

Configuration Manual

MSc Research Project
Data Analytics

Saigirish Palavarapu
Student ID: X17170401

School of Computing
National College of Ireland

Supervisor: Theo Mendonca

National College of Ireland
Project Submission Sheet
School of Computing



Student Name:	Saigirish Palavarapu
Student ID:	X17170401
Programme:	Data Analytics
Year:	2018
Module:	MSc Research Project
Supervisor:	Theo Mendonca
Submission Due Date:	20/12/2018
Project Title:	Configuration Manual
Word Count:	961
Page Count:	9

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	11th December 2019

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission , to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project , both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Sai Girish Palavarapu

X17170401

1 Introduction

The data for this project has been taken from 2 different locations. The processes of extraction is discussed in the below points.

- Initially the data has downloaded to the local system.
- This downloaded data is about 15GB of size and then they consist of test and train folders of images and two CSV files which consists of image ID with its diagnosis stage.
- To get the data which is available in Kaggle, we need to register into the official website¹. This Registration process can also be done using Gmail account credentials or else this process of registration can also be done by creating normal Kaggle email id and password.
- After successful login into website, click the links provided below to download the data²³
- Because of the unavailability of the results of test data in the second website, the test data has been taken from third link shown in the footnote.

Now we got the Data to the local system and because of its size and required computational power to process the data, Google's colab platform for building machine learning models have been used in this research. With help of images shown below the Kaggle registration with options for downloading the data is explained.

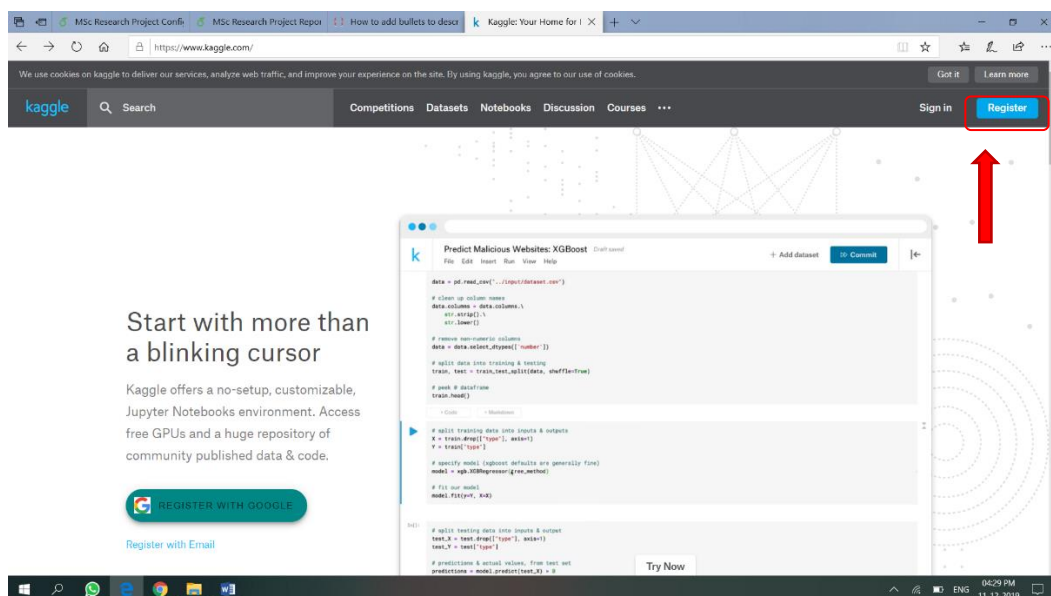


Figure 1: The official website of Kaggle

After clicking the registration button, you will be navigated to the next page as shown below.

