Universidad Austral - Facultad de Ingeniería
Maestría en Data Mining
Laboratorio de Implementacion I, Clases Gustavo Denicolay

Tarea para el Hogar 2021-09-17

Esta tarea para el hogar está dedicada a todos los alumnos que cursaron con el profesor Gustavo Denicolay la quinta clase de la materia, el viernes 17 de septiembre de 2021, ya sea presencialmente en la sede Cerrito o por Zoom.

La tarea esta narrada como una historia en la que se van probando mejoras a los modelos predictivos con el objetivo de generar cada vez mejores modelos e ir escalando posiciones en Kaggle.
Pasaremos desde los muy simples arboles de decisión sin ningún hiperparámetro hasta el estado del arte el algoritmo LightGBM en donde los hiperparámetros se optimizan con la técnica Bayesian Optimization.

Por ahora, solo entrenaremos en el mes de noviembre-2020 y aplicaremos el modelo a los datos de enero-2021 que es el mes sin clases.
A partir de la clase 6 de la materia, pasaremos a entrenar en la unión de varios meses del pasado y la ganancia en Kaggle aumentará notablemente.

El objetivo de esta tarea para el hogar es:

- Mejorar la ganancia de la predicción, ascender en el ranking.
- Correr arboles de decisión simple
- Presentar el algoritmo Random Forest
- Presentar el algoritmo LightGBM
- Presentar el Feature Engineering
- Presentar la planilla de experimentos llamada Andamios
- Ejercitarse en el manejo de Git y GitHub
- Ejercitarse en correr scripts en Google Cloud, preparándose para las grandes corridas que deberá hacer las próximas semanas.

Todas las corridas deben hacerse Google Cloud, al comienzo de cada script están la cantidad de vCPU, memoria RAM y espacio en disco que necesita cada script.
Para la creación de las máquinas vrituales siga el punto 4.2 del instructivo Google Cloud.

Si usted aún no tiene todo instalado en Google Cloud, por favor comuníquese por Zulip con el profesor Gustavo Denicolay y solicítele ayuda personalizada, ya que estará imposibilitado de correr.

Conceptualmente probaremos combinaciones de lo siguiente:

- Algoritmo
  - Arboles de Decision, algoritmo CART, librería rpart
  - Ensembles con Bagging, algoritmo Random Forest, librería ranger
  - Gradient Boosting of Decision Trees, algoritmo LightGBM, librería lightgbm
- Estimación de la ganancia
  - 5-fold cross validation
- Optimización de Hiperparámetros
  - Bayesian Optimization
- Clase
  - binaria1   1={BAJA+2}   0={BAJA+1, CONTINUA}
  - binaria2   1={BAJA+2, BAJA+1}   0={CONTINUA}
- Data Drifting
  - eliminar combinaciones de las variables `mpasivos_margen`, `mactivos_margen`, `mrentabilidad_annual` y alguna otra que se le ocurra probar.
- Feature Engineering
  - Feature Engineering propuesto por la cátedra
  - Variables de sentido común que se le ocurran a usted y las agregue al script de Feature Engineering
  - Variables que surjan de leer trabajos existentes en internet donde se resuelva un problema parecido.
- Período de entrenamiento
  - En esta tarea únicamente estaremos entrenando en noviembre-2020 [202011]

Usted verá que la planilla de registro de experimentos `andamios.ods` está llena de opciones que no hemos visto en la materia. Son opciones para mejorar los modelos predictivos, y a algunas las veremos en las clases restantes.

Todos los scripts que tienen menos de 100 lineas, usted ya está en condiciones de leerlos, entenderlos, y animarse a realizarles pequeñas modicaciones.

Los scripts mas grandes, que sonlas Optimizaciones Bayesianas, serán explicados en la clase por Zoom del martes 21 de septiembre a las 18:30

Tenga en cuenta que realizar toda esta tarea para el hogar demanda de subir más de 20 predicciones a la plataforma Kaggle, con lo cual no es posible realizarla en un solo dia. Además, tendrá corridas de optimizaciones bayesianas que le demandarán varias horas.

Tiene para entretenerse toda una semana ! Aprenderá que es lo que funciona mejor .

