PROMiDAT Iberoamericano
Machine Learning con Python III
Métodos Supervisados

- Las tareas tienen fecha de entrega una semana después a la clase y deben ser entregadas antes del inicio de la clase siguiente.
- Cada día de atraso en implicará una pérdida de 10 puntos.
- Las tareas son estrictamente de carácter individual, tareas iguales se les asignará cero puntos.
- En nombre del archivo debe tener el siguiente formato: `Tarea1_nombre_apellido.pdf`. Por ejemplo, si el nombre del estudiante es Luis Pérez: `Tarea1_luis_perez.pdf`. Para la tarea número 2 sería: `Tarea2_luis_perez.pdf`, y así sucesivamente.
- Esta tarea tiene un valor de un 25 % respecto a la nota total del curso.

Tarea Número 2

- **Pregunta 1:** [25 puntos] [**no usar** `Python`] Considere los datos de entrenamiento que se muestran en la siguiente Tabla para un problema de clasificación binaria.

ID Cliente	Género	Tipo-Carro	Talla	Clase
1	M	Familiar	Pequeño	C0
2	M	Deportivo	Mediano	C0
3	M	Deportivo	Mediano	C0
4	M	Deportivo	Grande	C0
5	M	Deportivo	Extra Grande	C0
6	M	Deportivo	Extra Grande	C0
7	F	Deportivo	Pequeño	C0
8	F	Deportivo	Pequeño	C0
9	F	Deportivo	Mediano	C0
10	F	Lujo	Grande	C0
11	M	Familiar	Grande	C1
12	M	Familiar	Extra Grande	C1
13	M	Familiar	Mediano	C1
14	M	Lujo	Extra Grande	C1
15	F	Lujo	Pequeño	C1
16	F	Lujo	Pequeño	C1
17	F	Lujo	Mediano	C1
18	F	Lujo	Mediano	C1
19	F	Lujo	Mediano	C1
20	F	Lujo	Grande	C1

1. Calcule el índice de Gini para la tabla completa, observe que el 50 % de las filas son de la clase C0 y el 50 % son de la clase C1.
2. Calcule el índice de Gini Split para la variable Género.
3. Calcule el índice de Gini Split para la variable Tipo-Carro.

4. Calcule el índice de Gini Split para la variable Talla.
5. ¿Cuál variable es mejor Género, Tipo-Carro o Talla?

- **Pregunta 2:** [25 puntos] En este ejercicio usaremos los datos (`voces.csv`). Se trata de un problema de reconocimiento de género mediante el análisis de la voz y el habla. Esta base de datos fue creada para identificar una voz como masculina o femenina, basándose en las propiedades acústicas de la voz y el habla. El conjunto de datos consta de 3.168 muestras de voz grabadas, recogidas de hablantes masculinos y femeninos.

  El conjunto de datos tiene las siguientes propiedades acústicas (variables) de cada voz:

  - `meanfreq`: frecuencia media (en kHz).
  - `sd`: desviación estándar de frecuencia.
  - `median`: frecuencia mediana (en kHz).
  - `Q25`: primer cuantil (en kHz).
  - `Q75`: tercer cuantil (en kHz).
  - `IQR`: rango intercuantile (en kHz).
  - `skew`: sesgo (ver nota en la descripción de specprop).
  - `kurt`: kurtosis (ver nota en la descripción de specprop).
  - `sp.ent`: entropía espectral.
  - `sfm`: planitud espectral.
  - `mode`: modo frecuencia.
  - `centroide`: centroide de frecuencia (ver specprop).
  - `peakf`: frecuencia de pico (frecuencia con mayor energía).
  - `meanfun`: promedio de la frecuencia fundamental medida a través de la señal acústica.
  - `minfun`: frecuencia mínima fundamental medida a través de la señal acústica.
  - `maxfun`: máxima frecuencia fundamental medida a través de la señal acústica.
  - `meandom`: promedio de la frecuencia dominante medida a través de la señal acústica.
  - `mindom`: mínimo de la frecuencia dominante medida a través de la señal acústica.
  - `maxdom`: máximo de la frecuencia dominante medida a través de la señal acústica.
  - `dfrange`: rango de frecuencia dominante medido a través de la señal acústica.
  - `modindx`: índice de modulación. Calculado como la diferencia absoluta acumulada entre las mediciones adyacentes de las frecuencias fundamentales dividida por la gama de frecuencias.
  - `género`: Masculino o Femenino (variable a predecir).

