Universidad Austral
Maestría en Ciencia de Datos
Laboratorio de Implementación I , 2023 Rosario

Tarea para el Hogar TRES

Esta es la última Tarea para el Hogar, luego seguirán los Experimentos Colaborativos en donde alumnos organizados en grupos optimizarán distintas etapas del workflow y colectivamente se encontrarán las mejores prácticas, que se aplicarán directamente a la entrega final de la Competencia Kaggle. Por último el video de 5 minutos.

Todo lo que hacemos es para mejorar la ganancia de la predicción en los datos nuevos; los Experimentos Colaborativos tienen como objetivo mejorar la ganancia, decididamente no son experimentos académicos.

Usted terminará de ver todos los videos referentes al workflow de trabajo y hará una primer corrida del workflow completo.
Llevará las clases 06 y 07 entender conceptualmente la funcionalidad de cada etapa del workflow, sus parámetros y la forma en que cada etapa afecta a las que siguen.
La determinación de los parámetros óptimos de cada etapa será empírica ya que depende de las caracteristicas de estos datos y de la correlación entre las variables independientes y la clase.
Los alumnos realizarán minuciosos experimentos formando grupos en la actividad *Experimentos Colaborativos* para encontrar empíricamente cual es la mejor configuración de cada script, y en su conjunto, lo que demandará a cada grupo *cientos de horas* de procesamiento en la nube.

Deberá cargar los parámetros y resultados de la corrida del workflow en la Planilla Colaborativa https://docs.google.com/spreadsheets/d/1m0jK-JFXdRfc3FNLhpLqhmz_KdN2DHF7WGKJ01d1bFA/edit?usp=sharing en la solapa workflow-inicial
Es la idea que los alumnos prueben configuraciones distintas de los parámetros de los scripts, así la planilla colaborativa se completa con enriquecedoras estrategias, y de esos resultados surgen nuevas ideas para mejorarlos.

En todos los scripts que deba correr, SIEMPRE reemplace las semillas que aparecen como parámetros por sus propias semillas, esté o no indicado expresamente en el ejercicio.

Cuando llegue a los scripts del workflow observe como se conectan entre ellos utilizando `PARAM$exp_input` y `PARAM$experimento`

Sección Deseable

1. Solicitar al profesor acceso al documento compartido de Experimentos Colaborativos
   https://docs.google.com/presentation/d/1yvSYokD43JTwA6mC_BcI54Y-oPrmoGEaycNhgo0vBDw/edit?usp=sharing

   tiempo humano estimado **1 minuto**

2. Videos Prioritarios clase 06
   ver los videos que aún no ha visto
   https://campusvirtual.austral.edu.ar/course/view.php?id=13075&section=4
   https://campusvirtual.austral.edu.ar/course/view.php?id=13075&section=5
   1. Workflow de trabajo
   2. Catastrophe Analysis
   3. Data Drifting
   4. Feature Engineering Intra Mes
   5. Feature Engineering Histórico
   6. Training Srategy
   7. Hyperparameter Tuning
   8. Etapas Finales

   tiempo humano estimado si debe verlos todos: **80 minutos** (a 1.5x)

3. Lectura de Reglas de Experimentos Colaborativos
   Se ha actualizado el documento pdf El Libro de la Asignatura , vuelva a cargarlo y lea el capítulo “5 Experimentos Colaborativos”
   tiempo humano estimado **10 minutos**

4. Lectura de Google Slides de Experimentos Colaborativos
   Una vez que reciba el email notificándole que ya tiene acceso al Google Slides de Experimentos Colaborativos, léala, reúnase con los compañeros con los que suele hacer grupo, y decida con cual experimento va a participar. Negocie en Zulip con el resto del curso en caso de haber colisiones.
   tiempo humano estimado **30 minutos**

5. Sincronización con el Repositorio Oficial de la Asignatura

   Ingrese a la máquina virtual de Google Cloud llamada `desktop` que está en Sao Paulo y sincronice su repositorio con el oficial de la asignatura, le deberán aparecer dentro de `~/labo2023r/src` las carpetas

   - `CatastropheAnalysis`
   - `workflow-inicial`

   Si además usted posee una copia de su repo en su laptop también sincronícela con el repo oficial

   tiempo humano estimado **10 minutos**

6. Análisis Variables Rotas

   Analizará un problema que el sector de DataWarehouseing ha tenido en la generación de los datos, tal cual se mostró en el video Catastrophe Analysis

   Desde la máquina virtual Desktop que está en Sao Paulo ingrese a RStudio tal cual lo ha hecho antes
   Ya en RStudio abra el script
   `~/labo2023r/src/CatastropheAnalysis/z505_graficar_zero_rate.r`

   Ponga a correr el script
   El proceso demorará alrededor de 20 minutos

   La salida del script queda en su bucket, en la carpeta `~/buckets/b1/exp/CA5050`
   Analice los archivos, en particular `zeroes_ratio.pdf`

   tiempo computacional: **20 minutos**
   tiempo humano estimado : **15 minutos**
   dificultad : **baja**
   creatividad requerida : 1**0%**

7. Modificaciones a scripts para hacer su corrida

   En los siguientes pasos usted modificará los seis scripts en Desktop para luego poder correrlos un una potente máquina virtual-

8. Modificaciones al script de CA Catastrophe Analysis

   Corregirá el problema de las variables rotas (pisadas en cero)

   Hay dos métodos disponibles.

