

Práctica 1 - Manejo visualización Iris Setosa

Guerrero Prado Issac Alexander
Piña Rossette Marco Antonio

Sánchez Rojo Juan Pablo
Ugalde Velasco Armando

Universidad Nacional Autónoma de México
Ciudad de México

Resumen— En esta práctica, se mostrarán las distintas formas de desplegar una imagen en distintos formatos, así como algunas manipulaciones básicas para el procesamiento de imágenes.

Palabras clave— Reconocimiento de patrones, manipulación de imágenes, procesamiento de imágenes, Iris setosa, .

I. OBJETIVO

Al término de esta práctica, el alumno conocerá las distintas formas de desplegar y analizar información de una base de datos sencilla.

II. INTRODUCCIÓN

Base de datos

Una base de datos es una colección de datos relacionados que se almacena de forma sistemática para que se pueda acceder a ellos, gestionarlos y actualizarlos fácilmente. Los datos de una base de datos se organizan en tablas, que se componen de filas y columnas. Cada fila representa un único registro y cada columna representa un único dato sobre ese registro.

Ejemplos:

- Una base de datos de clientes almacena información sobre los clientes, como sus nombres, direcciones e historial de compras.
- Una base de datos de inventario de productos almacena información sobre los productos, como sus precios, descripciones y cantidades en stock.
- Una base de datos financiera almacena información sobre transacciones financieras, como pagos, depósitos y retiros.

Histograma

Un histograma es una representación gráfica de la distribución de datos. Es un gráfico de barras que muestra la frecuencia o el número de observaciones dentro de diferentes rangos numéricos, llamados contenedores. Los contenedores generalmente se especifican como intervalos consecutivos y no superpuestos de una variable. La altura de cada barra representa el número de observaciones en el contenedor correspondiente.

Formato de imagen

Partes del histograma:

Datos: los datos representados por el histograma pueden ser discretos o continuos. Los datos discretos son datos que solo pueden tomar un número finito de valores, como el número de estudiantes en una clase. Los datos continuos son datos que pueden tomar cualquier valor dentro de un rango determinado, como la altura de las personas.

Bins: Los bins son los intervalos en los que se dividen los datos. El número de contenedores y el ancho de los mismos pueden variar según el conjunto de datos. Una buena regla general es tener suficientes contenedores para capturar las características importantes de la distribución, pero no tantos como para que el histograma quede demasiado desordenado.

Altura de las barras: La altura de cada barra representa el número de observaciones en el contenedor correspondiente. La altura de las barras se puede mostrar como la frecuencia o la frecuencia relativa. La frecuencia es el número de observaciones en el contenedor y la frecuencia relativa es la frecuencia dividida por el número total de observaciones.

III. DESARROLLO DE LA PRÁCTICA

A continuación, se muestran algunas capturas de pantalla mostrando el código utilizado para resolver los ejercicios presentados, incluyendo los resultados de su ejecución:

```
count 150.000000 150.000000 150.000000 150.000000
mean   5.843333  3.054000  3.758667  1.198667
std    0.828066  0.433594  1.764420  0.763161
min    4.300000  2.000000  1.000000  0.100000
25%    5.100000  2.800000  1.600000  0.300000
50%    5.800000  3.000000  4.350000  1.300000
75%    6.400000  3.300000  5.100000  1.800000
max    7.900000  4.400000  6.900000  2.500000
0      float64
1      float64
2      float64
3      float64
4      object
dtype: object
0  5.1  3.5  1.4  0.2  Iris-setosa
1  4.9  3.0  1.4  0.2  Iris-setosa
2  4.7  3.2  1.3  0.2  Iris-setosa
3  4.6  3.1  1.5  0.2  Iris-setosa
4  5.0  3.6  1.4  0.2  Iris-setosa
5  5.4  3.9  1.7  0.4  Iris-setosa
6  4.6  3.4  1.4  0.3  Iris-setosa
7  5.0  3.4  1.5  0.2  Iris-setosa
8  4.4  2.9  1.4  0.2  Iris-setosa
9  4.9  3.1  1.5  0.1  Iris-setosa
```

Fig. 1. Ejercicio 3.1: Descripción de datos, tipo y primeras 10 filas.

```

llaves: Int64Index([0, 1, 2, 3, 4], dtype='int64')
Numero de renglones: 150
Numero de columnas: 5

Numero de Nan: 0

```

Fig. 2. Ejercicio 3.2 y 3.3: Llaves, datos faltantes, números de filas y columnas.

```

Matriz dispersa:
(0, 0)    1
(1, 1)    1
(2, 2)    1
(3, 3)    1
(4, 4)    1

count  0      1      2      3
mean   5.843333 3.054000 3.758667 1.198667
std     0.828066 0.433594 1.764420 0.763161
min     4.300000 2.000000 1.000000 0.100000
25%     5.100000 2.800000 1.600000 0.300000
50%     5.800000 3.000000 4.350000 1.300000
75%     6.400000 3.300000 5.100000 1.800000
max     7.900000 4.400000 6.900000 2.500000