  Realice lo siguiente:

  1. Cargue la tabla de datos `voces.csv` en **Python**.

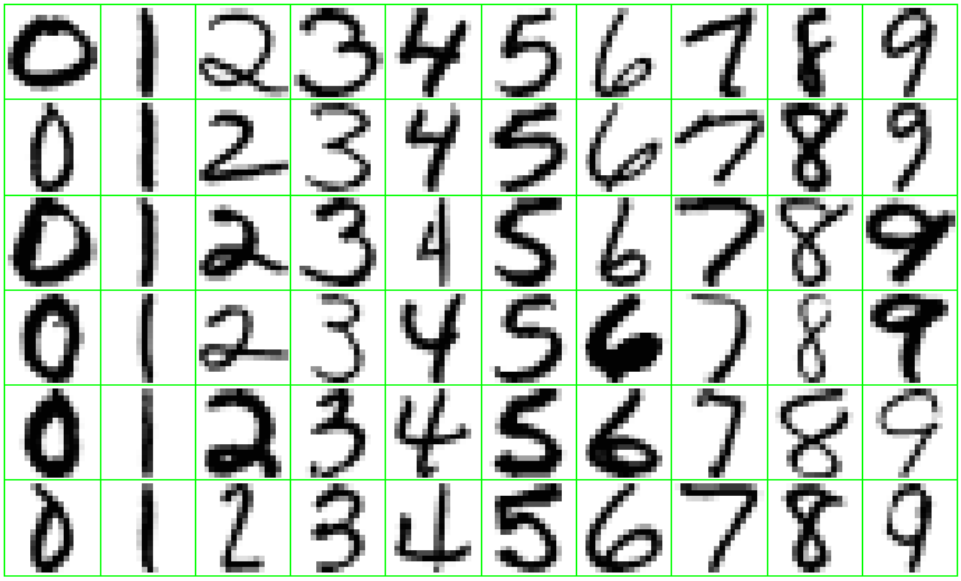
2. Use Árboles de Decisión en **Python** (con los parámetros por defecto) para generar un modelo predictivo para la tabla `voces.csv` usando el 80 % de los datos para la tabla aprendizaje y un 20 % para la tabla testing, luego calcule para los datos de testing la matriz de confusión, la precisión global y la precisión para cada una de las dos categorías. ¿Son buenos los resultados? Explique.
3. Usando la función programada en el ejercicio 1 de la tarea anterior, los datos `voces.csv` y los modelos generados arriba construya un `DataFrame` de manera que en cada una de las filas aparezca un modelo predictivo y en las columnas aparezcan los índices *Precisión Global, Error Global Precisión Positiva (PP), Precisión Negativa (PN), Falsos Positivos (FP), los Falsos Negativos (FN), la Asertividad Positiva (AP) y la Asertividad Negativa (AN)*. ¿Cuál de los modelos es mejor para estos datos?
4. Grafique el árbol generado e interprete al menos dos reglas que se puedan extraer del mismo. Si es necesario pode el árbol para que las reglas sean legibles.
5. Repita los ejercicios 1-4, pero esta vez use 2 combinaciones diferentes de los parámetros del método `DecisionTreeClassifier`. ¿Mejora la predicción?.
6. Repita los ejercicios 1-4, pero esta vez use 2 combinaciones diferentes de selección de 6 variables predictoras. ¿Mejora la predicción?.

