TP: Building a Streamlit App

Objective

In this TP, we will learn how to create a simple application using Streamlit. We will explore the basics of Streamlit and see how to use it to build an interactive interface.

Prerequisites

- Basic knowledge of Python
- A working Python environment (e.g., Anaconda, virtualenv, or system-wide installation)

---

1. Install Streamlit

Streamlit needs to be installed before we can use it. Run the following command:

pip install streamlit

2. Create a Streamlit Script

Create a new Python script file (e.g., `app.py`) and add the following code:

import streamlit as st

st.title("First Streamlit Application")
st.write("Welcome to this Streamlit demonstration!")

3. Run the Application

To run the Streamlit app, open a terminal in the same directory as `app.py` and run:

streamlit run app.py

This will launch a local web server and provide a URL to access the application.

4. Adding Interactive Widgets

Modify `app.py` to include interactive elements like buttons and text inputs:

import streamlit as st

st.title("Interactive Streamlit Application")

name = st.text_input("Enter your name:")
if st.button("Say Hello"):
   st.write(f"Hello {name}!")

Restart the application by running `streamlit run app.py` again to see the changes.

5. Adding a Chart with Matplotlib

Let's enhance the app by displaying a simple chart using Matplotlib.

First install matplotlib and numpy with this line of code:

pip install numpy matplotlib

You can now code your streamlit app with the code below:

import streamlit as st
import matplotlib.pyplot as plt
import numpy as np

st.title("Plotting with Matplotlib")

x = np.linspace(0, 10, 100)
y = np.sin(x)

fig, ax = plt.subplots()
ax.plot(x, y)
st.pyplot(fig)

Restart the application again to see the graph in action.

6. Conclusion

In this TP, we covered:

- Installing and running Streamlit
- Creating an interactive interface
- Adding widgets and charts

The aim of this TP is not to become streamlit pros - we don't have enough time for that. However, we've covered the basics that will enable you to create simple web front ends, ideal for creating prototypes and starting to make your models available to users.

If you want to compexify your streamlit app with components not covered in this course, don't worry, feel free to use ChatGPT, which is very useful for writing this type of code.