

Javier Josué Fong Guzmán
100437994
Programming in R

Task 1 Google Colab

Google Colab (short for Collaboratory) is a free-to-use, cloud-based service developed at Google Research. It allows anybody (with a google account) to run arbitrary python and R code in the browser. It is specially well suited to developed machine learning and data analysis projects.

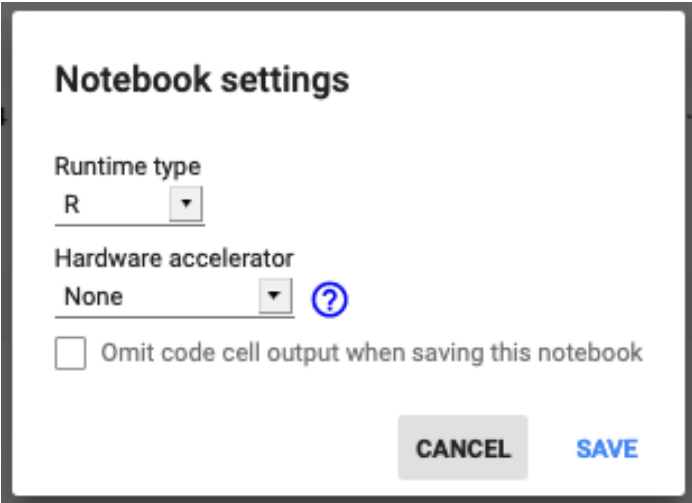
Based on Jupyter notebooks, the main difference between the 2 is that Colab does not required any setup, installation or management from the user side. It is a plug-and-play service. It inherits some characteristics from the Google platform, such as that the notebooks can be simultaneously edited by team members, similar to a Google Docs File.

As usual, the free version (the one tested for this assignment) does have some limitations. R Resources such as processing power or available memory are not always guaranteed and they're not unlimited. The usage limit sometimes fluctuates. Having said that, all these issues can be avoided by obtaining a Colab Pro Licence.

Although Google Colab was created to run only python code at the beginning, now it is possible to create R notebooks and use native R commands on them. To do so, you only need to click the following link:

<https://colab.research.google.com/#create=true&language=r>

This link will create a new notebook in your Google Colab, and it will look exactly like any other python notebook. But if you check in the *Runtime* tab and go to *Change runtime type*, you'll see that under Runtime type setting will be set on **R**.



The image shows a 'Notebook settings' dialog box. It has a title bar at the top. Below the title, there are two dropdown menus. The first is labeled 'Runtime type' and has 'R' selected. The second is labeled 'Hardware accelerator' and has 'None' selected, with a blue question mark icon to its right. Below these is a checkbox labeled 'Omit code cell output when saving this notebook', which is currently unchecked. At the bottom right, there are two buttons: 'CANCEL' in a grey box and 'SAVE' in a blue box.

You can easily install and import any library you need, as if you were in any other R environment:

Library installation

```
[1] install.packages('dplyr')
     install.packages('mlbench')
```

Installing package into '/usr/local/lib/R/site-library'
(as 'lib' is unspecified)

Installing package into '/usr/local/lib/R/site-library'
(as 'lib' is unspecified)

```
[3] library(dplyr)
     library(mlbench) # Contains several benchmark data sets (especially the Boston Housing dataset)
     library(ggplot2)
```

You can manipulate data. Use native and imported functions.

```
▶ setosa.petal.data = iris %>%
  select(Petal.Length, Petal.Width, Species) %>%
  filter(Species == "setosa") %>%
  mutate(Petal.ratio = Petal.Length / Petal.Width)

setosa.petal.data %>% head(10)
```



A data.frame: 10 × 4

	Petal.Length	Petal.Width	Species	Petal.ratio
	<dbl>	<dbl>	<fct>	<dbl>
1	1.4	0.2	setosa	7.000000
2	1.4	0.2	setosa	7.000000
3	1.3	0.2	setosa	6.500000
4	1.5	0.2	setosa	7.500000
5	1.4	0.2	setosa	7.000000
6	1.7	0.4	setosa	4.250000
7	1.4	0.3	setosa	4.666667
8	1.5	0.2	setosa	7.500000
9	1.4	0.2	setosa	7.000000
10	1.5	0.1	setosa	15.000000

You can do whatever you can do in any other R IDE. From data manipulation and visualization. To machine learning model design and creation.



```
ggplot(iris, aes(x = Petal.Width, y = Petal.Length, color = Species)) +  
geom_point()
```

