ y su métrica g, compacta y sin frontera de dimensión d, y denotemos d_g la distancia cpte. Sea X un elemento aleatorio (e.a.) con soporte en $\mathcal{M}$ y función de densidad f, y $\{X_1, ..., X_N\}$ una muestra de ee. aa. i.i.d. a X. Sean, además, K una «función núcleo» y $h > 0$ un «ancho de banda». Entonces, la estimación de f por KDE es

$$\hat{f}(x) = N^{-1} \sum_{i=1}^{N} \frac{1}{h^d} \frac{1}{\theta_{X_i}(x)} K\left(\frac{d_g(x, X_i)}{h}\right) \tag{1}$$

donde $\theta_p(q)$ es la *función de densidad volumétrica* en $\mathcal{M}$ alrededor de p. Obsérvese que cuando $\mathcal{M} = \mathbb{R}^d$ y g es la métrica euclídea, $\theta_p(q) = 1 \forall (p, q)$, y $\hat{f}$ se reduce a la más conocida

$$\hat{f}(x) = N^{-1} \sum_{i=1}^{N} \frac{1}{h^d} K\left(\frac{\|x - X_i\|}{h}\right) \tag{2}$$

El «núcleo gaussiano» $K(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}x^2\right)$ es casi universal; la elección de h es crítica y está ampliamente tratada en la literatura, no así la elección de la distancia d_g.

ii. Clasificación por KDE

Sean ahora $k \in \mathbb{N}$ «clases», y la muestra $\{(X_1, Y_1), ..., (X_N, Y_N)\}, Y_i \in \{1, ..., k\}$ de N elementos separados en k submuestas de tamaño $N_1, ..., N_k$, cada una soportada en su propia variedad (no ecesariamente la misma) con densidad $f_j, j \in \{1, ..., k\}$. Sea p_j la proporción poblacional de la clase j, aproximada por $\hat{p}_j = \frac{n_j}{N}$, y $\hat{f}_j$ el estimador por KDE de f_j ya descrito. Loubes & Pelletier [1], basándose en el criterio de Bayes, plantean como regla de clasificación para un nuevo (x, y)

$$\hat{y} = \arg\max_{j \in 1, ..., k} \widehat{f_j}(x) \hat{p}_j = \sum_{i=1}^{N} \mathbb{1}\{Y_i = j\} K_h(x, X_i) \tag{3}$$

donde $\mathbb{1}\{\cdot\}$ es la función indicadora, y $K_h(x, X_i) = \frac{1}{h^d} \frac{1}{\theta_{X_i}(x)} K\left(\frac{d_g(x, X_i)}{h}\right)$. Implementar la regla de Ecuación 3 requiere conocer la geometría de la(s) variedad(es) involucradas, que rara vez es factible. Una alternativa es *aprender la distancia de los datos.*

iii. Aprendizaje de Distancias: Isomap, Distancia de Fermat

Si los elementos muestrales $X_i \in \mathcal{M}$, y la variedad es «suficientemente regular», el segmento $\overline{X_i X_j}$ también pertenece a $\mathcal{M}$. Isomap (Tenenbaum et al [3]), pionero en esta tónica, plantea esencialmente aproximar la distancia en $\mathcal{M}$ por la geodésica en el grafo geométrico de k o ε vecinos más cercanos. En una propuesta tal vez superadora, Groisman et al [2] proponen la «Distancia de Fermat», una distancia propiamente dicha en $\mathcal{M}$, y muestran cómo ésta se puede aproximar «microscópicamente». Sea Q el grafo completo de la muestra, y $\alpha \geq 1$, luego

$$D_{Q,\alpha}(x, y) = \inf\left\{ \sum_{i=1}^{K} \|q_{i-1} - q_i\|^{\alpha} : (q_0, ..., q_K) \text{ es un camino de x a y} \right\} \tag{4}$$

es la «distancia muestral de Fermat. Nótese que usar el grafo completo obvia la necesidad de elegir (k/ε), mientras que $\alpha > 1$ «infla» el espacio y desalienta los «saltos largos» por «espacio vacío» fuera de $\mathcal{M}$. Cuando $\alpha = 1$, la distancia de Fermat se reduce a la distancia euclídea.

