Uso de XGBoost para clasificación en el mercado financiero Forex.

MTI. Mawrer Amed Ramírez Martínez

Instituto Tecnológico de Estudios Superiores de Occidente

Jalisco, México

ms723737@iteso.mx

Dr. J. Guadalupe Olascuaga Cabrera

Instituto Tecnológico de Estudios Superiores de Occidente

Jalisco, México

jolascua@iteso.mx

Dr. Luis Fernando Gutiérrez Preciado

Instituto Tecnológico de Estudios Superiores de Occidente

Jalisco, México

lgutierrez@iteso.mx

*Resumen*—La investigación de distintas técnicas para predecir movimientos en el mercado de divisas interbancarias Forex es muy extensa, va desde aplicación de métodos de control, pasando por identificación de patrones en gráficas, hasta la aplicación de redes neuronales y distintos tipos de algoritmos de aprendizaje automatizado actuales. Múltiples investigaciones muestran resultados favorables aplicando algoritmos y técnicas de aprendizaje automatizado para regresiones y clasificaciones de series de tiempo. El predecir una clase o un valor de un sistema financiero es un problema complejo con gran cantidad de variables, se han desarrollado distintitos enfoques para atacar el problema como el análisis técnico, análisis fundamental, análisis de sentimientos y combinaciones de estos. En este trabajo se aborda el problema desde un análisis técnico, donde se estudian las características principales de los datos históricos del par interbancario EUR/USD que va desde el 1 de enero de 2014 hasta el 30 de mayo de 2020, el autor realiza un etiquetado de clasificación en base al porcentaje de cambio al cierre de cada periodo y se realizan experimentos de optimización para identificar los mejores valores de clasificación; como parte del análisis de datos de entrada se utilizan herramientas estadísticas. Para la selección de nuevas características se hace uso de la librería de Python Technical Analysis (AT) para obtener indicadores a partir de la serie de tiempo original, para la selección de características se emplea el modelo Clasificador XGBoost, a continuación se obtiene un subconjunto de características relevantes que serán utilizadas por los modelos finales, posteriormente se entrena el modelo clasificador XGBoost para finalmente utilizarlo en la predicción del subconjunto de datos de prueba, los cuales serán evaluados mediante simulación de inversión.

*Palabras Clave*— ForReign Exchange Market, FOREX, Exchange Rates, Machine Learning, XGBoost, Predicción.

# INTRODUCCIÓN

El mercado de divisas es uno de los comercios de mayor impacto económico a nivel mundial . El comercio de compra venta de monedas (*trading*) logra generar flujos de hasta 5 billones de dólares diarios, la dinámica de este flujo de capital está relacionado con aspectos macroeconómicos, sociales y políticos. En el mercado *‘ForReign Exchange’* (FOREX) las divisas y su comportamiento son la base para la evaluación de gestión de riesgos y portafolios de inversión. El comportamiento de las divisas se clasifica dentro de los pilares más importantes a de la economía global ; bajo esta afirmación surge la motivación de desarrollar una propuesta que proporcione respuesta a la siguiente pregunta: ¿Es posible anticipar, definir y obtener ganancias de los precios en las divisas en el mercado FOREX con el uso de algoritmos de aprendizaje automatizado?

El aprendizaje automatizado y las redes neuronales actualmente tienen una fuerza importante en muchas áreas de la ciencia de la computación y en la toma de decisiones en negocios no solo tecnológicos [1]. Dentro del aprendizaje automatizado se dividen diferentes métodos, los llamados métodos supervisados y los métodos no supervisados [2]. El aprendizaje supervisado es uno de los métodos de aprendizaje automático basado en la función de predicción f: x→y donde *‘x’* representa a la entrada de datos conocidos y *‘y’* representa el valor a predecir [3].

En los mercados financieros muchas de las grandes instituciones bancarias realizan investigación constante utilizando los avances tecnológicos para evaluar condiciones de riesgo en inversiones, predecir tendencias y generar ganancias de inversiones en distintos mercados [1]. Las investigaciones van desde predicción en movimientos de acciones, futuros, monedas, criptomonedas, etc. [4]. El trading en el mercado Forex se entiende como la acción de compraventa de activos cotizados con mucha liquidez en un mercado financiero electrónico a través entidad llamada broker quién a su vez realiza transacciones de gran volumen con un mercado global . El objetivo del trading es obtener un beneficio económico a partir de activos que obtenga o pierdan valor. Una de las razones por las que el trading se ha popularizado en los últimos años es debido a la entrada de nuevas tecnologías online que permiten ‘operar’ a partir de poco capital de dinero en plataformas administrada por brokers.

Esta investigación hace uso de datos históricos del mercado interbancario FOREX y modelos de aprendizaje automatizando para la predicción de movimientos clasificados en tres tipos con el fin de identificar los mejores parámetros y el mejor manejo de datos en el objetivo de obtener ganancias con un esquema simulado (*backtesting*).

