

Instalación y uso
de los lenguajes
Julia, Python y R
y entornos
RStudio JupyterLab
en la nube
Google Cloud

Instalación de R , Python Julia; RStudio y JupyterLab en entorno Google Cloud

El presente documento es un instructivo de como instalar la última versión de Julia, Python y R RStudio Server y Jupyter Lab sobre Ubuntu 21.04 minimal en el entorno Google Cloud corriendo sobre una máquina virtual *preemptible* que representa el costo más bajo de todas al opciones disponibles en la nube a agosto-2021 para el tipo de procesamiento de generación de modelos predictivos.

Las principales alternativas a este servicio son :

- Amazon AWS EC2 <https://aws.amazon.com/ec2/instance-types/>
- Microsoft Azure <https://azure.microsoft.com/en-us/services/virtual-machines/>
- Google Cloud <https://cloud.google.com/compute/docs/instances/>
- IBM Cloud <https://www.ibm.com/cloud/virtual-servers>
- Oracle Cloud <https://www.oracle.com/cloud/compute/>
- Digital Ocean <https://www.digitalocean.com/products/droplets/>
- Alibaba Cloud <https://www.alibabacloud.com/product/computing>

Google Cloud provee máquinas virtuales comunes, y a un precio que se reduce entre 4 y 5 veces en máquinas virtuales preemptible <https://cloud.google.com/compute/pricing> .

Una máquina virtual preemptible <https://cloud.google.com/compute/docs/instances/preemptible> tiene una vida maxima de 24 horas, puede ser apagada por Google en cualquier momento. Esas máquinas preemptibles pueden crearse y viven en la medida que Google tenga poder de computo ocioso en el datacenter donde esta corriendo esa maquina virtual, cosa que suele suceder a la noche de ese datacenter, o los fines de semana.

Tenga en cuenta que lo mas probable es que Google le mate su máquina preemptible cada 8/10 horas, siéntase muy afortunado si su máquina vive 24 horas. Esto le demandará estar muy atento, verificando periódicamente cuales máquinas virtuales le han apagado. No subestime lo dicho en este párrafo.

Se puede ver las locaciones de los datacenters en <https://cloud.google.com/compute/docs/regions-zones/regions-zones> y una aplicación que muestra en tiempo real el dia y la noche es <https://www.timeanddate.com/worldclock/sunearth.html>

Vale la pena leer esta documentacion <https://cloud.google.com/compute/docs/concepts>
Los alumnos más técnicos podrian leer sobre esta alternativa <https://blog.codinghorror.com/the-cloud-is-just-someone-elses-computer/>

Se recomienda enfaticamente que configure el idioma ingles en Google Cloud durante toda la instalación, ya que la versión en español de Google Cloud es, al menos ... ambigua.

Esta es la razón por la cual se elige usar máquinas virtuales con sistema operativo Ubuntu de tipo preemptible:

Google Cloud, costo por hora en <u>dólares</u>				
maquina virtual de 32 vCPU y 32 GB de RAM, 30GB HDD, US Central1				
	Linux			Windows
	Ubuntu	SUSE	Red Hat	
Preemptible <i>mortal</i>	0.239	0.349	0.369	1.715
Normal <i>inmortal</i>	0.793	0.903	0.923	2.269

De la siguiente tabla se observa que la cantidad de vCPU y memoria RAM están ligadas entre si en lo que cobra Google Cloud .

Notar que si por ejemplo se necesita correr un proceso con una demanda 32 GB de memoria RAM, entonces conviene una máquina virtual con 4vCPU antes que con 2 o con 1vCPU.

Google Cloud, Virtual Machine, Ubuntu, Preemptive , 30GB HDD, costo en centavos Us\$ por hora							
	GB RAM memory						
vCPU	4	8	16	24	32	48	64
1	1.2	1.7	3.3	4.9	6.5	9.8	13.0
2	1.9	2.2	3.3	4.9	6.5	9.7	12.9
4	3.2	3.5	4.3	5.0	6.4	9.6	12.8
8	--	6.2	6.9	7.6	8.3	9.8	12.5
12	--	--	9.6	10.3	11.0	12.4	13.9
16	--	--	12.2	13.0	13.7	15.1	16.5
32	--	--	--	--	24.3	25.7	27.2

¿ Qué genera costo en Google Cloud ?

Máquina Virtual ENCENDIDA, en función de la cantidad de vCPU reservadas, memoria RAM reservada, tipo de disco local SSD o normal, tamaño disco local reservado, tráfico de red saliente real, accesos a disco reales, costo del Sistema Operativo si este es pago (Windows, Linux Red Hat), zona donde es creada .

Se paga aunque no se esté procesando nada .

Se paga aunque el proceso haya terminado.

Máquina Virtual APAGADA, en función del espacio en disco alocado del sistema operativo y datos locales. Es adicional a la imagen.

Almacenar en discos SSD es muy costoso, aunque no se estén utilizando.

Bucket, espacio en disco realmente utilizado, GB por segundo en los que existieron los archivos.

Transferencias salientes del Bucket a otros lugares, ya sea lecturas desde maquinas virtuales en otras regiones de Google, o hacia PCs que residen fuera de Google.

Imágenes de discos según la cantidad de GB de disco utilizada, por más que ninguna máquina virtual esté utilizando dicha imagen.

https://cloud.google.com/compute/all-pricing?hl=en_US&_ga=2.249918022.-1998581785.1564797469

Para que al finalizar la materia Google Cloud jamás cobre nada, se debe proceder de la siguiente forma :

- Borrar todas las imágenes de disco creadas
- Borrar todos los datos dentro de los Buckets creados (o el único Bucket que se creó)
- Eliminar todos los Buckets creados
- Apagar todas la máquinas virtuales creadas
- Eliminar todas las máquinas virtuales creadas
- En caso de haber creado algún otro objeto en Google Cloud que no fue mencionado en la materia, eliminarlo .

Al comienzo y por única vez vamos a crear una imagen de sistema operativo Ubuntu, con R y RStudio instalados, y todos los paquetes necesarios.

Esa imagen es la que van a utilizar siempre TODAS las máquinas virtuales.

