ITESO - Maestría en Sistemas Computacionales

Asesor: J. Guadalupe Olascuaga Cabrera

Co asesor: Luis Fernando Gutiérrez Preciado

Alumno: Mawrer Amed Ramírez Martínez

Reporte de Avance de Trabajo de Obtención de Grado

Asesoría: 26 julio de 2020 – IDI4

**Reporte final de avances:**

* **- Agregar Oversampling - DONE**
* **- Hacer suffle para la selección del sub-set de Training. - DONE**
* **- Agregar Métricas RUC AUC. - DONE**
* **Pruebas con/sin Oversampling – DONE.**

**Con un pipeline completo se generaron funciones para cada uno de los pasos desde la clasificación del dataset, hasta el backtesting. Todo con valores parametrizables en forma de lista para poder hacer multiples ejecuciones. Totos los resultados se guardarán en un CSV con los valores de parámetros usados y resultados.**

**Experimento 1:**

**Input:** R-EURUSD\_FULL\_4H\_144F.csv

El objetivo es experimentar con una combinación de parámetros que han arrojado ganancias superior al 30% en pruebas pasadas para la clasificación. Obtener los resultados de esos parámetros principalmente y evaluar el número de registros pasados convertidos en features (PROFUNDIDAD) desde 1 hasta 24 valores anteriores.

**Ejecución de 98 pruebas con los siguientes parametros:**

MODEL\_NAME = "XGBoost"

FRECUENCIA = "4H"

INDEX\_TO\_READ = 5 #

LIMITE\_IZQ = [-0.0029]

LIMITE\_DER = [0.0029]

BALANCED = True

TARGET\_NAME = "Class"

PROFUNDIDAD = [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24]

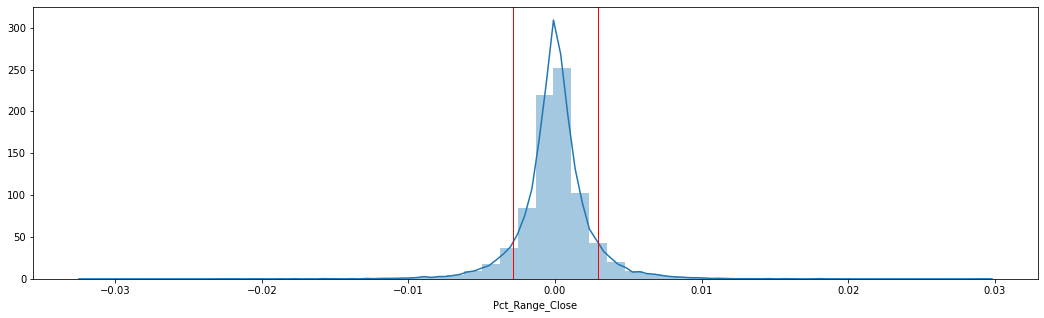
VAL\_PCT = [.2, .25]

TEST\_PCT = [.2, .15]

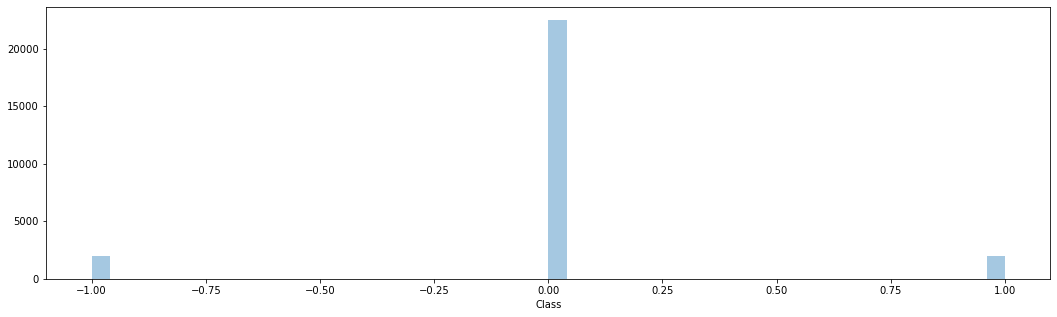
CASH = 1000

COMMISSION = 0.0001

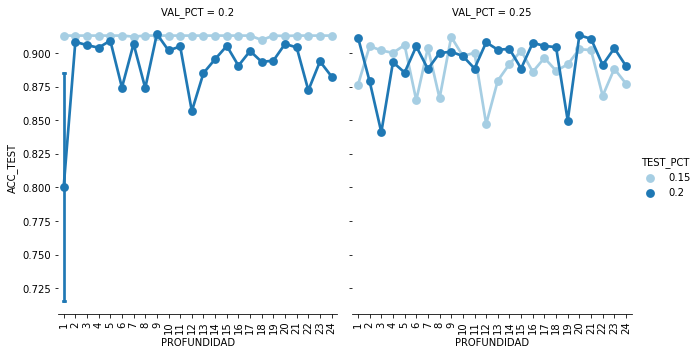
Distribución del dataset y clasificación con valores de LIMIT\_IZQ y LIMITE\_DER:

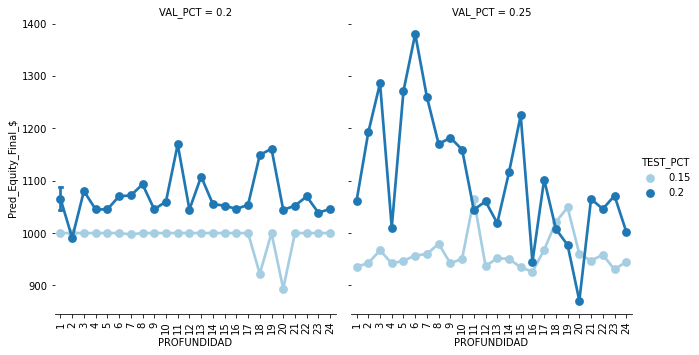


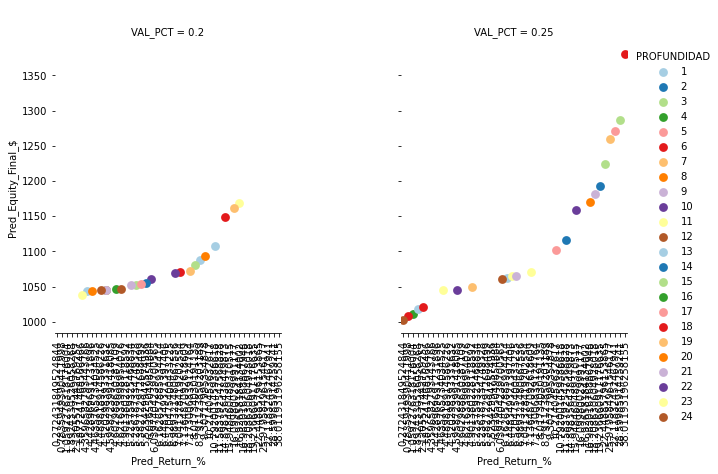
Distribución de clases:



Resultados:







Conclusiones del resultado:

En la primer grafica de resultados observamos como la combinación VAL\_PCT=2.0 y TEST\_PCT=.15 tienen un ACC de 0.91 en su mayoría, con esta predicción el backtesting no obtuvo operaciones, y por ende no hubo movimientos (grafica 2).

**La combinación de .25 de Val\_PCT y .2 de TEST\_PCT obtienen mejores resultados en el rango del 1 al 10 de profundidad en cuanto a backtesting se refiere. A partir de este resultado en los siguientes experimentos se utilizarán .25 de % para el subset de Validación, y .20 para el subset de testing. Probando únicamente en el rango de Profundidad de 1 a 10.**

La combinación .25 Val\_PCT y .15 TEST\_PCT no obtuvo Buenos resultados en backtesting.

En la tercer grafica se observan todas las ejecuciones con ganancia superior a 0.

A continuación los datos estadísticos de este subset de ejecuciones ganadoras:

count 48.000000

mean 9.585469

std 8.153566

min 0.272432

25% 4.520298

50% 6.305504

75% 12.412434

max 38.011932

Name: Pred\_Return\_%, dtype: float64

**Experimento 2:**

**Input:** R-EURUSD\_FULL\_4H\_144F.csv

El objetivo es experimentar con una combinación de parámetros que han arrojado mejores resultados en experimentos anteriores. VAL\_PCT =.25 , TEST\_PCT = .2, PROFUNDIDAD [0-10]. Con una combinación de diferentes limites, ya tengo los resultados de [-.0029 y 0.0029], 9 combinaciones principales evaluadas en una profundidad de 0 a 10.