¹ <https://www.kaggle.com/>

² <https://www.kaggle.com/c/aptos2019-blindness-detection/data>

³ <https://www.kaggle.com/c/diabetic-retinopathy-detection/data>

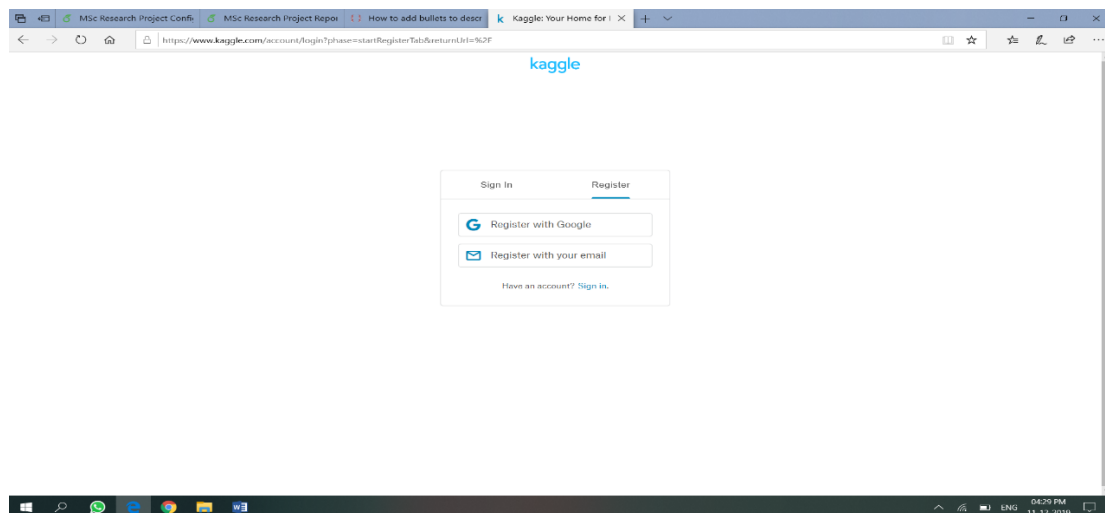
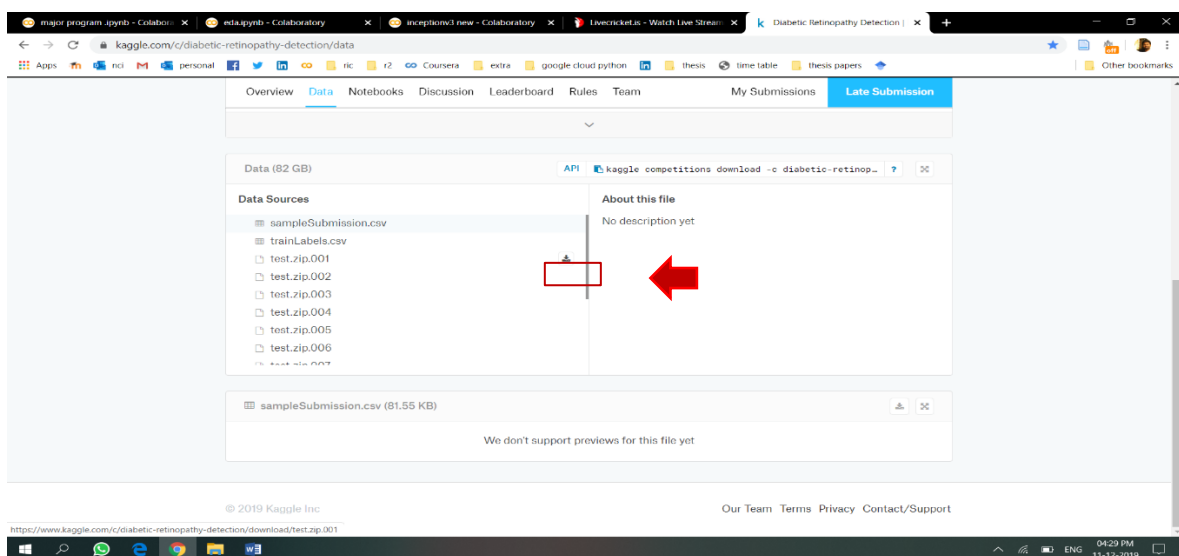
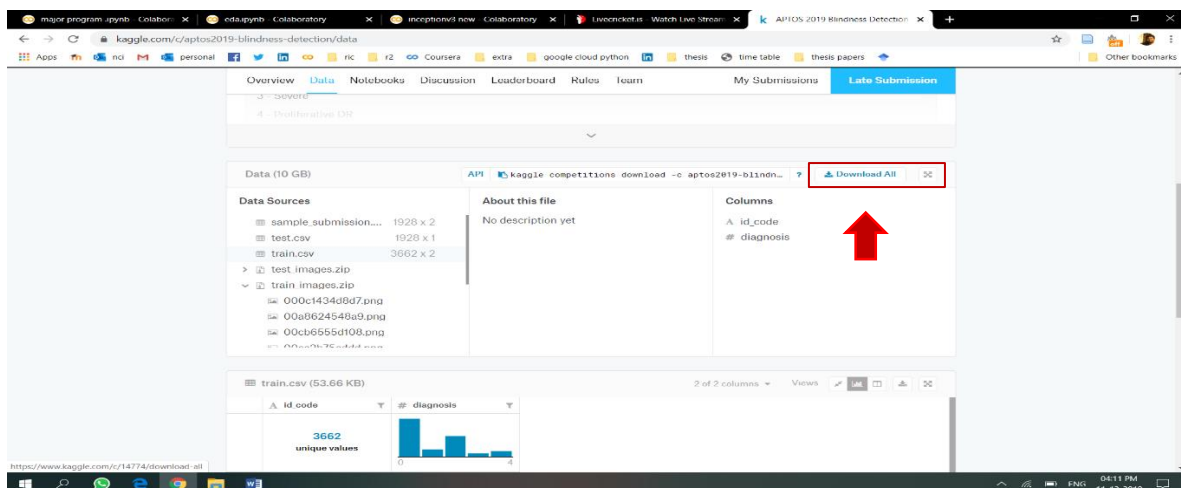


Figure2: Options for registration

- As explained above, after logging in successfully navigate to the links^{2,3} provided above. It will show as below images.



- The downloaded data is then uploaded to Google drive for further process.

