

Apéndice C: Manual de Usuario

Práctica 1: Implementación del algoritmo de clasificación K-NN

1. Objetivo del manual

Este manual describe el procedimiento para utilizar el programa desarrollado en Python con interfaz gráfica (GUI) para ejecutar el clasificador K-NN, así como el uso de la herramienta WEKA para replicar el proceso y comparar resultados.

2. Requisitos previos

Antes de ejecutar el sistema, el usuario debe contar con:

- Python 3 instalado.
- Los archivos de datos proporcionados: `DataTrained-iris.data`, `TestData-iris.data` y `NewData-iris.data`.
- WEKA instalado (para la parte de comparación).

3. Uso del programa en Python (GUI)

3.1. Inicio del programa

Para iniciar el sistema, el usuario debe ejecutar el archivo principal del programa. Al abrirse, se mostrará una ventana con el título “*Práctica 1: Clasificador K-NN*” y un panel principal con botones para cargar archivos y ejecutar el algoritmo.



Figura 1: Interfaz principal del clasificador K-NN en Python.

3.2. Carga del conjunto de entrenamiento

1. Presionar el botón “1. Cargar Entrenamiento”.
2. Seleccionar el archivo DataTrained-iris.data.
3. Confirmar la carga exitosa mediante el mensaje mostrado por el sistema.



Figura 2: Carga del archivo de entrenamiento.

3.3. Carga del conjunto de prueba

1. Presionar el botón “2. Cargar Prueba (Test)”.
2. Seleccionar el archivo TestData-iris.data.
3. Verificar el mensaje de confirmación mostrado por el sistema.



Figura 3: Carga del archivo de prueba.

3.4. Carga del archivo de datos nuevos

1. Presionar el botón “3. Cargar Nuevos Datos”.
2. Seleccionar el archivo NewData-iris.data.
3. Confirmar que el sistema muestre el mensaje de carga exitosa.



Figura 4: Carga del archivo de datos desconocidos.

3.5. Selección del valor de K

En la sección “Valor de K”, el usuario debe escribir un número entero positivo. En esta práctica se utilizaron los valores $K = 3$, $K = 5$ y $K = 7$, sin embargo el programa permite ingresar cualquier valor válido.

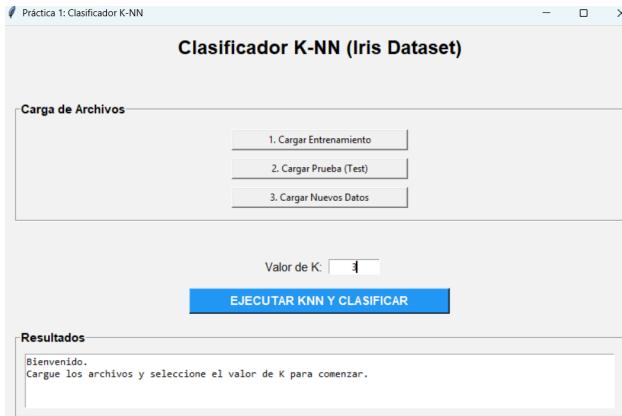


Figura 5: Ingreso del valor de K .

3.6. Ejecución del clasificador

1. Una vez cargados los archivos, presionar el botón “EJECUTAR KNN Y CLASIFICAR”.

2. El sistema calculará la exactitud del modelo (si se cargó el conjunto de prueba).
3. Posteriormente se mostrará la clasificación de los objetos desconocidos en el área de resultados.

```

Resultados

Exactitud del modelo con k=3: 0.9666666666666666

Clasificación de nuevos datos:
Objeto 1: 4.3, 2.0, 1.0, 0.1 → Iris-setosa
Objeto 2: 6.0, 3.5, 6.0, 1.5 → Iris-virginica
Objeto 3: 7.0, 4.1, 6.5, 0.8 → Iris-virginica
Objeto 4: 6.7, 3.2, 3.6, 1.2 → Iris-versicolor
Objeto 5: 6.2, 3.5, 4.0, 1.5 → Iris-versicolor
Objeto 6: 7.8, 2.8, 5.3, 1.8 → Iris-virginica
Objeto 7: 6.4, 2.8, 6.5, 2.3 → Iris-virginica
Objeto 8: 5.1, 3.2, 1.4, 0.5 → Iris-setosa
Objeto 9: 4.7, 4.0, 1.2, 0.2 → Iris-setosa
Objeto 10: 7.9, 4.4, 6.9, 2.5 → Iris-virginica

```

Figura 6: Resultados del clasificador K-NN en la GUI.