Una máquina virtual esta definida por lo siguiente

1. La imagen del disco previamente creada (SO, programas instalados, configuraciones)
2. La cantidad de vCPU
3. La cantidad de memoria RAM
4. El tamaño del disco local (que debe ser superior al tamaño de la imagen)
5. La disponibilidad de la máquina virtual (Preemptible o normal)
6. La ubicación geográfica

Si se apaga y vuelve a encender una máquina virtual, sigue estando disponible lo que quedó en el disco local. Si se borra una máquina virtual, se pierde toda la información del disco local.

Por la arquitectura de la solución, el disco local de las máquinas virtuales solamente se utiliza para almacenar información temporaria que selectivamente es copiada al bucket de datos.

El bucket de datos, sería algo como un “disco de red” al que todas las máquinas virtuales pueden acceder, leer y escribir al mismo tiempo. También podemos manualmente subirle información desde nuestras PCs por medio del Google Cloud Console.

Utilizamos el bucket de datos para:

- Almacenar los datasets, que son leídos por los programas R que corren en las máquinas virtuales. `~/buckets/b1/datasetsOri` y `~/buckets/b1/datasets`
- Almacenar los resultados intermedios y finales de cada una de las corridas, `~/buckets/b1/work` `~/buckets/b1/kaggle`

Es decir, en nuestra arquitectura la entrada y la salida siempre queda en el bucket.

Se podría decir que la copia central del bucket está ubicada en USA.

La información del bucket de datos es persistente y totalmente independiente de las máquinas virtuales. Por más que se apague o incluso borre una máquina virtual, la información del bucket de datos permanece disponible, y es accesible por medio de Google Cloud Console. No hace falta que una máquina virtual esté encendida para acceder al contenido del bucket.

Va a ser muy común tener varias máquinas virtuales corriendo al mismo tiempo, cada una corriendo un *experimento* distinto en distintos datacenters del planeta, y todas ellas leyendo y escribiendo del mismo bucket de datos al mismo tiempo.

Google busca constantemente su beneficio tentando a sus clientes para luego facturarles más de lo que ellos esperan.

Esta es la historia detrás de los USD 300 gratuitos que duran un máximo de 3 meses, y una forma de "ganarle" a Google.

El uso de los USD 300 tiene ciertas restricciones, que son estas

<https://cloud.google.com/free/docs/gcp-free-tier>

Es decir, tiene la limitación que solo se pueden tener corriendo al mismo tiempo 8 vCPU, ya sean todas en una sola máquina virtual o distribuidas a lo largo de varias.

Aquí hay dos alternativas :

1) Crear una máquina virtual con solo 8 vCPU y vivir en un mundo completamente seguro y feliz, donde jamás Google les va a cobrar nada, y lo único que va a pasar es que una vez consumidos los USD 300 se les borren las máquinas virtuales creadas y particularmente el bucket con todos los datos que tengan dentro, con todas las salidas de las Optimizaciones Bayesianas.

Es un mundo completamente seguro y feliz, que no requiere de atención, solo es necesaria paciencia para esperar a que terminen los procesos.

Una máquina de 16vCPU no funciona 2 veces más rápido que una de 8 vCPU, el escalamiento del procesamiento NO ES LINEAL, ya que los programas por más que estén escritos para correr en paralelo, hay partes inevitables que corren en forma secuencial.

2) Hacer algo más arriesgado que necesita un monitoreo diario:

2.1) Darle autorización a Google para que cuando se pasen de los USD 300, empiece a cobrar. De esa forma podrán crear tener corriendo más de 8 vCPU al mismo tiempo.

2.2) Controlar en forma diaria cuantos USD van gastando, y al llegar a los USD 270, apagar todas las máquinas virtuales y borrarlas, bajarse los datos que quieran del bucket, borrar el Bucket, borrar la imagen del sistema operativo, es decir dejar sin nada cosa que lo que Google Cloud quiera cobrarles por mes sea CERO .

En este caso les quedará la cuenta quizás con USD 15 a favor , y no les cobrará nada jamás.

El umbral de USD 270 es porque hay un delay de como un día entre el consumo que muestra Google Cloud en la página vs el real que terminará facturando.

El punto 2) implica estar alerta, y revisar todo el tiempo el consumo, en particular cuando se pone una máquina grande a funcionar.

Generalmente si el alumno es cuidadoso no hay problema, porque las máquinas preemptibles se mueren solas a las 24 horas, PERO si se crea una máquina normal, una inmortal, y no se controla, esa máquina sigue consumiendo todo el tiempo, durante días o semanas, y ahí si que uno se pasa de los USD 300 y le facturan a la tarjeta de crédito.

Si el alumno no tiene un computer literacy adecuado y no tiene una férrea disciplina de control es preferible que opte por la opción 1) (lo que tiene actualmente), para vivir más tranquilo en un

mundo seguro y feliz. Después de todo, se puede perfectamente hacer todo con 8 vCPU , a lo sumo se esperan mas horas.

Aclaración importante, Google corre un algoritmo para detectar a personas que probablemente luego de terminado el periodo gratis jamás paguen por nuevos servicios, y en esos casos NO permite pasar de las 8 vCPU. No está claro como funciona este algoritmo.

Adicionalmente, alumnos han reportado que haber sido contactados por Atención al Cliente de Google Cloud, en donde les hacen una campaña preventiva de retención de clientes, intentando lograr que luego de agotar los USD 300 pasen al modo pago y se transformen en clientes reales.

Jamás ha sido reportado por un alumno el evento que Google les bloquee una cuenta.

Todos los años, sucede con probabilidad 0.02 (1 en 50), que un alumno pasó al modo desprotegido, no controla diariamente si se olvidó una máquina virtual inmortal encendida, se pasa de los USD 300 gratuitos, y Google le factura a su tarjeta. En la gran mayoría de estos casos han podido revertirlo implorando piedad a Google haciéndose pasar por desvalidos estudiantes de un país del tercer mundo que no entiende muy bien que es la nube ...