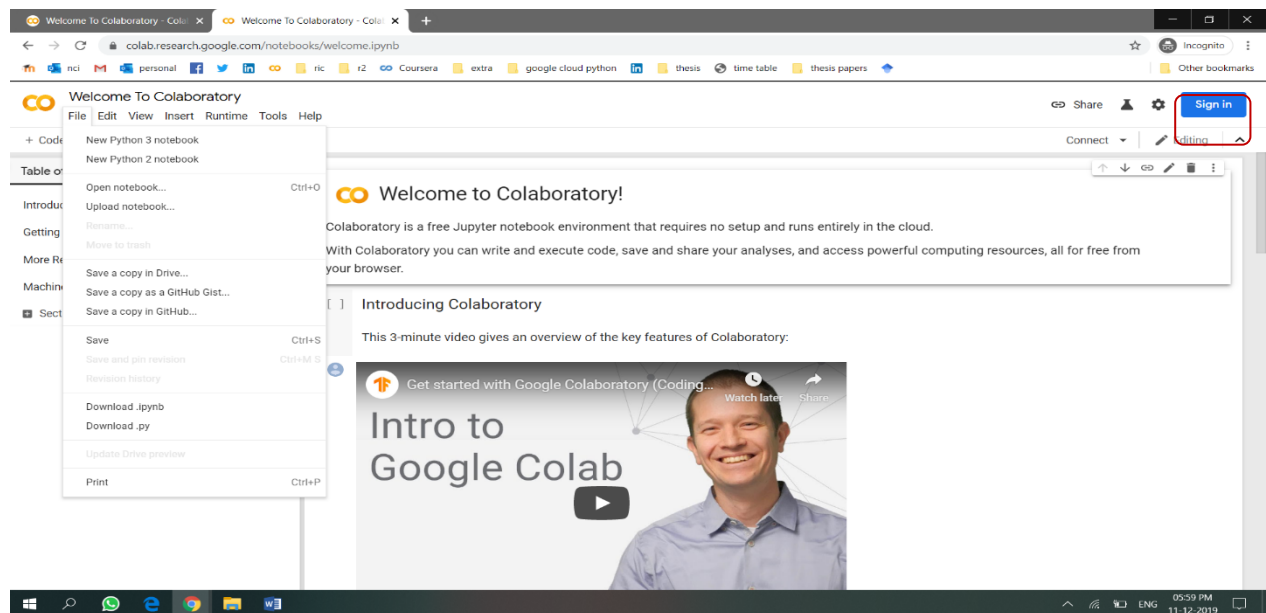
2. Usage of Colab Tool

The process of creating a file using colab and also converting it to GPU run-time and connecting google drive to colab has explained in below. The Python 3 version is used in this research.

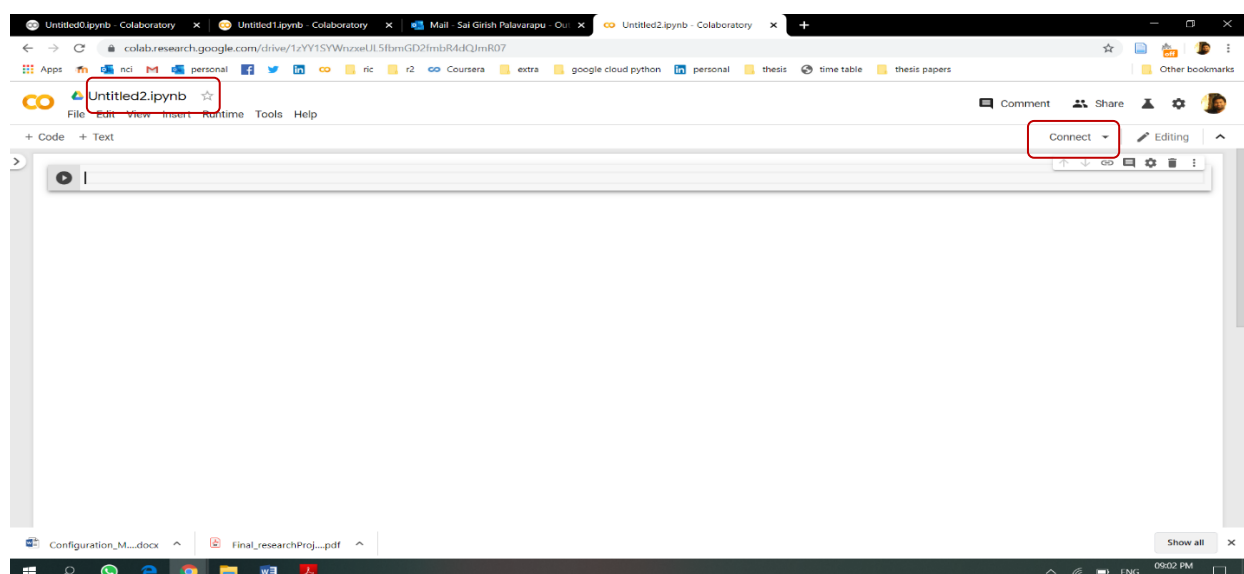
Creation of colab file

Follow the steps will allow us to create a GPU runtime colab file.

- 1) Click the link: <https://colab.research.google.com/>
- 2) Login to your Gmail where the file needs to be saved.
- 3) Click file → New Python3 Notebook.