El modelo es entrenado, validado y probado con datos históricos. La validación de las predicciones consiste en obtener las métricas de precisión, matrices de confusión y gráficas área bajo la curva; para la evaluación de los valores predichos estos son utilizados como indicadores de compra y venta en un ambiente de simulación de inversión (*backtesting*) obteniendo y comparando el retorno de inversión.

# Trabajo Relacionado

Existen numerosas investigaciones relacionadas a la predicción de los mercados financieros. Algunas técnicas basadas en gráficas, las cuales intentan identificar patrones a lo largo de la serie de tiempo, sin embargo, la volatilidad en los mercados genera ‘ruido’ y por consecuencia series de datos complejas, para resolver el problema del ruido y alta volatilidad Kenia et. al. [5] propone un enfoque para la transformación de series de tiempo en una representación de símbolos para posteriormente realizar un reconocimiento de patrones a partir de la similitud observada. Este enfoque es muy parecido al análisis de gráficas comúnmente utilizado por analistas y traders clásicos, donde mediante experiencia empírica logran identificar patrones en las gráficas representadas por velas japonesas [6]. Existen conceptos dentro del análisis de gráficas como Banderas, hombro cabeza hombro, formación de triángulos [7], etc. Análisis más complejos a partir de gráficas e indicadores como los niveles de Fibonacci suelen ser técnicas empleadas en el trading no automático.

Rupasinghe et al. [8] en su investigación utiliza el método soportes de vectores para clasificar, aplica estrategias de trading y analiza en que frecuencia de datos es posible obtener mayores ganancias. Por su parte Patil et. al. [9] propone un algoritmo hibrido basado en análisis técnico usando ‘*Long Short Term Memory’* (*LSTM*) y análisis de sentimientos basados en noticias utilizando ‘*Support Vector Machine’* (*SVM*) además hace una comparativa resultados utilizando múltiples pares de monedas.

Dentro del enfoque de análisis técnico existen numerosos indicadores financieros [10], investigaciones actuales desarrollan nuevos e incluso combinaciones de estos, como Liu et. al. [11] que sugiere un sistema de trading basado en una fusión de indicadores financieros (MACD, RSI, CCI).

Investigaciones con sistemas de aprendizaje automático que combinan técnicas de predicción aparecen en la actualidad como Lean [12] quién combina estrategias de soporte con redes neuronales para identificar movimientos de compra/venta en el mercado Forex.

Por su parte Lui Yang [13] utiliza el modelo XGBoost para la identificación de fuentes de perturbación en energías de calidad, es relevante pues se la estructura de los datos de entrada son series de tiempo. Algunas investigaciones relacionadas con el análisis de fuentes de energía consisten en la extracción de características e identificación de patrones. Para la extracción de características en series de tiempo comúnmente se emplean los métodos como la transformación de Fourier en tiempo corto, la transformación wavelet, la transformación Hilbert-Huan, etc. Y para la identificación de patrones comúnmente se emplean métodos como arboles de decisión[14, 15], redes neuronales profundas [16], maquinas de soportes de vectores[17].

# Trabajo propuesto

Este trabajo propone y evalúa el uso del modelo de clasificación XGBoost para la predicción del porcentaje de cambio de cierre en el mercado interbancario forex, para esta investigación se definieron 3 clases distintas a predecir [-1, 0, 1] donde la clase -1 será interpretada como movimiento de venta, para la clase 0 se interpreta que el porcentaje de cambio es mínimo y no representa operación en el y la clase 1 corresponde al de compra. En este trabajo se utilizó el siguiente ambiente: Python 3.7.3. y las siguientes versiones de librerías: Backtesting-0.1.4, TA- 0.5.25, xgboost-1.1.1, keras- 2.3.1 y tensorflow- 1.14.0.

Para el desarrollo de esta investigación se definió un flujo de trabajo que consta de distintas fases las cuales se muestran en Fig. 1.

A screenshot of a cell phone

Description automatically generated

Fig. 1. Flujo de datos principal, etapas principales del manejo de datos de entrada extracción y análisis de características, entrenamiento del modelo, validación y evaluación de las predicciones.

## Preparación de datos.

En esta etapa se obtienen y valida la integridad de los datos y los rangos en los que están distribuidos confirmando que no existan datos incongruentes en la serie de tiempo.

Los datos utilizados de entrada utilizados son historiales del mercado Forex del banco Dukascopy [18] del tipo de cambio EUR/USD (Euro – Dólar Americano) correspondientes al tipo de cambio EUR/USD, y van desde la fecha 1 de enero de 2014 hasta 30 de mayo de 2020. Las frecuencias utilizadas en esta investigación son: un minuto 1M, una hora 1H, cuatro horas 4H y un día 1D.

Para la creación del conjunto de datos inicial se utilizaron datos de compra y datos de venta correspondientes a las fechas mencionadas. Una vez unido se obtuvo el conjunto de datos principal con un total de once columnas [*timestamp, Open\_ask, High\_ask, Low\_ask, Close\_ask, Volume\_ask Open\_bid, High\_bid, Low\_bid, Close\_bid, Volume\_bid*].