III. Propuesta y Metodología

title: [Clasificación por KDE con Distancia de Fermat en variedades desconocidas: una aproximación empírica.], En la tesis desarrollamos un clasificador compatible con el *framework* de `scikit-learn` según los lineamientos de [1] que apodamos `KDC`. Luego, implementamos el estimador de Ecuación 4, y combinándolo con KDC, obtenemos la titular «Clasificación por KDE con Distancia de Fermat», `FKDC`. Evaluamos la *exactitud* («accuracy») de los clasificadores propuestos en diferentes *datasets*, relativa a técnicas bien establecidas:

- regresión logística (`LR`)
- clasificador de soporte vectorial (`SVC`)
- k-vecinos-más-cercanos (`KN`)
- Naive Bayes Gaussiano (`GNB`)

El criterio de evaluación consiste en (1) partir la muestra en entrenamiento y testeo; (2) elegiremos *hiperparámetros óptimos* por *validación cruzada en 5 pliegos* entre los datos de entrenamiento, y (3) medir la exactitud de cada algoritmo algoritmos en conjunto de testeo de (1).
Para tener una idea «sistémica» de la performance de los algoritmos, evaluaremos su performance con *datasets* que varíen en el tamaño muestral N, la dimensión p de las X_i y el nro. de clases k.

IV. Análisiss

i. Fantasías en $\mathbb{R}^2$

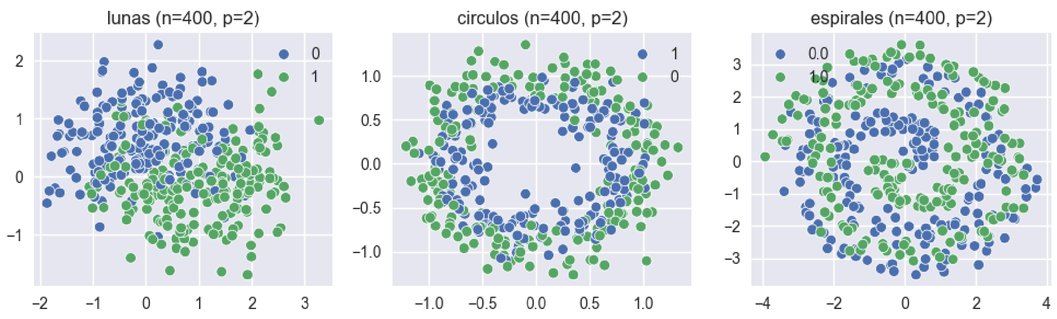

Figura 1: Datasets sintéticos en $\mathbb{R}^2$

Ruido	Dataset	FKDC	GNB	KDC	KN	LR	SVC
Alto	Circulos	67.2 (4.4)	63.5 (7.0)	67.0 (4.3)	67.3 (4.5)	44.8 (4.6)	71.3 (5.1)
	Espirales	76.2 (4.8)	48.5 (6.2)	76.6 (4.4)	76.0 (5.2)	48.6 (5.7)	78.7 (4.0)
	Lunas	79.7 (5.6)	80.4 (3.9)	81.3 (4.8)	80.9 (4.4)	80.7 (3.9)	81.2 (5.0)
Bajo	Circulos	78.4 (4.1)	67.7 (11.3)	78.5 (4.1)	79.1 (4.2)	45.0 (4.5)	81.2 (5.4)
	Espirales	90.0 (3.2)	49.6 (6.2)	90.4 (3.2)	90.3 (2.9)	49.5 (6.5)	92.9 (1.7)
	Lunas	88.0 (4.6)	83.6 (4.3)	88.1 (4.6)	87.8 (4.6)	83.9 (4.0)	88.0 (3.7)

Exactitud (en %), con sus respectivos desvíos estándares a lo largo de 16 repeticiones de cada experimento.

Comenzamos por 3 datasets, `lunas, circulos, espirales`, con $k = 2, p = 2, n = 400, n_1 = n_2 = 200$, que presentan variedades de dimensión intrínsica $d = 1$, a las cuales se les agrego «ruido» gaussiano con «bajo» y «alto» desvío estándar ($\sigma_{\text{alto}} \approx 1.5\sigma_{\text{bajo}}$). Las performances de SVC, KN, KDC y FKDC no son significativamente distintas, aunque SVC parece ligeramente superior. Es alentador ver que la performance de KDC es siempre competitiva, pero descorazonador ver que FKDC es sistemáticamente igual o ligeramente peor que KDC.

