Temas Selectos de Física Computacional III: Introducción a la ciencia de datos. Clase 1

Dr. Leonid Serkin

January 31, 2025

Introducción

El propósito de esta clase es familiarizarte con Python. Más específicamente, aprenderás a leer tu primer conjunto de datos.

1 Pasos preliminares

1.1 Disponibilidad de Python 3.7

Para asegurarte de que Python está correctamente instalado en el dispositivo, abre una terminal y ejecuta el comando `python3 --version` (o, `python --version`). Los resultados deben ser similares a los mostrados a continuación.

```
$ python --version
Python 3.7.6
```

Asegúrate de que Python 3 está instalado: la versión debe tener el formato 3.7.x.

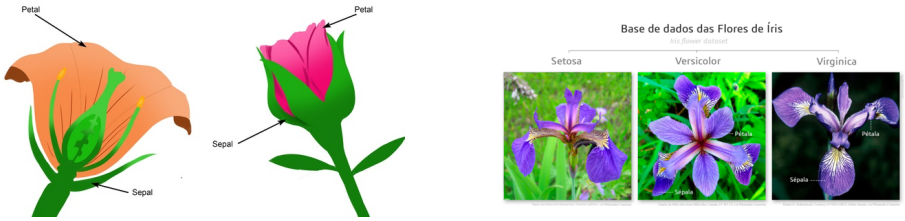
1.2 Disponibilidad de un entorno interactivo

Los scripts de Python (identificados por la extensión .py) pueden ejecutarse desde el intérprete de Python (`python script.py`) o en un entorno interactivo. Las opciones populares son 1) Jupyter Notebook, 2) Google Colab, 3) Anaconda, 4) Python instalado en una maquina virtual o en Windows. Decide e instala cualquiera siguiendo los pasos de instalación disponibles online.

2 Conjunto Iris

Iris es un conjunto de datos particularmente famoso. Es un conjunto de datos con un pequeño número de filas y columnas, utilizado principalmente para pruebas iniciales a pequeña escala y pruebas de concepto. Este conjunto de datos

específico contiene información sobre el Iris, un género que incluye entre 260 y 300 especies de plantas (puedes leer más sobre el Iris en Wikipedia). El conjunto de datos Iris contiene mediciones de 150 flores de Iris, cada una perteneciente a una de tres especies: Virginica, Versicolor y Setosa. (50 flores para cada una de las tres especies). Estas tres especies presentan flores similares, como puedes ver en la Figura 1.



(a) Diferencia entre pétalo y sépalo

(b) Flores de iris.

Figure 1

El conjunto de datos se describe con más detalle en el sitio web del Repositorio de Aprendizaje Automático de la UCI. El conjunto de datos se puede descargar directamente desde allí (archivo `iris.data`) o desde una terminal, utilizando la herramienta `wget`. Descarga el conjunto de datos desde la URL original y guárdalo en un archivo llamado `iris.csv`.

`https://archive.ics.uci.edu/dataset/53/iris`

Cada una de las 150 flores contenidas en el conjunto de datos Iris está representada por 5 valores:

- longitud del sépalo, en cm
- ancho del sépalo, en cm
- longitud del pétalo, en cm
- ancho del pétalo, en cm
- especie de Iris, una de: *Iris-setosa*, *Iris-versicolor*, *Iris-virginica*

Cada fila del conjunto de datos representa una flor distinta (por lo tanto, el conjunto de datos tendrá 150 filas). Cada fila contiene entonces 5 valores (4 mediciones y una etiqueta de especie).

El conjunto de datos está disponible como un archivo de Valores Separados por Comas (CSV). Estos archivos se utilizan típicamente para representar datos tabulares. Cada fila se representa en una de las líneas. Cada una de las filas

contiene un número fijo de columnas. Cada una de las columnas (en cada fila) está separada por una coma, de ahí el nombre.

Las siguientes son 3 líneas tomadas del conjunto de datos Iris. Debes revisar el contenido del archivo CSV tú mismo para tener una idea de cómo se ven los archivos CSV.

```
5.0,3.6,1.4,0.2,Iris-setosa
6.3,2.3,4.4,1.3,Iris-versicolor
7.2,3.0,5.8,1.6,Iris-virginica
```

Para leer archivos CSV, Python ofrece un módulo llamado `csv`. Este módulo permite usar `csv.reader()`, que lee un archivo fila por fila. Para cada fila, devuelve una lista de columnas que se pueden procesar según sea necesario. El siguiente es un ejemplo de cómo leer un archivo CSV en Python. Sin embargo, este código genera un error. Encuentra la causa del problema

```
import csv

with open("iris.csv") as f:
    for cols in csv.reader(f):
        print(cols[0], cols[2])
```

3 Ejercicios con el conjunto de datos Iris

1. Carga el conjunto de datos Iris usando el módulo `csv` presentado anteriormente.

2. Calcula e imprime la media y la desviación estándar para cada una de las 4 columnas de medición (es decir, la longitud y el ancho del sépalo, la longitud y el ancho del pétalo). Recuerda que, para una lista dada de n valores $x = (x_1, x_2, ..., x_n)$, la media μ y la desviación estándar σ se definen respectivamente como:

$$\mu = \frac{1}{n} \sum_i x_i$$

$$\sigma = \sqrt{\frac{1}{n} \sum_i (x_i - \mu)^2}$$

3. Calcula e imprime la media y la desviación estándar para cada una de las 4 columnas de medición, por separado para cada una de las tres especies de Iris (versicolor, virginica y setosa).

