Ciencia de Datos
Tarea 4

Para entregar el 4 de mayo de 2020

1. Para datos de clasificación binaria $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, considera la siguiente función de costo:

$$\mathcal{L} = \sum_i (\theta(y_i) - \beta' \mathbf{x}_i - \beta_0)^2 \tag{1}$$

Definimos n_+, n_- el número de observaciones con $y_i = 1$ y $y_i = -1$, respectívamente $\mathbf{c}_+, \mathbf{c}_-$ el centroide de las observaciones con $y_i = 1$, y $y_i = -1$ y $\mathbf{c}$ el centroide de todos los datos.

Como en clase, construimos las matrices:

$$\begin{aligned} \mathbf{S}_B &= (\mathbf{c}_+ - \mathbf{c}_-)(\mathbf{c}_+ - \mathbf{c}_-)' \\ \mathbf{S}_W &= \textstyle\sum_{i:y_i=1}(\mathbf{x}_i - \mathbf{c}_+)(\mathbf{x}_i - \mathbf{c}_+)' + \sum_{i:y_i=-1}(\mathbf{x}_i - \mathbf{c}_-)(\mathbf{x}_i - \mathbf{c}_-)' \end{aligned}$$

a) Verifica que

$$\mathbf{S}_W = \sum_{i:y_i=1} \mathbf{x}_i \mathbf{x}_i' + \sum_{i:y_i=-1} \mathbf{x}_i \mathbf{x}_i' - n_+ \mathbf{c}_+ \mathbf{c}_+' - n_- \mathbf{c}_- \mathbf{c}_-'$$

b) Verifica que el vector $\mathbf{S}_B \boldsymbol{\beta}$, es un múltiplo del vector $(\mathbf{c}_+ - \mathbf{c}_-)$.

c) Si definimos $\theta(1) = n/n_+$ y $\theta(-1) = -n/n_-$, verifica que en el mínimo de (1):

$$\beta_0 = -\boldsymbol{\beta}' \mathbf{c},$$

$$(\mathbf{S}_W + \frac{n_+ n_-}{n} \mathbf{S}_B)\boldsymbol{\beta} = n(\mathbf{c}_+ - \mathbf{c}_-) \tag{2}$$

d) Usando el resultado de inciso b, argumenta que (2) implica que en el mínimo:

$$\boldsymbol{\beta} \sim S_W^{-1}(\mathbf{c}_+ - \mathbf{c}_-),$$

es decir la solución coincide con la del Fisher Discriminant Analysis (FDA).

e) Lo anterior permite implementar FDA usando algún algoritmo de mínimos cuadrados, por ejemplo, el del módulo `sklearn.linear_model`. Ilustra cómo funciona el método con algunos conjuntos de datos en 2D bien elegidos.

f) Observamos que (1) muestra que FDA **no** es muy robusto a datos atípicos.
Una forma de hacerlo más robusto es usar mínimos cuadrados ponderados, con ciertos pesos $w_i, i = 1, \ldots, n$ para minimizar:

$$\sum_i w_i (\theta(y_i) - \boldsymbol{\beta}^t \mathbf{x}_i - \beta_0)^2.$$

¿Cómo elegirías estos pesos? Verifica tu propuesta con algunos ejemplos en 2D.

2. Considera los datos `MNIST` de dígitos escritos a mano que usamos anteriormente de 28×28 pixeles. En este ejercicio implementarás métodos de clasificación para los $k \in K = \{0, 1, \ldots, 9\}$ dígitos.

   a) Implementa el *baseline* que usaremos. Este será un método de regresión multivariada, es decir

   $$\mathbf{Y} = \mathbf{X}\widehat{\mathbf{B}},$$

   donde $\mathbf{Y}_{n\times|K|}$ es una matriz indicadora, donde cada renglón tiene ceros excepto en el lugar que corresponde al valor y_k, donde colocamos un 1. Por ejemplo, si alguna imagen corresponde al dígito "3", el renglón correspondiente en $\mathbf{Y}$ será $(0, 0, 0, 1, 0, 0, 0, 0, 0, 0)$.

