ITBA
Especialización en Ciencia de Datos
Data Mining 2021B

Tarea para el Hogar 2021-09-08

1. El dilema de Carla Olmo Descartando Hipotesis Semillas

Carla corrió un grid search con el script `250_crossvalidation.r` , estimando la ganancia con 5-fold cross validation y algo le llamó poderosamente la atención : el que funcionaba mejor en cross validation, su ganador del grid search, le iba muy mal en Kaggle. ¿Por qué se daba este fenómeno?

Durante la clase del miércoles consideremos dos alternativas:

- Fue un problema de la semilla que eligió para el 5-fold cross validation
- Es un problema de data drifting, es decir, algun/algunos campos cambian mucho de valor de noviembre a enero.

Para ello primero vamos a correr el script `src/rpart/260_CarlaOlmo_dilema.r` el que hace Montecarlo Estimation con 20 semillas. Para correr el script se deben cambiar las 20 semillas que están al comienzo del script por otras 20 propias.

param1 = < cp=-1, minsplit=200, minbucket=100, maxdepth=6 >
param2 = < cp=-1, minsplit=50, minbucket=10, maxdepth=6 >

(las ganancias están expresadas en millones de pesos)

Hiperparametros	Ganancia 20 Montecarlo	Public Leaderboard Kaggle
param1	8,981042	14.90453
param2	8,522500	17.87127

Al haber utilizado 20 semillas para la Montecarlo Estimation, descartamos que sea un tema de semillas, realmente está sucediendo que lo que funciona mejor en 20-Montecarlo empeora sospechosamente en Kaggle.

Nuestra hipótesis es que hay un problema de Data Drifting

2. El dilema de Carla Olmo Buscando Data Drifting

Leer en detalle intentando comprender profundamente el funcionamiento, y luego correr los siguientes dos scripts , que dejan el resultado en la carpeta work en formato pdf

- src/DataDrifting/390_graficar_densidades.r
- src/DataDrifting/391_graficar_delta.r

Observando la salida work/data_delta_01.pdf encontramos tres campos sospechosos, en los que los valores para cada cliente se han movido mucho del mes 202011 al mes 202101

- mrentabilidad_annual
- mactivos_margen
- mpasivos_margen

Pues bien, ahora pasamos a ejecutar pruebas sobre que sucede si quitamos algunos (o todos esos campos del dataset)
Se deben correr los siguientes muy artesanales scripts (evitando loops y funciones para no complicarle la vida a Raúl) , primero entendiéndolos, y luego subir a Kaggle las salidas

- src/rpart/261_aplicar.r
- src/rpart/262_quitar_20montecarlo.r

Campos Eliminados	20-Montecarlo param1	20-Montecarlo param2	Kaggle param1	Kaggle param2
mrentabilidad_annual	9.0	8.5	17.4	16.6
mactivos_margen	9.0	8.6	16.9	17.8
mpasivos_margen	8.9	8.8	18.3	19.0
mrentabilidad_annual mactivos_margen	9.0	8.6	17.4	16.6
mrentabilidad_annual mpasivos_margen	8.9	8.9	17.3	19.0
mactivos_margen mpasivos_margen	8.9	8.9	18.5	19.0
mrentabilidad_annual mactivos_margen mpasivos_margen	9.0	9.0	17.7	19.0

Se ha comprobado el data drifting. El modelo funciona mejor en datos nuevos si eliminamos algunas variables porque esas variables pegan saltos de noviembre a enero.

3. Optimización Bayesiana primera

   Correr el script `src/rpart/360_rpart_BO.r` haciéndole los siguientes cambios
   - linea 22, cambiar por la ruta de la PC local
   - linea 40, cambiar por su primer semilla
   - linea 124, poner `mc.cores= 1` si se tiene Windows
   - entre la linea 198 y 199 agregar lo siguiente
     - `dataset[ , mpasivos_margen := NULL ]`

Este script demorará 2 horas
El script va generando salidas en la carpeta `kaggle`, que puede ir subiendo a Kaggle a medida que van apareciendo.
El script va acumulando los resultados en un archivo `log` en la carpeta work

El script será visto en clase, y sus partes explicadas en zero2hero
Usted ha hecho lo siguiente: primero eliminar el campo mpasivos_margen, y luego lanzar una optimización bayesiana que busca los mejores hiperparámetros de rpart para este dataset.

¿ Obtiene buenas ganancias en Kaggle ?

4. Optimizacion Bayesiana segunda

   Correr el script `src/rpart/360_rpart_BO.r` haciéndole los siguientes cambios
   - linea 22, cambiar por la ruta de la PC local
   - linea 40, cambiar por su primer semilla
   - entre la linea 198 y 199 agregar lo siguiente
     - `dataset[ , mpasivos_margen := NULL ]`
     - `dataset[ , mactivos_margen := NULL ]`

   Este script demorará 2 horas
   El script va generando salidas en la carpeta `kaggle`, que puede ir subiendo a Kaggle a medida que van apareciendo.
   El script va acumulando los resultados en un archivo `log` en la carpeta work

¿ Obtiene buenas ganancias en Kaggle ?