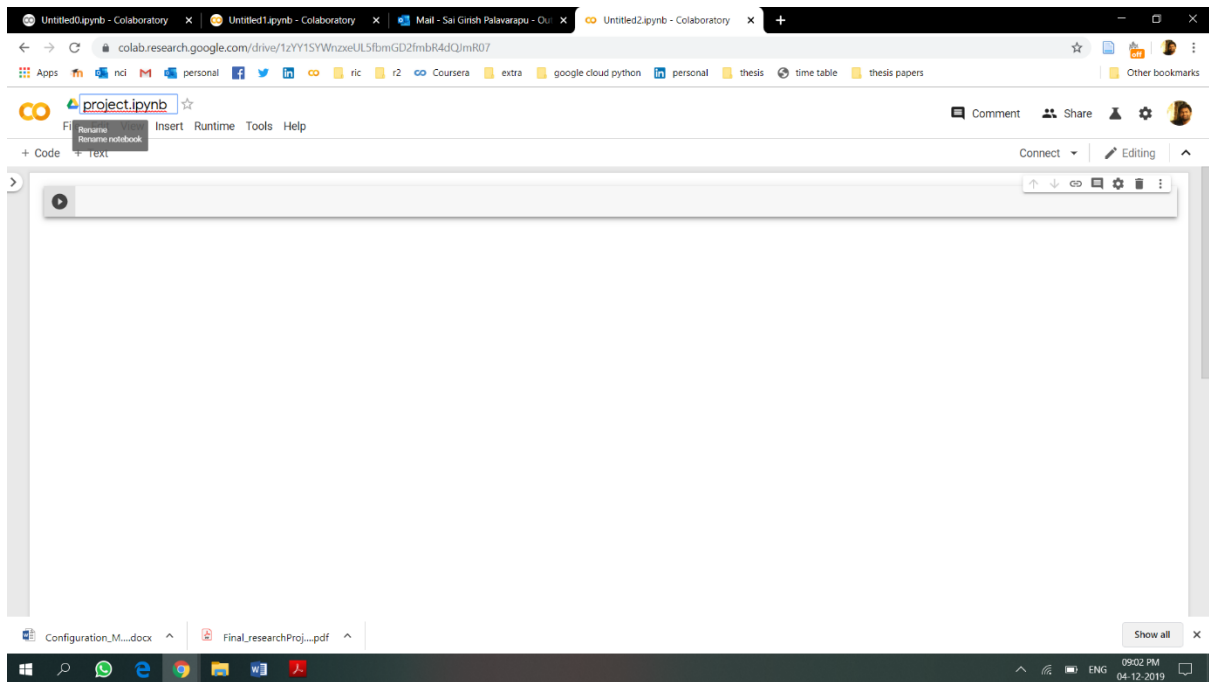


After that you will be created a blank file as shown in below figure. Now the created file does not have any name and set to localhost run time. This runtime will use the local system hardware to process the model which is not our moto in this research. To change the runtime, follow the below steps.

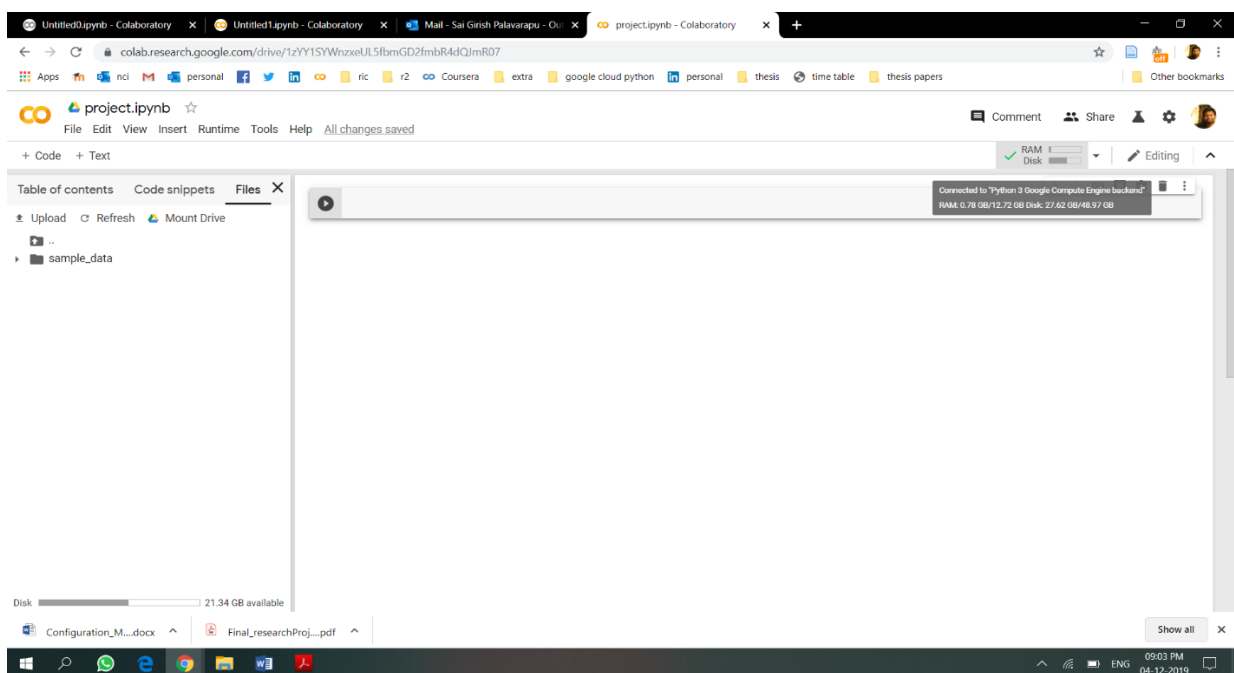


Blank file which is not connected to any run-time and can be connected to runtime using connect button as shown in a square box. File is named as Untitled.ipynb as default.

