Ciencia de Datos
Tarea 4

Para entregar el 1 de junio de 2018

1. Aunque hay varias extensiones de AdaBoost al caso multiclase, una de las mas usadas es la llamada SAMME (Stagewise Additive Modeling using a Multi-class Exponential loss function), ya que está basada en la caracterización estadística de Friedman et al. Implementa esta versión de AdaBoost y verifica su desempeño en un conjunto de datos con más de dos categorias.

   Incluye una breve descripción del método basandote en el artículo: Zhu J, Zou H, Rosset S, and Hastie T (2009). *Multi-Class AdaBoost.* Statistics and Its Interface,2, 349360. Puedes usar también los datos que ahí se muestran para reproducir los resultados.

2. Usando los datos de los dígitos escritos a mano y digitalizados, complementa el ejercicio que hiciste en la tarea 1 aplicando métodos de clasificación basados en

   - LDA
   - QDA
   - Redes neuronales
   - Máquinas de Soporte Vectorial
   - Arboles de clasificación
   - AdaBoost

   Utiliza $K-$Fold CV como criterio para elegir el mejor modelo, así como para compararlos. ¿Qué método elegirías?

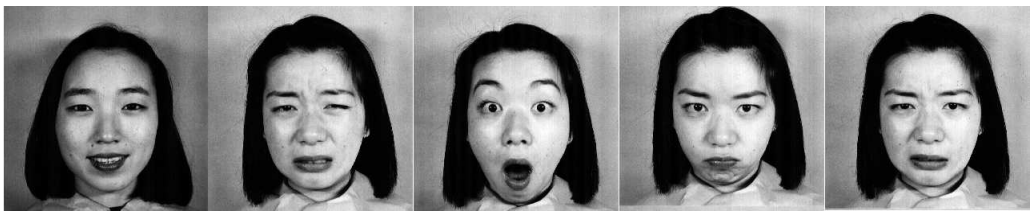
   Especifica los parámetros que usaste en cada método de clasificación. Incluye gráficos *informativos* sobre el desempeño de cada método. Actualiza tu aplicación interactiva, si es que la implementaste en la primera tarea.

3. Repite el ejercicio 2 para los datos de frutas que usaste en la tarea 2. Utiliza la representación en el espacio HSV con la mediana y los cuartiles centrales.

   Puntos extra: verifica el desempeño en del clasificador que elegiste en ejemplos "reales". Toma algunas fotos de frutas y realiza un preprocesamiento básico para clasificarlas. Puedes usar el código en `C` (cortesía de Karen) para quitar el fondo de tu foto. Leé las instrucciones que vienen documentadas.

   ¿Cómo funciona tu clasificador? ¿Qué tan sensible es a las condiciones de la imagen (tamaño, rotación, etc.)?

4. Considera los datos contenidos en `img_expression.zip`, que corresponden a fotos (256×256 pixeles) de mujeres japonesas con diferentes tipos de expresión.



El archivo `class_img_exp.dat` contiene las etiquetas para cada imagen. En este caso, tenemos dos tipos de etiquetas:

- `file.expression`: corresponde básicamente a la expresión que *se le pidió* hacer a la persona. La etiqueta NEU es un rostro inexpresivo. Estas etiquetas son:

  `HAP (hapiness), SAD (sadness), SUR (surprise) ANG (anger),`
  `DIS (disgust) y NEU (neutral)`

- `semantic.expression`: corresponde a una calificación semántica asignada de acuerdo a un experimento psicológico donde se le pidió a varias personas clasificar cada imagen. En este caso, la clase `neutral` desaparece, ya que se asignó a alguna de las otras etiquetas. La etiquetación se realizó según la calificación máxima.

Para mas detalles, puedes consultar el archivo `README`, que describe los datos originales.

a) Repite el ejercicio 1 para estos datos usando Eigenfaces y las etiquetas `semantic.expression`. Especifica además cuántos componentes principales usaste y el critero que adoptaste.

b) Ahora hazlo considerando las etiquetas `file.expression`. ¿Qué diferencias notas en el desempeño?

c) Prueba el clasificador que elegiste en imágenes tuyas para estimar tu expresión. Prueba con distintos tipos de fondo, luminosidad y posición para verificar qué tan sensible es a las características del entorno.

¿Qué recomendarías para mejorar el clasificador?

5. **Puntos extra: Competencia IMDB.**

Considera los datos `movie_metadata.csv`, que contiene datos extraídos del sitio `www.imdb.com` de poco mas de 5000 películas. El objetivo es analizar la base de datos y construir métodos de predicción (regresión, clasificación) para dos variables de interés: ganancias (`gross`) y calificación (`imdb_score`) de las películas.

a) Utiliza un (unos) método (métodos) para estimar las ganancias basado en las características de las películas que consideres *convenientes*. Indica cuál es el criterio que usaste para decidir qué variables usar.

b) Define la variable ordinal `calificacion` basandote en

`calificacion`	`imdb_score`
Excelente	≥ 8
Buena	$[7, 8)$
Regular	$[5, 7)$
Mala	< 5

Utiliza uno o algunos métodos de clasificación para estimar la `calificacion` de las películas.

Implementa **PRank**, el método de clasificación (ranking) propuesto por Cramer y Singer (el paper es Pranking with Ranking, y se encuentra en la página del curso) y compáralo con el o los métodos que usaste antes. ¿Qué diferencia notas? (la implementación de PRank es bastante sencilla).

Observaciones: Los datos tienen varios detalles. Antes todo, revisa las notas del sitio `https://www.kaggle.com/deepmatrix/imdb-5000-movie-dataset` y `https://blog.nycdatascience.com/student-works/machine-learning/movie-rating-prediction/`.

Hay algunos datos faltantes que pueden confundirse con 0. Leé bien las observaciones del sitio web de los datos y decide cómo tratarás estos datos.

Los datos tienen variables de diferente tipo, decide también en qué forma las analizas y las incluyes en los modelos que uses.

Hay un premio especial al que tenga el menor error de predicción. Se requieren mínimo 2 participantes...