**Ejecución de 90 pruebas con los siguientes parámetros:**

MODEL\_NAME = "XGBoost"

FRECUENCIA = "4H"

INDEX\_TO\_READ = 5 #

LIMITE\_IZQ = [-0.0025, -.002, -.0018]

LIMITE\_DER = [0.0025, .002, .0018]

BALANCED = True

TARGET\_NAME = "Class"

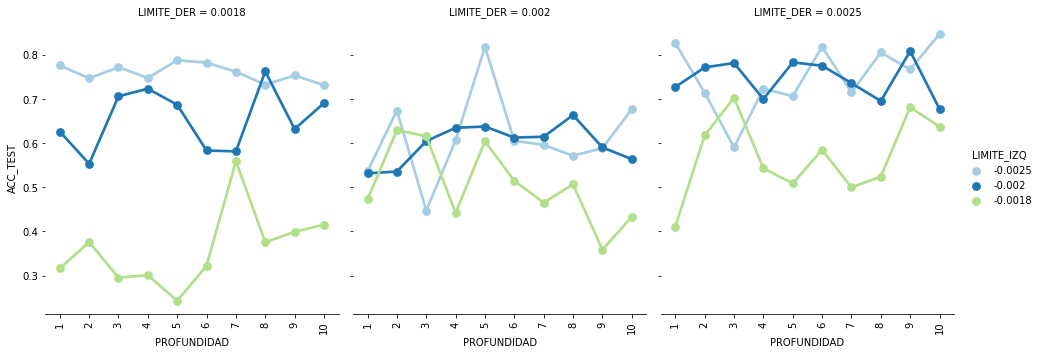
PROFUNDIDAD = [1,2,3,4,5,6,7,8,9,10]

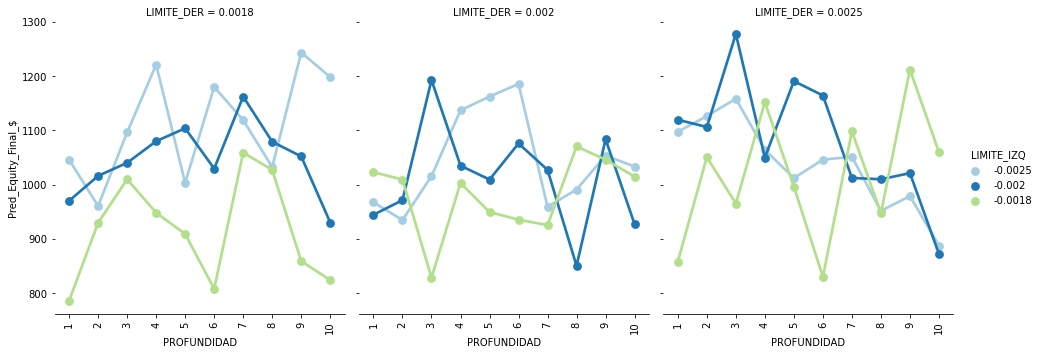
VAL\_PCT = [.25]

TEST\_PCT = [.2]

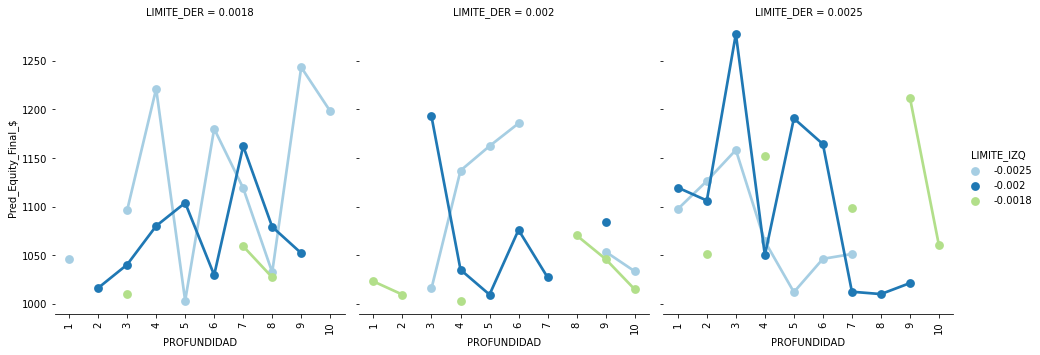
CASH = 1000

COMMISSION = 0.0001

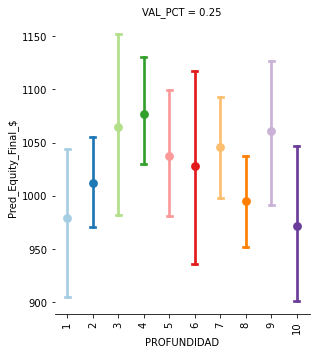
****

****

Grafica3 del subset de ejecuciones con ganancias mayores a 0:



Grafica4



Conclusiones:

En la primer grafica se puede observar el total de 9 ejecuciones principales a lo largo de 10 valores de profundidad. como ciertas combinaciones son peores en accuracy que otras. Se distingue la combinación de LIMITE\_DER 0.0018 con LIMITE\_IZQ=-0.0025 como una de las mejores.