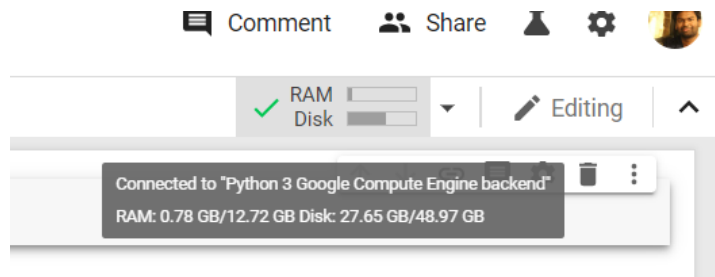
- 4) Change the name of the file to project by clicking on the title box.



- 5) Click on the connect button and it connects to the local runtime which have a ram of 12GB and hard disk of 50GB. As shown in the below image the blank file is using 0.75GB of local runtime.

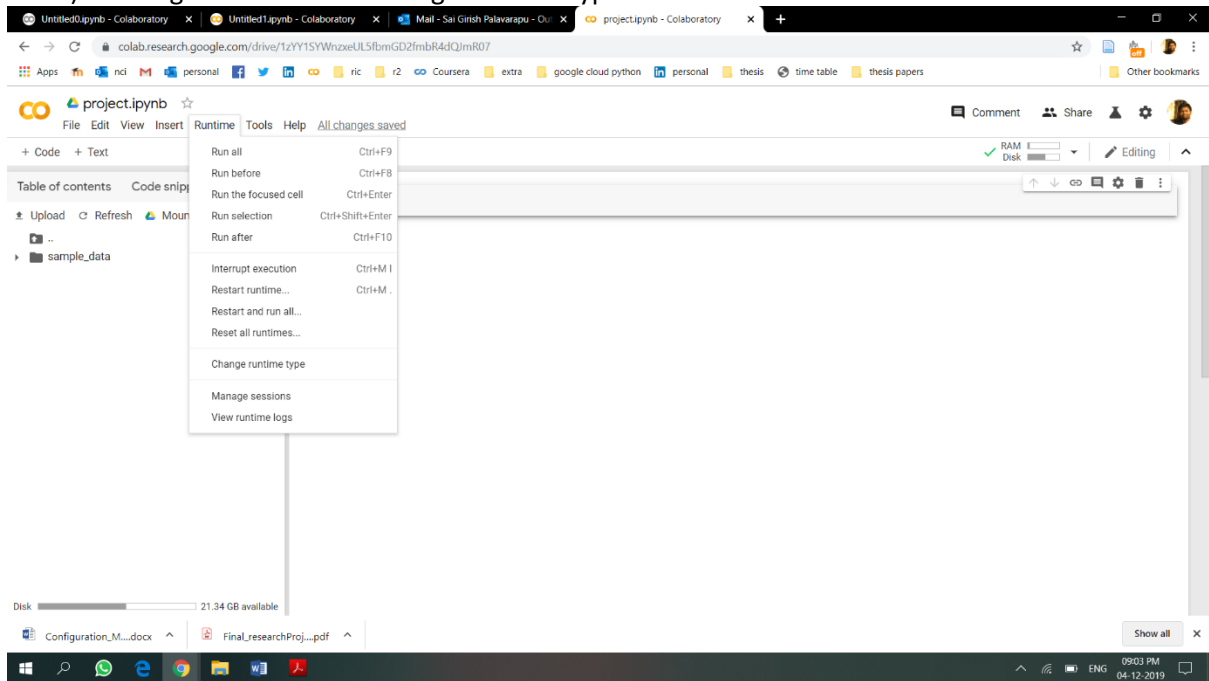


The ram and memory usage of local runtime is shown below.



