



INSTITUTO POLITECNICO NACIONAL



Escuela Superior de Cómputo PLab1.- Carga de Datos

Unidad de aprendizaje: Fundamentos de Inteligencia
Artificial

Alumnos:

Hernández Antaño Eduardo

Delgado Acosta Luis Bernardo

Flores Lara Alberto

Profesor:

Catalán Salgado Edgar armando

Grupo:

4BV1

Este código es la implementación de la carga de datos de un archivo de texto txt o csv, donde mostraremos a través de un archivo de texto, información relevante de nuestro dataset:

```
import numpy as np
import pandas as pd
```

Utilizaremos la biblioteca numpy de Python para operaciones numéricas y manipulación de matrices.y la biblioteca pandas para la lectura del archivo.

```
# Función para cargar el archivo de texto plano
def cargardatos(archivo, delimitador):
    data=pd.read_csv(archivo, delimiter=delimitador)
    return data
```

cargardatos es la función que utilizamos para cargar el archivo de texto utilizando la función read_csv de Pandas. Recibe los parámetros de nombre del archivo a cargar y el delimitador seleccionado.

```
#Funcion para imprimir en el txt
def imprimirtxt(datos_array, tipos_de_datos, vector_atributos,
matriz_patrones, matriz_patrones_reducida):
    with open('impresion.txt', 'w') as archivo:
        archivo.write('Informacion completa del dataset\n')
        archivo.write(str(datos_array))
        archivo.write("\n\nInformacion de el tipo de datos de los atributos
del dataset\n")
        archivo.write(str(tipos_de_datos))
        archivo.write("\n\nAtributos seleccionados del dataset\n")
        archivo.write(str(vector_atributos))
        archivo.write("\n\nMatriz que muestra los patrones seleccionados\n")
        archivo.write(str(matriz_patrones))
        archivo.write("\n\nMatriz que muestra los patrones seleccionados
reducidos con base en los atributos seleccionados\n")
        archivo.write(str(matriz_patrones_reducida))
        archivo.write("\n")
```

La función imprimirtxt crea un archivo de texto impresion.txt y escribe:

- Información completa del dataset.
- Tipos de datos de los atributos.
- Atributos seleccionados por el usuario.
- Matriz de patrones seleccionados por el usuario.
- Matriz de patrones seleccionados reducidos con base en los atributos seleccionados por el usuario.

```
def main():
    archivo=input("Escriba el nombre del archivo de donde obtendremos la
informacion: ")
    delimitador=input("Seleccione cual es el signo delimitador del archivo:
")
    datos=cargardatos(archivo,delimitador)
```

```

    num_filas, num_columnas = datos.shape
    print(f"El DataFrame tiene {num_filas} patrones y {num_columnas} atributos.")
    tipos_de_datos = datos.dtypes
    #Seleccionamos atributos al azar para generar el vector
    limite_inferior_1 = int(input(f"Seleccione el limite inferior (Valores entre 0 y {num_columnas}) para generar el vector de atributos: "))
    limite_superior_1 = int(input(f"Ahora el limite superior (Valores entre {limite_inferior_1} y {num_columnas}): "))
    vector_atributos=
list(datos.columns[limite_inferior_1:limite_superior_1])
    #Seleccionamos patrones al azar para generar la matriz
    limite_inferior_2 = int(input(f"Seleccione el limite inferior (Valores entre 0 y {num_filas}) para generar la matriz de patrones: "))
    limite_superior_2 = int(input(f"Ahora el limite superior (Valores entre {limite_inferior_2} y {num_filas}): "))
    matriz_patrones= datos.iloc[limite_inferior_2:limite_superior_2]
    matriz_patrones=matriz_patrones.values
    matriz_patrones_reducida = matriz_patrones[:,
limite_inferior_1:limite_superior_1]
    datos_array=np.array(datos)
    imprimirtxt(datos_array,
tipos_de_datos,vector_atributos,matriz_patrones,matriz_patrones_reducida)
    print("Se ha impreso correctamente la informacion en el archivo de texto!")
main()

```

El código comienza ejecutando la función `main()`. Después el usuario proporciona el nombre del archivo y el delimitador para cargar los datos, luego se muestra la cantidad de filas (patrones) y columnas (atributos) en el DataFrame cargado. Posteriormente se pide al usuario seleccionar límites para generar el vector de atributos y la submatriz de patrones. Se extraen los datos requeridos según los límites establecidos.

Se convierten los datos del dataset en un array para su posterior impresión en un archivo de texto usando la función `imprimirtxt()`. Y por último, se imprime un mensaje indicando que la información se ha guardado correctamente en el archivo de texto.