```

Fig. 3. Ejercicio 3.4 y 3.5: Matriz dispersa y estadísticas básicas.

```

Numero de muestras para Iris Setosa: 50
Numero de muestras para Iris Versicolor 50
Numero de muestras para Iris Virginica 50

Data frame con encabezados:
class      sepal length  sepal width  petal length  petal width
Iris-setosa      5.1         3.5         1.4         0.2
Iris-setosa      4.9         3.0         1.4         0.2
Iris-setosa      4.7         3.2         1.3         0.2
Iris-setosa      4.6         3.1         1.5         0.2
Iris-setosa      5.0         3.6         1.4         0.2
...
Iris-virginica    6.7         3.0         5.2         2.3
Iris-virginica    6.3         2.5         5.0         1.9
Iris-virginica    6.5         3.0         5.2         2.0
Iris-virginica    6.2         3.4         5.4         2.3
Iris-virginica    5.9         3.0         5.1         1.8

[150 rows x 4 columns]

```

Fig. 4. Ejercicio 3.6 y 3.7: Número de muestras para cada clase y datos con encabezado.

```

Numero de muestras para Iris Setosa: 50
Numero de muestras para Iris Versicolor 50
Numero de muestras para Iris Virginica 50

Primeros 10 renglones de primeras 2 columnas:
class      sepal length  sepal width
Iris-setosa      5.1         3.5
Iris-setosa      4.9         3.0
Iris-setosa      4.7         3.2
Iris-setosa      4.6         3.1
Iris-setosa      5.0         3.6
Iris-setosa      5.4         3.9
Iris-setosa      4.6         3.4
Iris-setosa      5.0         3.4
Iris-setosa      4.4         2.9
Iris-setosa      4.9         3.1

```

Fig. 9. Ejercicio 3.8: Primeras diez filas y dos primeras columnas con sus índices

```

Primeros 10 renglones de primeras 2 columnas:
sepal length  sepal width
class
Iris-setosa      5.1         3.5
Iris-setosa      4.9         3.0
Iris-setosa      4.7         3.2
Iris-setosa      4.6         3.1
Iris-setosa      5.0         3.6
Iris-setosa      5.4         3.9
Iris-setosa      4.6         3.4
Iris-setosa      5.0         3.4
Iris-setosa      4.4         2.9
Iris-setosa      4.9         3.1

```

Fig. 10. Ejercicio 3.9: Gráfica de barras que muestra la media, mínimo y máximo de los datos.

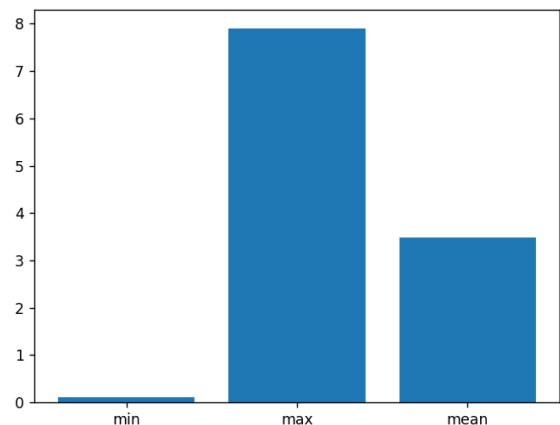


Fig. 11. Ejercicio 3.10: Tres especies.

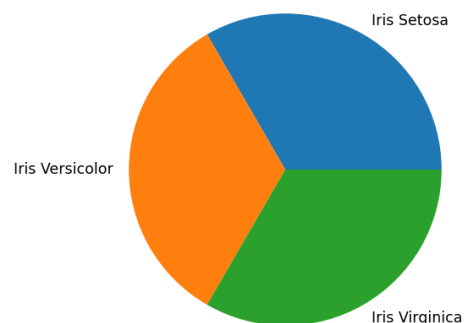


Fig. 12. Ejercicio 3.11: Relación entre longitud y ancho del sépal de las tres especies conjuntamente.

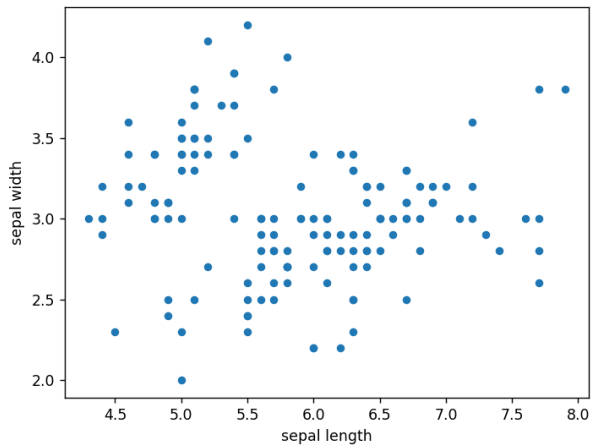


Fig. 13. Ejercicio 3.12: Histograma SepalLength.