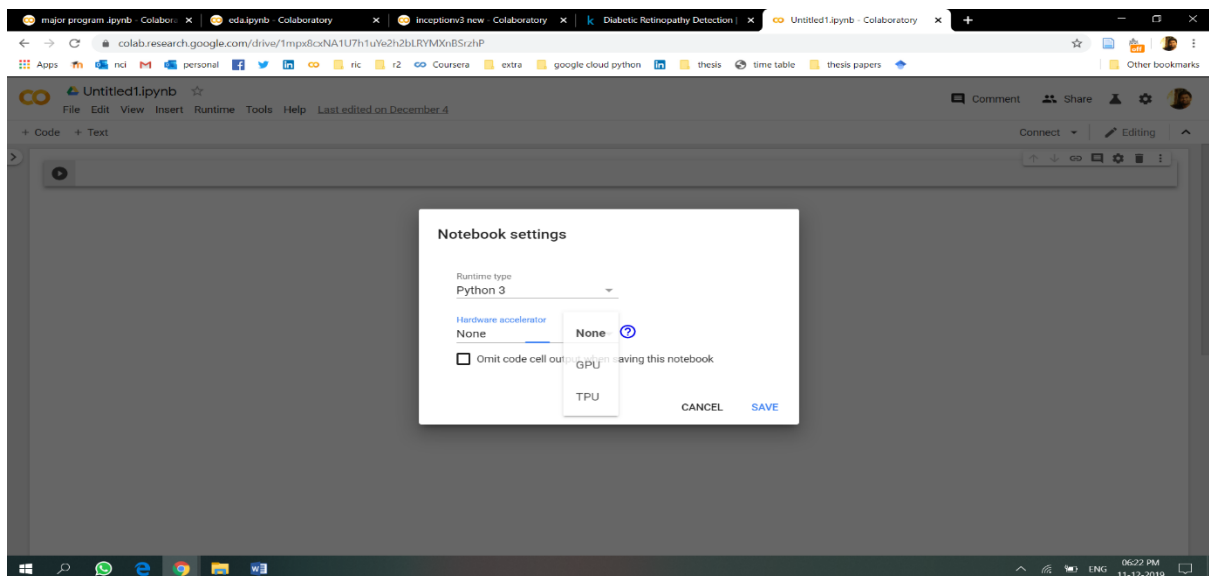
6) To change the local runtime of the file, go to Runtime → reset all runtimes.

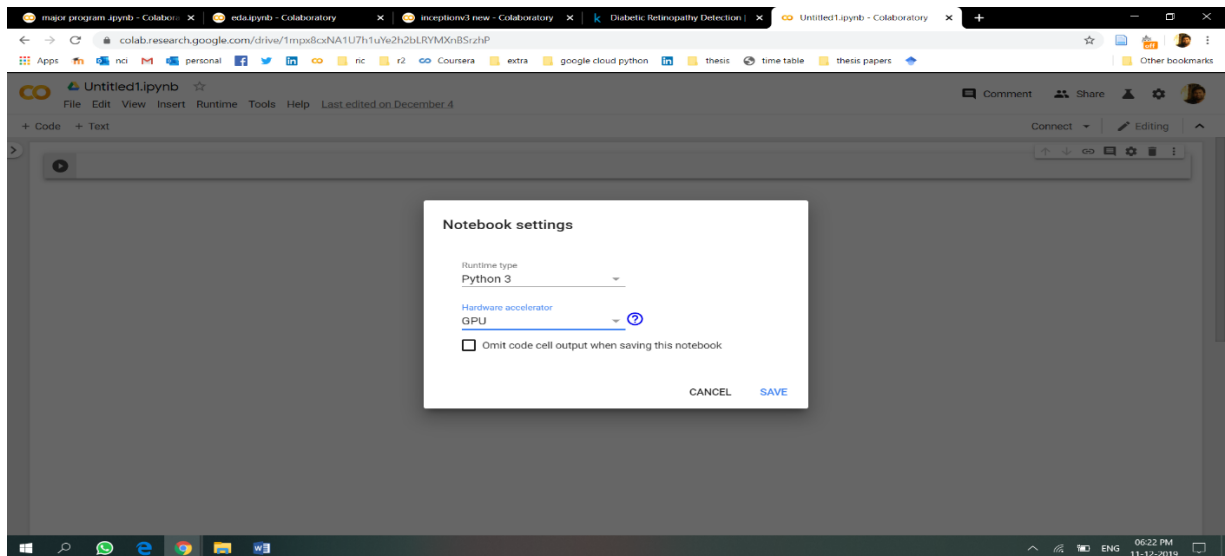
7) Then go to Runtime → change runtime type.



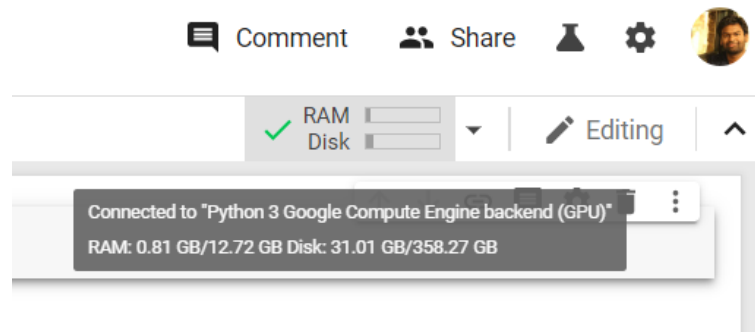
After this step you will be navigated to below page.

8) Click the Hardware accelerator → select GPU and then save the file.