En la segunda grafica Podemos distinguir que combinación de valores no son Buenos para obtener buenos resultados en backtesting.

En la tercer grafica se observa cómo hay 3 combinaciones de las 9 que en su mayoría de resultados a través de la profundidad de 0 a 10 obtienen ganancias:

Limite\_der: 0.0018 y Limite\_izq: -.0025

Limite\_der: 0.0018 y Limite\_izq: -.002

Limite\_der: 0.0025 y Limite\_izq: -.002

En la gráfica 4, se observa como la **profundidad de valor 4** obtuvo los mejores resultados de backtesting del total de ejecuciones. De hecho es el único que siempre obtuvo ganancias en las 9 combinaciones de pruebas.

Los datos estadísticos del subset de ejecuciones con ganancia mayor a 0 del total de 90 ejecuciones:

count 59.000000

mean 8.572538

std 7.058879

min 0.307049

25% 2.832975

50% 6.004899

75% 13.169421

max 27.770372

Name: Pred\_Return\_%, dtype: float64

**Experimento 3:**

**Input:** R-EURUSD\_FULL\_4H\_144F.csv

El objetivo es experimentar con una combinación de parámetros que han arrojado mejores resultados en experimentos anteriores. VAL\_PCT =.25 , TEST\_PCT = .2, LIMITE\_IZQ = [0.002], LIMITE\_DER = [-0.0025], PROFUNDIDAD [0,1,2,3,4]. Aplicando balanceo de clases y sin aplicar. La idea es determinar en qué porcentaje se mejora utilizando Oversampling.

**Ejecución de 2 pruebas con los siguientes parámetros (una balanceada y otra no):**

MODEL\_NAME = "XGBoost"

FRECUENCIA = "4H"

INDEX\_TO\_READ = 5 #

LIMITE\_IZQ = [-0.002]

LIMITE\_DER = [0.0025]

BALANCED = True

TARGET\_NAME = "Class"

PROFUNDIDAD = [0,1,2,3,4]

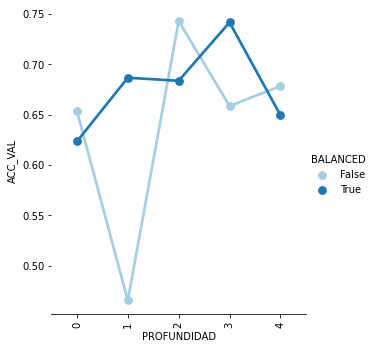
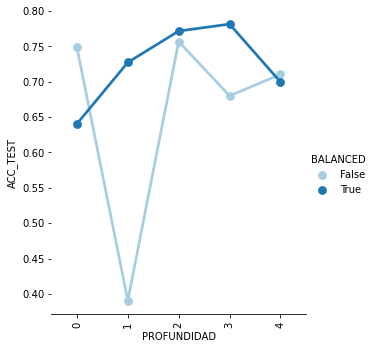
VAL\_PCT = [.25]

TEST\_PCT = [.2]

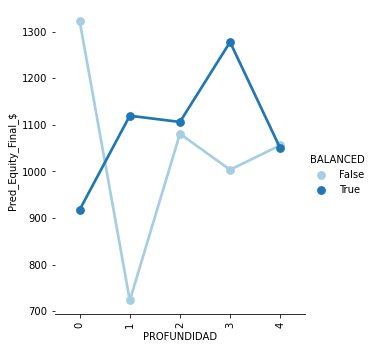
CASH = 1000

COMMISSION = 0.0001

Graficas1 de Accuracy para subset de Validación y Subset de Testing.

Grafica 3:



Conclusiones de experimento:

Tanto en la gráfica de validación como en la de pruebas se aprecia una mejoría cuando se balancean las clases.

Para los resultados de backtesting es curioso observar que cuando no existe balanceo, ni profundidad se obtienen mejores resultados. Hasta de 30%.