4. Basado en los resultados de los ejercicios 2 y 3, ¿cuál de las 4 mediciones considerarías como la más característica para las tres especies? En otras palabras, ¿qué medición considerarías "mejor", si tuvieras que adivinar la especie de Iris basándote solo en esos cuatro valores?

5. Ahora crea histogramas de densidad para cada característica (longitud del sépalo, ancho del sépalo, longitud del pétalo y ancho del pétalo), separados por especie de Iris. El código para crear un histograma de densidad se muestra a continuación:

   ```
   import matplotlib.pyplot as plt
   plt.hist(values, density=True, alpha=0.2, color=color)
   ```

   - `values`: Lista de valores para una característica específica de una especie de Iris.
   - `density=True`: Normaliza el histograma para que represente una función de densidad de probabilidad.
   - `alpha=0.2`: Ajusta la transparencia del histograma, para que las superposiciones sean más fáciles de ver.
   - `color=color`: Define el color del histograma, diferente para cada especie.

6. Una vez que tengas el histograma, puedes ajustar una curva de densidad de probabilidad normal sobre él. Para crear esta curva, primero necesitas calcular la media (`mean`) y la desviación estándar (`std`) de los valores de cada especie para la característica en cuestión.

   Luego, genera una serie de valores `x` que cubran un rango alrededor de la media:

   ```
   x = np.linspace(u - 5*s, u + 5*s, 100)
   ```

   - `u`: Media de los valores para la característica y especie.
   - `s`: Desviación estándar de los valores.
   - `np.linspace(u - 5*s, u + 5*s, 100)`: Genera 100 puntos `x` entre `u - 5*s` y `u + 5*s` para cubrir un rango amplio alrededor de la media.

7. Finalmente, puedes usar la función de densidad de probabilidad normal (`norm.pdf()`) para generar la curva y superponerla al histograma:

   ```
   plt.plot(x, norm(u, s).pdf(x), label=iris_type, color=color)
   ```

   - `norm(u, s).pdf(x)`: Calcula la densidad de probabilidad para los puntos `x` usando la media `u` y la desviación estándar `s`.
   - `label=iris-type` : Etiqueta la curva con el nombre de la especie.

- `color=color` : Usa el mismo color que el histograma para la curva.

8. Para hacer que la gráfica sea más informativa, es importante agregar un título y etiquetas a los ejes:

```
plt.title(m)
plt.xlabel(f"{m} (cm)")
plt.ylabel("densidad")
```

   - `plt.title(m)`: Coloca el nombre de la característica (`m`) como título de la gráfica.
   - `plt.xlabel(f"m (cm)")`: Etiqueta el eje `x` con la característica medida en centímetros.
   - `plt.ylabel("densidad")`: Etiqueta el eje `y` como "densidad".
   - Agrega una leyenda:

     ```
     plt.legend()
     ```

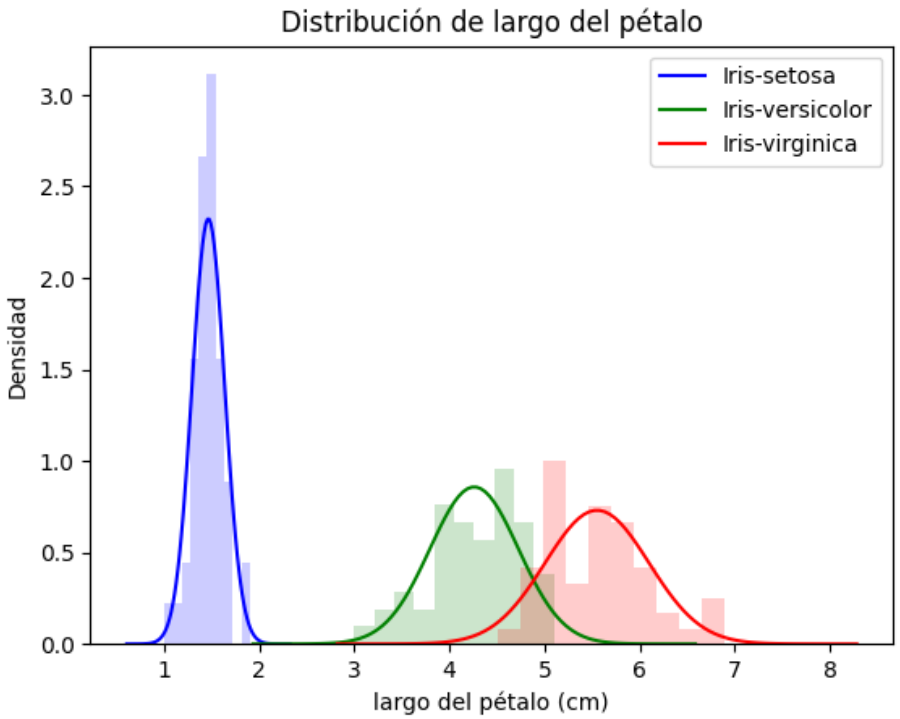


Figure 2: Distribución del ancho del pétalo para las tres especies de Iris.