Course:	**Maschinelles Lernen und künstliche Intelligenz/ Machine Learning and Artificial Intelligence**		
(Group) Member:	*Nr.*	*Name*	*Matrikel*
	1	**Miguel Angel Lopez Mejia**	**5197617**
	2		
	3		

Task Reporting Template

Please read the given Task carefully. You can find all necessary information in Moodle.

1. Definition

1.1. Introduction and Explanations

Task 2.1: Understanding the Iris Data Set ML problem

The goal is to familiarize yourself with the terminology used in ML. Please review the Iris dataset and answer the following questions:

a. What are the labels?
b. What are the features?
c. How could I use this data to create a supervised learning python program?
d. What is / are the targets for the above scenario in one sentence?

Task 2.2: Visualizing the Iris dataset

Create a python application “IrisPlot.py”. Make use of the Iris.csv file to load data and create plots to visualize the Iris dataset. Make a single plot each time that shows the data for all three species at once, in different colours.

1.2. Allowed Tools

- The script is going to be executed using ***Python version 3.13.2***
- ***Visual Studio Code*** is going to be used as IDE (Integrated Development Environment)
- Libraries: pandas, matplotlib, numpy

2. Your Preparation/ Concept

2.1. Structure your Problem

- **Task 2.1 - I have to work on:**
  - My task is to answer the following questions:
    - What are the labels?
    - What are the features?
    - How could I use this data to create a supervised learning python program?
    - What is / are the targets for the above scenario in one sentence?
  - In order to answer these questions I'm going to use the 'pandas' library to read the csv file and analyze the data that it contains.

- **Task 2.2 - I will work on following points for first part:**
  - The problem in task 2.2 is to create plots to visualize the Iris dataset. Make a single plot each time that shows the data for all three species at once, in different colours.
  - I'm going to use the library matplotlib to create the plots.

Flows:

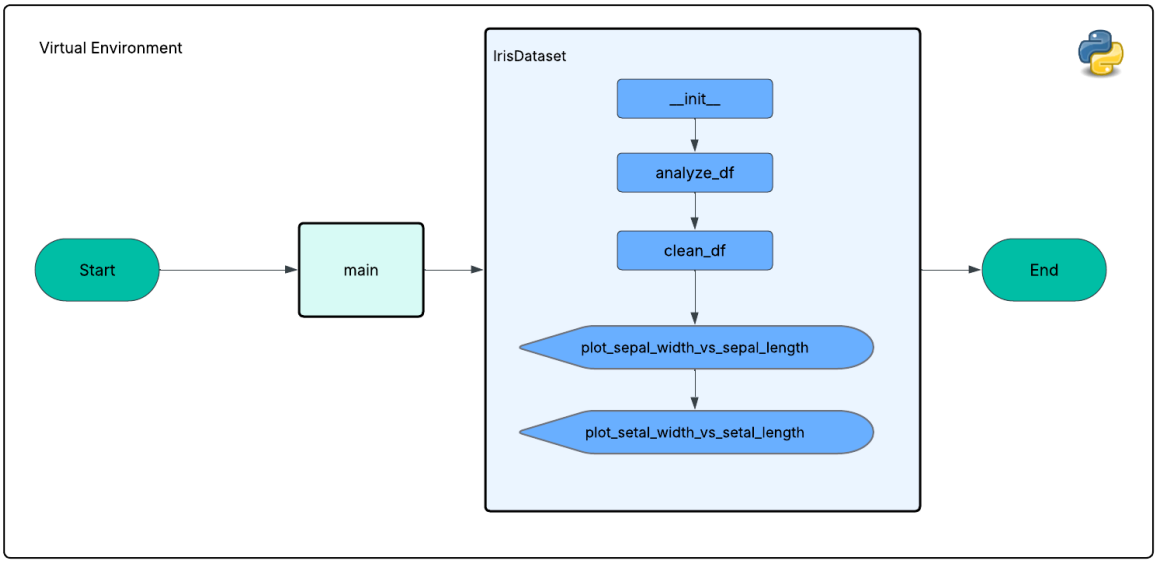


Figure 1: Iris analysis diagram

Libraries:

- **matplotlib** is a comprehensive library for creating static, animated, and interactive visualizations.
- **logging**: this module defines functions and classes which implement a flexible event logging system for applications and libraries.

