

CAS ETH ML in Finance and Insurance

BLOCK I: Intro to Machine Learning

Our first Google Colab (Notebook)

Version February 6th 2024

Author: Andrea Ferrario

In this short document, we will create our first Google Colab notebook. We will go through a basic machine learning exercise to get in touch with the Python syntax, nothing more. Prerequisite: you have a functioning Google Colab environment. For all details on how to prepare your Google Colab environment, I refer to the short guide “*2014_Pre_reading_material_Google_Colab_FINAL.pdf*” in the folder *Pre-reading: Google Colab Environment* contained in the Moodle page of our lecture.

Our first Google Colab notebook is described in Table 1.

Table 1. Our first Google Colab notebook.

Step	Description
1. A quick check: Wi-Fi connectivity	<p>You need a connection to the web to work with Google Colab. When at the ETH, check your Wi-Fi connectivity by clicking on the Wi-Fi icon on the bottom-right of your screen. As an ETH student you should have access to eduroam-5 / eduroam and eth-5 / eth Wi-Fi networks.</p> <p>More details are found here: https://ethz.ch/content/dam/ethz/associates/services/Service/IT-Services/files/catalogue/networks/wifi-en.pdf</p>
3. Creating the notebook “first_colab” and start working with it	<ol style="list-style-type: none">1. Go to https://colab.google/ and click on “New Notebook”2. Wait a few seconds. The notebook <code>Untitled0.ipynb</code> loads.3. Rename the notebook by clicking on its name and typing <code>first_colab.ipynb</code>4. Click on the first cell and type the shortcut <code>CTRL+M+M</code>. It will change the type of the cell from “code” to “text”. Typing the shortcut <code>CTRL+M+Y</code> does the opposite.5. Double click on the first cell and write some introductory text. Choose the formatting you prefer. You may want to write something like this: <div><pre>Hello World! This is my first Notebook in Google Colab for the CAS ETH ML in Finance and Insurance. I will try out a few functionalities and get ready for a toy example of machine learning modelling.</pre></div><div>Add some more text...add a picture...</div>6. Create a new cell by clicking on the button “+Code” under the first cell or using the shortcut <code>Ctrl+M+B</code>.7. General: If you want to update the content of a cell, double-click on it

4. Our first Python machine learning exercise

8. Double click on the code cell and write (the # symbol is important):

```
# We start with a Python exercise
# Let us import some libraries we need
!pip install numpy pandas matplotlib scikit-learn
```

9. Let us execute the code in this cell by clicking **Ctrl+Enter**

10. We are ready to start with our machine learning exercise. Let us create a new text cell (clicking on the button “+Code” under the current cell or using the shortcut **Ctrl+M+M**). We can write the title of the next section of the notebook. For instance, “Data Processing and Machine Learning: Testing Google Colab”. Choose your formatting.

11. Create a new code cell. Double click on the cell and write:

```
# Some naming conventions to speed-up writing code
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
```

12. Run the cell.

13. Create a new code cell. Write the following:

```
# Let us start by generating synthetic data with numpy
# We want two arrays called X, y. Each array has 100
entries generated as follows:
X = 2 * np.random.rand(100, 1)
Y = 4 + 3 * X
```

14. Create a new code cell. Write the following:

```
# Let us check the results of our computations
# First, we check what X and y 'are' in Python. This
means to check what their 'type' is.
# Then, we check the shape of X and Y
# Finally, we show their values.
```

```
print('What is the type of X and Y?', type(X),
      type(Y))
print('Shape of X:', X.shape)
print('Shape of Y:', y.shape)
```

15. Create a new code cell. Let us visualize data that we have just generated by writing:

```
# Let us visualize data that we have just generated.
# To do so, we use the library matplotlib that we
imported as 'plt'.
# We call different functions of matplotlib and
compose our first diagram.
plt.scatter(X, Y)
plt.xlabel('Independent Variable - X')
```

	<pre>plt.ylabel('Dependent Variable - Y') plt.title('Scatter Plot of Data') plt.show()</pre> <p>16. The visualization is a bit boring and quite uninspiring. Let us make it more interesting by create a new code cell and writing the following code:</p> <pre># Let us inject some randomness into our exercise. # We introduce the new array Y_bis: err = np.random.normal(0, 1, 100) Y_bis = 4 + 3 * X + err.reshape(100,1) # Let us plot X vs. Y_bis plt.scatter(X, Y_bis) plt.xlabel('X') plt.ylabel('Y') plt.title('Scatter Plot of Data') plt.show()</pre> <p>17. Let us add a “trend line” in a new cell:</p> <pre># Trend line (whatever it is...) model = LinearRegression() model.fit(X, Y_bis) # Collecting results from trend line intercept = model.intercept_ slope = model.coef_[0] print('We collect results:', intercept, slope)</pre> <p>18. Create a new code cell. Let us compute a trend line to our plot:</p> <pre># Let us plot X vs. Y_bis and add the trend line. # We can change a few parameters to make the plot nicer. # Examples for scatter: color = 'lightblue', 'red' marker: '.', 'd' # Examples for the trend line: color = 'green', linestyle = 'dotted', 'dashed' plt.scatter(X, Y_bis, color = 'blue', marker='o') plt.xlabel('X') plt.ylabel('Y') plt.title('Scatter Plot of Data and Trend Line') plt.plot(X, model.predict(X), color='red', linestyle='solid', linewidth=1.0) plt.show()</pre> <p>We are done with this exercise.</p>
<p>*IMPORTANT* Integrating Google Drive in Google Colab</p>	<p>With Google Colab you can save notebooks, data and models directly in your Google Drive. Similarly, you can use Google Drive to load notebooks, data and models in Google Colab. Linking Google Drive to Google Colab is called “mounting”. It is similar to attaching an external drive to your computer. We will use “mounting” in our machine learning exercise. For your information, to mount Google Drive in Google Colab:</p>

a. In the notebook `first_colab`, run the cell containing this code:

```
# mounting Google Drive in Colab
from google.colab import drive
drive.mount('/content/drive')
```

b. Follow the prompt to authorize Colab to access your Drive.