- **Ejercicio 3:** [25 puntos] Esta pregunta utiliza los datos (`tumores.csv`). Se trata de un conjunto de datos de características del tumor cerebral que incluye cinco variables de primer orden y ocho de textura y cuatro parámetros de evaluación de la calidad con el nivel objetivo. La variables son: Media, Varianza, Desviación estándar, Asimetría, Kurtosis, Contraste, Energía, ASM (segundo momento angular), Entropía, Homogeneidad, Disimilitud, Correlación, Grosor, PSNR (Pico de la relación señal-ruido), SSIM (Índice de Similitud Estructurada), MSE (Mean Square Error), DC (Coeficiente de Dados) y la variable a predecir `tipo` (1 = Tumor, 0 = No-Tumor).

  Realice lo siguiente:

  1. Use el método de Árboles de Decisión en **Python** para generar un modelo predictivo para la tabla `tumores.csv` usando el 70 % de los datos para la tabla aprendizaje y un 30 % para la tabla testing.
  2. Usando la función programada en el ejercicio 1 de la tarea anterior, los datos `tumores.csv` y los modelos generados arriba construya un `DataFrame` de manera que en cada una de las filas aparezca un modelo predictivo y en las columnas aparezcan los índices *Precisión Global, Error Global Precisión Positiva (PP), Precisión Negativa (PN), Falsos Positivos (FP), los Falsos Negativos (FN), la Asertividad Positiva (AP) y la Asertividad Negativa (AN)*. ¿Cuál de los modelos es mejor para estos datos?
  3. Grafique el árbol generado e interprete al menos dos reglas que se puedan extraer del mismo. Si es necesario pode el árbol para que las reglas sean legibles.

- **Pregunta 4:** [25 puntos] En este ejercicio vamos a predecir números escritos a mano (Hand Written Digit Recognition), la tabla de aprendizaje está en el archivo `ZipDataTrainCod.csv` y la tabla de testing está en el archivo `ZipDataTestCod.csv`. En la figura siguiente se ilustran los datos:



Los datos de este ejemplo vienen de los códigos postales escritos a mano en sobres del correo postal de EE.UU. Las imágenes son de 16×16 en escala de grises, cada pixel va de intensidad de -1 a 1 (de blanco a negro). Las imágenes se han normalizado para tener aproximadamente el mismo tamaño y orientación. La tarea consiste en predecir, a partir de la matriz de 16×16 de intensidades de cada pixel, la identidad de cada imagen $(0, 1, \ldots, 9)$ de forma rápida y precisa. Si es lo suficientemente precisa, el algoritmo resultante se utiliza como parte de un procedimiento de selección automática para sobres. Este es un problema de clasificación para el cual la tasa de error debe mantenerse muy baja para evitar la mala dirección de correo. La columna 1 tiene la variable a predecir `Número` codificada como sigue: 0=‘cero’; 1=‘uno’; 2=‘dos’; 3=‘tres’; 4=‘cuatro’; 5=‘cinco’;6=‘seis’; 7=‘siete’; 8=‘ocho’ y 9=‘nueve’, las demás columnas son las variables predictivas, además cada fila de la tabla representa un bloque 16×16 por lo que la matriz tiene 256 variables predictoras.

1. Usando Árboles de Decisión más cercanos un modelo predictivo para la tabla de aprendizaje.
2. Con la tabla de testing calcule la matriz de confusión, la precisión global, el error global y la precisión en cada unos de los dígitos. ¿Son buenos los resultados? Además compare respecto a los resultados obtenidos en la tarea anterior.
3. Repita los ejercicios 1, 2 y 3 pero usando solamente los 3s, 5s y los 8s. ¿Mejora la predicción?
4. **Optativo:** [10 puntos] Repita los ejercicios 1, 2 y 3 pero reemplazando cada bloque 4×4 de píxeles por su promedio. ¿Mejora la predicción? Recuerde que cada bloque 16×16 está representado por una fila en las matrices de aprendizaje y testing. **Despliegue la matriz de confusión resultante**.
5. **Optativo:** [10 puntos] Repita los ejercicios 1, 2 y 3 pero reemplazando cada bloque $p \times p$ de píxeles por su promedio. ¿Mejora la predicción? (pruebe con algunos valores de p). **Despliegue las matrices de confusión resultantes**.

PRØMiDAT

IBEROAMERICANO

Programa Iberoamericano de
Formación en Minería de Datos