   $\mathbf{X}_{n\times 784}$ es la matriz de características y $\widehat{\mathbf{B}}$ es la matriz cuyas columnas contienen los $|K|$ coeficientes correspondientes $\widehat{\boldsymbol{\beta}}_k$.

   Con esta formulación, asumimos un modelo lineal para cada respuesta $\mathbf{y}_k$:

   $$\widehat{\mathbf{y}}_k = \mathbf{X}\widehat{\boldsymbol{\beta}}_k,$$

   y la clasificación para alguna observación $\mathbf{x}$ se obtiene mediante

   $$\widehat{C}(\mathbf{x}) = \arg\max_{k\in K} \hat{y}_k.$$

   Utiliza las tuplas `(x_train, y_train), (x_test, y_test)` que usamos en clase para ajustar y probar el modelo, respectívamente. Puedes restringir el número de observaciones de cada conjunto, pero procura que el conjunto de entrenamiento sea más grande que el de prueba. Obten el error de entrenamiento y prueba.

   b) Utiliza `LDA` y `QDA`. Verifica si puedes superar al baseline. ¿Crees que ayudaría tener otra representación de los dígitos? Explica tu respuesta e impleméntala.

   c) **Opcional (puntos extra)**. Programa una aplicación interactiva donde dibujes un número y te diga qué dígito es usando los clasificadores del inciso anterior. Puedes usar y modificar el applet que usamos en el curso.

3. Este ejercicio es sobre análisis de textos y de sentimientos.

   Considera los datos que se encuentran en `spanish_reviews.zip`, que corresponden a opiniones de usuarios en los siguientes productos: automóviles, hoteles, lavadoras, libros, teléfonos celulares, música, computadoras y películas [1].

   Contiene 3 carpetas. `all_files` tiene todas las 400 opiniones, `train` y `test` corresponde a un conjunto de datos de entrenamiento (80 %) y prueba (20 %) que escogí aleatoriamente. Para el sentimiento tenemos dos categorias: `yes` y `no`, que indican las opiniones positivas y negativas, respectívamente. Incluí también el archivo `reviews_text_caract.csv` que indica el nombre de archivo para cada texto junto con su categoría y sentimiento.

   a) Obtén una representación vectorial de los textos mediante Bag Of Words (BOW).

[1] Julian Brooke and Maite Taboada. `https://www.sfu.ca/~mtaboada/SFU_Review_Corpus.html`.

Realiza el preprocesamiento que creas conveniente. Puedes usar el código que vimos en clase, el cual puedes aumentar/modificar si es necesario. Elije el tamaño de vocabulario que creas conveniente y justifica.

b) Obtén una matriz de similaridades de los textos usando tu representación BOW y la distancia del coseno ¿Puedes identificar algún patrón en los textos?

c) Aplica algún (o algunos) método de clustering apropiado. ¿Puedes identificar las categorías de los textos (ya sean productos o sentimiento)?

d) Implementa un método de *recuperación de información* para éste corpus de tal forma que, dado un texto de consulta (*query*), te devuelva los 5 documentos del corpus más relacionados (ordenados). Prueba por ejemplo con éste query: *"Los saltos con breves reflexiones, de lectura ágil y sin problemas de comprensión de una historia que se cuenta hacia atrás y en peequeñas dosis. Al libro le sobran páginas, divagaciones que no aportan demasiado, sin embargo me gustó bastante"*. ¿Cómo funciona? Prueba con varias queries.

e) Basado en BOW, y una representación adecuada de los textos, utiliza LDA y QDA para clasificar las opiniones positivas y negativas. Usa los datos train y test para ajustar y verificar los resultados de tu clasificador, respectívamente. ¿Cuál funciona mejor? Reporta el error de entrenamiento y de prueba.