Profesor: Dr. Oldemar Rodríguez Rojas
Minería de Datos 1
Fecha de Entrega: Jueves 25 de mayo - 8am
Instrucciones:

- Las tareas serán revisadas en clase, 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.

Tarea Número 10

- **Ejercicio 1:** [30 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. Cargue la tabla de datos `tumores.csv` en `R` y genere en **R** usando la función `createDataPartition(...)` del paquete `caret` la tabla de testing con una 25 % de los datos y con el resto de los datos genere una tabla de aprendizaje. Investigue cómo se hace la separación en training-testing con el paquete `caret` ¿Cuál es la ventaja respecto a usar `sample`?
  2. Usando Redes Neuronales (`nnet`) con el paquete `traineR` genere un modelo predictivo para la tabla de aprendizaje usando 2, 4 y 20 capas ocultas ¿Qué pasa en cada uno de los casos? Lo anterior con un número máximo de iteraciones igual a 1000 (recuerde adaptar el número máximo de pesos `MaxNWts`.
  3. Repita los ejercicios anteriores usando `neuralnet` desde el paquete `traineR` con 3 capas ocultas, es decir, use `hidden = c(k1, k2, k3)` (determine usted el número adecuado para `k1, k2` y para `k3`.
  4. Compare todos los resultados de los ejercicios anteriores. ¿Cuál es mejor?

- **Ejercicio 2:** [30 puntos] Esta pregunta utiliza los datos sobre la conocida historia y tragedia del Titanic, usando los datos `titanicV2020.csv` de los pasajeros se trata de predecir la supervivencia o no de un pasajero.

  La tabla contiene 12 variables y 1309 observaciones, las variables son:

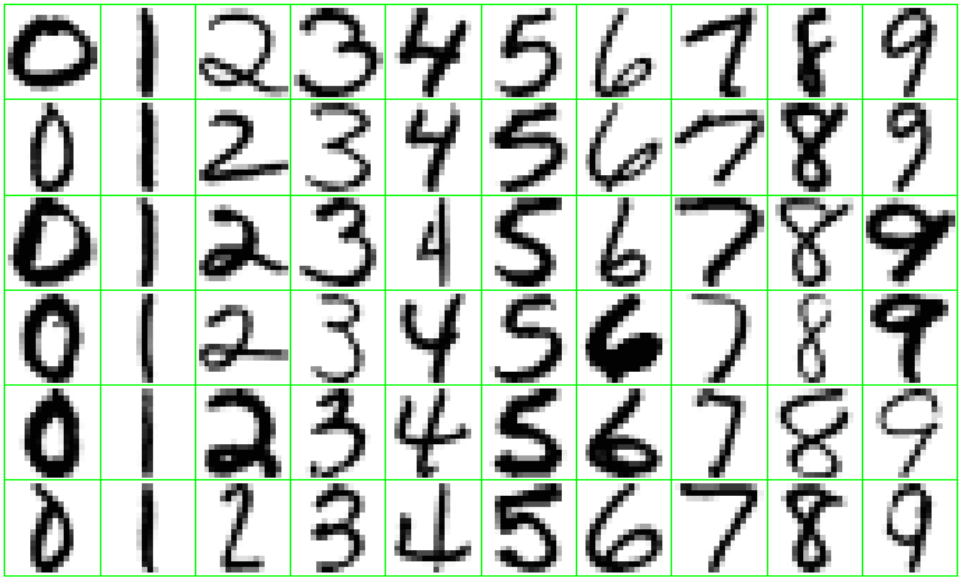
  - `PassegerId`: El código de identificación del pasajero (valor único).
  - `Survived`: Variable a predecir, 1 (el pasajero sobrevivió) 0 (el pasajero no sobrevivió).

- `Pclass`: En que clase viajaba el pasajero (1 = primera, 2 = segunda , 3 = tercera).
- `Name`: Nombre del pasajero (valor único).
- `Sex`: Sexo del pasajero.
- `Age`: Edad del pasajero.
- `SibSp`: Cantidad de hermanos o cónyuges a bordo del Titanic.
- `Parch`: Cantidad de padres o hijos a bordo del Titanic.
- `Ticket`: Número de tiquete (valor único).
- `Fare`: Tarifa del pasajero.
- `Cabin`: Número de cabina (valor único).
- `Embarked`: Puerto donde embarco el pasajero (C = Cherbourg, Q = Queenstown, S = Southampton).