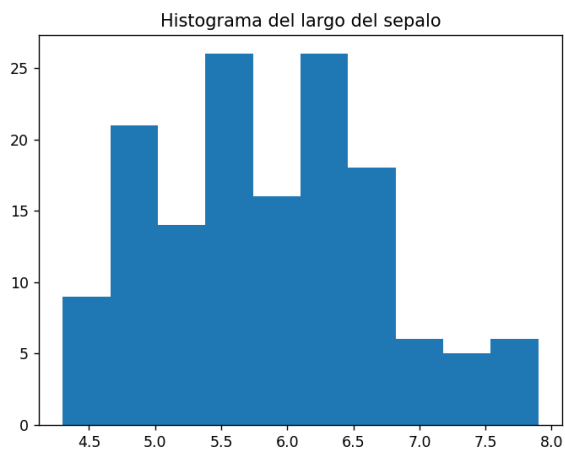


Fig. 14. Ejercicio 3.12: Histograma SepalWidth.

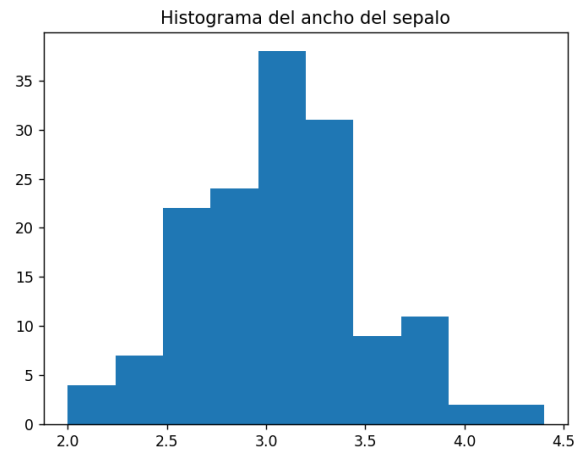


Fig. 15. Ejercicio 3.12: Histograma PetalLength.

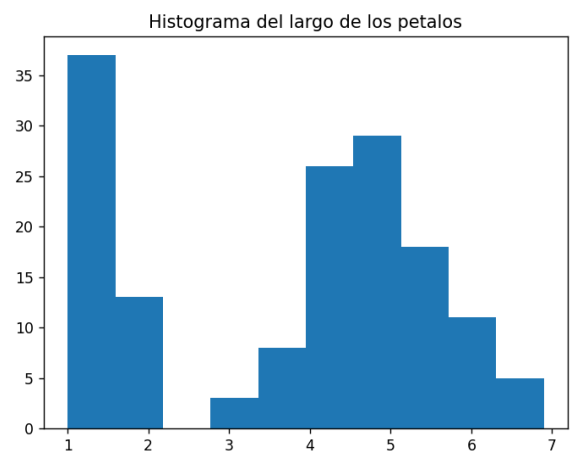


Fig. 16. Ejercicio 3.12: Histograma PetalWidth.

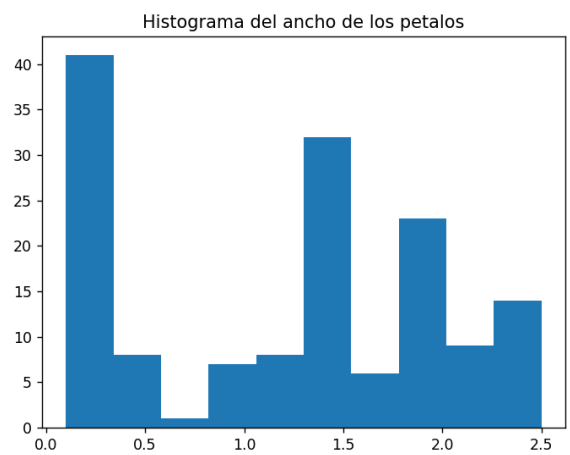


Fig. 17. Ejercicio 3.13: Gráfica de dispersión de las tres especies.