- **numpy** is a Python library providing a multidimensional array object, derived objects like masked arrays and matrices, and tools for fast array operations, including math, logic, shape manipulation, sorting, I/O, Fourier transforms, linear algebra, statistics, and random simulation.

```
# region Libraries
import matplotlib.pyplot as plt
import logging
import numpy as np
# endregion
```

2.2. Your Implementation

Virtual Environment

- In order to run the script, a virtual environment must be created, open a new terminal and create it as it is shown:

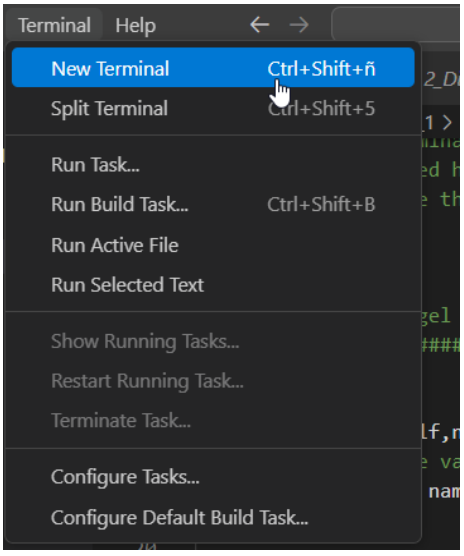


Figure 2: New terminal

- Execute the following command in the console in order to create a new virtual environment: ***python -m venv venv*** once the virtual environment is created, it will be displayed in the explorer section:

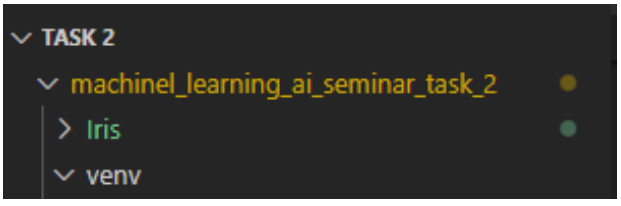


Figure 3: Virtual environment

- In order to activate the virtual environment, execute this script in the console: ***venv/scripts/activate*** .

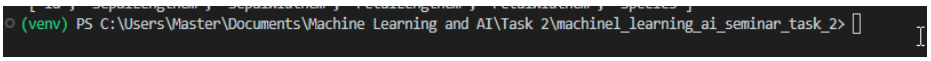
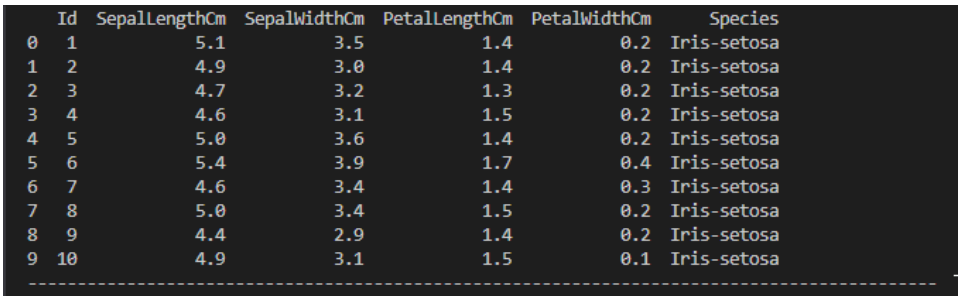


Figure 4: activation of venv

Task 2.1 - Understanding the Iris Data Set ML problem

In order to execute the script, run this command in the console: ***python .\IrisPlot.py***, Once executed, the script will read the csv file and create a dataframe, it uses:

- ***df.head(10)*** to show the first 10 rows, in that way we can visualize the data as well as the column names.



	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
5	6	5.4	3.9	1.7	0.4	Iris-setosa
6	7	4.6	3.4	1.4	0.3	Iris-setosa
7	8	5.0	3.4	1.5	0.2	Iris-setosa
8	9	4.4	2.9	1.4	0.2	Iris-setosa
9	10	4.9	3.1	1.5	0.1	Iris-setosa

Figure 5: Dataframe first 10 rows

- ***df.info()*** shows the summary of the dataframe, including columns, data type, row number and memory usage.