En la Figura 2 se observan los primeros 100 registros del conjunto de datos correspondientes a la compra del conjunto de datos 4H.

A close up of a map

Description automatically generated

Fig. 2. Muestra del conjunto de datos de entrada en frecuencia de 4 horas: Valor de Apertura de venta, valor máximo de venta, valor mínimo de venta y valor de cierre de venta.

## Extracción de características

En esta fase a partir del conjunto de datos iniciales y mediante aplicación de métodos estadísticos se generaron nuevos datos que contienen información relevante de la serie de tiempo como valores de tipo movimiento promedio, osciladores estocásticos entre otros indicadores financieros utilizados en trading [20].

En la tabla 1 se muestran la lista de características agregadas de forma inicial en el conjunto de datos a utilizar con el modelo base Clasificador XGBoost. El conjunto de datos se compone de 138 características.

Las características de media móvil son frecuentemente utilizadas en series de tiempo, nos permite ‘*suavizar*’ series promediando el valor actual junto con *n* periodos anteriores, no existe una regla de cuantos periodos deben utilizarse. Estos valores a menudo son utilizados para definir tendencias. Existen cuatro tipos de medias móviles: simples, exponenciales, lineal ponderado y suavizado ponderado.

La librería *Technical Analysis (TA*) de Python genera toda una colección de indicadores a partir de una serie de datos de tipo Open, High, Low, Close y Volume. Con el uso de la librería TA fue posible obtener 75 características adicionales. Generando un total de 138 características totales.

TABLA I

Lista Incrementada De Características

|  |  |  |
| --- | --- | --- |
| Núm. | Descripción | Comentarios |
| 1-11 | Datos de entrada | Time, OHLC\_ask, OHLC\_bid, Volume\_ask, Volume\_bid. |
| 12 | SPREAD | Diferencia de precio de venta –  precio de compra. |
| 13 | WEEKDAY | Número del día de la semana [0-6]. |
| 14 | PCT\_RANGE | Porcentaje de cambio de Close\_ask. |
| 15 | PCT\_RANGE \_D | Diferencia de PCT\_RANGE y su  valor previo. |
| 16-20 | MA[5,25,50,100, 150, 175, 200] | Movimiento promedio. |
| 21 | EMA[5] | Movimiento promedio exponencial. |
| 22-42 | MOM[1-15,25,35, 50,75,80,100] | Momento. |
| 43-47 | RSI[5,14,25,50,70] | Indicador RSI. |
| 48-50 | BBANDS | Bandas de Bolinger (valor alto, valor bajo y promedio[25]). |
| 51 | STOK | Oscilador estocástico K. |
| 52-53 | STOD | Oscilador estocástico D. |
| 54-59 | TRIX[2,5,15,25, 50,75] | Indicador TRIX. |
| 60-62 | MACD[2,25] | Indicador MACD. |
| 63 | MASS | Indicador MASS. |
| 64-138 | TA\_FEATURES | Características generadas por la librería de Python TA. |

OHLC = Open, High, Low, Close.

SPREAD, PCT\_RANGE, PCT\_RANGE\_D, AQM\_PRC, AQM\_PRC\_D, MA, MOM obtenidos a partir del valor Close\_ask.

En la Figura 3 observamos la relación entre las características cierre, volumen, porcentaje de cambio y diferencia de porcentaje de cambio. En esta gráfica se muestra la relación de tipo regresión lineal para visualizar si existe una correlación directa entre estas.

A close up of text on a white background

Description automatically generated

Fig. 3. Gráfica con la relación de tipo regresión lineal de las características de venta ‘Cierre’, ‘Volumen’, ‘porcentaje de cambio de cierre’, ‘diferencia de porcentaje de cambio de cierre’, en una distribución .

Esta investigación busca predecir el precio de cierre de cada registro, por lo tanto, se procede a obtener el porcentaje de cambio entre registros, porcentaje de cambio (*PCT\_RANGE*) mostrado en la Figura 4 este valor nos permite identificar el tipo de movimiento en la serie de tiempo.

A screenshot of a cell phone

Description automatically generated

Fig. 4. Distribución de las características PCT\_RANGE y PCT\_RANGE \_D ambos del conjunto de datos de compra.

A screenshot of a cell phone

Description automatically generated

Fig. 5. Gráfica de distribución normal del valor PCT\_RANGE.

A partir del valor PCT\_RANGE es posible clasificar que registros tienen un incremento en el valor de Cierre, así como identificar que registros tuvieron un decremento en el valor de Cierre el cual será un valor negativo, en la Figura 5 podemos observar la distribución normal de esta característica.