Estos son los tiempos de corrida en Google Cloud en función de la cantidad de vCPU. Observar que costo óptimo está cuando se utilizan aproximadamente 4 o 6 o 8 vCPU

Como se ve, **no** se escala linealmente !

Xgboost elemental				
Al menos 20GB RAM, máquina preemptive				
vCPU	RAM mínima	Costo x hora	Tiempo Corrida (segundos)	Costo Total
1	20.00	0.0410	575	0.0065
2	20.00	0.0410	418	0.0048
4	20.00	0.0470	245	0.0032
6	20.00	0.0610	188	0.0032
8	20.00	0.0750	157	0.0033
10	20.00	0.0890	143	0.0035
12	20.00	0.1030	127	0.0036
14	20.00	0.1170	121	0.0039
16	20.00	0.1310	110	0.0040
18	20.00	0.1450	87	0.0035
20	20.00	0.1590	88	0.0039
24	21.75	0.1890	98	0.0051
32	28.80	0.2410	87	0.0058
48	43.25	0.3760	71	0.0074

Una conclusión: jamas utilice máquinas de más de 16 vCPU, estará desperdiciando dinero.

Los pasos de este instructivo son :

1. Instalar lenguajes y entornos (se hace por única vez)
 1. Crear máquina virtual inmortal
 2. Ingresar a la terminal de la máquina virtual
 3. Crear el bucket por linea de comando
 4. Copiar archivo de la PC local a la máquina virtual
 5. Instalar (70 minutos, corre desatendido)

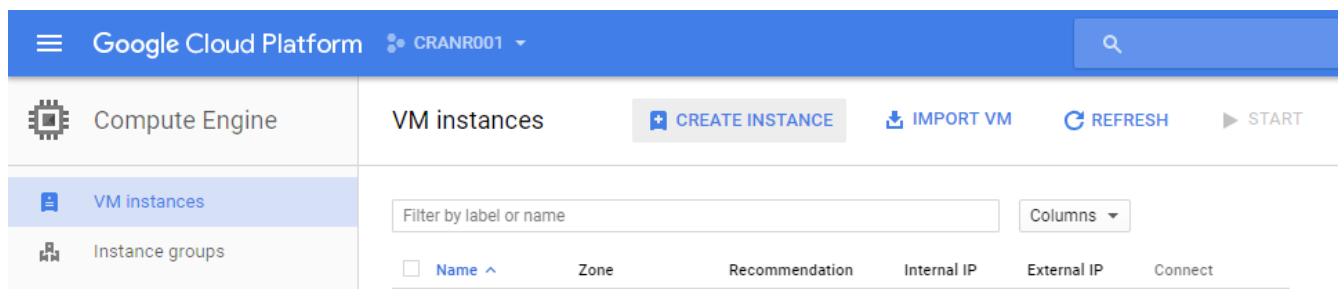
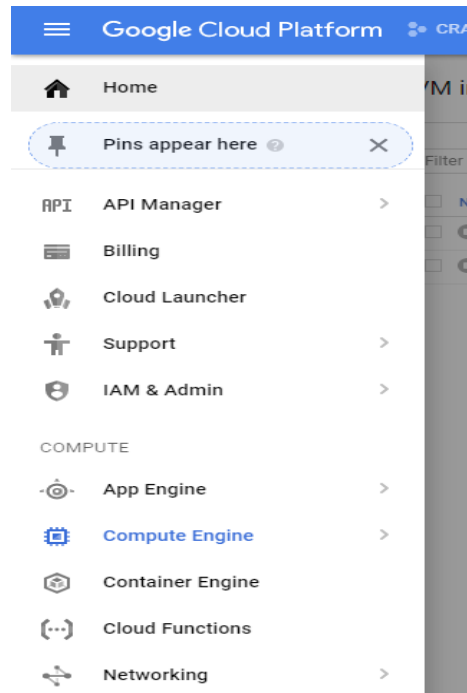
2. Crear Imagen (se hace por única vez)
 1. Cambiar la clave
 2. Verificar datasets
 3. Verificar R
 4. Verificar RStudio
 5. Verificar Jupyter lab
 6. Clonar el repositorio personal
 7. Cerrar la maquina virtual, y crear la *imagen*

3. Ejemplos de como correr un script R
 1. Metodología disciplinada de trabajo
 2. Crear máquina virtual
 3. RStudio en forma remota
 4. Desde una terminal Ubuntu con la consola de R correr scripts
 5. Usar Jupyter Lab en forma remota

Instalación de lenguajes y entornos

1.1 Crear Maquina Virtual

*Por favor, prestar atencion a todo lo que esté encerrado en una **elipse roja** en las impresiones de pantalla, es VITAL seguir esa instrucción.*



Name ?

Name is permanent

instance-instalacion

Labels ? (Optional)

+ Add label

Region ?

Region is permanent

us-east4 (Northern Virginia)

Zone ?

Zone is permanent

us-east4-c

Machine configuration

Machine family

General-purpose

Compute-optimized

Memory-optimized

Machine types for common workloads, optimized for cost and flexibility

Series

E2

CPU platform selection based on availability

Machine type

e2-standard-4 (4 vCPU, 16 GB memory)



vCPUs
4

Memory
16 GB

GPUs
-

⌵ CPU platform and GPU

Confidential VM service ?

☐ Enable the Confidential Computing service on this VM instance.

Container ?

☐ Deploy a container image to this VM instance. [Learn more](#)

Boot disk ?



New 20 GB SSD persistent disk

Image

Ubuntu 21.04 Minimal

Change

Identity and API access ?

Service account ?

Compute Engine default service account

Access scopes ?

☐ Allow default access

☒ Allow full access to all Cloud APIs

☐ Set access for each API

Firewall ?

Add tags and firewall rules to allow specific network traffic from the Internet

☒ Allow HTTP traffic

☐ Allow HTTPS traffic

En la elección del Boot Disk deberá seleccionar estas opciones.

Por favor, siga estas instrucciones al pie de la letra, cualquier otra alternativa a Ubuntu 21.04 Minimal no funcionará, un tamaño menor a 20 GB de disco, fallará y dará crípticos mensajes de error.