1. Prerrequisito Storage Bucket de Google Cloud

Conéctese en su navegador al Google Cloud Console https://console.cloud.google.com/
Vaya al Cloud Storage Bowser https://console.cloud.google.com/storage/browser

Allí deberá ver su bucket, haga click sobre su bucket y navegando deberá ver el siguiente contenido

- `datasets`
- `datasetsOri`
  - `paquete_premium.csv.gz`
  - `paquete_premium_202011.csv`
  - `paquete_premium_202101.csv`
- `exp`
- `kaggle`
- `log`
- `modelitos`
- `work`

puede que además tenga archivos dentro de la carpeta `work` y la carpeta `kaggle` ya que en la clase del viernes 17 se hizo la corrida de un script
Si no tiene estar carpetas comuníquese por Zulip con el profesor Gustavo Denicolay para ser guiado a la solución.

2. Prerrequisito Imagen de la máquina Virtual

Desde el browser que tiene conectado a Google Cloud vaya al link
https://console.cloud.google.com/compute/images

Debería ver arriba de todo la imagen `image-dm` , le que fue reconstruida durante la clase del viernes 17.
Si no tiene la imagen image-dm comuníquese por Zulip con el profesor Gustavo Denicolay para ser guiado a la solución.

3. Actualización en su PC local de su repositorio

Primero actualice SU repositorio github con el oficial de la materia, si no recuerda como hacerlo siga este instructivo https://docs.github.com/es/github/collaborating-with-pull-requests/working-with-forks/syncing-a-fork

Luego vaya a su PC local y actualice la copia que tiene en su pc local de su repositorio GitHub, con el comando `git pull`
En particular abra en su PC local la planilla `Experimentos.ods` la irá completando con las siguientes corridas, se encuentra en la carpeta `labo2021 / clasesGustavo / TareasHogar / Tarea20210917`

4. Creación de Maquina Virtual para Corridas Rápidas

Siguiendo al detalle el capítulo 4.2 del interminable instructivo de Instalación de Google Cloud, yendo al link https://console.cloud.google.com/compute/instancesAdd cree una máquina virtual con las siguientes características:

Nombre: `instance-impaciente`
Region: `us-east4(Nothern Virginia)`
Zone: `us-east4-c`
Series: `E2`
Machine type: `e2-standard-4 ( 4vCPU, 16 GB memory )`

En Boot Disk elija en Custom Images nuestra querida `image-dm`, con un `Standard Persistent Disk` de `256 GB`

Por supuesto, en Identity and API access debe elegir la opción del medio `Allow full access to all Cloud APIs`

En Firewall debe marcar el cuadrado `Allow Http Traffic`

Y finalmente, en management, security, disks networking and sole tenancy debe elegir en la opción `Preemptibility On`

presione el botón azul Create que está abajo y espere a que en la página https://console.cloud.google.com/compute/instances aparezca su nueva máquina virtual con el tilde verde llamada `instance-impaciente`

Una vez que se encendió el tilde verde de la máquina virtual, ingrese a la terminal Ubuntu presionando el boton SSH que está a la derecha, espere unos 63 segundos a que se abra la pantalla de la terminal Ubuntu, y haga lo siguiente para actualizar el repositorio:
(recuerde que el $ NO se debe tipear)

```
$ cd  labo2021
$ git  pull
```

Finalmente, vuelva a https://console.cloud.google.com/compute/instances y haciendo click en la External IP de la máquina virtual instance-impaciente ingrese a RStudio, recuerde que le pedirá usuario y password, usted ya ha realizado estos pasos en la clase del 17 de septiembre.

Dentro de RStudio, vaya a la carpeta
`labo2021 / clasesGustavo / TareasHogar / Tarea20210917`
allí estarán todos los scripts R que usted deberá correr en esta tarea para el hogar

5. Script `111_rpart_default.r`

Esta es la llamada más trivial posible, un árbol de decision llamado con los parámetros por default.
Es nuestro primer andamio.
Esta corrida dará una ganancia paupérrima, ya que no tiene sentido llamar a un algoritmo con los parámetros por default.

