



# Manual de Interfaz Gráfica del Prototipo de Software para el Proyecto

## Optimización y Desarrollo de Métodos de Inteligencia Computacional Aplicados a la Solución de Problemas en Biomedicina

Responsable:  
Dr. Luis Carlos Padierna García

Los códigos del prototipo están disponibles en:

[https://github.com/padiernacarlos/CIIC-232-2019/tree/master/Interfaz\\_Gráfica\(Prototipo\\_Software\)](https://github.com/padiernacarlos/CIIC-232-2019/tree/master/Interfaz_Gráfica(Prototipo_Software))

El prototipo se encuentra programado en Python. El script principal se denomina **main\_app.py**

### Introducción

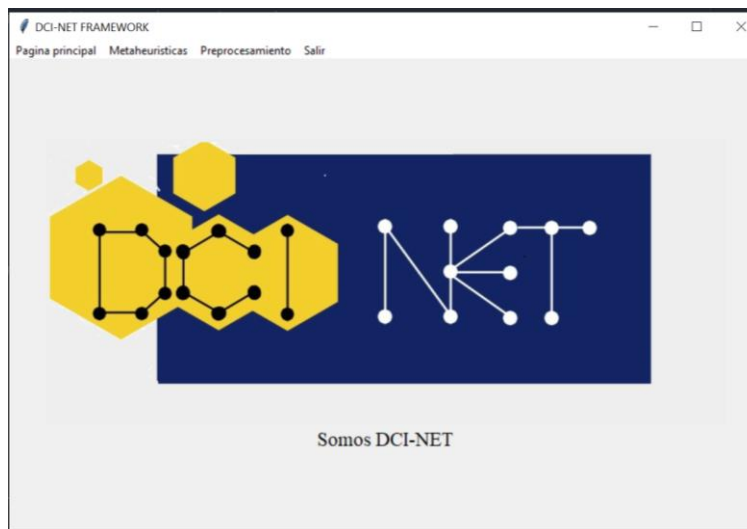
En este documento se describe el funcionamiento de la interfaz gráfica a través de la cual se accede al software desarrollado durante el proyecto. En esta versión del prototipo se ilustrará y dará acceso a algoritmos metaheurísticos y de preprocesamiento para clasificación. Para ejemplificar su uso, la interfaz muestra una animación del algoritmo Particle Swarm Optimization (PSO). También tiene dos funciones para preprocesar datos y permitir que sean compatibles con la librería LIBSVM. Estándar para el manejo de Máquinas de Soporte Vectorial.

Para darle una identidad al grupo de trabajo formado por los participantes del proyecto, decidimos nombrarlo como DCI-NET (División de Ciencias e Ingenierías)-NET. La parte -NET hace alusión a las arquitecturas de Redes Neuronales Convolucionales, encontradas en el área de Deep Learning y cuyos nombres típicos son U-Net, V-Net, etc.



## Pantalla Principal

En la pantalla principal se muestra el logo de DCI-Net junto con un pequeño saludo. Además es posible ver la barra de herramientas que contiene las opciones: “Metaheurísticas”, “ Preprocesamiento” y “Salir”. Estas pestañas se explican a continuación.



## Metaheurísticas.

En esta pestaña se mostrarán los parámetros necesarios para ejecutar el algoritmo PSO, así como algunos parámetros necesarios para generar la animación. A continuación se explica de manera breve cada uno de estos parámetros. Más información sobre el algoritmo se puede encontrar en [1].

- Número de Partículas: El número de agentes que realizarán la búsqueda en el espacio de soluciones. (Número entero).
- C1: Factor que determina qué tanto influye la mejor posición encontrada por cada partícula en la búsqueda de la solución. (Número flotante)
- C2: Factor que determina qué tanto influye la mejor posición encontrada por todas las partículas en la búsqueda de la solución. (Número flotante)
- Inercia máxima: El valor más grande que toma el término de inercia en las ecuaciones de movimiento del algoritmo. (Número flotante)
- Inercia mínima: El valor más chico que toma el término de inercia en las ecuaciones de movimiento del algoritmo. (Número flotante)
- Número máximo de iteraciones: (Número entero)
- Ángulo azimutal: Ángulo azimutal de la animación. (Número entero)

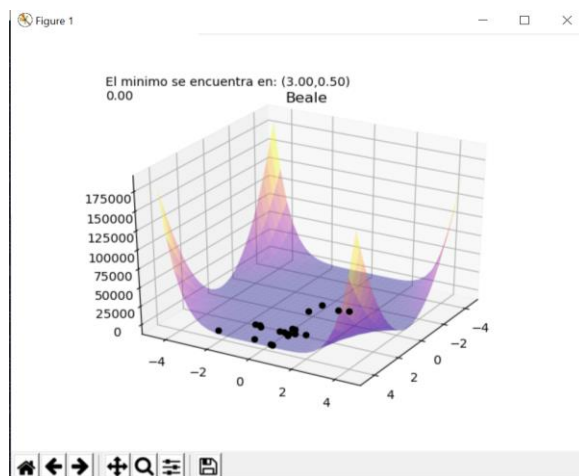
**DEPARTAMENTO DE INGENIERÍAS QUÍMICA, ELECTRÓNICA Y BIOMÉDICA**  
**DIVISIÓN DE CIENCIAS E INGENIERÍAS**