Boot disk

Select an image or snapshot to create a boot disk; or attach an existing disk. Can't find wh

Public images	Custom images	Snapshots	Existing disks
---------------	---------------	-----------	----------------

Operating system

Ubuntu

Version

Ubuntu 21.04 Minimal

amd64 hirsute minimal image built on 2021-08-26, supports Shielded VM features

Boot disk type

SSD persistent disk

Size (GB)

20

Por favor, es MUY IMPORTANTE que el disco tenga 20GB de espacio, en caso contrario no se podrá instalar todo.

1.2 Ingresar a la terminal de la maquina virtual recién creada

<input type="checkbox"/> Name ^	Zone	Recommendation	Internal IP	External IP	Connect
<input type="checkbox"/> instance-1	us-west1-c		10.138.0.2	None	SSH ▾ ⋮
<input type="checkbox"/> instance-2	europa-west1-d		10.132.0.2	None	SSH ▾ ⋮
<input type="checkbox"/> instance-3	us-central1-c		10.128.0.3	None	SSH ▾ ⋮
<input checked="" type="checkbox"/> instance-instalacion	us-central1-c		10.128.0.4	35.192.64.53	SSH ▾ ⋮
<input type="checkbox"/> instance-itba	us-central1-c		10.128.0.2	None	SSH ⋮

Open in browser window

Open in browser window on custom port

View gcloud command

Use another SSH client

```
ssh.cloud.google.com/projects/cranr001-150425/zones/us-east4-c/instances/instance-instalacion?authuser=0&hl=

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

0 updates can be applied immediately.

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

gustavo_denicolay@instance-instalacion:~$
```

1.3 Crear bucket por linea de comandos

desde la terminal Ubuntu (dar enter al final)

```
$ gsutil mb gs://buko666
```

Donde **buko666** debe ser un nombre único, no repetido en todo Google Cloud

Si le aparece que ya existe ese nombre de bucket, volver a intentar el mismo comando con tro

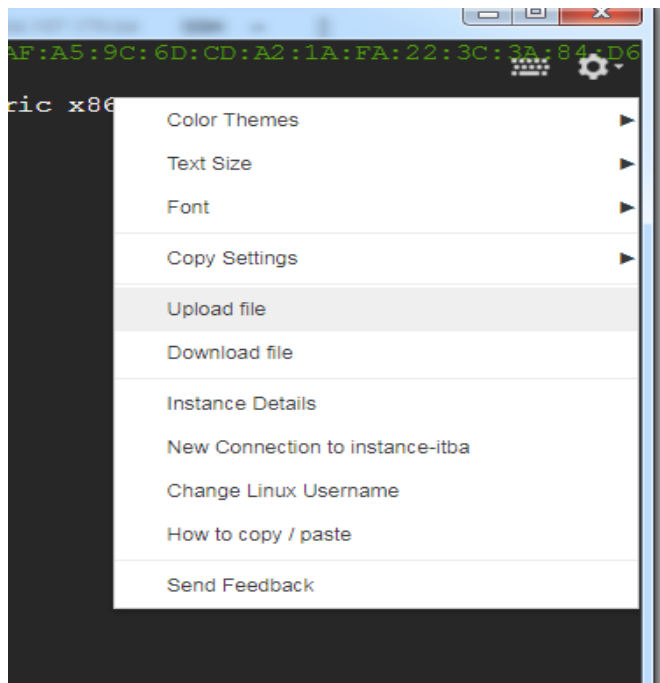
Si le sale el siguiente mensaje, es que debe elegir otro nombre y volver a correr el comando

```
ServiceException: 409 A Cloud Storage bucket named 'buko666' already exists. Try another name. Bucket names must be globally unique across all Google Cloud projects, including those outside of your organization.
```

1.4 Subir archivo de la instalación

Baje a su PC local el archivo `instalar_01.sh` del repositorio GitHub de donde bajó este manual de instalación.

Subir el archivo recién bajado a la máquina virtual



1.5 Instalar

desde la terminal Ubuntu

```
$ chmod +x *.sh  
$ ./instalar_01.sh
```

este paso es completamente automático, puede llevar alrededor de 70 minutos, se verán pasar miles de líneas por la pantalla; algunos alumnos manifestaron sentirse dentro de una Matrix pero es nada más que una humilde, aburrida y típica instalación, la mayor parte de tiempo se compilan los paquetes de R.

Atencion, ante cualquiera de estos eventos usted deberá correr todo desde cero, recreando la maquina virtual

- Si se le corta internet durante la instalación
- Si se le apaga su PC
- Si su PC entra en modo ahorro de energía (no la deje totalmente desatendida, mueva el mouse cada algunos minutos).

Crear
Imagen
(del sistema operativo)

2.1 Asignar password al usuario

desde la terminal

(el \$ inicial en color negro representa el prompt de Ubuntu, y **no** debe ser tipeado)

```
$ cd
```

Cambiar la clave

```
$ ./cambiar_claves.sh
```

aparecerá en la pantalla “New password:” , se debe tipear una password la que **no** es mostrada en pantalla mientras se va tipeando (parece como que esta todo “colgado”). Se debe presionar la tecla Enter el final. Luego solicita nuevamente la password.

Dado que es posible que en algun momento quiera compartir la máquina virtual con un compañero de equipo, se sugiere utilizar una password distinta a sus preferidas. Se piden DOS passwords, una se utiliza para RStudio y la otra para Jupyter Lab, se sugiere que sean las mismas.

Con este usuario y password se ingresará al RStudio, por favor recordar ambos .

2.2 Verificar Datasets

corriendo desde la consola Ubuntu

```
$ ls -l ~/buckets/b1/datasetsOri
```

se deberán observar los siguientes archivos

- paquete_premium_202009.csv
- paquete_premium_202009.csv.gz
- paquete_premium_202011.csv
- paquete_premium_202011.csv.gz

Ir a la página <https://console.cloud.google.com/storage/browser> , hacer click en su bucket, luego en la carpeta datasetsOri, y verificar que se ven los mismos 4 archivos.

