Tarea 2 Reconocimiento de Patrones

Fecha de entrega: domingo 25 de feb, 22PM

Comentarios generales:

1. Si quieren hacer PCA en Python, una opción alternativa es `https://erdogant.github.io/pca/pages/html/index.html`
   No es libreria oficial; hay que instalarlo a mano con `pip install pca`

2. Como se comentó en clase, un método que hoy es bastante popular y que surgió junto con T-SNE es UMAP. Contrario a T-SNE tiene una teoria matemática extensa detrás pero es fuera del alcance de la clase (usa muchos conceptos de topologia). No es dificil entender la idea general. Ver por ejemplo: `https://pair-code.github.io/understanding-umap/`
   Tiene dos parámetros: uno asociado al espacio original y otro al espacion nuevo. Los invita leer y jugar con esta página.

Ejercicios:

1. Mencionamos en clase la divergencia de Kullback Leibler $d_{KL}(P, Q)$ entre dos distribuciones:

   Caso discreto: $\sum_k P_k \log P_k/Q_k$
   Caso continuo: $\int_x f_P(x) \log f_P(x)/f_Q(x)dx$ con $f()$ la densidad.

   Calcula $d_{KL}(P, Q)$ para el caso discreto con $P \sim Bern(\theta_1)$ y $Q \sim Bern(\theta_2)$. Muestra con una gráfica como cambia $d_{KL}(P, Q)$ si θ_2 se alega de θ_1.

   Misma pregunta para el caso continuo con $P \sim \mathcal{N}(\mu_1, \sigma^2)$ y $Q \sim \mathcal{N}(\mu_2, 1)$. Haz uso del hecho que $\int_x f_P(x) \log f_P(x)/f_Q(x)dx = E_P \log f_P(X) - E_P \log f_Q(X)$.

   Se puede motivar la forma de la divergencia de muchas maneras. Además de lo que se comentó en clase, un camino alternativo es tomar como punto de partida la distancia de $Pearson$ χ^2 que es muy intuitvo:

$$\chi^2(P,Q) := \sum_k \frac{(P_k - Q_k)^2}{Q_k}$$

Verifica que lo anterior es igual a:

$$\sum_k P_k(\frac{P_k}{Q_k} - 1).$$

Se puede generalizarlo introduciendo un parámetro λ:

$$\frac{2}{\lambda(\lambda+1)} \sum_k P_k((\frac{P_k}{Q_k})^\lambda - 1).$$

Se puede demostrar que si $\lambda \to 0$ eso converge a $d_{KL}(P,Q)$ (poner $\lambda = 0$ conduce a $0/0$, así se debe hacer unos pasos más).

Observa la siguiente conexión con el estimador de máxima verosimilitud: si Q es una distribución discreta P_θ con θ un parámetro por estimar y $\hat{P}$ es la distribución empirica de una muestra $\{x_i\}$, entonces buscar θ que maximiza la verosimilitud de la muestra bajo P_θ es equivalente a buscar θ que minimiza la distancia de Kullback-Leibler entre P_θ y la empírica de la muestra $\hat{P}$.
Verifica eso (hint: la distribucion empirica de una muestra se define como $\hat{P}_k = n(k)/n$ con $n(k)$ el número de veces que k ocurre en la muestra $\{x_i\}$; la función de log verosimilitud se puede escribir como $\sum_k log(P_\theta)_k^{n(k)}$).

Finalmente, solamente como comentario para aquellos familiarizados con la entropia (los del posgrado): no es dificil mostrar que la divergencia entre una distribución bivariada P y el producto de sus marginales P^1P^1 (lo que se espera baja independencia), $d_{KL}(P, P^1P^2)$, es igual a la información mutua entre P y P^1P^2, asi se puede usar también como una medida de (in)dependencia.

2. Vimos que en *Local linear embedding* se resuelvan dos problemas;
El primer paso es un problema de regresión con restricciones para encontrar $\{w_{i,j}\}$ que minimiza:

$$\sum_i ||x_i - \sum_{j \in vec(i)} w_{i,j}x_j||^2 \text{ con } \sum_j w_{i,j} = 1$$

y después, dadas $\{w_{i,j}\}$, se buscan las $\{x_i^*\}$ que minimizan:

$$\sum_i ||x_i^* - \sum_{j \in vec(i)} w_{i,j} x_j^*||^2$$

Verifica que $\sum_j w_{i,j} = 1$ garantiza que la solución no cambia al hacer una translación de los datos originales $\{x_i\}$, es decir, $\{x_i + a\}$ para algun vector a tiene la misma solución que $\{x_i\}$.

Nos enfocamos ahora al segundo paso. Para simplificarlo, vamos a suponer que $x_i^* \in \mathcal{R}$, asi se convierte en:

$$\sum_i (x_i^* - \sum_{j \in vec(i)} w_{i,j} x_j^*)^2$$

Verifica que lo anterior se puede escribir como

$$(X^*)^t X^* - (X^*)^t (W X^*) - (W X^*)^t (X^*) + (W X^*)^t (W X^*)$$

donde W es la matriz $[w_{i,j}]$ donde $w_{i,j} = 0$ si $j \notin vec(i)$, y X^* el vector $[x_i^*]$. Verifica que se puede escribir lo anterior como

$$(X^*)^t M (X^*) \text{ con } M = (I - W)^t (I - W) \text{ y } I \text{ la matriz idéntica} \tag{1}$$

Minimizar la forma cuadrática (1) con la restricción $||X^*||^2 = 1$, conduce a un cociente de Rayleigh como lo vimos con PCA.

Verifica que el vector con unos 1 es un vector propio con valor propio 0 de M (hint: calcula $(I - W)1$).

Como estamos minimizando (y no maximizando como pasa en PCA) se puede mostrar que la solución es el vector propio con valor propio más chiquito. Eso es 0 y es claramente no util. Por eso uno se queda con el segundo vector propio más chico de M.

3. (no entregar) Vimos el Teorema de Rao en clase:
Si $\mathbb{F}$ es una matriz simétrica de rango d y con SVD:

$$\mathbb{F} = \sum_1^d \lambda_i v_i v_i^t$$

La matriz simétrica de rango $p < d$ que minimiza $||\mathbb{F} - \mathbb{G}||_F$ es:

$$\mathbb{G} = \sum_{1}^{p} \lambda_i v_i v_i^t$$

Muestra que para esta elección, el error $||\mathbb{F} - \mathbb{G}||_F^2$ es igual a $\sum_{i=p+1}^{d} \lambda_i^2$. (hint: usa las propiedades de v_i y recuerda que $||\mathbb{A}||_F^2 = traza(\mathbb{A}^t\mathbb{A})$).

4. La base de datos Animales con Atributos (Animals with Attributes) contiene información sobre 50 animales. Para cada uno, se tienen 85 características de valor real que capturan varias propiedades del animal: dónde vive, qué come, etc.

   Usa ISOMAP, LLE, T-SNE y SOMs para encontrar visualizaciones informativas de los datos y encontrar grupos.

   Se usan 3 archivos:
   `classes.txt` los nombres de cada animal
   `predicates.txt` los nombres de las caracteristicas (columnas)
   `predicate-matrix-continuous.txt` la matriz de datos

5. (después de la clase de miércoles)
   Toma de la base `https://faces.mpdl.mpg.de/imeji/` las caras de una mismas persona. Implementa KernelPCA con kernel lineal para aproximar las caras con matrices de menor rango. Visualizalos. Solamente se puede usar una función que calcula la SVD, no las funciones de (kernel)PCA. No olvides de centrar los datos.