Este script corre en segundos.
En el Rstudio de su máquina virtual vaya primero a Home (icono de la casa, en la solapa Files)
navegue hasta `labo2021/clasesGustavo/TareasHogar/Tarea20210917`
y cargue el script `111_rpart_default.r`

Córralo desde Rstudio de su máquina virtual en Google Cloud
Al terminar habrá generado el archivo `111_rpart_default.csv` en la carpeta `kaggle` del bucket
https://console.cloud.google.com/storage/browser

Baje el archivo `111_rpart_default.csv` a su PC, a una carpeta que habrá llamado kaggle.
Hage el submit de ese archivo a la plataforma Kaggle
Cargue en la planilla `Andamios.ods` la ganancia del Public Leaderboard.
Tenga en cuenta que usted no tendrá disponible el valor del Private Leaderboard ni para este script ni para ninguno de los siguientes.

No apague la máquina virtual, ya que la utilizará para el siguiente script.

6. Script `112_rpart_buenos.r`

Aqui la idea es contar con un buen conjunto de hiperparámetros, que en clase se calcularon con la lentísima técnica de Grid Search.
Esos hiperparámetros son

```
param  <- list( "cp"= -1,
                "minsplit"=   50,
                "minbucket"=  10,
                "maxdepth"=    6 )
```

Este script corre en segundos.
En el Rstudio de su máquina virtual vaya primero a Home (icono de la casa, en la solapa Files)
navegue hasta `labo2021/clasesGustavo/TareasHogar/Tarea20210917`
y cargue el script `112_rpart_bueno.r`

Córralo desde Rstudio de su máquina virtual en Google Cloud
Al terminar habrá generado el archivo `112_rpart_bueno.csv` en la carpeta `kaggle` del bucket
https://console.cloud.google.com/storage/browser

Baje el archivo `112_rpart_bueno.csv` a su PC, a una carpeta que habrá llamado kaggle.
Hage el submit de ese archivo a la plataforma Kaggle
Cargue en la planilla `Andamios.ods` la ganancia del Public Leaderboard.

No apague la máquina virtual, ya que la utilizará para el siguiente script.

7. Data Drifting

En una clase anterior se corrieron dos scripts para comparar las densidades de cada una de los atributos del dataset para 202011 y 202101 , generándose los archivos que se pueden encontrar en el repositorio donde está la presente tarea para el hogar

- `densidades_01.pdf`
- `data_delta_01.pdf`

de esos archivos son algunas de las variables que parecen problemáticas:

- `mrentabilidad_annual`
- `mactivos_margen`
- `mpasivos_margen`

Es posible que a su criterio existan más varibles con data drifting. Si este fuera el caso, modifique los scripts que vendrán para tener en cuenta a esas variables, posiblemente sea una buena oportunidad de sacar ventaja a sus compañeros y subir en el ranking.

Por ahora, se probará solamente quitarlas del dataset-

8. Script `113_rpart_drift_R.r`

El objetivo es correr un experimento para ver si quitando una variable que detectamos con data drifting, mejora la ganancia del modelo.

Se usan los mismos hiperparámetros que en el script 112

```
param  <- list( "cp"= -1,
                "minsplit"=   50,
                "minbucket"=  10,
                "maxdepth"=    6 )
```

pero ahora se elimina la variable `mrentabilidad_annual` del dataset antes de entrenar.

Este script corre en segundos.
En el Rstudio de su máquina virtual vaya primero a Home (icono de la casa, en la solapa Files)
navegue hasta `labo2021/clasesGustavo/TareasHogar/Tarea20210917`
y cargue el script `113_rpart_drift_R.r`

Córralo desde Rstudio de su máquina virtual en Google Cloud
Al terminar habrá generado el archivo `113_rpart_drift_R.csv` en la carpeta `kaggle` del bucket
https://console.cloud.google.com/storage/browser

Baje el archivo `113_rpart_drift_R.csv` a su PC, a una carpeta que habrá llamado kaggle.
Hage el submit de ese archivo a la plataforma Kaggle
Cargue en la planilla `Andamios.ods` la ganancia del Public Leaderboard.

No apague la máquina virtual, ya que la utilizará para el siguiente script.

9. Script `114_rpart_drift_A.r`

El objetivo es correr un experimento para ver si quitando una variable que detectamos con data drifting, mejora la ganancia del modelo.

Se usan los mismos hiperparámetros que en el script 112

```
param  <- list( "cp"= -1,
                "minsplit"=   50,
                "minbucket"=  10,
                "maxdepth"=    6 )
```

pero ahora se elimina la variable `mactivos_margen` del dataset antes de entrenar.