2.3 Verificar R

Ejecutar el siguiente comando desde la terminal Ubuntu (la letra R debe estar en mayúsculas)

```
$ R
```

Le deberá aparecer la consola de R

```
gustavo_denicolay@instance-instalacion:~$ R
R version 4.1.1 (2021-08-10) -- "Kick Things"
Copyright (C) 2021 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> █
```

para salir de esta pantalla, debe tipear `quit(save="no")` y luego la tecla ENTER

2.4 Verificar RStudio

Ejecutar el siguiente comando desde la terminal Ubuntu

```
$ sudo rstudio-server status
```

Le deberá aparecer algo de este estilo, donde es fundamental la parte en color verde `active (running)`

Si no le aparece esto, deberá hacer de nuevo la instalación, desde cero.

```
● rstudio-server.service - RStudio Server
   Loaded: loaded (/lib/systemd/system/rstudio-server.service; vendor preset: enabled)
   Active: active (running) since Sat 2021-09-11 09:20:11 UTC; 1min 1s ago
     Process: 20931 ExecStart=/usr/lib/rstudio-server/bin/rserver (code=exited, status=0/SUCCESS)
    Main PID: 20932 (rserver)
      Tasks: 3 (limit: 19193)
     Memory: 2.0M
    CGroup: /system.slice/rstudio-server.service
            └─20932 /usr/lib/rstudio-server/bin/rserver

Sep 11 09:20:11 instance-instalacion systemd[1]: Starting RStudio Server.
Sep 11 09:20:11 instance-instalacion systemd[1]: Started RStudio Server.
```

Para salir de este comando debe presionar la letra `q`

2.5 Verificar Jupyter lab

Ejecutar el siguiente comando desde la terminal Ubuntu

```
$ systemctl status jupyterlab
```

Le deberá aparecer algo de este estilo, donde es fundamental la parte en color verde **active (running)**

```
gustavo_denicolay@instance-instalacion:~$ systemctl --user status jupyterlab
• jupyterlab.service - JupyterLab
   Loaded: loaded (/home/gustavo_denicolay/.config/systemd/user/jupyterlab
   Active: active (running) since Sat 2021-09-11 12:48:33 EDT; 9min ago
   Process: 64753 ExecStartPre=/bin/sleep 10 (code=exited, status=0/SUCCESS)
   Main PID: 64756 (jupyter-lab)
   CGroup: /user.slice/user-1001.slice/user@1001.service/app.slice/jupyterlab
           └─64756 /usr/bin/python3 /home/gustavo_denicolay/.local/bin/jupyterlab

Sep 11 12:48:34 instance-instalacion jupyter-lab[64756]: [I 2021-09-11 12:48:34.123]
Sep 11 12:48:34 instance-instalacion jupyter-lab[64756]: [I 2021-09-11 12:48:34.123]
Sep 11 12:48:34 instance-instalacion jupyter-lab[64756]: [I 2021-09-11 12:48:34.123]
Sep 11 12:48:34 instance-instalacion jupyter-lab[64756]: [I 2021-09-11 12:48:34.123]
Sep 11 12:48:34 instance-instalacion jupyter-lab[64756]: [I 2021-09-11 12:48:34.123]
Sep 11 12:48:34 instance-instalacion jupyter-lab[64756]: [I 2021-09-11 12:48:34.123]
Sep 11 12:48:34 instance-instalacion jupyter-lab[64756]: [I 2021-09-11 12:48:34.123]
Sep 11 12:48:34 instance-instalacion jupyter-lab[64756]: [I 2021-09-11 12:48:34.123]
Sep 11 12:48:34 instance-instalacion jupyter-lab[64756]: [I 2021-09-11 12:48:34.123]
Sep 11 12:48:34 instance-instalacion jupyter-lab[64756]: [I 2021-09-11 12:48:34.123]
lines 1-18/18 (END)
```

Para salir de este comando debe presionar la letra **q**

2.6 Clonar el repositorio personal

Usted debe clonar SU propio repositorio de github, , donde usted trabajará (que NO es el oficial de la materia)

```
$ cd
$ git clone https://github.com/xxxxxx/dmeyf
```

donde **xxxxxx** es SU nombre de usuario de github
y **dmeyf** es el nombre del repositorio de la asignatura

2.7 Crear Imagen

actualizo la distribucion de Ubuntu a la ultima versión

```
$ sudo apt-get --yes update
$ sudo apt-get --yes dist-upgrade
```

aparecerá la siguiente pantalla, escribir 1 2 3 4 5 6 7 y luego ENTER

```
Restarting services...
Daemons using outdated libraries
-----
1. dbus.service           4. polkit.service        7. unattended-upgrades.service
2. networkd-dispatcher.service 5. rstudio-server.service 8. none of the above
3. packagekit.service     6. rsyslog.service

(Enter the items or ranges you want to select, separated by spaces.)

Which services should be restarted? █
```






Finalmente esta instruccion apagara la máquina

```
$ sudo apt-get --yes autoremove
$ sudo poweroff
```

cerrar la terminal (la que se ha puesto con fondo gris) y esperar a que se apague la maquina virtual (90 segundos, no se impaciente), que pase del tilde verde al cuadrado blanco en fondo naranja. Refrescar con F5 el browser para que se actualice.

Tilde verde encendido, la maquina virtual está aun en funcionamiento

Si está el tilde verde, entonces Google nos está facturando por segundo.






