ITBA
Especialización en Ciencia de Datos
Data Mining 2021B

Tarea para el Hogar 2021-09-15

El objetivo de esta tarea para el hogar es:

- Mejorar la ganancia de la predicción, ascewnder en el ranking.
- Presentar el algoritmo LightGBM
- Presentar el Feature Engineering
- 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 y fue visto en gran detalle en la clase del miércoles 15 de septiembre.

En la misma infinita clase del miércoles 15 de septiembre, usted ya ha visto como es el manejo entre:

- el repositorio GitHub de la Materia
- su repositorio GitHub
- la copia a su disco local de su repositorio GitHub y los cambios que hace en el, y como los sube a GitHub
- como actualiza su repositorio en la máquina virtual

Conceptualmente probaremos combinaciones de lo siguiente:

- 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.

1. Script `src/lightgbm/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 `lightgbm_611.csv` en la carpeta kaggle, súbalo y vea la ganancia.
No apague la máquina virtual, ya que la utilizará para el siguiente script.

2. Script `src/lightgbm/612_lightgbm_default.r`

Este script corre en segundos.
Este script llama a lightgbm con los hiperparámetros por default salvo que ahora se utiliza

- `min_data_in_leaf= 4000`
- `se elimina la variable "mpasivos_margen"`

Primero entienda en detalle lo que hace el script, córralo en Google Cloud.
El script genera el archivo `lightgbm_612.csv` en la carpeta kaggle, súbalo y vea la ganancia.

¿Ha mejorado la ganancia en Kaggle respecto al script anterior?
No apague la máquina virtual, ya que la utilizará para el siguiente script.

3. Script `src/lightgbm/613_lightgbm_default.r`

Este script corre en segundos.
Este script llama a lightgbm con los hiperparámetros por default salvo que ahora se utiliza

- `min_data_in_leaf= 4000`
- `se eliminan la variablen "mpasivos_margen"  "mactivos_margen"`

Primero entienda en detalle lo que hace el script, córralo en Google Cloud.
El script genera el archivo `lightgbm_613.csv` en la carpeta kaggle, súbalo y vea la ganancia.

¿Ha mejorado la ganancia en Kaggle respecto al script anterior?
Ahora SI puede apagar y eliminar esta máquina virtual.

4. Script `src/lightgbm/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

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.

Por favor, no se angustie por el siguiente mensaje.
`[LightGBM] [Warning] No further splits with positive gain, best gain: -inf`
Sino puede vivir sin conocer el significado, busquelo en internet.

5. Script `src/lightgbm/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

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.

6. Script `src/lightgbm/672_lgb_binaria2.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

   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.

7. Script `src/lightgbm/672_lgb_binaria2.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 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.

8. Script `src/FeatureEngineering/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

9. Script `src/lightgbm/672_lgb_binaria2.r`

Este script corre en varias horas.

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 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.