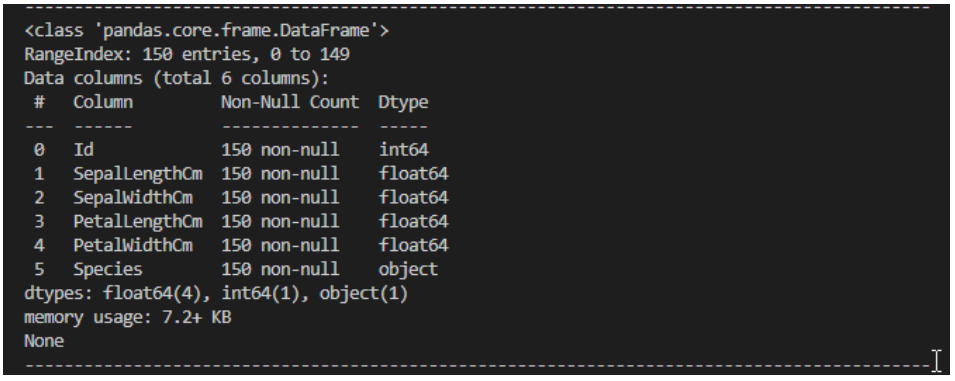


Figure 6: Dataframe information

- ***df.columns*** to extract the name of the columns

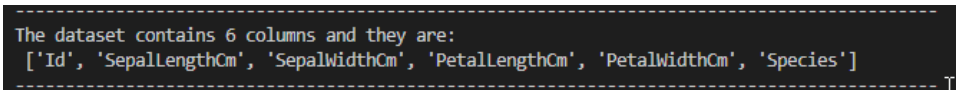


Figure 7: Dataframe columns

Analysis

- **What are the labels?**
  I extracted all the unique values from the column 'Species' as they represent the labels for each sample.

```
----------------------------------------------------------------------------------------
The labels are:
 ['Iris-setosa' 'Iris-versicolor' 'Iris-virginica']
----------------------------------------------------------------------------------------
```

Figure 8: Labels

According to Figure 8, the labels are ***'Iris-setosa' 'Iris-versicolor' 'Iris-virginica'***

- **What are the features?**
  According to *Figure 6* and *Figure 7*, the features are ***'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm'*** as these are the columns that contain information for each sample that can help us to classify them.

- **How could I use this data to create a supervised learning python program?**
  Supervised learning relies on labeled data to train models and when the model's predictions don't align with the labels, we adjust it.
  Having our labeled data and features ready, we can now use them to feed a supervised learning algorithm, which will learn and can be further refined if its output is not as expected.

- **What is / are the targets for the above scenario in one sentence?**

  The target is to classify each sample to its corresponding category, in this case *'Iris-setosa' 'Iris-versicolor' 'Iris-virginica'.*

Task 2.2 - Visualizing the Iris dataset

In order to execute the script, run this command in the console: ***python .\IrisPlot.py***.

Plot 1: The relationship between sepal width and length of the three classes of flowers.

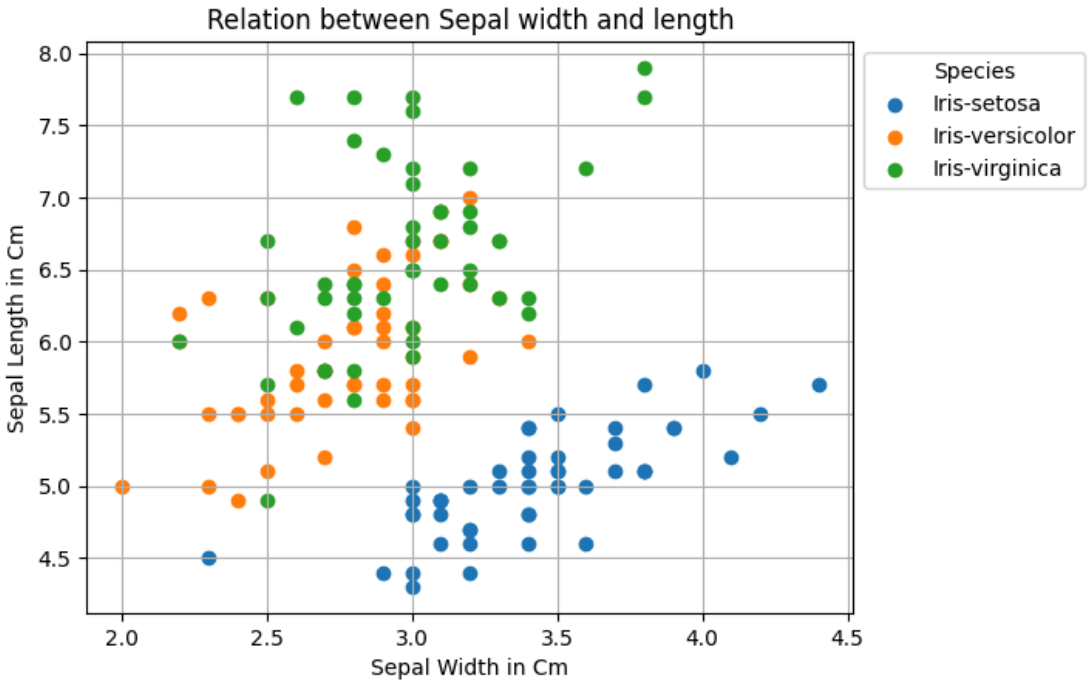


Figure 9: Relation between Sepal width and length

Plot 2: The relationship between petal width and length of three classes of flowers.