<input type="checkbox"/> Name ^	Zone	Recommendation	Internal IP	External IP	Connect
<input type="checkbox"/>  instance-1	us-west1-c		10.138.0.2	None	SSH ▾ ⋮
<input type="checkbox"/>  instance-2	europa-west1-d		10.132.0.2	None	SSH ▾ ⋮
<input type="checkbox"/>  instance-3	us-central1-c		10.128.0.3	None	SSH ▾ ⋮
<input type="checkbox"/>  instance-instalacion	us-central1-c		10.128.0.4	35.192.191.153	SSH ▾ ⋮
<input type="checkbox"/>  instance-	us-central1-c		10.128.0.2	None	SSH ▾ ⋮

el tilde verde ya no está más, la maquina virtual está apagada y NO se factura

VM instances

[+ CREATE INSTANCE](#)[IMPORT VM](#)[REFRESH](#)[▶ START](#)

☰ Filter VM instances

<input type="checkbox"/> Name ^	Zone	Recommendation	Internal IP	External IP	Connect
<input type="checkbox"/>  instance-1	us-west1-c		10.138.0.2	None	SSH ▾ ⋮
<input type="checkbox"/>  instance-2	europa-west1-d		10.132.0.2	None	SSH ▾ ⋮
<input type="checkbox"/>  instance-3	us-central1-c		10.128.0.3	None	SSH ▾ ⋮
<input type="checkbox"/>  instance-instalacion	us-central1-c		10.128.0.4	None	SSH ▾ ⋮
<input type="checkbox"/>  instance-	us-central1-c		10.128.0.2	None	SSH ▾ ⋮

una vez que está apagada vaya al link <https://console.cloud.google.com/compute/images> y presione **[+] CREATE IMAGE** en el menú de la parte superior de la pantalla

Siga las instrucciones de la siguiente página.

La imagen tardará unos interminables 4 minutos en crearse.

Create an image

Name

Name is permanent

image-dm

Source

Disk

Source disk

instance-instalacion

Location

☒ Multi-regional

☐ Regional

us (United States) (default)

Family (Optional)

Description (Optional)

Labels (Optional)

+ Add label

Encryption

Data is encrypted automatically. Select an encryption key management solution.

☒ Google-managed key

No configuration required

☐ Customer-managed key

Manage via Google Cloud Key Management Service

☐ Customer-supplied key

Manage outside of Google Cloud

You will be billed for this image. [Compute Engine pricing](#)

Create

Cancel

Correr Scripts en R

3.1 Metodología disciplinada de trabajo

- Usted tiene su propio repositorio en GitHub
- En su PC local tiene clonado su repositorio.
- Todos los scripts se desarrollan (escriben y/o modifican) en su PC local, no se usa el RStudio de la nube ni tampoco Jupyter Lab de la nube para escribir scripts
- Todos los scripts se prueban primero en su PC local, con un dataset más pequeño que el disponible en Google Cloud, probar significa dejarlos correr varios minutos hasta que empiecen a generar una salida.
- Cuando está seguro que su script funciona hace un commit en su PC local y luego un push a su repositorio personal en GitHub
- Jamás su script escribirá a disco fuera de su `setwd()` “directorio de trabajo”, que en el caso de correr en la nube está en el Google Bucket. Ningun resultado deberá ser guardado jamás en el disco local de la Virtual Machine
- Utilizar las máquinas virtuales preemptibles (mortales) hacen rendir los USD 300 gratuitos como si fueran USD 1200, pero le demandarán un gran esfuerzo de estar atento a que Google Cloud le mate la máquina y usted deba poner a correr el proceso nuevamente.
- Para cada nueva corrida, cada optimización bayesiana, se procede de esta forma:
 - Primero se prueba que el script funciona en la PC local
 - Hace un commit en su PC local, un push a su repositorio personal de GitHub
 - Se crea una nueva máquina virtual preemptible (la de los pobres) con la memoria RAM, espacio en disco, y vCPU adecuados
 - Se ingresa al RStudio de esa máquina virtual
 - Desde la terminal de RStudio hace un pull de su repositorio al disco local de la máquina virtual. También lo puede hacer desde la terminal de Jupyter Lab, y por supuesto desde la terminal Ubuntu
 - Se levanta el script en RStudio desde el disco local de la máquina virtual
 - De ninguna forma usted puede verse tentado a hacer un cambio de ultimo momento en el script y que eso no quede reflejado en el repositorio GitHub, ha decenas de casos de buenas corridas que no se han podido volver a reproducir por no haber guardado la version del script que corrió.
 - Se pone a correr el script, se espera unos minutos para ver que levantó bien el dataset y empezó a escribir en los logs
 - Un tema delicado es saber si usted le asignó la suficiente cantidad de memoria RAM a su máquina virtual. Luego de lanzar a correr su script desde el RStudio, puede ingresar a la terminal Ubuntu, correr el comando `htop` y monitoree el uso de memoria RAM los primeros minutos.
 - Toda optimización Bayesiana demandará sobre varios meses al menos 128GB de memoria

RAM, y si trabaja con los datasets ext o exhist necesitará 256 GB de RAM. Si la memoria RAM es insuficiente, deberá BORRAR esa máquina virtual y volver a crear otra con EL DOBLE DE MEMORIA; no sea tacaño, ir subiendo de apenas unos pocos GB para probar es caminar sobre espinas.

- Monitorear periodicamente que la máquina virtual no fue apagada por Google y que además de estar encendida nuestro script continua corriendo (se puede monitorear desde la app de Google Cloud Console para móviles)
 - Si la máquina se apagó, se vuelve a encender esa misma máquina, y se pone a correr nuevamente el script, los scripts están diseñados para retomar la Optimización Bayesiana desde donde había quedado.
 - Podría llegar a suceder que sea de día en ese datacenter, se estén utilizando todos los recursos, y usted vea que la máquina virtual no se enciende, o se apaga enseguida. En este caso, va a tener que eliminarla, y crear una nueva en otro datacenter de Google.
 - Si la máquina está encendida pero el script ya no esta corriendo se verifica si hemos sido tan afortunados que ya terminó.
 - Si el proceso abortó por algun error, primero debe prepararse un café bien cargado y luego empezará la batalla por entender el motivo y como corregirlo.
 - Si dejó corriendo un proceso durante la noche, al despertar verifique su status.
 - Una vez que está todo terminado, se verifican que los archivos de salida estén en las carpetas kaggle, work, modelitos, y si era un script de feature engineering en la carpeta datasets
 - apagar la máquina virtual
 - verificar que realmente quedó apagada, que no está el tilde verde
 - eliminar la máquina virtual. Por favor, no reutilice las máquinas virtuales, solo encontrará angustia y sufrimiento; para cada corrida debe crear una nueva máquina virtual
 - Verifique que la máquina virtual ya no aparece en en VM Instances. Una máquina virtual apagada continua consumiendo dinero, ya que Google cobra por el espacio de disco local.
 - Bajar del Bucket los archivos que se generaron como salida a su PC .
- Los resultados de una corrida se analizan sobre los archivos que se bajó del Bucket a su PC.
 - Asegúrese que todas las máquinas virtuales que cree sea del tipo preemptible, las baratas que siempre se apagan a las 24 horas.
 - Monitoree diariamente que no quedó encendida ninguna máquina virtual que no corresponde.
 - Desestime la recomendación de Google de aumentar la cantidad de vCPU, ellos ven que su

proceso está al 100% y solo quieren facturar más. No le están diciendo que su proceso NO escala en forma lineal con la cantidad de vCPU.