3.7. Archivo de salida generado

Al finalizar el proceso, el sistema genera automáticamente un archivo Excel con la clasificación final de los datos desconocidos. El archivo se guarda en el mismo directorio donde se ejecuta el programa con el nombre:

New-Best-K.xlsx

	A	B	C
1	Objeto	Clase asignada	
2	4.3, 2.0, 1.0, 0.1	Iris-setosa	
3	6.0, 3.5, 6.0, 1.5	Iris-virginica	
4	7.0, 4.1, 6.5, 0.8	Iris-virginica	
5	6.7, 3.2, 3.6, 1.2	Iris-versicolor	
6	6.2, 3.5, 4.0, 1.5	Iris-versicolor	
7	7.8, 2.8, 5.3, 1.8	Iris-virginica	
8	6.4, 2.8, 6.5, 2.3	Iris-virginica	
9	5.1, 3.2, 1.4, 0.5	Iris-setosa	
10	4.7, 4.0, 1.2, 0.2	Iris-setosa	
11	7.9, 4.4, 6.9, 2.5	Iris-virginica	
12			

Figura 7: Archivo Excel generado con la clasificación de datos desconocidos.

3.8. Mensajes y posibles errores

El sistema muestra mensajes de error en los siguientes casos:

- Si no se cargan los archivos necesarios antes de ejecutar.
- Si el valor de K no es un entero.
- Si el formato del archivo seleccionado no corresponde al esperado.

4. Uso de WEKA (Herramienta de minería de datos)

4.1. Objetivo del uso de WEKA

WEKA fue utilizada como herramienta de minería de datos para ejecutar el algoritmo K-NN (clasificador IBk) sobre el mismo conjunto de datos Iris, con el fin de comparar los resultados obtenidos con la implementación realizada en Python. Esta comparación permite validar el correcto funcionamiento del algoritmo y analizar el desempeño del clasificador con distintos valores de K .

4.2. Inicio de WEKA

Para comenzar el proceso, se debe abrir la aplicación WEKA desde el menú de programas del sistema operativo. Al iniciar, se mostrará la pantalla principal con diferentes módulos disponibles.

1. Abrir WEKA.
2. En la ventana principal, seleccionar la opción **Explorer**.

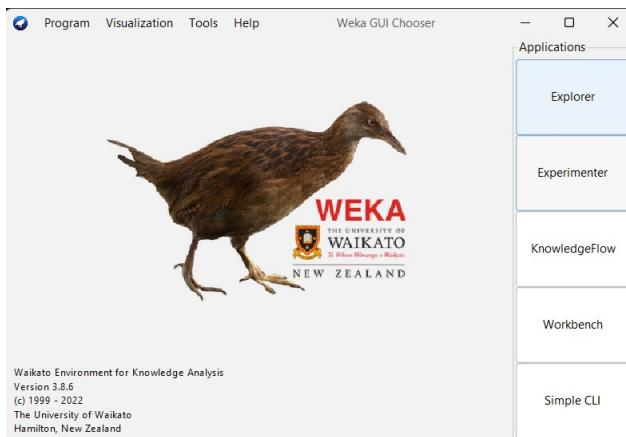


Figura 8: Pantalla inicial de WEKA.

4.3. Carga del conjunto de datos Iris

Una vez dentro del módulo **Explorer**, se procede a cargar el conjunto de datos Iris.

