Instructions for the use of google colaboratory

This document contains information of interest related to use of the material contained in the workbooks (NoteBooks) hosted on the Google Colaboratory service, referred to here as Google Colab for short.

What is Google Colab?

Google Colab is a cloud-based tool made by Google to run Python or R code and create machine learning models. It is like other better-known resources such as Google Docs—an online text editing service. The main advantage of this tool is that it provides us with a completely ready-to-use programming environment within our web browser. This means that we do not need to install any software on our personal computers to be able to program or run any programs we develop.

Using Google Colab is also helpful because it frees our computer from having to execute these programs. That is, it allows us to save resources on our computer and perhaps execute codes/programs for which our own computer might have insufficient resources to execute (e.g., in terms of CPU power or memory).

To use this service, we must connect it with a personal or business Google account, and it can be accessed from any browser (e.g., Chrome, Explorer, Firefox, or Safari, etc.). In short, you only need basic computer equipment with a web browser, an internet connection, and a Google account. You will not be asked to save any files or code on your computer and so, completing the activities in this course will be safe, efficient, and transparent.

Access and use of Google Colab

Inside each capsule there is a link that will take you directly to a page from Google Collab. Remember that to use this resource, you must connect to it through a Google account. This page will have a format like the one in the following image:

This page will show different blocks, such as the following:

- Access to program menus. At the top, just below the document name (ending in '.ipynb') you will find the menus file, edit, insert, execution environment, tools, and help. In principle, you will not need to access any of them but, if you'd like, you can change the style of the page by clicking on 'Tools → Settings' to select the 'Dark' theme if you find it more comfortable for your eyes.
- Index. On the left side you can access the full content of the NoteBook and can jump to the different sections if you need to.
- Content. The right block on the page presents information about the learning material for each capsule. It is divided into a series of blocks, some of which are text while others contain source code (in Python or R). Its use is explained in more detail in the next section.

NoteBook format description. Once you access the Google Colab page/service, you can view the so-called Jupyter NoteBook (a Python NoteBook). This NoteBook combines blocks of text with blocks or 'cells' of code. The great advantage of these cells is their interactivity because they can be executed to directly check their results. Therefore, this

NoteBook-based material will guideyou step-by-step, from uploading a set of data to the descriptive analysis of its contents.

VERY IMPORTANT: the order of the instructions (code blocks) is fundamental and so, each cell in this NoteBook must be executed sequentially. If you omit any, the program may throw an error (an output block with a red message written in English). In this case, you will have to start again from the beginning. To make this step easier, access the 'Settings environment' menu, click 'Execution' and then click 'Execute previous'.

<u>Using NoteBooks in Google Colab</u>

Code cells are highlighted with a darker background and contain characters from a different source code. To execute them, you must click on the triangular button on the left side of each code cell. The lines that start with a hashtag (#) are comments and do not affect the execution of the program. When the first blocks are executed, the following message will appear:

"Warning: This notebook was not created by Google. The creator of this notebook is <autor>@go.ugr.es. You may request access to your data stored in Google or read data and credentials from other sessions. Check the code source before running this notebook. If you have any questions, get in touch with the creator of this notebook by sending an email to <autor>@go.ugr.es."

Don't worry, this is normal. You will need to trust the contents of this NoteBook by clicking on 'Run anyway'. All the code runs on an external computer server and will not affect your computer equipment in any way. No information or credentials are ever shared and therefore, you can continue with this course safely.

To make it easier to execute each block of code, you can also click on each cell and use the keyboard shortcut 'Ctrl+Enter' (or 'Cmd+Enter' on computers running macOS). By executing the first code blocks, the system will connect to the Google resources (runtime environment) and the upper right corner of the Google Colab window will change from its appearance in Figure 2 ('Not Connected') to its appearance in Figure 3 ('Connected'). There will be a slight delay before you will be able to see the results of the execution in your browser.
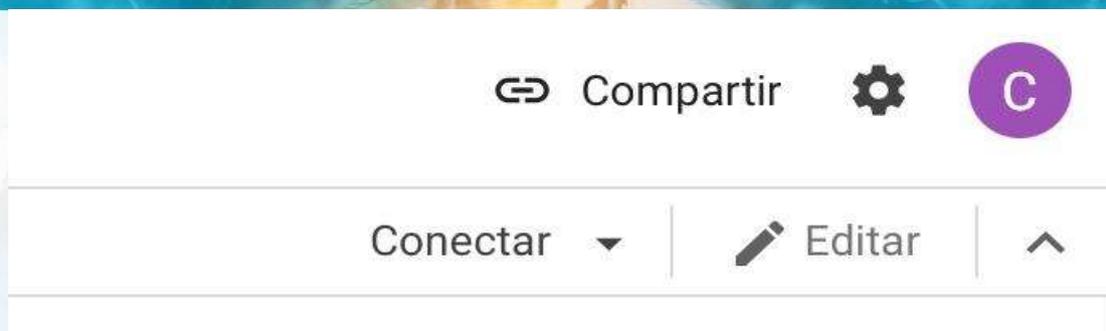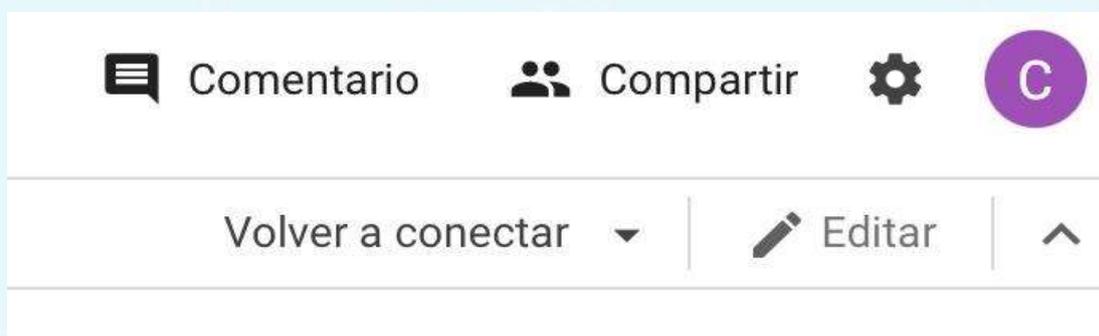
Figure 2. The top right corner of the Google Colab window, showing the status of the build environment startup execution (displaying the 'Not connected'/'Connect' option).

Figure 3. The top right corner of the Google Colab window, showing the status of the build environment running when connected (displaying the RAM and Disk usage indicators).

This environment will be active temporarily and so if you access the NoteBook again later, you may need to reload the runtime resources and re-execute the earlier blocks of code so that the subsequent blocks will work. If your Google Colab environment goes offline, the status shown in Figure 4 will be displayed.

Figure 4. Top right corner of the Google Colab window, showing the status of the build environment execution (displaying the 'Offline' status).

Every time you execute a block, you can see the output just below it. The information almost always relates to the last instruction, together with all the 'print()' (print) commands in the code. You will see a number in brackets on the left side of each executed cell. This number shows the order in which each cell was executed. You can also consult information about who executed the cell, when, and how long the execution took by examining the cursor to the left of the cell. If you have any doubts or queries, remember that you will always have direct access to the MOOC coordination team through the forum available on the platform.