No le de importancia a estos warnings

```
[LightGBM] [Warning] No further splits with positive gain, best gain: -inf
```

(simplemente esta diciendo que no puede construir un nuevo árbol de decisión en el ensemble ya que no son lo suficientemente buenos)

```
[LightGBM] [Warning] Auto-choosing col-wise multi-threading, the overhead of testing was 0.962394 seconds.
```

```
You can set `force_col_wise=true` to remove the overhead.
```

(simplemente esta informando sobre una optimización)

```
[LightGBM] [Warning] verbosity is set=-1, verbose=-1 will be ignored. Current value: verbosity=-1
```

(no hay ningún tipo de problema)

Lo que sigue no es obligatorio correrlo, simplemente muestra diversas formas de correr scripts en R usando las Virtual Machines de Google Cloud, basándose en la imagen recién creada.

3.2 Crear una nueva maquina virtual con mas memoria y cpus

Según el proceso que se deba correr, se especificara la cantidad de vCPU, memoria RAM y espacio en el disco local.

Al crear una máquina virtual para procesar es muy importante crearla con el suficiente espacio en disco local.

Si la máquina virtual no es creada con el suficiente espacio en disco local, el dataset puede llegar a leerse en forma incompleta, resultando en comportamientos extraños y erróneos.

La formula para el espacio es disco es la siguiente :

$\text{MAX}(2 * \text{memoria_ram}, 256\text{GB})$

Ejemplo 1:

se quiere crear una máquina virtual con 80 GB de memoria RAM, entonces el espacio en disco deberá ser de $\text{MAX}(2 * 80\text{GB}, 256\text{GB}) = \text{MAX}(160\text{GB}, 256\text{GB}) = 256\text{GB}$

Ejemplo 2:

se quiere crear una máquina virtual con 300 GB de memoria RAM, entonces el espacio en disco deberá ser de $\text{MAX}(2 * 300\text{GB}, 256\text{GB}) = \text{MAX}(600\text{GB}, 256\text{GB}) = 600\text{GB}$

Si usted especifica poco espacio en el disco local, los scripts le fallarán misteriosamente, ya que posiblemente no puede cargarse completamente el dataset. Esta situación es muy común en alumnos

El inicio elegir lo siguiente

Name ?
Name is permanent

instance-xgboost

Labels ? (Optional)

+ Add label

Region ?
Region is permanent

us-east4 (Northern Virginia)

Zone ?
Zone is permanent

us-east4-c

Esta es la pantalla que aparecerá cuando se debe cambiar el Boot Disk

Boot disk

Select an image or snapshot to create a boot disk; or attach an existing disk. Can't find what you're looking for? See [hundreds of VM solutions in Marketplace](#).

Public images Custom images Snapshots Existing disks

Show images from

CRANR001

☐ Show deprecated images

Image

image-dm

Created on 2020. May 28. 23:32:21

Boot disk type

Standard persistent disk

Size (GB)

256


Esta pantalla aparecerá cuando elija el tamaño de la máquina virtual

Machine configuration

Machine family

General-purpose Compute-optimized Memory-optimized GPU

Machine types for common workloads, optimized for cost and flexibility

Series 

N2

SECOND GENERATION

E2
CPU platform selection based on availability

☒ N2
Powered by Intel Cascade Lake CPU platform

N2D
Powered by AMD EPYC Rome CPU platform

FIRST GENERATION

N1
Powered by Intel Skylake CPU platform or one of its predecessors

Co

Machine configuration

Machine family

General-purpose Compute-optimized Memory-optimized GPU

Machine types for common workloads, optimized for cost and flexibility

Series

N2

Powered by Intel Cascade Lake CPU platform

Machine type ←

Custom

☒ Custom
Select vCPU cores and memory

Standard

- n2-standard-2
2 vCPU, 8 GB memory
- n2-standard-4
4 vCPU, 16 GB memory
- n2-standard-8
8 vCPU, 32 GB memory
- n2-standard-16
16 vCPU, 64 GB memory
- n2-standard-32
32 vCPU, 128 GB memory
- n2-standard-48
48 vCPU, 192 GB memory
- n2-standard-64
64 vCPU, 256 GB memory
- n2-standard-80
80 vCPU, 320 GB memory

Custom es el primero de la lista arriba de todo !

Series

N2

Powered by Intel Cascade Lake CPU platform

Machine type

Custom

Cores

8 vCPU 2 - 80

Memory

128 GB 4 - 640

☒ Extended memory ?

Marcar Extended Memory

Así quedara la primer parte

Name ?
Name is permanent
instance-xgboost

Labels ? (Optional)
[+ Add label](#)

Region ?
Region is permanent
us-east4 (Northern Virginia)

Zone ?
Zone is permanent
us-east4-c

Machine configuration

Machine family
General-purpose Compute-optimized Memory-optimized
Machine types for common workloads, optimized for cost and flexibility

Series
N2
Powered by Intel Cascade Lake CPU platform

Machine type
Custom

Cores
8 vCPU 2 - 80


Memory
128 GB 4 - 640

☒ **Extend memory** ?

⌵ **CPU platform and GPU**

Confidential VM service ?
☐ Enable the Confidential Computing service on this VM instance.

Container ?
☐ Deploy a container image to this VM instance. [Learn more](#)

Boot disk ?
 **New 256 GB standard persistent disk**
Image
image-dm [Change](#)

Identity and API access ?

Service account ?

Compute Engine default service account

Access scopes ?

☐ Allow default access

☒ Allow full access to all Cloud APIs

☐ Set access for each API

Firewall ?