   - El llamado “EstadisticaClasica” al valor que fue pisado en cero para ese mes, lo imputa como el promedio del mes anterior y siguiente.
   - El llamado “MachineLearning” al valor que fue pisado en cero para ese mes, lo imputa con un NA, ha leído bien, somos tan diabólicos que agregamos nulos al dataset.

   Quizas usted piense que esta última estrategia es imposible que funcione, pero recuerde la máxima de esta asignatura:

   Un experimento no se le niega a nadie.

   Habrá un equipo que en Experimentos Colaborativos comparará ambos métodos.
   Siempre podrá usted modificar el código, agregar nuevas formas de corregir las variables que están rotas.

   - En la máquina vitual Desktop haga su copia de trabajo del script
     `~/labo2023r/src/workflow/z611_CA_reparar_dataset.r`
   - Si su nombre pertenece a este conjunto
     { “Eduardo Philipp”, “Marcelo Giordano”, “Jiang Mamani”, “Rodrigo Estevez”, “Maira Lignini”,””Josefina Perez”}

     esta primera vez cambie en la actual donde dice
     - `PARAM$metodo <- "MachineLearning"`
   - por
     - `PARAM$metodo <- “EstadisticaClasica”`

     `( el resto de los alumnos debe elegir o “MachineLearning” o “Ninguno”)`

   - Agregue el nuevo archivo al repositorio, haga el commit, y sincronícelo con su repositorio en GitHub
   - Cargue en la planilla colaborativa sus parametros elegidos

     `tiempo humano estimado : 5 minutos`

9. Modificaciones al script de DR Data Drifting

Intentará corregir el data drifting existente en el dataset.
Habrá un equipo que en Experimentos Colaborativos comparará en detalle como funcionan los métodos disponibles aplicados a este dataset.
Siempre podrá usted modificar el código, agregar nuevas formas de corregir el drifting.

- Haga su copia de
  `~/labo2023r/src/workflow-inicial/z621_DR_corregir_drifting.r`

- Si usted posee una carrera de grado que pertenece a
  `{ Contador, Administración de Empresas, Economia, Ingeniería Industrial }`

  cambie donde dice:
  - `PARAM$metodo  <- "rank_cero_fijo"`
    por
  - `PARAM$metodo  <- "deflacion"`

  en caso contrario, elija alguno de los métodos disponibles

- Agregue el nuevo archivo al repositorio, haga el commit, y sincronícelo con su repositorio en GitHub
- Cargue en la planilla colaborativa sus parametros elegidos

`tiempo humano estimado : **5 minutos**`

10. Modificaciones al script FE Feature Engineering

Agregará nuevas variables históricas al dataset.
Habrá varios equipos que en Experimentos Colaborativos compararán los métodos disponibles.

Se usted posee conocimientos previos, se lo invita a que modifique el código, a agregar nuevas variables históricas, por ejemplo, la derivada segunda.

- Haga su copia de `~/labo2023r/src/workflow-inicial/z631_FE_historia.r`

- Parámetros del script que puede cambiar a gusto
  - `PARAM$lag1` agrega para cada variable el valor del mes anterior
  - `PARAM$lag2` agrega para cada variable el valor de dos meses antes
  - `PARAM$lag3` agrega para cada variable el valor de tres meses antes
  - `PARAM$Tendencias1` agrega para cada variable la pendiente que ajusta por cuadrados minimos el valor de ese mes y los cinco meses anteriores
  - `PARAM$RandomForest` agrega variables nuevas a partir de los arboles de un Random Forest de baja profundidad.
  - `PARAM$CanaritosAsesinos` elimina variables que son menos importantes que los canaritos.

- Agregue el nuevo archivo al repositorio, haga el commit, y sincronícelo con su repositorio en GitHub
- Cargue en la planilla colaborativa sus parametros elegidos

`tiempo humano estimado :` **`5 minutos`**

11. Modificaciones al script TS Training Strategy

Decidirá en que meses se < entrena, valida, testea> y en que meses se realiza el <train_final>
Si coinciden los meses de entrenamiento y validacion aguas abajo la Optimización de Hiperparámetros se realizará utilizando cross validation.