**Experimento 4:**

**Input:** R-EURUSD\_FULL\_4H\_144F.csv

El objetivo es experimentar con una combinación de parámetros que han arrojado mejores resultados en experimentos anteriores. VAL\_PCT =.25 , TEST\_PCT = .2, LIMITE\_IZQ = [0.002], LIMITE\_DER = [-0.0025], PROFUNDIDAD [6].

Aplicar un suffle para determinar el inicio del subset de pruebas, que ira desde el inicio del subset total hasta el (final del subset total – el tamaño del subset de validación).

**Ejecución de 101 pruebas con los siguientes parámetros**

MODEL\_NAME = "XGBoost\_4"

FRECUENCIA = "4H"

INDEX\_TO\_READ = 5

LIMITE\_IZQ = [-.0029]

LIMITE\_DER = [.0029]

CICLOS = 100

BALANCED = True

WITH\_SUFFLE = True

IDX\_TEST\_START = 0

TARGET\_NAME = "Class"

PROFUNDIDAD = [6]

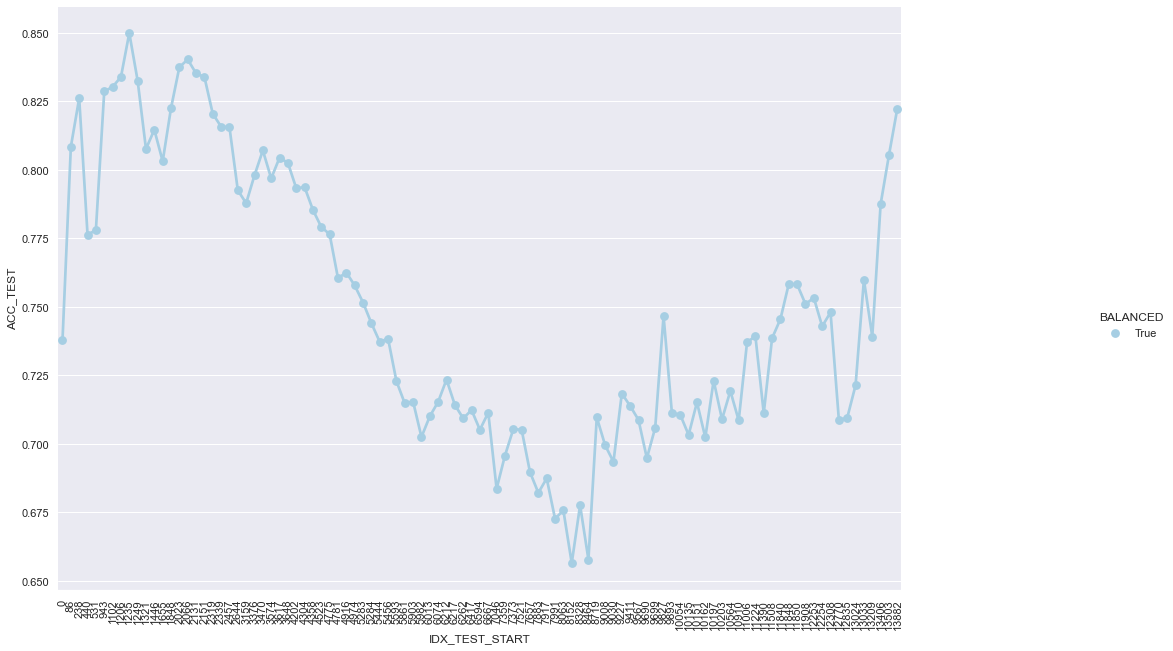
VAL\_PCT = [.25]

TEST\_PCT = [.2]

