

Google Colaboratory

IIC3697 - Aprendizaje Profundo

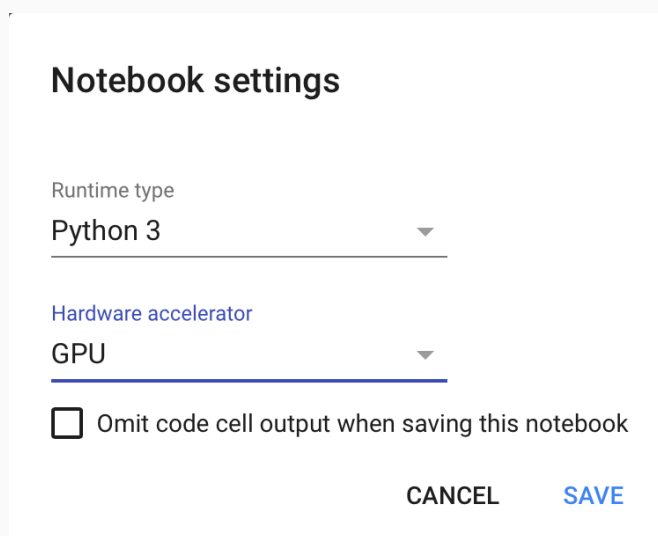
Departamento de Ciencia de la Computación, Pontificia Univerdad Católica de Chile

Para el desarrollo de las tareas se recomienda el uso de la plataforma [Google Colaboratory](#). Este es un proyecto de [Google](#) orientado a diseminar la educación y la investigación en el área de aprendizaje de máquina, y nos permitirá utilizar [GPUs](#) para ejecutar las tareas del curso.



ACTIVACIÓN DE LA GPU

Para poder usar una GPU en tu notebook, debes activarla. En el menú **Edit**, has click sobre **Notebook settings**. En la opción **Hardware accelerator**, elige la opción **GPU**.



CONFIGURACIÓN DEL AMBIENTE DE TRABAJO

Cada vez que se crea una conexión a un **instancia** en **Google Colaborative**, se crea un entorno vacío¹. Esto significa que cada vez hay que instalar todas las herramientas que se van a utilizar. Se recomienda crear una celda de configuración al principio de tu **Notebook**.

Para instalar programas en la instancia, debes escapar el comando que pondrías en la terminal, con un **!**. Por ejemplo, para instalar **antigravity**, tienes que escribir:

```
!pip install antigravity
```

¹En realidad viene con varios paquetes típicos, podrías no necesitar instalar nada.

RESTRICCIONES DE GOOGLE COLABORATORY

Google Colaboratory es una herramienta gratuita, por lo que existen restricciones de uso:

- Tu **instancia** es interrumpida luego de algún rato de inactividad o 12 horas, por lo que se recomienda que guardes **avances parciales** para evitar perder tiempo de cómputo.
- Las **GPUs** se comparten entre varias instancias de Google Colaboratory, por lo que no necesariamente vas a disponer de toda la memoria que tiene la **GPU**.

MEMORIA DISPONIBLE EN GPU

El siguiente [snippet](#) te permite ver la cantidad de memoria disponible en la GPU. Revisa que tengas suficiente memoria asignada antes de empezar a trabajar.

```
!ln -sf /opt/bin/nvidia-smi /usr/bin/nvidia-smi
!pip install gputil
!pip install psutil
!pip install humanize
import psutil
import humanize
import os
import GPUtil as GPU
GPUs = GPU.getGPUs()
gpu = GPUs[0] # Only one GPU on Colab and isn't guaranteed
def printm():
    process = psutil.Process(os.getpid())
    print("Gen RAM Free: " + humanize.naturalsize(
        ↪ psutil.virtual_memory().available ), " I Proc size: " +
        ↪ humanize.naturalsize( process.memory_info().rss))
    print("GPU RAM Free: {0:.0f}MB | Used: {1:.0f}MB | Util {2:3.0f}% |
        ↪ Total {3:.0f}MB".format(gpu.memoryFree, gpu.memoryUsed,
        ↪ gpu.memoryUtil*100, gpu.memoryTotal))
printm()
```