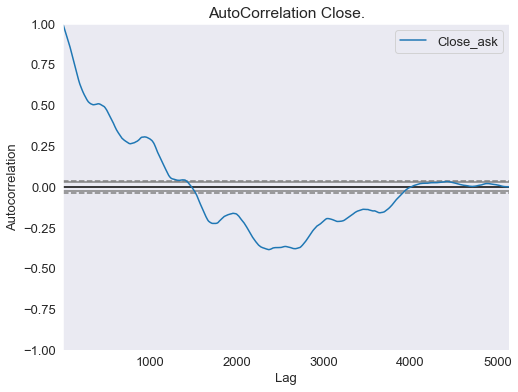


Fig. 6. Autocorrelación del valor de cierre a la venta en conjunto de datos.

El análisis de autocorrelación del precio de Cierre en la Figura 6 muestra una de sus propiedades principales de las series de tiempo que son secuencia de datos sucesivos; por otro lado, en la Figura 7 se observa la autocorrelación de los valores PCT\_RANGE correspondientes al porcentaje de cambio del precio de cierre, la autocorrelación en estos datos es muy cercana a 0, lo cual permitirá en la fase de entrenamiento del modelo poder utilizar técnicas como oversampling para el balanceo de clases.

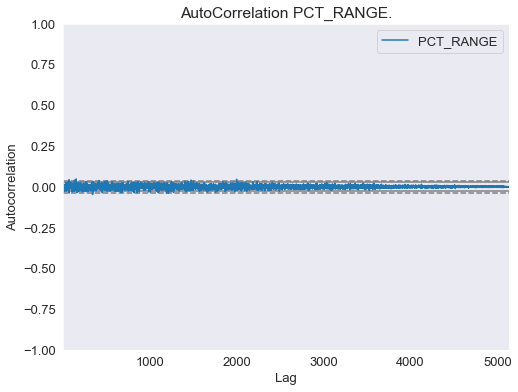


Fig. 7. Autocorrelación del valor de PCT\_RANGE en conjunto de datos.

En esta investigación se propone la clasificación en 3 clases mostradas en la Figura 8; a partir de dos valores de umbral se propone separar las clases de tipo -1 (venta), 0 (sin operación) y 1 (compra). Dichos valores umbral se obtuvieron mediante la optimización con la herramienta backtesting.

A screenshot of a social media post

Description automatically generated

Fig. 8. Representación de clases del valor PCT\_RANGE utilizando dos valores de umbral: -0.0029 y -0.0029.

## Análisis y selección de características

Con el fin de obtener resultados precisos en la predicción de movimientos del mercado, este articulo establece un modelo de clasificación basado en XGBoost. Para evitar la selección empírica y de forma aleatoria de los parámetros usados en los datos de entrada, se utilizó el modelo de clasificación XGBoost como base para la selección de las características más importantes. XGBoost permite organizar las características relacionadas con su grado de influencia en el modelo de predicción. La importancia está basada en un puntaje denominado puntaje F [19] que indica la cantidad de veces que fue utilizada dicha característica para predecir un valor. Esta métrica muestra específicamente cuantas veces fue utilizada esta característica para el uso de árboles de decisión y promediada con el total de árboles de decisión existentes para este modelo.

En la Tabla II se observa como con un umbral de decisión (*threshure*) de 0.006 y con un número de características de 118 obtiene mayor exactitud (*accuracy*) en un modelo de clasificación base XGBoost.

TABLA II

Resultados del Selector de Características usando XGBoost

|  |  |  |
| --- | --- | --- |
| n. | Thresh. | Accuracy: |
| 131 | 0.000 | 80.63% |
| 121 | 0.004 | 78.58% |
| 119 | 0.005 | 81.79% |
| **118** | **0.006** | **84.89%** |
| 95 | 0.007 | 84.03% |
| 43 | 0.008 | 81.88% |
| 21 | 0.009 | 83.22% |
|  |  |  |

Acc. base: 78.58% con 131 características.

n. = cantidad de características usadas para evaluar el modelo XGBoost.

## Entrenamiento del modelo

El algoritmo XGBoost consiste en un clasificador compuesto de múltiples arboles de decisiones. Comparado con otros algoritmos, XGBoost tolera bien ruido y tiene buen performance. Además, XGBoost provee una funcionalidad espacial para la selección de características, en esta investigación se utiliza el clasificador XGBoost de tipo regresión logística para el análisis de características y para la predicción de las clases.

### B. Arboles de decisión

Un árbol de decisión clasificador es una estructura que describe la clasificación de instancias. El árbol de decisión consiste en nodos y enlaces directos. Existen dos tipos de nodos en un árbol de decisión: nodos internos y nodos hoja. Los nodos internos representan las características de los datos o atributos y los nodos hoja representan la clase.

### AlgoritmoXGBoost

XGBoost es un algoritmo que construye un conjunto de arboles de decisión. El algoritmo XGBoost surge a partir de la teoría del gradiente descendente, la diferencia es que XGBoost utiliza una combinación de arboles débiles para generar colectivamente un árbol fuerte. XGBoost es un algoritmo supervisado lo que significa que se le proporcionan una serie de etiquetas durante la fase de entrenamiento con la intención de que prediga correctamente cuando se ejecute con información sin etiquetar. Para este trabajo el tipo de algoritmo utilizado es un clasificador XGBoost de tipo regresión logística.