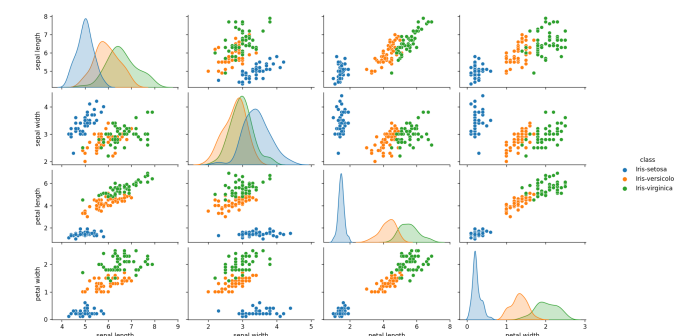


Fig. 18. Ejercicio 3.14: Dispersión entre longitud y ancho del sépalo y sus distribuciones.

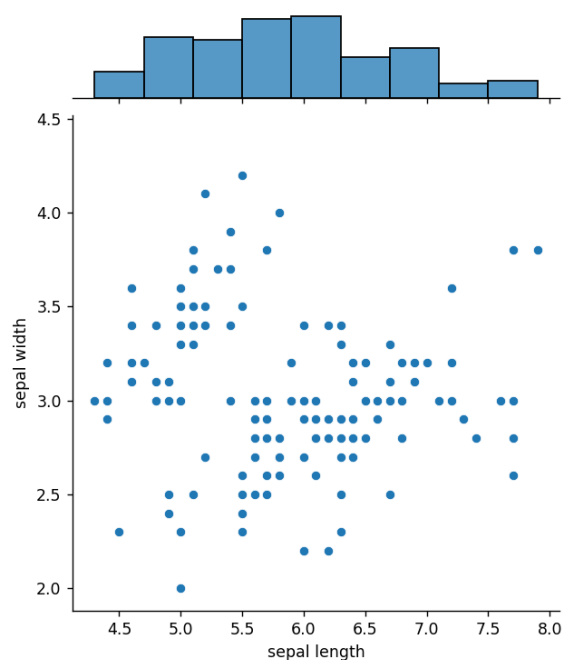
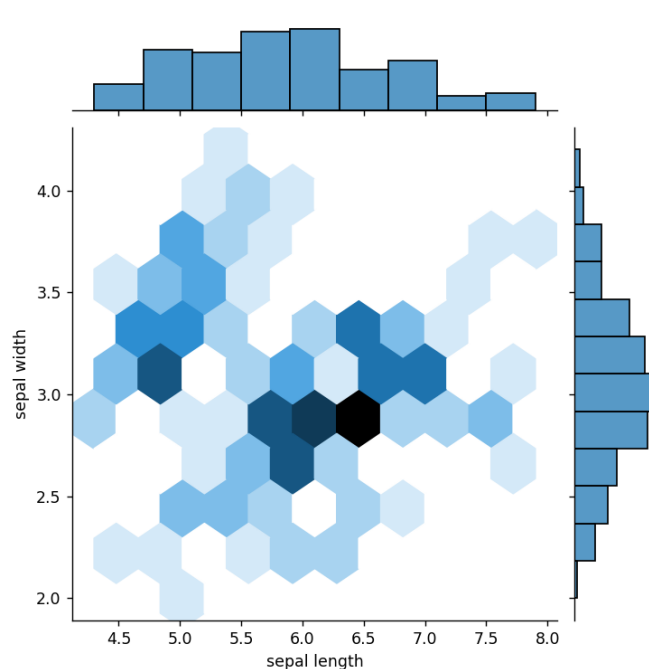


Fig. 19. Ejercicio 3.15: Dispersión entre longitud y ancho del sépalo y sus distribuciones con `kind="hex"`.



IV. CONCLUSIONES

Gracias a la realización de la práctica, logramos familiarizarnos con las herramientas necesarias para la manipulación de imágenes en Python, haciendo uso de bibliotecas como matplotlib, OpenCV, PIL, entre otras.

Lo anterior nos permitirá manipular la información de mejor manera para poder realizar un análisis de datos correcto mediante el uso de herramientas de python.

REFERENCIAS

- [1] A. C. a. C. Revision, «Pillow (PIL Fork),» Read the Docs, 2010. [En línea]. Available: <https://pillow.readthedocs.io/en/stable/reference/Image.html>. [Último acceso: 12 Agosto 2019].
- [2] S. Dey, Hands-On Image Processing with Python: Expert techniques for advanced image analysis and effective interpretation of image data, Birmingham, UK: Packt Publishing, 2018.
- [3] R. Chityala y S. Pudipeddi, Image Processing and Acquisition using Python, Boca Raton, USA: CRC Press, 2014.
- [4] P. Joshi, OpenCV with Python By Example: Build real-world computer vision applications and develop cool demos using OpenCV for Python, Birmingham, UK: Packt Publishing, 2015.
- [5] C. Solomon y T. Brekon, Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab, Oxford, UK: Wiley's global Scientific, Technical and Medical business with Blackwell Publishing., 2011.
- [6] E. Moya Albor, Práctica 2: Manejo Básico de imágenes con Matlab, CDMX, 2015.