5. Clase Binaria Trivial

Esta es una tarea de programación sencilla.
Hasta ahora hemos trabajado siempre con la clase_ternaria que tiene los valores { BAJA+1, BAJA+2, CONTINUA }
Ahora trabajaremos con una clase binaria, que solo tomara dos valores.
La transformación será la siguiente:

Valores Originales	Valores Nuevos
BAJA+2	BAJA+2
BAJA+1	NEGATIVO
CONTINUA	NEGATIVO

lo anterior se logra de la siguiente forma, agregando apenas una sola nueva linea, enseguida después de cargar el dataset

```
dataset[ clase_ternaria!="BAJA+2" ,   clase_ternaria := "NEGATIVO" ]
```

realizar el cambio en el script `src/rpart/360_rpart_BO.r`
tener en cuenta que se debe haber eliminado justo antes el campo `mpasivos_margen` con la instrucción `dataset[ , mpasivos_margen := NULL ]`

finalmente, con este cambio, volver a correr el script `src/rpart/360_rpart_BO.r`

¿ Obtiene buenas ganancias en Kaggle ?

6. Corrida Ciega

   A pesar de no haber visto en clase aún el Algoritmo Random Forest, con el fin de tener ya resultados para poderlos analizar en clase luego que sea explicada la teoría del algotimos, hacer lo siguiente con el script `src/ranger/560_ranger_BO.r`
   procediendo de la siguiente forma

   - Instalar todas las librerías que estan entre las lineas 9 y 19, en particular `ranger` y `randomForest`
   - linea 23, cambiar por la ruta de la PC local
   - linea 39, cambiar por su primer semilla
   - entre la linea 211 y 212 agregar lo siguiente
     - `dataset[ , mpasivos_margen := NULL ]`
     - `dataset[ , mactivos_margen := NULL ]`

Este script será muy pesado, utilizará TODOS los nucleos de su procesador y demorará más de una noche. Córralo lo mas que pueda.

7. Clase Binaria Compleja

Esta es una tarea de programación muy compleja. Si usted está aún intentando entender *zero2hero*, no intente hacer esta tarea en casa ya puede resultar severamente lastimado.

En el ejercicio 5, trabajamos con una clase binaria en la que los BAJA+2 son los positivos, y los BAJA+1 y CONTINUA los negativos.
Sin embargo, desde el punto de vista de lo saludable, los clientes BAJA+1 son los que están en peor estado, van a "fallecer" durante el próximo mes. Luego, un poco menos graves, están los BAJA+2 que fallecen recién dentro de dos mes. Finalmente los CONTINUA van a sobrevivir.

La idea es cambiar el script `src/rpart/360_rpart_BO.r` y ejecutarlo para que durante el entrenamiento trabaje con esta clase binaria

Nueva clase binaria parausar durante el entrenamiento	
Valores Originales	Valores Nuevos
BAJA+2	POS
BAJA+1	POS
CONTINUA	NEG

Sin embargo, cuando usted mida la ganancia en testing, solo sumarán 48750 los "BAJA+2"

Conceptualmente trabajar con esta nueva clase binaria, la que es más natural, presenta un enorme desafío. Ya no es la probabilidad de corte el 0.0.25 debido a que ahora tambien son positivos para el momento del entrenamiento los BAJA+1. La probabilidad de corte será mayor a 0.025, pero como no podemos determinarla, será un nuevo hiperparámetro del algoritmo.

Dejo algunos tips para la tarea:

- Debe cambiar la ruta del setwd() a la ruta de su PC local
- Debe cambiar por su semilla
- En la linea 32 debe agregar un nuevo hiperparámetro a la busqueda bayesiana que sea makeNumericParam("pcorte" , lower= 0.020 , upper= 0.060)

- A continuación de la linea 198 debe agregar lo siguiente:
  - `dataset[ , ganancia:= ifelse( clase_ternaria=="BAJA+2", 48750, -1250) ]`
  - `dataset[ ,  clase_binaria := ifelse( clase_ternaria=="CONTINUA", "NEG","POS") ]`
  - `dataset[ ,  clase_ternaria := NULL ]`

- Por supuesto, debe eliminar los campos mpasivos_margen y mactivos_margen enseguida despues de leer el dataset

- En la funcion Arbol_Simple, la formula del llamado a rpart debe ser "clase_binaria ~ . -ganancia)
- En la funcion ArbolSimple, donde dice 0.025 debe cambiarlo por el nuevo hiperparámetro param$pcorte
- En la funcion ArbolSimple, cambiar la linea
  `ganancia_testing <- sum( data[ fold==fold_test ][ prob_baja2 >0.025, ifelse( clase_ternaria=="BAJA+2", 48750, -1250 ) ] )`
  Por una que en lugar del ifelse tenga el nuevo campo `ganancia`

- En la funcion Estimar Ganancia, reemplazar la formual de la llamada a rpart por
  `"clase_binaria ~ . -ganancia"`

8. Lectura Obligatoria
   Es importante leer el siguiente artículo para prepararse para la clase del lunes 13 de septiembre el algoritmo Random Forest
   https://www.all-about-psychology.com/the-wisdom-of-crowds.html

9. Leer las primeras 5 páginas del paper
   http://www2.math.uu.se/~thulin/mm/breiman.pdf

10. Ver el video de marzo de 2021 https://www.youtube.com/watch?v=06-AZXmwHjo donde Andrew Ng, considerado por muchos la máxima autoridad en ciencia de datos, da su punto de vista sobre centrarse más en los datos que en algoritmos.

11. Bibliografía Bayesian Optimization
    Esta bibliografía es solamente para los alumnos que han solicitado profundizar en el tema:
    - https://towardsdatascience.com/bayesian-optimization-concept-explained-in-layman-terms-1d2bcdeaf12f
    - en el repositorio GitHub la materia, carpeta bibliografia/`HyperparameterOptimization` hay papers y presentaciones. Primero leeria `TakingtheHumanOutoftheLoop.pdf`, luego `HyperparameterOptimization_lecture.pdf`, despues BayesianOptimization_Tutorial.pdf, para finalmente intentar con mlrMBO.pdf