Profesor: Dr. Oldemar Rodríguez Rojas
PF-1319 y PF-1320 Análisis de Datos II
Fecha de Entrega: Domingo 2 de octubre a las 12 media noche
Instrucciones:

- La solución a cada tarea se debe subir en el aula virtual, no pueden ser enviadas por correo.
- Las tareas son estrictamente individuales.
- Tareas idénticas se les asignará cero puntos.
- Todas las tareas tienen el mismo valor en la nota final del curso.
- Las tareas se pueden entregar tarde, pero cada día de atraso tendrá un rebajo de 20 puntos.

Tarea Número 6

- **Ejercicio 1:** [25 puntos] En este ejercicio vamos a usar la tabla de datos `raisin.csv`, que contiene es resultado de un sistema de visión artificial para distinguir entre dos variedades diferentes de pasas (Kecimen y Besni) cultivadas en Turquía. Estas imágenes se sometieron a varios pasos de preprocesamiento y se realizaron 7 operaciones de extracción de características morfológicas utilizando técnicas de procesamiento de imágenes.

  El conjunto de datos tiene 900 filas y 8 columnas las cuales se explican a continuación.

  - `Area`: El número de píxeles dentro de los límites de la pasa..
  - `MajorAxisLength`: La longitud del eje principal, que es la línea más larga que se puede dibujar en la pasa.
  - `MinorAxisLength`: La longitud del eje pequeño, que es la línea más corta que se puede dibujar en la pasa.
  - `Eccentricityl`: Una medida de la excentricidad de la elipse, que tiene los mismos momentos que las pasas.
  - `ConvexArea`: El número de píxeles de la capa convexa más pequeña de la región formada por la pasa.
  - `Extent`: La proporción de la región formada por la pasa al total de píxeles en el cuadro delimitador.
  - `Perimeter`: Mide el entorno calculando la distancia entre los límites de la pasa y los píxeles que la rodean.
  - `Class`: Tipo de pasa Kecimen y Besni (Variable a predecir).

  Realice lo siguiente:

  1. Use Máquinas de Soporte Vectorial en **Python** para generar un modelo predictivo para la tabla `raisin.csv` usando el 80 % de los datos para la tabla aprendizaje y un 20 % para la tabla testing. Obtenga los índices de precisión e interprete los resultados.
  2. Repita el ítem anterior pero intente identificar el mejor núcleo (Kernel) y valor para el parámetro de regularización C. ¿Mejora la predicción?

3. Construya un `DataFrame` que compare el mejor modelo generado arriba contra los mejores modelos construidos en tareas anteriores para la tabla `raisin.csv`. Para esto en cada una de las filas debe aparecer un modelo predictivo y que en las columnas aparezcan los índices *Precisión Global, Error Global, Precisión Positiva (PP) y Precisión Negativa (PN)*. ¿Cuál de los modelos es mejor para estos datos?

- **Ejercicio 2:** [25 puntos] En este ejercicio usaremos la tabla de datos `abandono_clientes.csv`, que contiene los detalles de los clientes de un banco.

  La tabla contiene 11 columnas (variables), las cuales se explican a continuación.

  - `CreditScore`: Indica el puntaje de crédito.
  - `Geography`: País al que pertenece.
  - `Gender`: Género del empleado.
  - `Age`: Edad del empleado.
  - `Tenure`: El tiempo del vínculo con la empresa.
  - `Balance`: La cantidad que les queda.
  - `NumOfProducts`: Los productos que posee.
  - `HasCrCard`: Tienen tarjeta de crédito o no.
  - `IsActiveMember`: Es un miembro activo o no.
  - `EstimatedSalary`: Salario estimado.
  - `Exited`: Indica si el cliente se queda o se va.

  Realice lo siguiente:

  1. Cargue en **Python** la tabla de datos `abandono_clientes.csv`.
  2. Use Máquinas de Soporte Vectorial en **Python** (con los parámetros por defecto) para generar un modelo predictivo para la tabla `abandono_clientes.csv` usando el 75 % de los datos para la tabla aprendizaje y un 25 % 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. Repita el ítem anterior pero intente identificar el mejor núcleo (Kernel) y valor para el parámetro de regularización C. ¿Mejora la predicción?
  4. Con los mejores parámetros identificados en el ítem anterior realice un nuevo modelo pero haciendo selección de 6 variables. ¿Mejoran los resultados?
  5. Construya un `DataFrame` que compare el mejor modelo generado arriba contra los mejores modelos construidos en tareas anteriores para la tabla `abandono_clientes.csv`. Para esto en cada una de las filas debe aparecer un modelo predictivo y en las columnas aparezcan los índices *Precisión Global, Error Global, Precisión Positiva (PP) y Precisión Negativa (PN)*. ¿Cuál de los modelos es mejor para estos datos?
  6. Utilizando el mejor modelo construido prediga los nuevos individuos que se encuentran en el archivo `nuevos_abandono_clientes.csv`. Recuerde que si estandarizó los datos para entrenar el modelo debe guardar valores como la media y desviación estándar para estandarizar los nuevos individuos.

- **Ejercicio 3:** [25 puntos] Según el ejemplo de los hiperplanos visto en clase realice lo siguiente:

  1. Escriba la regla de clasificación para el clasificador con margen máximo. Debe ser algo como lo siguiente: $w = (w_1, w_2, w_3)$ se clasifica como Rojo si $ax + by + cz + d > 0$, de otra manera se clasifica como Azul.
  2. Indique la medida del margen entre el hiperplano óptimo de separación y los vectores de soporte.
  3. Explique por qué un ligero movimiento de la octava observación no afectaría el hiperplano de margen máximo.

- **Ejercicio 4:** [25 puntos] Pruebe que si la función objetivo a minimizar es:

$$f(w) = \frac{\|w\|^2}{2} + C\left(\sum_{i=1}^{n} \xi_i\right)^2,$$

donde C es un parámetro del modelo, entonces Lagrangiano Dual para la Máquina Vectorial de Soporte lineal con datos no separables es:

$$L_D = \sum_{i=1}^{n} \lambda_i - \frac{1}{2}\sum_{i,j} \lambda_i \lambda_j y_i y_j x_i \cdot x_j - C\left(\sum_{i=1}^{n} \xi_i\right)^2.$$