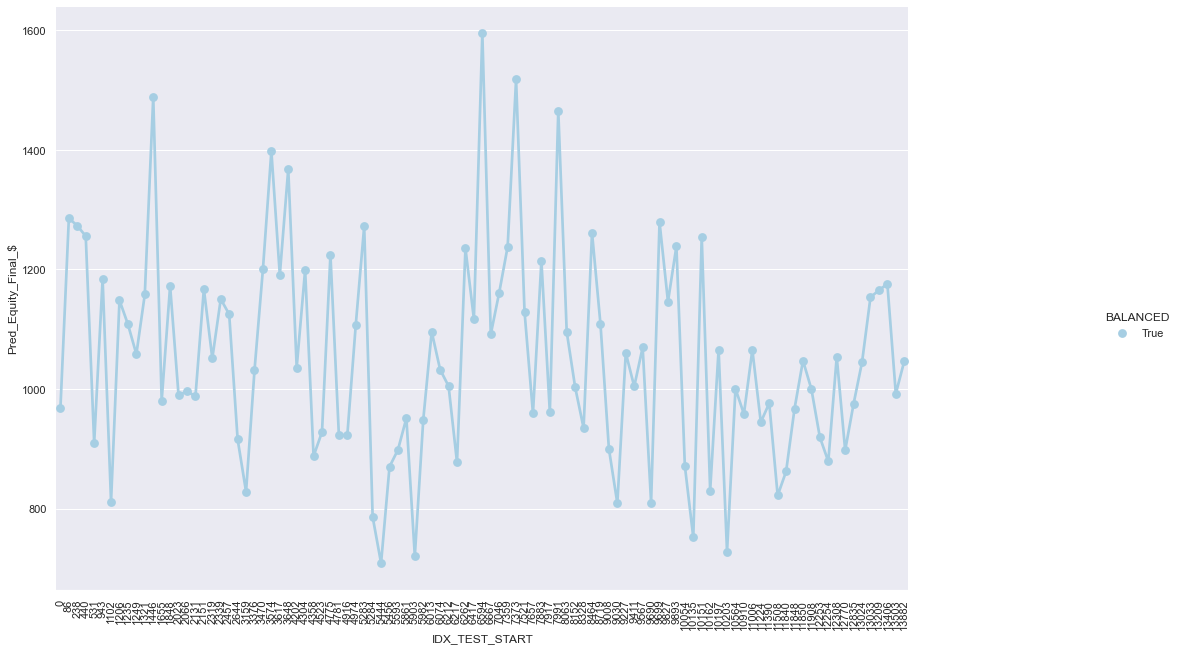
CASH = 1000

COMMISSION = 0.0001

Grafica 1, las 101 ejecuciones, donde ejeX muestra el índice donde inicio el Subset de Pruebas. Eje Y muestra el accuracy de Test.



Grafica 2, eje de las Y con el valor de retorno después del backtesting:



Conclusiones del experimento:

De la grafica 1 se observa claramente como se reduce el accuracy en el subset de test cuando el subset de pruebas se encuentra al centro del total del dataset. Al final y al inicio muestra mejores resultados.

De la grafica 2 podemos observar que no existe una correlación aparente con los valores de accuracy de la grafica pasada. Básicamente se visualizan ejecuciones ganadoras en mayor proporción a operaciones de perdida.

Datos estadísticos del subset de ejecuciones ganadoras:

count 58.000000

mean 17.060530

std 13.287437

min 0.042416

25% 6.561643

50% 15.201858

75% 23.736381

max 59.540153

Name: Pred\_Return\_%, dtype: float64

Datos estadísticos del subset de ejecuciones perdedoras:

count 43.000000

mean -10.327088

std 8.008899

min -29.125971

25% -15.386488

50% -8.395742

75% -3.986683

max -0.032736

Name: Pred\_Return\_%, dtype: float64

De los datos estadísticos observamos un 58% de pruebas ganadoras VS un 42 % de pruebas perdedoras.

Con los valores suffle se alcanzo hasta un retorno de inversión de 59% (~1590 USD).

Mientras en las ejecuciones con perdida la mayor perdida fue de -29%.

**Experimento 5:**

**Input:** R-EURUSD\_FULL\_4H\_144F.csv

El objetivo es experimentar con una combinación de parámetros que han arrojado mejores resultados en experimentos anteriores. VAL\_PCT =.25 , TEST\_PCT = .2, LIMITE\_IZQ = [0.002], LIMITE\_DER = [-0.0025], PROFUNDIDAD [0].

Aplicar un suffle para determinar el inicio del subset de pruebas, que ira desde el inicio del subset total hasta el (final del subset total – el tamaño del subset de validación).

**Ejecución de 101 pruebas con los siguientes parámetros**

MODEL\_NAME = "XGBoost\_4"

FRECUENCIA = "4H"

INDEX\_TO\_READ = 5

LIMITE\_IZQ = [-.002]

LIMITE\_DER = [.0025]

CICLOS = 100

BALANCED = False

WITH\_SUFFLE = True

IDX\_TEST\_START = 0

TARGET\_NAME = "Class"

PROFUNDIDAD = [0]