1. Ir a la pestaña **Preprocess**.
2. Presionar el botón **Open file**.
3. Seleccionar el archivo del conjunto de datos Iris en un formato compatible (por ejemplo **iris.arff**).
4. Esperar a que WEKA cargue el conjunto de datos y muestre los atributos en pantalla.

En esta sección es importante verificar que:

- Los atributos numéricos se cargaron correctamente.
- El atributo de clase corresponde a la última columna.
- La clase aparece como nominal con sus tres categorías (setosa, versicolor y virginica).

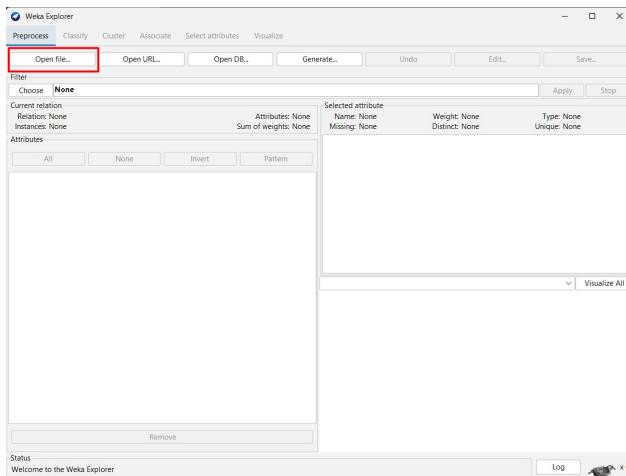


Figura 9: Carga del conjunto Iris en WEKA (Preprocess).

4.4. Selección del clasificador K-NN (IBk)

Para ejecutar el algoritmo K-NN dentro de WEKA, se utiliza el clasificador **IBk**, el cual implementa el método de vecinos más cercanos.

1. Ir a la pestaña **Classify**.
2. En la sección **Classifier**, presionar el botón **Choose**.
3. Seleccionar la ruta:

lazy → IBk

Al seleccionar el clasificador, este aparecerá en el apartado de configuración.

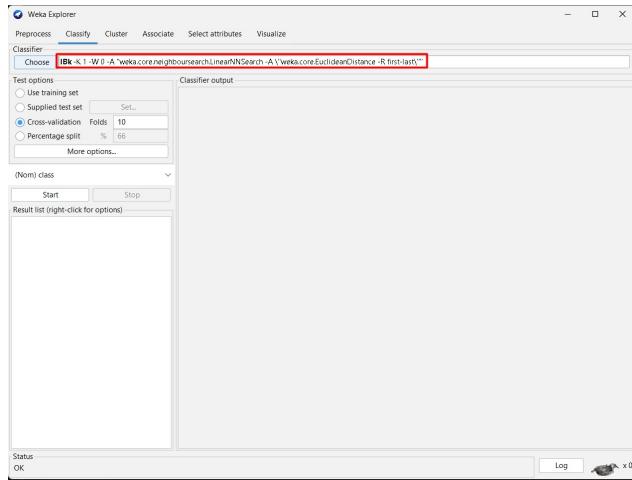


Figura 10: Selección del clasificador IBk en WEKA.

4.5. Configuración del valor de K

Para que el algoritmo funcione como K-NN, se debe configurar el número de vecinos más cercanos.

1. En la pestaña **Classify**, hacer clic sobre el nombre del clasificador **IBk**.
2. Se abrirá una ventana con parámetros del algoritmo.
3. Ubicar el parámetro **kNN**.
4. Asignar el valor deseado de K .

Para esta práctica se realizaron pruebas con:

$$K = 3, 5, 7$$

Esto implica que se ejecutó el clasificador tres veces, cambiando únicamente el parámetro **kNN**.