Dados los largos tiempos de procesamiento de la Optimización de Hiperparámetros posterior que depende de la cantidad de meses donde se entrena, habrá varios equipos que en Experimentos Colaborativos realicen experimentos para determinar que es lo que funciona mejor.

Haga su copia del script
`~/labo2023r/src/workflow-inicial/z641_TS_training_strategy.r`

Cambie por *su* semilla el parámetro correspondiente

Generalmente entrenar en más meses genera un mejor modelo predictivo, en la medida que no se incluyan los meses más duros de la pandemia que en Argentina significaron muy estrictas restricciones a la circulación.

- Agregue el nuevo archivo al repositorio, haga el commit, y sincronícelo con su repositorio en GitHub
- Cargue en la planilla colaborativa sus parametros elegidos

`tiempo humano estimado : 5 minutos`

12. Modificaciones al script HT Hyperparameter Tuning

- Haga su copia del script
  `~/labo2023r/src/workflow-inicial/z651_HT_lightgbm.r`

- Cambie el parámetro de la semilla por su semilla

- Agregue el nuevo archivo al repositorio, haga el commit, y sincronícelo con su repositorio en GitHub
- Cargue en la planilla colaborativa sus parametros elegidos

`tiempo humano estimado : 5 minutos`

13. Modificaciones al script ZZ Pasos Finales

- Haga su copia del script
  `~/labo2023r/src/workflow-inicial/z661_ZZ_final.r`

- Cambie en `PARAM$semillas` por sus cinco semillas
- No debe hacer ningun otro cambio

- Agregue el nuevo archivo al repositorio, haga el commit, y sincronícelo con su repositorio en GitHub
- Cargue en la planilla colaborativa sus parametros elegidos

- Las salidas quedan en el bucket `./exp/ZZ6610`
  - Archivos con los modelos de LightGBM en formato binario, **FM** Final Models
    - `modelo_01_xxx.model` y `modelo_02_yyy.model`
  - Archivos con la importancia de variables
    - `impo_01_xxx.txt` y `impo_02_yyy.txt`
  - Archivos con las probabilidades **SC** Scoring
    - `pred_01_xxx.csv` y `impo_02_yyy.txt`
  - Archivos generados para **KA** Kaggle
    - `ZZ6610_01_xxx_09500.csv` al `ZZ5910_01_xxx_11500.csv`
    - `ZZ5910_02_yyy_09500.csv` al `ZZ5910_02_yyy_11500.csv`
    - El `01` y `02` indican que son el mejor y el segundo mejor modelo encontrados en la optimización bayesiana
    - `xxx` e `yyy` son el numero de iteración de esos modelo en la optimización bayesiana
    - `09500`, `10000`, 10500, `11000`,`11500` son la cantidad de estímulos a enviar

tiempo humano estimado : **5 minutos**

14. Modificaciones al script `correr_workflow`

Haga su copia del script
`~/labo2023r/src/workflow-inicial/z601_RUN_correr_workflow.r`

Haga los cambios en las lineas `source()` para que se reflejen sus nombres de scripts, generalmente alcanzará con que elimine la letra z
por ejemplo

```
source( "~/labo2023r/src/workflow-inicial/z611_CA_reparar_dataset.r")
```

quitando la "z" de "z611" pasa a :

```
source( "~/labo2023r/src/workflow-inicial/611_CA_reparar_dataset.r")
```

- Agregue el nuevo archivo al repositorio, haga el commit, y sincronícelo con su repositorio en GitHub

tiempo humano estimado : **8 minutos**

15.Finalmente Corrida del Workflow

Deberá seguir los pasos que están el el documento de Instalación de Google Cloud capítulo *3.4 Crear una máquina virtual desde template*
Cree una virtual machine con 256 GB de memoria RAM y 8 vCPU
Asigne el nombre `workflow01` elija la región `us-west4 (Las Vegas)`

Una vez que se creó la máquina virtual deberá seguir los pasos que están en el capítulo *3.5 RStudio en forma remota* para correr el RStudio en forma remota en la máquina virtual

Una vez que ingresó al RStudio busque su script creado en el paso anterior
`~/labo2023r/src/workflow-inicial/601_RUN_correr_workflow.r`
y córralo linea a linea hasta `TS_training_strategy.r` inclusive

finalmente, ponga a correr de ahí en adelante

Los resultados se irán generando dentro de `~/buckets/b1/exp` en las siguientes carpetas

- `CA6110`
- `DR6210`
- `FE6310`
- `TS6410`
- `HT6510`
- `ZZ6610`

En cada carpeta queda un archivo llamado `output.yml`

Una vez que finalizó todo, desde la máquina virtual de Sao Paulo `desktop` suba a Kaggle los archivos generados que quedan en el ~/buckets/b1/exp/ZZ6610/ , archivos de la forma `ZZ6610_*.csv`

`tiempo humano estimado :` ***20 minutos***
`tiempo computacional :` **12 horas**