Add tags and firewall rules to allow specific network traffic from the Internet

☒ Allow HTTP traffic

☐ Allow HTTPS traffic

deberá hacer click en
[Management, security, disks, networking, sole tenancy](#)
y se le desplegará para que la marque como Preemptible

Metadata (Optional)

You can set custom metadata for an instance or project outside of the server-defined metadata. This is useful for passing in arbitrary values to your project or instance that can be queried by your code on the instance. [Learn more](#)

shutdown-script suicidio.sh

+ Add item

Availability policy

Preemptibility

A preemptible VM costs much less, but lasts only 24 hours. It can be terminated sooner due to system demands. [Learn more](#)

On

On-host maintenance

When Compute Engine performs periodic infrastructure maintenance it can migrate your

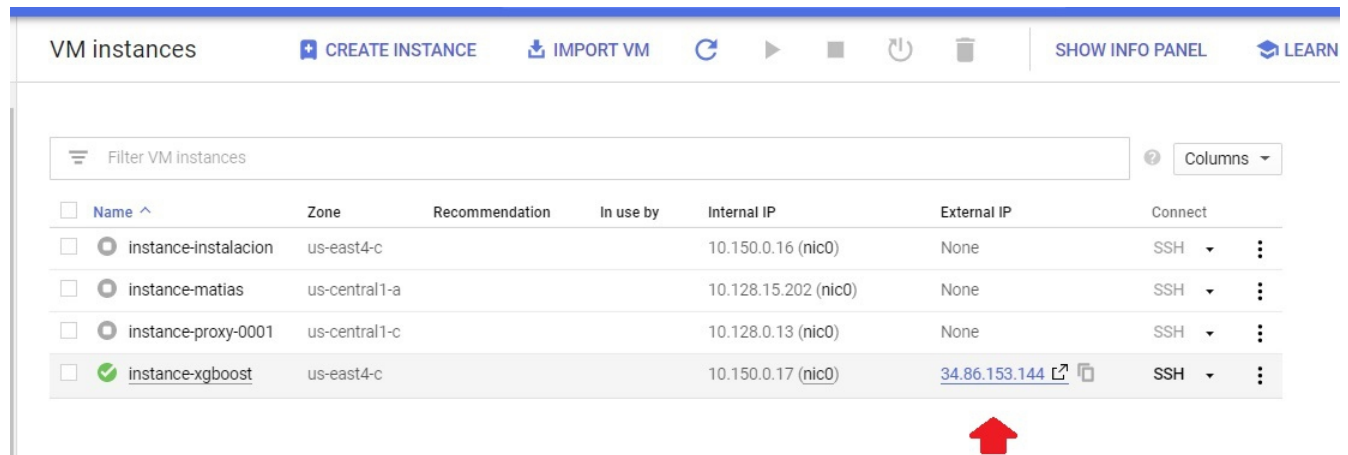
Muy Importante

3.3 RStudio en forma remota

RStudio fue configurado para trabajar con el puerto 80, de forma de poder ser utilizado detrás del firewall de una universidad, empresa, pabellón de cuarentena, etc


Para RStudio se debe utilizar el usuario y claves vistos en el punto 3.1

La forma de ingresar al RStudio es haciendo doble click en la dirección External IP (pública)



<input type="checkbox"/>	Name ^	Zone	Recommendation	In use by	Internal IP	External IP	Connect
<input type="checkbox"/>	instance-instalacion	us-east4-c			10.150.0.16 (nic0)	None	SSH ▾ ⋮
<input type="checkbox"/>	instance-matias	us-central1-a			10.128.15.202 (nic0)	None	SSH ▾ ⋮
<input type="checkbox"/>	instance-proxy-0001	us-central1-c			10.128.0.13 (nic0)	None	SSH ▾ ⋮
<input checked="" type="checkbox"/>	instance-xgboost	us-east4-c			10.150.0.17 (nic0)	34.86.153.144 ↗ 📄	SSH ▾ ⋮

y se abrirá en el browser de la PC local la página de ingreso al RStudio



Sign in to RStudio

Username:

Password:

☐ Stay signed in

Sign In

Es muy importante notar, que el acceso a RStudio es mediante una conexión insegura del tipo `http://` y NO mediante la conexión segura de `https://`

Dado que en el año 2021 hay algunos browsers que por default fuerzan una conexión segura, puede llegar a ser necesario desde el browser poner la siguiente url

<http://34.86.153.144/> (reemplazar por la ip publica real)

Atención, si usted pudo crear esta máquina virtual de prueba, y le funcionó el RStudio, ya está en condiciones de borrar definitivamente la máquina virtual **instance-instalacion**

Una vez dentro, navegar a la carpeta del repositorio, buscar el script, y comenzar a ejecutarlo.

3.4 Desde una terminal Ubuntu con la consola de R correr scripts

No vaya por este camino a menos que usted tenga más de 3000 (tres mil) horas de experiencia en programación generadas en los últimos 5 años.

Esta es una forma de muy bajo nivel para correr un script, se le recomienda permanecer en la zona de confort del RStudio y su hermosa y segura interfase gráfica.

En primera instancia se debe abrir la terminal Ubuntu

luego, para correr desde la terminal se debe tipear (es una sola linea)

```
$ nohup Rscript --vanilla ~/dmeyf/src/rpart/101_PrimerModelo.R .r &
```

(decididamente **vanilla** ni es un script ni un paquete de R)

3.5 Jupyter Lab en forma remota

Desde el navegador de la PC local, utilizando la External IP (pública) de la máquina virtual escribir <http://35.44.55.66:8888/lab>

(reemplazar lo rojo el la IP que corresponda)

notar aqui lo mismo, es una conexión insegura del tipo http:// ya que decididamente NO funciona con https://

Dentro de JupyterLab navegar dentro de la carpeta `dmeyf` hasta donde este el notebook que se desea ejecutar, cargarlo, y ejecutarlo

Han habido casos de alumnos que utilizan la laptop provista por su empleador en donde el puerto 8888 está bloqueado y no tienen privilegios para cambiarlo. **Game Over.**