### Datos de entrenamiento, validación y pruebas.

El conjunto de datos inicial se dividió en tres subconjuntos correspondientes al proceso de entrenamiento, al proceso de validación y proceso de pruebas. El porcentaje correspondiente al conjunto de pruebas y conjunto de datos se obtuvo a partir de experimentación y comparación de mejores resultados explicado en la sección de experimentos.

Debido a que la cantidad de registros pertenecientes a la clase venta y compra no estaba balanceada con la clase ‘0’ se procedió a aplicar la técnica de Oversampling [20] en la cual se genera misma cantidad de valores de cada clase a partir de los datos que ya se tienen. Aplicar esta técnica mejora la exactitud del modelo [21].

## Validación del modelo

En la etapa de validación se utilizaron diferentes métricas como la precisión absoluta, la matriz de confusión y el área bajo la curva para determinar los modelos que mejor predijeron los subconjuntos de datos correspondientes a validación y pruebas.

Una curva ROC (curva de característica operativa del recepto) es un gráfico que muestra el rendimiento de un modelo de clasificación en todos los umbrales de clasificación.

Esta curva representa dos parámetros:

##### Tasa de verdaderos positivos

##### Tasa de falsos positivos

Tasa de verdaderos positivos (TPR) es sinónimo de exhaustividad y, por lo tanto, se define de la siguiente manera:

A picture containing object, clock

Description automatically generated

Tasa de falsos positivos (FPR) se define de la siguiente manera:

A picture containing object, clock

Description automatically generated

Una curva ROC representa TPR frente a FPR en diferentes umbrales de clasificación. Reducir el umbral de clasificación clasifica más elementos como positivos, por lo que aumentarán tanto los falsos positivos como los verdaderos positivos.

## Evaluación de predicciones

En esta etapa se utilizó la librería *Backtesting* de Python para la simulación de trading en base a las predicciones realizadas a partir del modelo XGBoost.

La librería *Backtesting* permite simular un valor inicial del portafolio de inversión, definir el precio de comisión para cada operación, y permite definir una estrategia de compra/venta (*trading)* de divisas.

La estrategia en el caso de que no exista operación previa el algoritmo ingresará una operación de venta si el valor de la clase predicha es -1 correspondiente al siguiente registro, para el caso de 1 la estrategia agregara una operación de compra. Si ya existe una operación previa y la operación no es la misma que la actual, entonces el algoritmo cerrará la operación actual y abrirá una nueva, en caso de que el valor de la clase sea 0 o sea del mismo tipo entonces continuara al siguiente valor.

Al finalizar el backtesting se obtienen valores como porcentaje de retorno, capital final, el pico máximo de capital, número de operaciones totales, mejor operación, peor operación, duración del backtesting, tasa de operaciones con ganancia, etc. Estos valores serán utilizados para comparar los resultados predichos de los modelos XGBoost generados. La librería *Backtesting* cuenta con gráficas de resultados como la mostrada en la Figura 10.

A close up of a map

Description automatically generatedFig. 10. Gráfica de resultados de backtesing a partir de datos clasificados.

# Experimentos y Resultados

A partir del flujo de trabajo descrito anteriormente se realizaron experimentos con el fin de encontrar los mejores valores y parámetros para el tratamiento de los datos y selección de características.

El termino *Profundidad* se utilizará para la cantidad de registros anteriores en la serie de tiempo, los cuales serán usados como características adicionales a las 138 antes descritas.

En todos los experimentos descritos en esta sección para el backtesting se utilizó un capital inicial de $1000 con una comisión de .0001 por operación.

## Experimento 1:

### Objetivos:

#### Determinar si existe una relación con la exactitud y el % de Retorno resultante del backtesting con el valor de Profundidad.

#### Determinar que porcentajes son mejores para dividir el conjunto de datos en subconjunto de validación y subconjunto de pruebas.

### Ambiente de pruebas:

Ejecución de combinación de 98 pruebas con los siguientes parámetros, frecuencia=4H, Clases balanceadas, subconjunto de validación=[20%,25%], subconjunto de pruebas=[20%,15%], umbral de clase -1=-0.0029, umbral de clase 1=0.0029, profundidad = [1-24].

### Resultados:

A screen shot of a social media post

Description automatically generated

Fig. 11. Resultados de exactitud del subconjunto de datos de prueba respecto a las 98 pruebas realizadas.

A picture containing sitting

Description automatically generated

Fig. 12. Resultados de capital final de backtesting del subconjunto de datos de prueba respecto a las 98 pruebas realizadas.