ii. `vino`, `pinguinos`, `iris` *y* `anteojos` *(*$k = 3$*)*

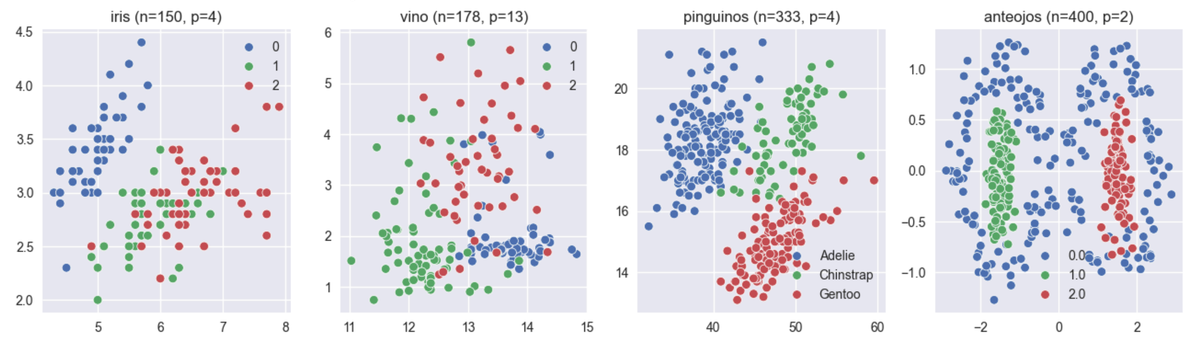

Figura 2: Datasets con $k = 3$. Salvo por `anteojos`, todos los datasets son pequeños pero reales.

Dataset	FKDC	GNB	KDC	KN	LR	SVC
Anteojos	97.5 (1.4)	97.0 (1.8)	97.4 (1.4)	97.7 (1.4)	50.5 (5.0)	97.7 (1.8)
Iris	94.4 (4.3)	94.6 (5.0)	94.0 (4.4)	95.4 (4.0)	97.5 (2.3)	94.2 (5.8)
Pinguinos	84.0 (4.2)	97.8 (1.6)	84.1 (4.2)	85.2 (3.8)	66.6 (4.5)	98.2 (1.0)
Vino	71.9 (7.1)	96.9 (2.2)	73.8 (6.3)	71.0 (6.5)	66.0 (6.7)	95.3 (2.6)

En los datasets de `anteojos` e `iris`, se observa el mismo fenómeno que en los datasets «2D»: (F)KDC es competitivo con los mejores métodos (SVC y LR, resp.), pero no superador. En los datasets de `pinguinos` y `vino`, la *performance* de los métodos propuestos es significativamente peor. En todos los casos, no conseguimos mejoras significativas sobre KDC con FKDC.

iii. digitos

Los ee.aa. son imágenes de 8x8 (*id est*, en $\mathbb{R}^{64}$) que representan dígitos manuscritos. Es de esperar que la variedad donde yacen los trazos sea de menor dimensión, para hacer buen uso de la «estimación de la variedad» que promete FKDC. Consideramos dos regímenes de evaluación: sobre el 80% de entrenamiento («escaso») y sobre el 20%(«denso»), para ver si FKDC destaca en alguno.

Eval.	FKDC	GNB	KDC	KN	LR	SVC
20%	98.8 (0.7)	92.1 (1.1)	98.9 (0.6)	98.9 (0.4)	96.7 (0.6)	99.0 (0.6)
80%	97.0 (0.4)	90.2 (0.6)	96.9 (0.5)	96.6 (0.8)	94.5 (0.7)	97.5 (0.4)

Una vez más, FKDC no se distingue de KDC, y a su vez ambos andan tan bien pero no mejor que KN y SVC. La diferencia entre ambos regímenes de evaluación es leve: pareciera que con sólo el 20% de los datos de entrenamiento, la muestra ya es suficientemente «densa».

V. FKDC versus KDC

El hecho de que la performance de FKDC sea casi idéntica a la de su primo euclídeo, se encuentra parcialemente por el hecho de que en la mayoría de los casos $\alpha_{\text{opt}} \approx 1$ (y el h_{opt} de ambos métodos es similar, lo que indica la coherencia interna de FKDC), como se observa en la tabla a derecha con hiperparámetros óptimos para una semilla al azar en c/ experimento. Lo que no queda claro aún, es por qué la performance de FKDC tampoco mejora cuando $\alpha_{\text{opt}} > 1$.