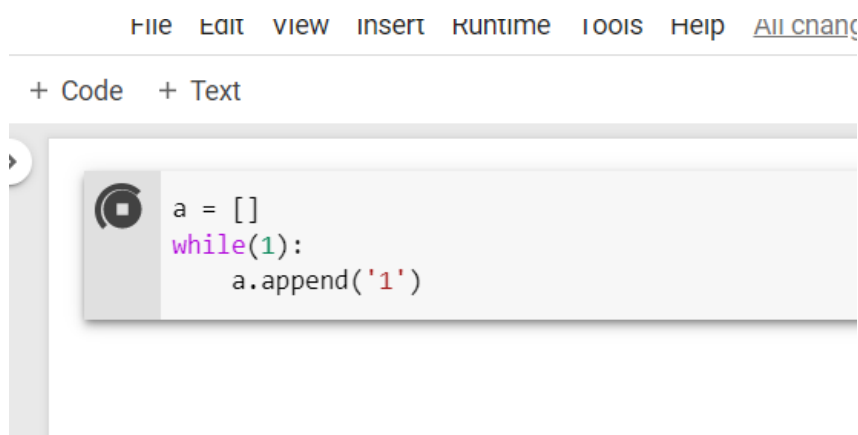


If the above steps are followed, then the allocation of ram and hard disk by the host runtime is shown below. RAM:12.72GB Disk:358.27GB

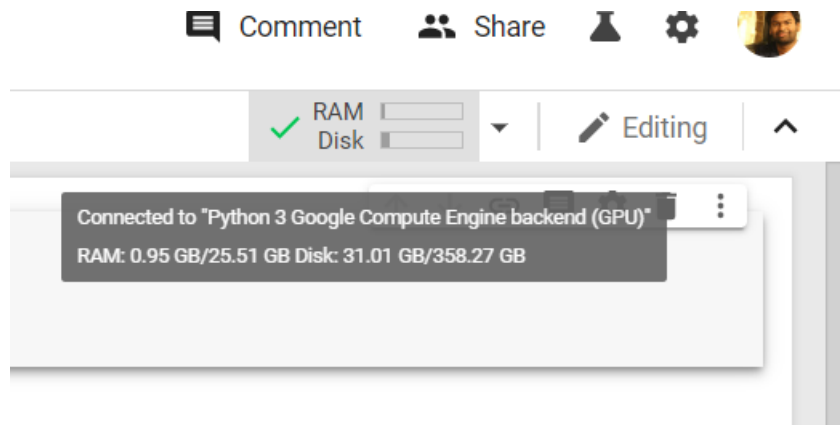


The RAM size is not fixed to 12.72 GB, it can increase up to 25GB if the model required. To increase the RAM size the entire kernel will be killed and allocated freshly. There is problem in this process that if the kernel is killed then the entire data stored in RAM will be removed. This may Lead to execution of entire model from first. To avoid such issues a small code will be executed which is shown in below.

9) Execute the below code.



- The increases RAM size will be reflected immediately as shown in the below image.

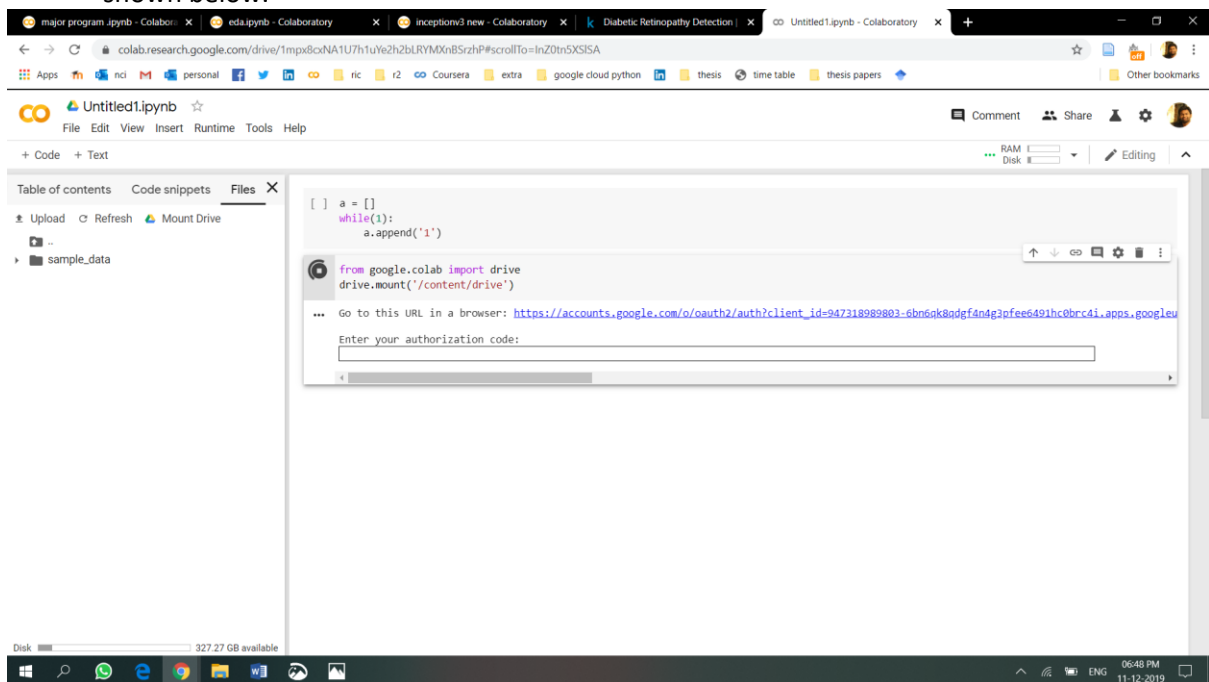


- The RAM size is increased to 25.51GB and Disk size will be remained same. This size will be remained permanently to this file.