TABLA III

Resultados Experimento 1- Máximos % de Retorno.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| % Val. | % Pruebas | Acc. Pruebas | Acc. Validación | % de Retorno | # Op. | | Win  Rate |
| 20 | 15 | 0.904888 | 0.857871 | 0 | 1 | 0 | |
| 20 | 20 | 0.914135 | 0.857871 | 16.92 | 14 | 50.0 | |
| 25 | 15 | 0.900101 | 0.870149 | 6.44 | 14 | 50.0 | |
| **25** | **20** | **0.904888** | **0.840707** | **38.01** | **24** | **66.6** | |
|  |  |  |  |  |  |  | |

# Op = Número de Operaciones

Los resultados de la Tabla III muestran que la combinación de 25% de subconjunto de validación y 20% del subconjunto de pruebas se obtuvo mejor retorno de inversión, un 38%. En esta misma serie de resultados se obtuvieron resultados positivos en el rango de 0 a 10 de profundidad mostrados en las Figuras 11 y 12.

## Experimento 2:

### Objetivos:

#### Tomando como base los resultados del Experimento 1 determinar el mejor rango [0-10] la cantidad de registros anteriores para ser usados como entrada.

#### Determinar el mejor umbral para clasificar el target.

### Ambiente de pruebas:

Ejecución de combinación de 90 pruebas con los siguientes parámetros, frecuencia=4H, Clases balanceadas, subconjunto de validación=25%, subconjunto de pruebas=20%, umbral de clase -1=[-0.0025, -.002, -.0018], umbral de clase 1=[0.0025, .002, .0018], profundidad = [1-10].

### Resultados:

A picture containing building, sitting, front, screen

Description automatically generated

Fig. 13. Resultados de exactitud del subconjunto de datos de prueba respecto a las 90 pruebas realizadas.

A picture containing standing, playing, person, game

Description automatically generated

Fig. 14. Resultados de exactitud del subconjunto de datos de prueba respecto a las 90 pruebas realizadas.

TABLA IV

Resultados Experimento 2- Máximos % De Retorno.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Lim.  -1. | Lim  1 | | Acc. Pruebas | Acc. Val. | % de Retorno | # Op. | | Win  Rate |
| -0.0025 | | 0.0025 | 0.590489 | 0.615280 | 15.82 | 340 | 55.88 | |
| -0.0025 | | 0.0020 | 0.605209 | 0.638985 | 18.57 | 394 | 52.0 | |
| -0.0025 | | 0.0018 | 0.753350 | 0.679601 | 24.34 | 203 | 56.0 | |
| **-0.002** | | **0.0025** | **0.781468** | **0.741658** | **27.77** | **163** | **58.28** | |
| -0.002 | | 0.0020 | 0.604642 | 0.635513 | 19.27 | 410 | 55.60 | |
| -0.002 | | 0.0018 | 0.581242 | 0.552167 | 16.22 | 399 | 48.12 | |
| -0.0018 | | 0.0025 | 0.681261 | 0.695304 | 21.20 | 297 | 55.55 | |
| -0.0018 | | 0.0020 | 0.506888 | 0.555639 | 7.02 | 653 | 49.46 | |
| -0.0018 | | 0.0018 | 0.559162 | 0.576023 | 5.9 | 566 | 48.05 | |
|  | |  |  |  |  |  |  | |

Lim -1 = Umbral utilizado para clase de tipo -1

Lim 1 = Umbral utilizado para clase de tipo 1

# Op = Número de Operaciones

La Tabla IV muestra que con la combinación de Lim -1=-0.002 y Lim 1=0.0025 obtuvo un mayor % de retorno de 27.77 y una tasa de operaciones ganadas de 58.28%, series de resultados mostrados en las Figuras 13 y 14.

## Experimento 3:

### Objetivos:

#### Tomando como base los resultados del Experimento 1 y 2 determinar si los resultados de ganancia se mantienen aún cuando la distribución del conjunto de datos de prueba esta ubicado de forma aleatoria a lo largo la serie de tiempo.

### Ambiente de pruebas:

Ejecución de 100 pruebas con los siguientes parámetros, frecuencia=4H, Clases balanceadas, subconjunto de validación=25%, subconjunto de pruebas=20%, umbral de clase -1=[-.002], umbral de clase 1=[0.0025], profundidad = [0].