Loma del Bosque #103, Col. Lomas del Campestre. León, Gto. C.P. 37150  
Tel. +52 (477) 7 88 51 00 ext. 8522  
lc.padierna@ugto.mx  
[www.dci.ugto.mx/depinguim](http://www.dci.ugto.mx/depinguim)



- Nivel de elevación: Nivel de elevación de la animación (Número entero)
- Función a evaluar: Función objetivo que se quiere minimizar

Una vez que se han llenado todos los parámetros se mostrará una animación del funcionamiento del algoritmo. En caso de que algún parámetro falte o no sea válido se mostrará un mensaje indicándolo.



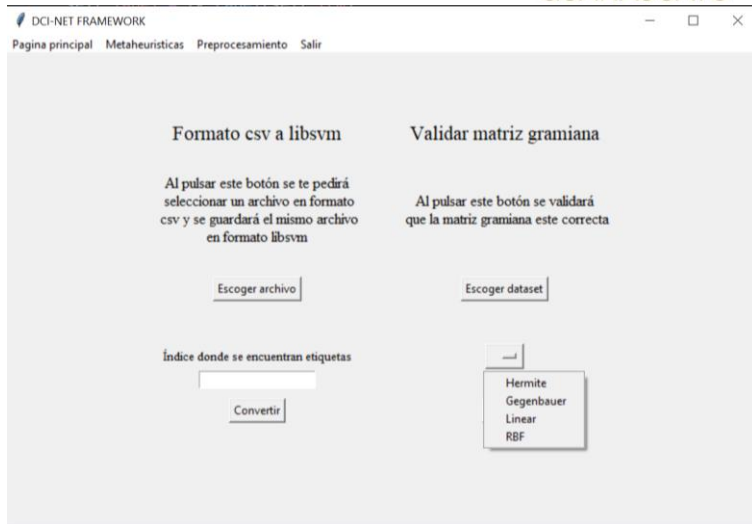


## Preprocesamiento

En la pestaña de preprocesamiento se pueden observar dos funciones. La primera función permite convertir un archivo de tipo csv a uno de tipo libsvm para que pueda ser utilizado por la misma librería.

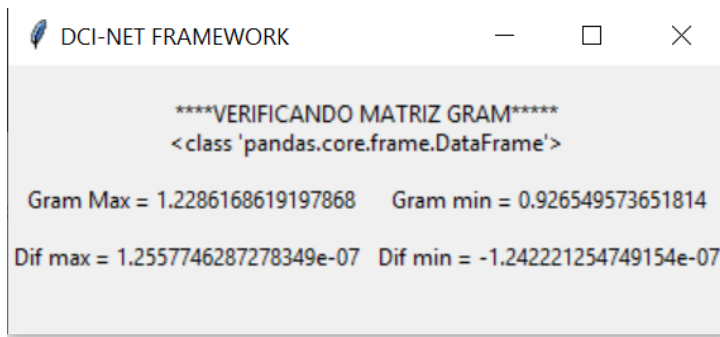
Para poder utilizar esta función es necesario indicar la ruta del archivo csv que se desea convertir, a través del botón que dice: “Escoger archivo”. Una vez seleccionado el archivo, se mostrará en pantalla la ruta relativa del archivo a modo de confirmación. Una vez corroborado que es el archivo correcto, es necesario indicar el índice del arreglo donde se encuentran las etiquetas. Es importante tomar en cuenta que la cuenta de este índice debe empezar en cero.

Una vez que todo esté completo se debe pulsar el botón que dice: “Convertir”. Este botón generará un archivo con el mismo nombre, en la misma ubicación, pero con extensión libsvm.



La última función de esta interfaz nos permite validar matrices gramianas. Para poder hacer esto es necesario indicar el folder donde se encuentran las matrices gramianas, utilizando el botón que dice: “Escoger dataset”. Es importante mencionar que lo que se debe seleccionar es el folder donde se encuentran las matrices y no algún archivo.

Después de esto se debe seleccionar el tipo de kernel que tiene la matriz a través de la lista desplegable. Una vez que se haya seleccionado la opción deseada se debe pulsar el botón con el mensaje: “Validar”. Cuando este botón sea pulsado se abrirá una ventana que muestra los datos de validación.



## Referencias

[1] Venter, G., & Sobieszczanski-Sobieski, J. (2003). Particle swarm optimization. AIAA journal, 41(8), 1583-1589.