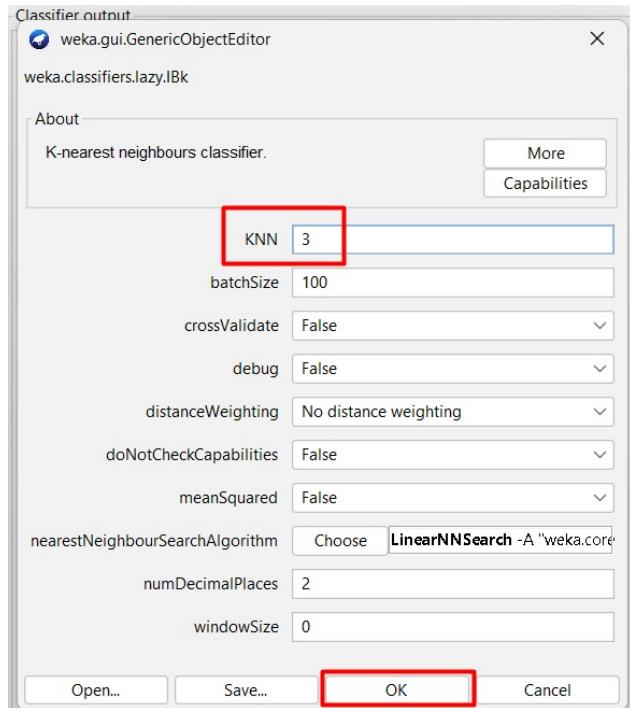


Figura 11: Configuración del parámetro K en IBk.

4.6. Configuración del método de evaluación

Para evaluar el modelo, WEKA permite diferentes opciones. En esta práctica se utilizó la evaluación mediante un conjunto de prueba, para que fuera equivalente al proceso realizado en Python.

1. En la pestaña **Classify**, ubicar el apartado **Test options**.
2. Seleccionar la opción **Supplied test set**.
3. Presionar el botón **Set....**
4. Seleccionar el archivo de prueba correspondiente (en formato compatible, por ejemplo **TestData-iris.arff**).

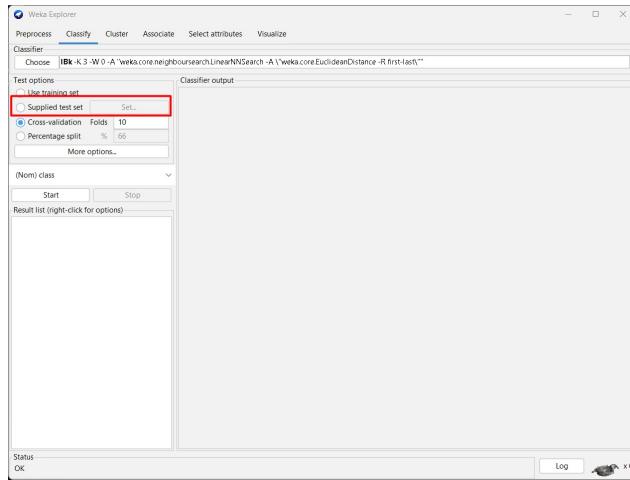


Figura 12: Configuración del conjunto de prueba en WEKA.

4.7. Ejecución del clasificador

Una vez seleccionado el clasificador, configurado K y definido el método de evaluación, se procede a ejecutar el algoritmo.

1. Verificar que el clasificador seleccionado sea **IBk**.
2. Confirmar que la opción **Supplied test set** esté activa.
3. Presionar el botón **Start**.

WEKA procesará los datos y mostrará los resultados en el panel de salida.

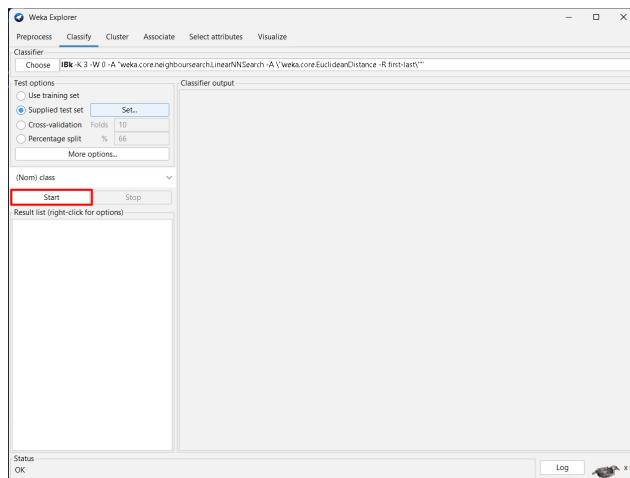


Figura 13: Ejecución del clasificador en WEKA.