### Resultados:

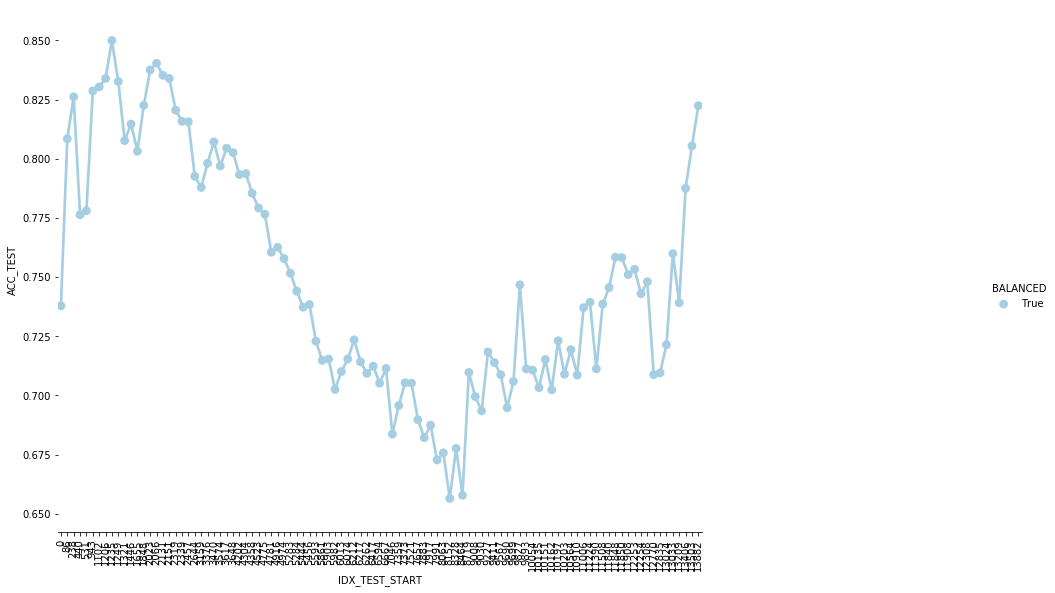


Fig. 15. Resultados de exactitud del subconjunto de datos de prueba respecto a las 90 pruebas realizadas.

A picture containing object, television, room

Description automatically generated

Fig. 16. Resultados de exactitud del subconjunto de datos de prueba respecto a las 90 pruebas realizadas.

TABLA V

Resultados Experimento 3- Valores Estadísticos.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Acc. Pruebas | | Acc.  Val. | | % de Retorno | | # Op. | Win  Rate |
| Promedio | | 0.749470 | | 0.852055 | | **5.40** | 212.86 | **50.51** |
| Std | | 0.050124 | | 0.028480 | | 17.68 | 128.52 | 4.13 |
| Min | | 0.656474 | | 0.769382 | | -29.12 | 31.00 | 39.39 |
| 25% | | 0.709513 | | 0.831948 | | -7.19 | 100 | 47.79 |
| 50% | | 0.739147 | | 0.857942 | | 4.50 | 156 | 50.60 |
| 75% | | 0.793696 | | 0.874868 | | 16.51 | 338 | 53.12 |
| Max | | 0.849943 | | 0.892852 | | **59.40** | 507 | **64.51** |
|  | |  | |  | |  |  |  |

# Op = Número de Operaciones

## En la Figura 15 se observa como la exactitud del subconjunto de datos de prueba muestra mejores resultados en los extremos de la serie de tiempo.

La figura 16 no muestra una correlación alta con los resultados de la exactitud en la figura X.

La Tabla V muestra resultados de hasta 59% de retorno de inversión. La tasa de operaciones ganados que se alcanzó fue de 64.51.

El 58% de las pruebas obtuvieron ganancias y un 42% de las pruebas aleatorias reflejaron perdidas.

# Conclusiones Y Trabajo Futuro

En este articulo se utilizó el algoritmo XGBoost para clasificar una serie de tiempo correspondiente al tipo de cambio EUR/USD del mercado financiero Forex. Después de un proceso de experimentación de generación, selección de características, entrenamiento, validación de resultados y evaluación de resultados de predicción, se logró obtener resultados positivos en un ambiente de simulación de trading de hasta un 59.4% de retorno de inversión.

A partir de los experimentos con localización del conjunto de datos de pruebas de forma aleatoria se logró demostrar la relación que existe de la ubicación del conjunto de datos con la precisión del modelo clasificador. Los resultados de las pruebas aleatorias muestran un promedio de 50% de la tasa de operaciones ganadas, lo cual demuestra que a pesar de tener ejecuciones de hasta un retorno de inversión de 59.40% los modelos desarrollados aun no obtienen ganancias constantes.

A partir de este flujo de trabajo se puede continuar explorando las capacidades del modelo XGBoost mediante la combinación de hiperparámetros, así como evaluar la metodología en múltiples pares de divisas. Esta investigación baso su exploración en el retorno de capital a partir del procedimiento de backtesting, sin embargo, se puede proponer enfocar las optimizaciones para mejorar ]la tasa de ganancias (*win rate*).