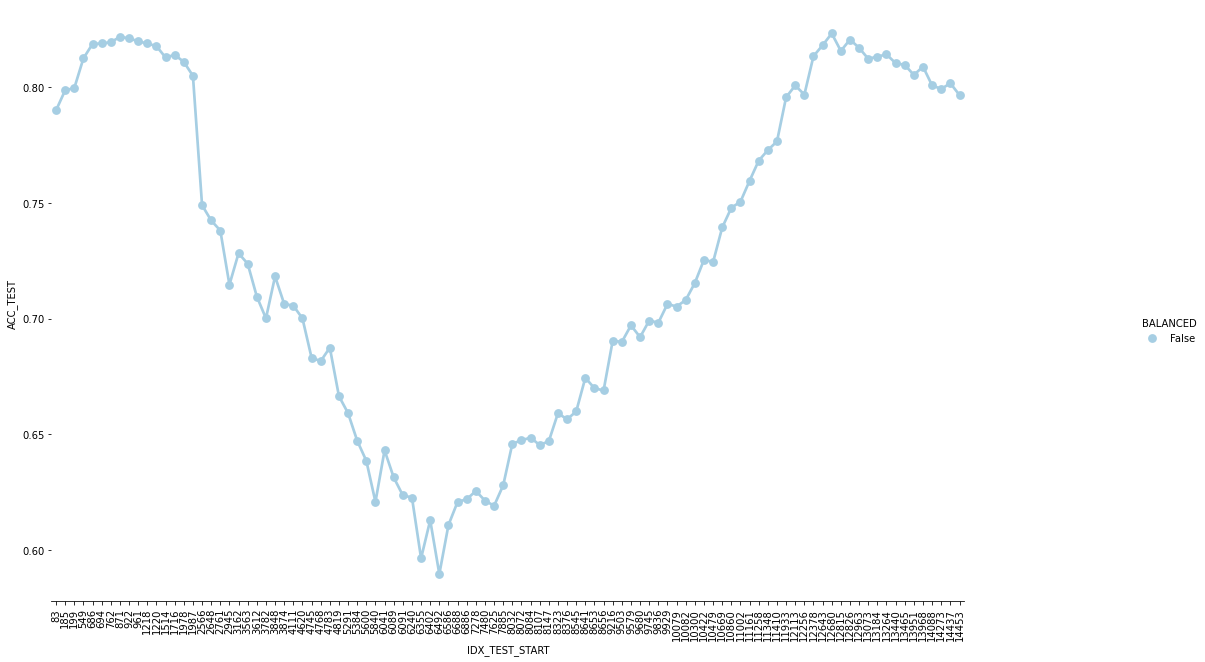
VAL\_PCT = [.25]

TEST\_PCT = [.2]