Realice lo siguiente:

1. Cargue la tabla de datos `titanicV2020.csv`, asegúrese re-codificar las variables cualitativas y de ignorar variables que no se deben usar.
2. Usando Redes Neuronales, con `nnet` del paquete `traineR` y con 80 % de los datos para tabla aprendizaje y un 20 % para la tabla testing, genere un modelo predictivo para la tabla de aprendizaje, usando 4, 15 y 20 nodos en la capa oculta ¿Qué pasa en cada uno de los casos? Lo anterior con un número máximo de iteraciones igual a 1000, recuerde adaptar el número máximo de pesos `MaxNWts`.
3. Repita los ejercicios anteriores usando `neuralnet` del paquete `traineR` con 4 capas ocultas, es decir, use `hidden = c(k1, k2, k3, k4)` (determine usted el número adecuado para `k1, k2, k3` y para `k4`. ¿Mejoran los resultados?
4. Con la tabla de testing calcule la matriz de confusión, la precisión, la precisión positiva, la precisión negativa, los falsos positivos, los falsos negativos, la acertividad positiva y la acertividad negativa para los modelos anteriores (los que fue posible generar). ¿Cuál es mejor? Compare además los resultados con los obtenidos en la tarea anterior ¿Cuál es mejor?

- **Ejercicio 3:** [30 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 píxel 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 píxel, 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 predictivas.

Para esto realice lo siguiente (podría tomar varios minutos los cálculos):

1. Cargue las tablas aprendizaje y testing en **R** de los archivos `ZipDataTrainCod.csv` y `ZipDataTestCod.csv` respectivamente.
2. Use el método de Redes Neuronales con el método y los parámetros que usted considere más conveniente para generar un modelo predictivo para la tabla `ZipDataTrainCod.csv`, luego calcule para los datos de testing, en el archivo `ZipDataTestCod.csv`, la matriz de confusión, la precisión global y la precisión para cada una de las categorías. ¿Son buenos los resultados? Explique.
3. Compare los resultados con los obtenidos en la tarea anterior.
4. Repita los ejercicios 1, 2 y 3 pero usando solamente los 3s, 5s y los 8s, ¿Mejoraron los resultados? La respuesta en el método KNN debería ser:

```
$`Matriz de Confusion`
         prediccion
observado cinco ocho tres
    cinco   149    4    7
    ocho      3  158    5
    tres     10    0  156

$Precision
[1] 0.9411

$Error
[1] 0.05894
```

Nota: En su solución debe desplegar la matriz de confusión para todos los métodos.

5. Repita los ejercicios 1, 2 y 3 pero reemplazando cada bloque 4×4 de píxeles por su promedio, ¿Mejoraron los resultados? Recuerde que cada bloque 16×16 está representado por una fila en las matrices de aprendizaje y testing. La respuesta en el método KNN debería ser:

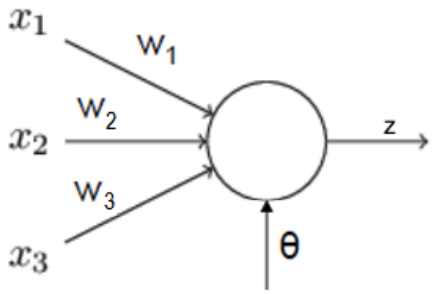
```
$`Matriz de Confusion`
         prediccion
observado cero cinco cuatro dos nueve ocho seis siete tres uno
   cero    343     0      0   2     0    4    2     0    2   6
   cinco     9   121      1   0     3    6    0     1   19   0
   cuatro    0     0    168   5    21    0    2     0    0   4
   dos       4     2      0 184     0    5    0     1    1   1
   nueve     0     0      2   0   172    1    0     2    0   0
   ocho      5     4      1   2     1  129    1     0    2  21
   seis      7     2      2   2     0    0  157     0    0   0
   siete     0     0      5   1     9    0    0   132    0   0
   tres      3     8      0   3     4    5    0     2  140   1
   uno       1     1      5   0     0    2    2     1    0 252

$Precision
[1] 0.8959

$Error
[1] 0.1041
```

Nota: En su solución debe desplegar la matriz de confusión para todos los métodos.

- **Ejercicio 4:** [10 puntos] Para la Tabla de Datos que se muestra seguidamente donde x^j para $j = 1, 2, 3$ son las variables predictoras y la variable a predecir es z diseñe y programe a pie una Red Neuronal de una capa (Perceptron):

x^1	x^2	x^3	z
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Es decir, encuentre todos los posibles pesos w_1, w_2, w_3 y umbrales θ para la Red Neuronal que se muestra en el siguiente gráfico:



Use una función de activación tipo Tangente hiperbólica, es decir:

$$f(x) = \frac{2}{1 + e^{-2x}} - 1.$$

Para esto escriba una función en `R` que haga variar los pesos w_j con $j = 1, 2, 3$ en los siguientes valores `v=(-1,-0.9,-0.8,...,0,...,0.8,0.9,1)` y haga variar θ en `u=(0,0.1,...,0.8,0.9,1)`. Escoja los pesos w_j con $j = 1, 2, 3$ y el umbral θ de manera que se minimiza el error cuadrático medio:

$$E(w_1, w_2, w_3) = \frac{1}{4} \sum_{i=1}^{4} \left[I \left[f \left(\sum_{j=1}^{3} w_j \cdot x_i^j - \theta \right) \right] - z_i \right]^2,$$

donde x_i^j es la entrada en la fila i de la variable x^j e $I(z)$ se define como sigue:

$$I(t) = \begin{cases} 1 & \text{si } t \geq 0 \\ 0 & \text{si } t < 0. \end{cases}$$