Referencias

1. P. Vats and K. Samdani, "Study on Machine Learning Techniques In Financial Markets," 2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN), Pondicherry, India, 2019, pp. 1-5, doi: 10.1109/ICSCAN.2019.8878741.
2. K. Pahwa and N. Agarwal, "Stock Market Analysis using Supervised Machine Learning," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 2019, pp. 197-200, doi: 10.1109/COMITCon.2019.8862225.
3. D. Pradeepkumar and V. Ravi, "FOREX Rate prediction using Chaos and Quantile Regression Random Forest," 2016 3rd International Conference on Recent Advances in Information Technology (RAIT), Dhanbad, 2016, pp. 517-522, doi: 10.1109/RAIT.2016.7507954.
4. J. Sun, Y. Zhou and J. Lin, "Using machine learning for cryptocurrency trading," 2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS), Taipei, Taiwan, 2019, pp. 647-652, doi: 10.1109/ICPHYS.2019.8780358.
5. Kania, Krzysztof & Juszczuk, Przemysław & Kozak, Jan. (2019). Enhanced Symbolic Description in Analyzing Patterns and Volatility on the Forex Market. Vietnam Journal of Computer Science. 6. 10.1142/S2196888819500180.
6. Chiung-Hon Leon Lee, WenSung Chen and Alan Liu, "An implementation of knowledge based pattern recognition for financial prediction," IEEE Conference on Cybernetics and Intelligent Systems, 2004., Singapore, 2004, pp. 218-223 vol.1, doi: 10.1109/ICCIS.2004.1460415.
7. Y. L. Yong, Y. Lee and D. Ngo, "An investigation into the recurring patterns of forex time series data," 2015 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS), Langkawi, 2015, pp. 313-317, doi: 10.1109/IRIS.2015.7451631.
8. M. Rupasinghe, M. N. Halgamuge and N. Tran Quoc Vinh, "Forecasting Trading-Time based Profit-Making Strategies in Forex Industry: Using Australian Forex Data," 2019 11th International Conference on Knowledge and Systems Engineering (KSE), Da Nang, Vietnam, 2019, pp. 1-9, doi: 10.1109/KSE.2019.8919432.
9. V. Patil, N. Somani, A. Tadvi and V. Attar, "Algorithmic Forex Trading using Combination of Numeric Time Series and News Analysis," 2018 4th International Conference for Convergence in Technology (I2CT), Mangalore, India, 2018, pp. 1-5, doi: 10.1109/I2CT42659.2018.9058285.
10. C. Zhou et al., "Fine-grained Sentiment Analysis of Foreign Exchange News," 2019 5th International Conference on Information Management (ICIM), Cambridge, United Kingdom, 2019, pp. 279-284, doi: 10.1109/INFOMAN.2019.8714715.
11. Z. Liu and D. Xiao, "An Automated Trading System with Multi-indicator Fusion Based on D-S Evidence Theory in Forex Market," 2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery, Tianjin, 2009, pp. 239-243, doi: 10.1109/FSKD.2009.395.
12. Lean Yu, Kin Keung Lai and Shouyang Wang, "Designing a hybrid AI system as a forex trading decision support tool," 17th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'05), Hong Kong, 2005, pp. 5 pp.-93, doi: 10.1109/ICTAI.2005.56.
13. Y. Liu, H. Luo, B. Zhao, X. Zhao and Z. Han, "Short-Term Power Load Forecasting Based on Clustering and XGBoost Method," 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 2018, pp. 536-539, doi: 10.1109/ICSESS.2018.8663907.
14. Li Xiaona, Shen Xinglai, Xue Xue, et al. Power Quality Disturbance Based on Improved HHT and Decision Tree[J]. Electric Power Construction, 2017, 38(2): 114-21.
15. Huang Nantian, Peng Hua, Cai Guowei, et al. Feature Selection and Optimal Decision Tree Construction of Complex Power Quality Disturbance [J]. Proceedings of the CSEE, 2017, 03): 776-86.
16. Guan Chun, Zhou Weiwei, Lu Weiguo. Recognition of Multiple Power Quality Disturbances Using Multi-Label RBF Neural Networks[J]. Transactions of China Electrotechnical Society, 2011, 08): 198-204.
17. Qin Ye, Yuan Haiwen, Yuan Haibin, et al. Classification of Power Quality Disturbances Based on Optimized Least Squares Support Vector Machine[J]. Transactions of China Electrotechnical Society, 2012, 08): 209-14.
18. “Historical Data Feed :: Dukascopy Bank SA | Swiss Forex Bank | ECN Broker | Managed accounts | Swiss FX trading platform,” Dukascopy.com, 2020. [Online]. Available: https://www.dukascopy.com/swiss/english/marketwatch/historical/.
19. H. P. S. D. Weerathunga and A. T. P. Silva, "DRNN-ARIMA Approach to Short-term Trend Forecasting in Forex Market," 2018 18th International Conference on Advances in ICT for Emerging Regions (ICTer), Colombo, Sri Lanka, 2018, pp. 287-293, doi: 10.1109/ICTER.2018.8615580.
20. L. Jidong and Z. Ran, "Dynamic Weighting Multi Factor Stock Selection Strategy Based on XGboost Machine Learning Algorithm," 2018 IEEE International Conference of Safety Produce Informatization (IICSPI), Chongqing, China, 2018, pp. 868-872, doi: 10.1109/IICSPI.2018.8690416.
21. A. D. Chakravarthy, S. Bonthu, Z. Chen and Q. Zhu, "Predictive Models with Resampling: A Comparative Study of Machine Learning Algorithms and their Performances on Handling Imbalanced Datasets," 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), Boca Raton, FL, USA, 2019, pp. 1492-1495, doi: 10.1109/ICMLA.2019.00245.