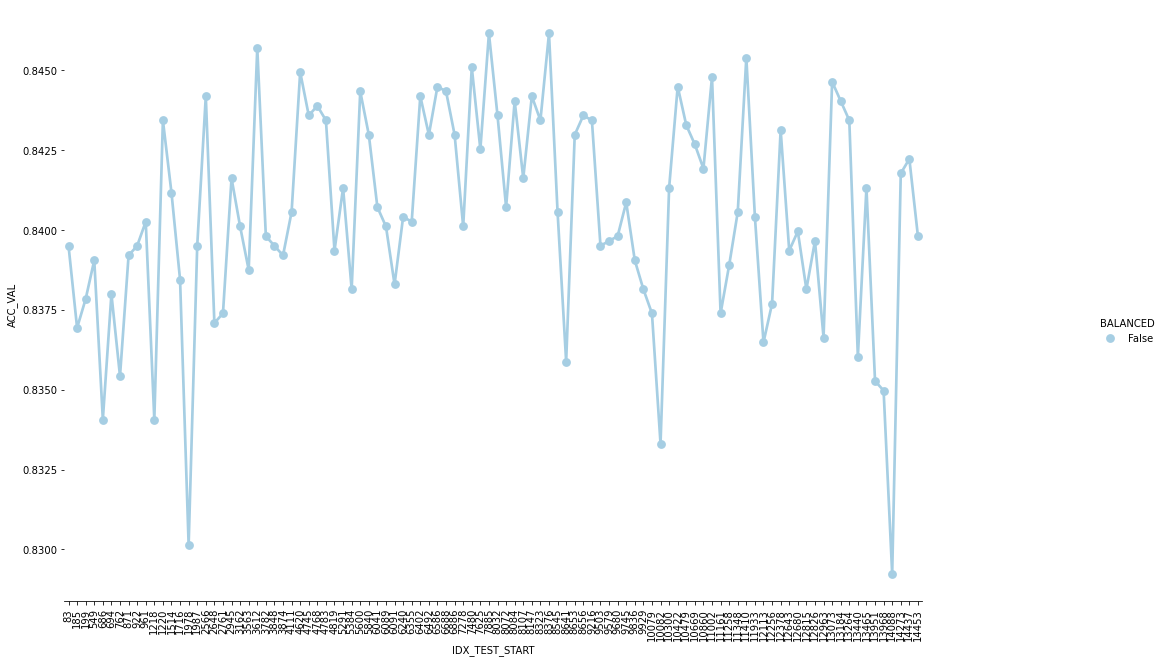
CASH = 1000

COMMISSION = 0.0001

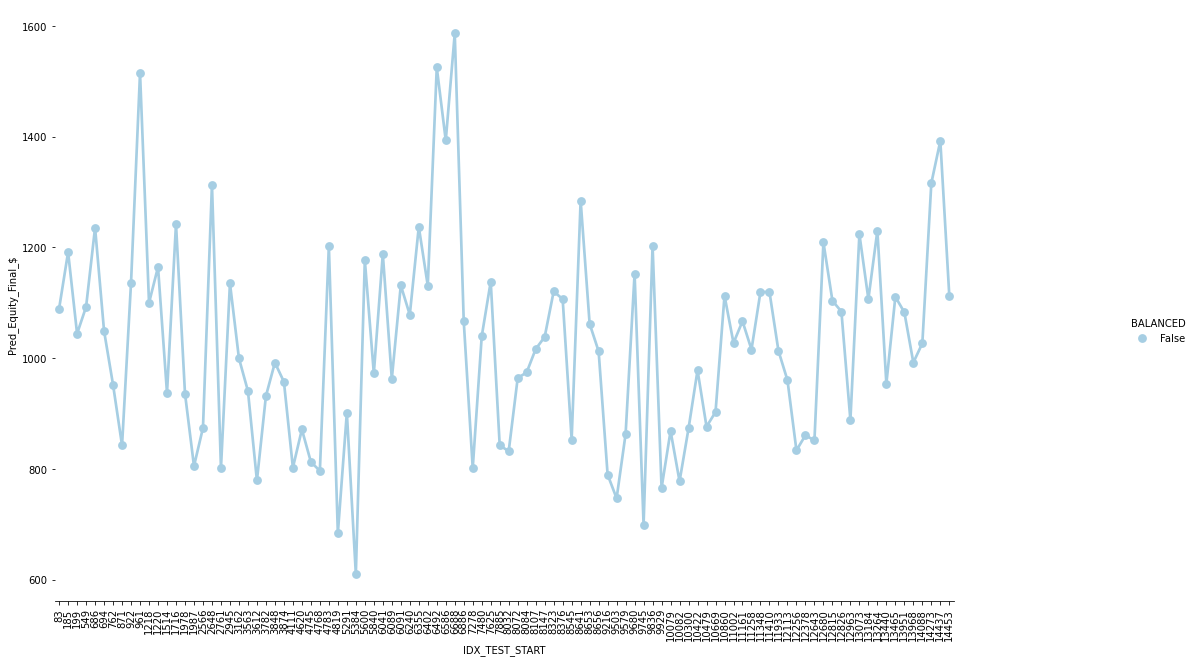
Grafica 1, Acc del subset de Test

****

Grafica 2, Acc del subset de validación

****

Grafica 3, retorno de inversión:



Estadísticas de resultados de pruebas con ganancia superior a 0:

count 54.000000

mean 16.067852

std 13.082041

min 1.242117

25% 7.942343

50% 12.004243

75% 20.773333

max 58.654386

Name: Pred\_Return\_%, dtype: float64

Estadísticas de resultados de pruebas con pérdidas:

count 46.000000

mean -13.247110

std 8.935954

min -38.992932

25% -19.704083

50% -13.013665

75% -5.094665

max -0.030881

Name: Pred\_Return\_%, dtype: float64

Conclusiones del experimento5:

Se confirman las conclusiones del experimento 4, existe una tendencia de mejor accuracy cuando está en algún extremo del dataset.

La mayoría de las pruebas fueron ganadoras con un porcentaje de 54%. Y un 46% de pruebas con pérdidas.

**Conclusión de experimentos:**

**Con parámetros seleccionados en base experimentación, se pudo obtener un mayor número de pruebas con ganancias (58% de pruebas) de hasta un 59% de retorno de capital aun utilizando un subset random y no continuo de una serie de tiempo.**