Este script corre en segundos.
En el Rstudio de su máquina virtual vaya primero a Home (icono de la casa, en la solapa Files)
navegue hasta `labo2021/clasesGustavo/TareasHogar/Tarea20210917`
y cargue el script `114_rpart_drift_A.r`

Córralo desde Rstudio de su máquina virtual en Google Cloud
Al terminar habrá generado el archivo `114_rpart_drift_A.csv` en la carpeta `kaggle` del bucket
https://console.cloud.google.com/storage/browser

Baje el archivo `114_rpart_drift_A.csv` a su PC, a una carpeta que habrá llamado kaggle.
Hage el submit de ese archivo a la plataforma Kaggle
Cargue en la planilla `Andamios.ods` la ganancia del Public Leaderboard.

No apague la máquina virtual, ya que la utilizará para el siguiente script.

10. Script `115_rpart_drift_P.r`

El objetivo es correr un experimento para ver si quitando una variable que detectamos con data drifting, mejora la ganancia del modelo.

Se usan los mismos hiperparámetros que en el script 112

```
param  <- list( "cp"= -1,
                "minsplit"=   50,
                "minbucket"=  10,
                "maxdepth"=    6 )
```

pero ahora se elimina la variable `mpasivos_margen` del dataset antes de entrenar.

Este script corre en segundos.
En el Rstudio de su máquina virtual vaya primero a Home (icono de la casa, en la solapa Files)
navegue hasta `labo2021/clasesGustavo/TareasHogar/Tarea20210917`
y cargue el script `115_rpart_drift_P.r`

Córralo desde Rstudio de su máquina virtual en Google Cloud
Al terminar habrá generado el archivo `115_rpart_drift_P.csv` en la carpeta `kaggle` del bucket
https://console.cloud.google.com/storage/browser

Baje el archivo `115_rpart_drift_P.csv` a su PC, a una carpeta que habrá llamado kaggle.
Hage el submit de ese archivo a la plataforma Kaggle
Cargue en la planilla `Andamios.ods` la ganancia del Public Leaderboard.

Enhorabuena, usted deberia habe llegado a los 19M en el Public Leaderboard

No apague la máquina virtual, ya que la utilizará para el siguiente script.

11. Script `116_rpart_drift_AP.r`

El objetivo es correr un experimento para ver si quitando dos variables al mismo tiempo que detectamos con data drifting, mejora la ganancia del modelo.

Se usan los mismos hiperparámetros que en el script 112

```
param  <- list( "cp"= -1,
                "minsplit"=   50,
                "minbucket"=  10,
                "maxdepth"=    6 )
```

pero ahora se eliminan las variables `mactivos_margen` y `mpasivos_margen` del dataset antes de entrenar.

Este script corre en segundos.
En el Rstudio de su máquina virtual vaya primero a Home (icono de la casa, en la solapa Files)
navegue hasta `labo2021/clasesGustavo/TareasHogar/Tarea20210917`
y cargue el script `116_rpart_drift_AP.r`

Córralo desde Rstudio de su máquina virtual en Google Cloud
Al terminar habrá generado el archivo `116_rpart_drift_AP.csv` en la carpeta `kaggle` del bucket https://console.cloud.google.com/storage/browser

Baje el archivo `116_rpart_drift_AP.csv` a su PC, a una carpeta que habrá llamado kaggle.
Hage el submit de ese archivo a la plataforma Kaggle
Cargue en la planilla `Andamios.ods` la ganancia del Public Leaderboard.

No apague la máquina virtual, ya que la utilizará para el siguiente script.

12. Script `511_ranger.r`

Esta es una llamada básica al algoritmo Random Forest implementado por la librería ranger.
Ya sabemos que llamar a un algoritmo con los parámetros quasi default no es muy buena idea.

Este script corre en 6 minutos

En el Rstudio de su máquina virtual vaya primero a Home (icono de la casa, en la solapa Files)
navegue hasta `labo2021/clasesGustavo/TareasHogar/Tarea20210917`
y cargue el script `511_ranger.r`

Córralo desde Rstudio de su máquina virtual en Google Cloud
Al terminar habrá generado el archivo `511_ranger.csv` en la carpeta `kaggle` del bucket
https://console.cloud.google.com/storage/browser

Baje el archivo `511_ranger.csv` a su PC, a una carpeta que habrá llamado kaggle.
Hage el submit de ese archivo a la plataforma Kaggle
Cargue en la planilla `Andamios.ods` la ganancia del Public Leaderboard.