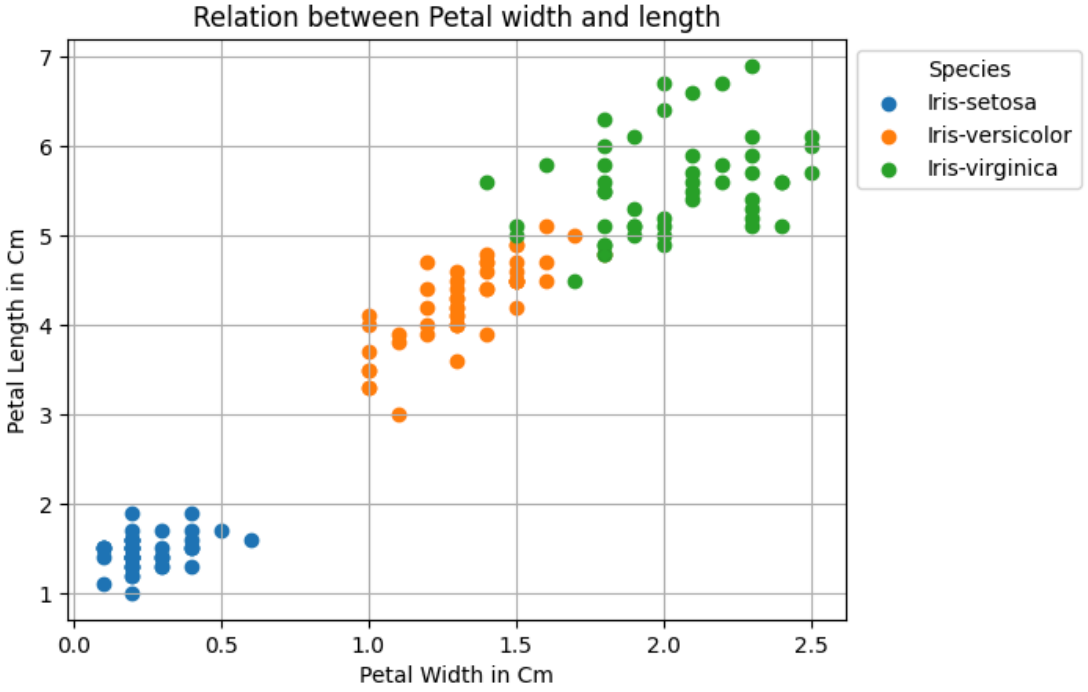


Figure 10: Relation between Petal width and length

Code

The basic idea to create these plots is:

- Extract the labels from the data set, by using the ***df['column_name'].unique()*** function from pandas.
- With a ***FOR*** loop, create a subset of data for each specie/label and create the scatter plot

```
def plot_sepal_width_vs_sepal_length(self):
    plt.figure(1)
   self.labels = self.df["Species"].unique()
    for species in self.labels:
        subset = self.df[self.df['Species'] == species]
        plt.scatter(subset['SepalWidthCm'], subset['SepalLengthCm'],label=species)
    plt.xlabel('Sepal Width in Cm')
    plt.ylabel('Sepal Length in Cm')
    plt.legend(title='Species', loc='upper left', bbox_to_anchor= (1, 1))
    plt.title('Relation between Sepal width and length')
    plt.grid(True)
```

Analysis

a. Which comparison is easier to use to make predictions?

According to FIgure 9 and FIgure 10, the relation between the Petal width and the Petal length is better to compare the data, we can see that for ***Iris-setosa***, all the petal values are in **lower left section** of the graphic, while, for **Iris-versicolor**, the values tend to be in the **middle** with some interpolations with **Iris-virginica** that are located in the **upper left** section.

b. What is / are the ranges for the features to predict the target?

According to FIgure 10, the ranges are:

Iris-setosa:

- Petal width between **0 cm and 0.6 cm**
- Petal length between **0.8 cm and 2 cm**

Iris-versicolor:

- Petal width between **1 cm and 1.7 cm**
- Petal length between **3 cm and 5.2 cm**

Iris-virginica:

- Petal width between **1.4 cm and 2.5 cm**
- Petal length between **4.5 cm and 7 cm**

Conclusion

In this task I understood the importance of analyzing, cleaning, and visualizing the data that is going to be used to create a basic machine learning. Understanding the data is important as there are features that will allow the models to generate better results, in this case, the relation between the petal length and petal width is easier to use to make predictions, as the values in most of the cases are isolated for each species, which will lead the model to have more accuracy.

Github Repository

MiguelAnhalt/machinel_learning_ai_seminar_task_2

References, extra reading links, and resources

- Dua, D., & Graff, C. (2017). *UCI Machine Learning Repository* http://archive.ics.uci.edu/ml. Irvine, CA: University of California, School of Information and Computer Science.

- GeeksforGeeks. (2023, November 20). *Iris Dataset*. https://www.geeksforgeeks.org/iris-dataset/
- W3Schools. (n.d.). *Python Classes and Objects*. https://www.w3schools.com/python/python_classes.asp