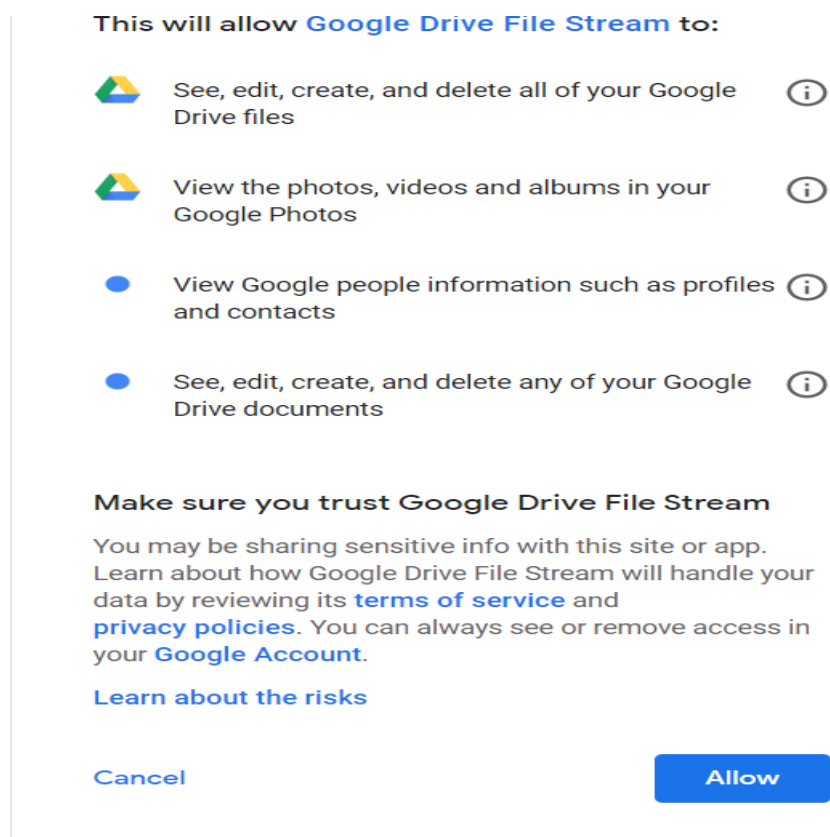
After increasing the size of the RAM, the next step is to connect colab to Google Drive for code to interact with the data.

The process begins with,

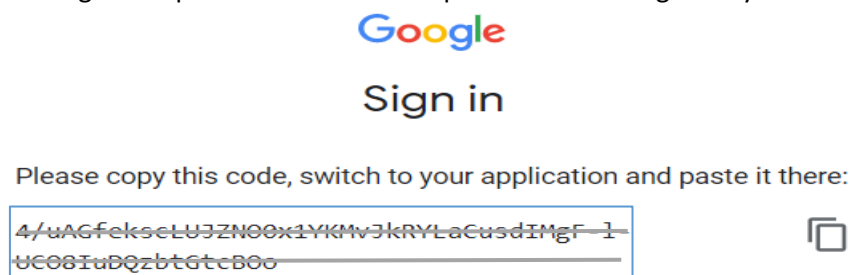
- 10) Go to file in the left panel → Mount Drive. The code for the connection will be appeared as shown below.



- 11) After executing the code, it asks for the one-time password to establish connection. To get this code click on the link given by the code and it will navigates to the other tab in the browser to login with Gmail and asks for required permission as shown below.

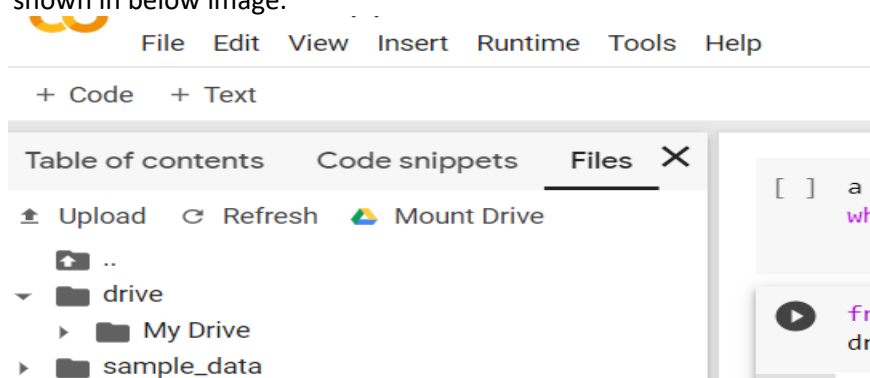


- 12) After granting the permissions to google, the password will be given by the browser as shown in below image. This password needs to be copied and pasted in the box given by the code.



This password will be active for only specific period of time and after that it will be expired. This password will serve as a gateway to access the data stored in google drive.

- 13) Now we have the tool and the dataset ready for building the model. The drive has been mounted at /content/drive/ folder in colab which can be reflected in the left panel of colab as shown in below image.



This are the steps needs to be followed begore building the model. Setting-up the tool, increasing the memory size, connecting colab to google drive is explained clearly with the help of images. But we need to know the location where the files are stored.

These files which consists of code will be saved in the Google drive which you have initially signed in second step. If you go to the drive you will find a new folder with colab as a default name and all the files related to colab code will be store in this folder.