No apague la máquina virtual, ya que la utilizará para el siguiente script.

13. Script `513_ranger_drift_P.r`

Aqui probamos corriendo con los mismos hiperparámetros que el script 511

```
param  <- list( "num.trees"=      500,  #cantidad de arboles
                "mtry"=             sqrt(ncol(dtrain)),
                "min.node.size"=    1,  #hoja mas chica
                "max.depth"=        0   # 0 significa profundidad infinita
              )
```

pero esta vez eliminando del dataset la variable mpasivos_margen

Este script corre en 6 minutos

En el Rstudio de su máquina virtual vaya primero a Home (icono de la casa, en la solapa Files)
navegue hasta `labo2021/clasesGustavo/TareasHogar/Tarea20210917`
y cargue el script `513_ranger_drift_P.r`

Córralo desde Rstudio de su máquina virtual en Google Cloud
Al terminar habrá generado el archivo `513_ranger_drift_P.csv` en la carpeta `kaggle` del bucket
https://console.cloud.google.com/storage/browser

Baje el archivo `513_ranger_drift_P.csv` a su PC, a una carpeta que habrá llamado kaggle.
Hage el submit de ese archivo a la plataforma Kaggle
Cargue en la planilla `Andamios.ods` la ganancia del Public Leaderboard.

No apague la máquina virtual, ya que la utilizará para el siguiente script.

14. Script `611_lightgbm_default.r`

Este script corre en segundos.

Este script llama a lightgbm con los hiperparámetros por default.
Primero entienda en detalle lo que hace el script.
Córralo en Google Cloud.
El script genera el archivo `611_lightgbm.csv` en la carpeta kaggle, búsquelo desde el bucket https://console.cloud.google.com/storage/browser, bájelo a su pc local, y finalmente súbalo a la página Kaggle de la competenencia y vea la ganancia.
Registre los resultados en la planilla Andamios

No apague la máquina virtual, ya que la utilizará para el siguiente script.

15. Script `612_lgb_drift_P.r`

Este script corre en segundos.

Entienda lo que hace, ya que está eliminando la variable `mpasivos_margen` al incio de la corrida.

El script genera el archivo `612_lgb_drift_P.csv` en la carpeta kaggle, búsquelo desde el bucket https://console.cloud.google.com/storage/browser, bájelo a su pc local, y finalmente súbalo a la página Kaggle de la competenencia y vea la ganancia.
Usted debería estar pasando los **20M** en el Public Leaderboard en esta corrida.
Registre los resultados en la planilla Andamios

No apague la máquina virtual, ya que la utilizará para el siguiente script.

16. Script `613_lgb_drift_AP.r`

Este script corre en segundos.

Entienda lo que hace, ya que está eliminando la variable `mpasivos_margen` y `mactivos_margen` al incio de la corrida.

El script genera el archivo `613_lgb_drift_AP.csv` en la carpeta kaggle, búsquelo desde el bucket https://console.cloud.google.com/storage/browser, bájelo a su pc local, y finalmente súbalo a la página Kaggle de la competenencia y vea la ganancia.
Registre los resultados en la planilla Andamios
No apague la máquina virtual, ya que la utilizará para el siguiente script.

17. Script `615_lgb_hero.r`

Este script corre en segundos.

LightGBM es llamado con estos hiperparámetros

```
param=  list( objective= "binary",
              max_bin= 31,
              learning_rate= 0.02,
              num_iterations= 200,
              feature_fraction= 1,
              num_leaves= 100,
              min_data_in_leaf= 2500 )
```

Entienda lo que hace, ya que está eliminando la variable `mpasivos_margen` y `mactivos_margen` al incio de la corrida.

El script genera el archivo `615_lgb_hero.csv` en la carpeta kaggle, búsquelo desde el bucket https://console.cloud.google.com/storage/browser, bájelo a su pc local, y finalmente súbalo a la página Kaggle de la competenencia y vea la ganancia.
Registre los resultados en la planilla Andamios

Usted está cerca de los 22M en el Public Leaderboard

Ahora ya puede apagar y eliminar la máquina virtual , han finalizado las corridas rápidas.