4.8. Interpretación de resultados

Después de ejecutar el clasificador, WEKA muestra en pantalla un reporte automático. Entre los resultados principales se encuentran:

- **Correctly Classified Instances:** porcentaje de exactitud.
- **Incorrectly Classified Instances:** porcentaje de error.
- **Kappa statistic:** medida de concordancia.
- **Confusion Matrix:** matriz de confusión.

En esta práctica, la métrica principal utilizada para comparación fue la exactitud del modelo.

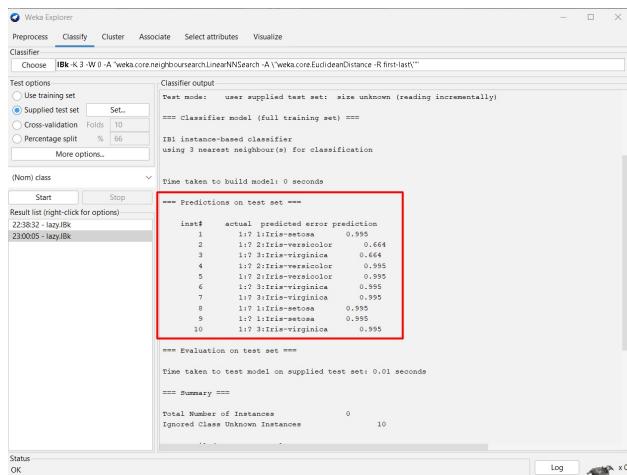


Figura 14: Resultados obtenidos en WEKA para un valor de K .

4.9. Repetición del proceso para distintos valores de K

Para completar la comparación solicitada, el proceso se repitió para cada valor de K .

1. Cambiar el valor del parámetro **kNN** en **IBk**.
2. Presionar nuevamente **Start**.
3. Registrar la exactitud mostrada por WEKA.

De esta forma se obtuvieron resultados para:

$$K = 3, K = 5, K = 7$$

4.10. Guardado de resultados

Finalmente, para documentar los resultados en el reporte, el usuario puede guardar el reporte generado por WEKA.

1. Guardar el archivo en formato .txt.

Los archivos guardados permiten respaldar la evidencia de resultados obtenidos con WEKA y compararlos con los resultados generados por la implementación en Python.

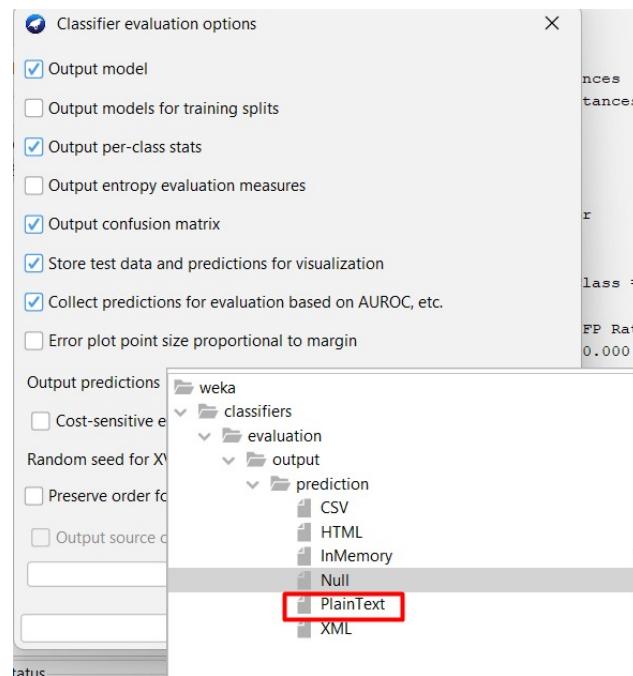


Figura 15: Exportación del reporte de resultados en WEKA.