Dataset	KDC	FKDC	
	h	h	α
Circulos (alto)	0.13	0.06	2.12
Espirales (alto)	0.03	0.16	2.12
Lunas (alto)	0.33	0.29	1.0
Circulos (bajo)	0.09	0.03	1.94
Espirales (bajo)	0.01	0.01	1.38
Lunas (bajo)	0.48	0.4	1.19
Anteojos	0.01	0.01	1.0
Iris	0.04	0.03	1.0
Pinguinos	57.54	73.56	1.0
Vino	33.11	29.29	1.0
Digitos	10.96	15.85	1.19

Cuando observamos la «accuracy» com función de h para distintos α en la etapa de testeo, pareciera ser que hay un «techo» a la performance, y aún cuando para cierto h exista un $\alpha_{\text{opt}} > 1$, lo cierto es que para *cualquier* α, existe un $h_{\text{opt}} = f(\alpha \mid \text{dataset})$ de performance equivalente.

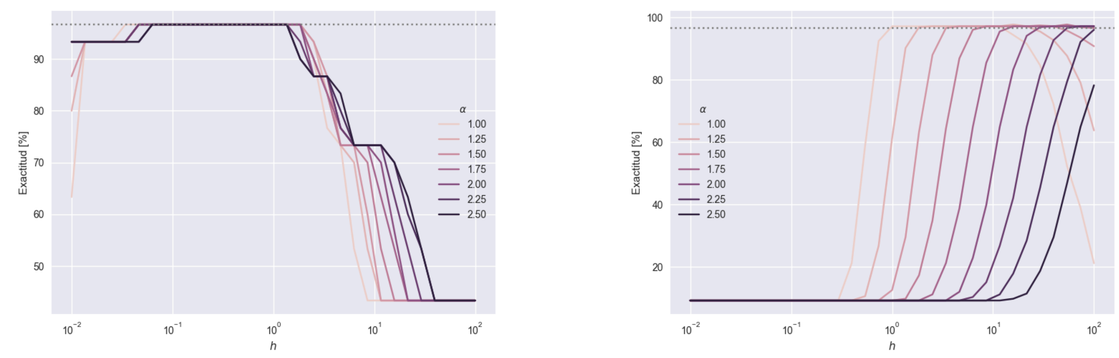

Figura 3: Exactitud en validación para `iris` (izq.) y `digitos` (der.)

VI. Conclusiones y Trabajo a Futuro

La sensación es «agridulce»: el algoritmo de clasificación por KDE resulta competitivo con métodos bien establecidos, pero no encontramos aún mejoras marginales por el uso de la distancia muestal de Fermat. ¿Por qué? Tal vez en los datasets considerados la variedad subyacente no difiera mucho del espacio euclídeo ambiente. Esta hipótesis es problemática en tanto las `lunas, circulos y espirales` son claramente unidimensionales en $\mathbb{R}^2$, y cuesta pensar en `digitos` como elementos de $\mathbb{R}^{64}$.
Otra alternativa - no la única - es que cuando $\mathcal{M}$ dufiere de su espacio ambiente, $\theta_p(q)$ (la *función de densidad volumétrica* en $\mathcal{M}$ alrededor de p) sea sumamente variable en el espacio, e ignorarla nos haga pesar incorrectamente las observaciones. Al autor del trabajo no le resulta familiar la geometría riemanniana, lo cual dificulta la corroboración de dicha hipótesis.

Bibliografía

[1] J.-M. Loubes y B. Pelletier, «A Kernel-Based Classifier on a Riemannian Manifold», *Statistics & Decisions*, vol. 26, n.º 1, pp. 35-51, mar. 2008, doi: 10.1524/stnd.2008.0911.

[2] P. Groisman, M. Jonckheere, y F. Sapienza, «Nonhomogeneous Euclidean First-Passage Percolation and Distance Learning», n.º arXiv:1810.09398. arXiv, diciembre de 2019.

[3] J. B. Tenenbaum, V. de Silva, y J. C. Langford, «A Global Geometric Framework for Nonlinear Dimensionality Reduction», *Science*, vol. 290, n.º 5500, pp. 2319-2323, dic. 2000, doi: 10.1126/science.290.5500.2319.