Ahora en adelante se correrán Optimizaciones Bayesianas de varias horas.

18. Script `371_rpart_BO.r`

Aqui haremos una optimización bayesiana de arboles de decision y veremos cual es la máxima ganancia que pueden alcanzar los árboles en este dataset.

Este script corre en varias horas.

Siguiendo al detalle el capítulo 4.2 del interminable instructivo de Instalación de Google Cloud, yendo al link https://console.cloud.google.com/compute/instancesAdd cree una máquina virtual con las siguientes características:

Nombre: `instance-371-rpart-BO`
Region: `us-east4(Nothern Virginia)`
Zone: `us-east4-c`
Series: `E2`
Machine type: `e2-standard-4 ( 4vCPU, 16 GB memory )`

En Boot Disk elija en Custom Images `image-dm`, con un `Standard Persistent Disk` de `256 GB`

Por supuesto, en Identity and API access debe elegir la opción del medio `Allow full access to all Cloud APIs`

En Firewall debe marcar el cuadrado `Allow Http Traffic`

Y finalmente, en management, security, disks networking and sole tenancy debe elegir en la opción `Preemptibility On`

Ingrese al script y cambie:

- La semila por SU primer semilla, en `ksemilla_azar <- 102191 #Aqui poner la propia semilla`

Correr el script, notará que en la carpeta kaggle van escribiendo salidas, subirlas a medida que se van generando a Kaggle fijarse cual es la mejor ganancia que obtiene.
Copie el archivo log y los archivos Kaggle a su PC local, como resguardo.

Apague y elimine la máquina virtual

Registre en una planilla Andamios los resultados de este experimento, tanto los de Kaggle, como el cross validation que queda en el archivo log.

19. Script 571_ranger_BO.r

Aqui haremos una optimización bayesiana de Random Forest y veremos cual es la máxima ganancia que pueden alcanzar en este dataset.

Este script demorará más de 1 día en correr, puede dejar esta corrida para el final.

Siguiendo al detalle el capítulo 4.2 del interminable instructivo de Instalación de Google Cloud, yendo al link https://console.cloud.google.com/compute/instancesAdd cree una máquina virtual con las siguientes características:

Nombre: `instance-571-ranger`
Region: `us-east4(Nothern Virginia)`
Zone: `us-east4-c`
Series: `E2`
Machine type: `e2-highcpu-8 ( 8vCPU, 8 GB memory )`

En Boot Disk elija en Custom Images `image-dm`, con un `Standard Persistent Disk` de `256 GB`

Por supuesto, en Identity and API access debe elegir la opción del medio `Allow full access to all Cloud APIs`

En Firewall debe marcar el cuadrado `Allow Http Traffic`

Y finalmente, en management, security, disks networking and sole tenancy debe elegir en la opción `Preemptibility On`

Ingrese al script y cambie:
La semila por SU primer semilla, en `ksemilla_azar <- 102191 #Aqui poner la propia semilla`

Correr el script, notará que en la carpeta kaggle van escribiendo salidas, subirlas a medida que se van generando a Kaggle fijarse cual es la mejor ganancia que obtiene.
Copie el archivo log y los archivos Kaggle a su PC local, como resguardo.

Apague y elimine la máquina virtual

Registre en una planilla Andamios los resultados de este experimento, tanto los de Kaggle, como el cross validation que queda en el archivo log.

20. Script `671_lgb_binaria1.r`

Este script corre en varias horas.

Ingrese al script y cambie:

- La semila por SU primer semilla, en `ksemilla_azar  <- 102191  #Aqui poner la propia semilla`
- Alrededor de la linea 30 cambie a la ruta que usted tiene en su PC, ya sea Windows, Mac o Linux, pero NO toque la ruta de Google Cloud, ya que para todos es la misma
- Ahora, alrededor de la linea 56,
  `campos_malos  <- c( "mpasivos_margen" )   #aqui se deben cargar todos los campos culpables del Data Drifting`
  agrregue alguna otra variable que usted ya indentificó como causante del Data Drifting

Usted deberá crear una máquina virtual específica para este script, fijese que al comienzo del script están los requerimientos de cVPU y memoria RAM.

Correr el script, ir subiendo los archivos de Kaggle y fijarse cual es la mejor ganancia que obtiene.
Copie el archivo log y los archivos Kaggle a su PC local, como resguardo.

Apague y elimine la máquina virtual
Es posible correr este script al mismo tiempo que el script anterior, por supuesto en una máquina virtual distinta.

Registre en una planilla los resultados de este experimento.

21. Script `672_lgb_binaria2.r`

Este script corre en varias horas, de aquí saldrán las mejores ganancias en Kaggle

Ingrese al script y cambie:

- La semila por SU primer semilla, en `ksemilla_azar <- 102191 #Aqui poner la propia semilla`
- Alrededor de la linea 30 cambie a la ruta que usted tiene en su PC, ya sea Windows, Mac o Linux, pero NO toque la ruta de Google Cloud, ya que para todos es la misma

Usted deberá crear una máquina virtual específica para este script, fijese que al comienzo del script están los requerimientos de cVPU y memoria RAM.

Correr el script, ir subiendo los archivos de Kaggle y fijarse cual es la mejor ganancia que obtiene. Copie el archivo log y los archivos Kaggle a su PC local, como resguardo.

Apague y elimine la máquina virtual

Registre en una planilla los resultados de este experimento.

22. Script `610_fe_simple.r`

Este script corre en menos de 15 minutos, sin embargo le llevará horas de su materia gris agregarle nuevas variables.

Antes que nada, lea en detalle el diccionario de datos del dataset de la asignatura.

Ingrese al script y léalo con gran atención, es muy facil de seguir.
Piense usted variables nuevas que le gustaría agregar al dataset y agréguelas en el script, aquí es donde empieza la magia.

Busque en internet artículos, tésis de maestria que hablen sobre el churn o attrition de clientes bancarios, y COPIE ideas de variables que otros encontraron como relevantes. Agregue esas variables al script 610

Los scripts de las corridas anteriores generan en la carpeta work unos archivos del tipo *imp*.txt los que tienen la importancia de variables.
No hace falta que entienda que son las columnas, le alcanza con saber que las variables están ordenadas por importancia, las más importantes son las que aparecen primero en el archivo.
Analícelos y cree variables nuevas que sean la combinación de las variables que aparecen como más importantes.

Toda variable nueva debe ser agregada al script `610_fe_simple.r`

Este ejercicio es tremendamente difuso, por favor no caiga en ataque de panico. Experimente !
Aquí es donde usted podrá diferenciarse de sus compañeros de curso gracias a su ingerio.

Finalmente, debe correr el script `610_fe_simple.r`

Este script escribirá estos archivos en la carpeta `datasets`

- `paquete_premium_202011_ext.csv`
- `paquete_premium_202101_ext.csv`

verifique que dichos archivos se generaron

23. Script `682_lgb_binaria2.r`

Este script corre en varias horas.

Copia el script `672_lgb_binarai2.r` a `682_lgb_binarai2.r`

Ingrese al script y cambie:

- Reemplace la linea
  `karch_generacion  <- "./datasetsOri/paquete_premium_202011.csv"`
  por
  `karch_generacion  <- "./datasets/paquete_premium_202011_ext.csv"`

- Reemplace la linea
  `karch_aplicacion  <- "./datasetsOri/paquete_premium_202101.csv"`
  por
  `karch_aplicacion  <- "./datasets/paquete_premium_202101_ext.csv"`

- Reemplace la linea kscript <- "672_lgb_binaria2"
  por kscript <- "682_lgb_binaria2"

- Reemplace la semila por SU primer semilla, en `ksemilla_azar  <- 102191  #Aqui poner la propia semilla`

- Alrededor de la linea cambie a la ruta que usted tiene en su PC, ya sea Windows, Mac o Linux, pero NO toque la ruta de Google Cloud, ya que para todos es la misma

- Ahora, alrededor de la linea 58,
  `campos_malos  <- c( "mpasivos_margen" )  #aqui se deben cargar todos los campos culpables del Data Drifting`
  agrregue alguna otra variable que usted ya indentificó como causante del Data Drifting

Correr el script, ir subiendo los archivos de Kaggle y fijarse cual es la mejor ganancia que obtiene.
Copie el archivo log y los archivos Kaggle a su PC local, como resguardo.

Apague y elimine la máquina virtual
Es posible correr este script al mismo tiempo que el script anterior, por supuesto en una máquina virtual distinta.

Registre en una planilla los resultados de este experimento.