

TradeForest

Predicción de Acciones Colombianas (BVC) usando Random Forest

Integrantes:

Alan Cheyne

Fabian Espitia

Julian Pira

Lector:

Arles Ernesto Rodriguez

Universidad Nacional de Colombia

10 de diciembre de 2025

1 Definición del problema (Antecedentes)

El mercado bursátil se caracteriza por su alta volatilidad y complejidad, lo que dificulta la predicción precisa de los precios de las acciones. En el contexto colombiano, acciones líderes como **Ecopetrol**, **ISA**, **Grupo Argos** y **GEB** presentan comportamientos influenciados por diversos factores macroeconómicos y técnicos.

El problema abordado en este proyecto es la **predicción del cambio porcentual diario** de estas cuatro acciones principales de la Bolsa de Valores de Colombia (BVC). El objetivo fundamental es utilizar un modelo de aprendizaje supervisado para inferir el comportamiento futuro de los activos en el año 2025, basándose en el aprendizaje de patrones históricos recolectados desde el año 2007 hasta finales de 2024.

Para la ejecución técnica, se utiliza la librería `yfinance` para la obtención de datos históricos y `scikit-learn` para la implementación del modelo predictivo.

2 Definición de estados - Estado inicial

El estado inicial del sistema se define por el conjunto de datos históricos crudos obtenidos y procesados mediante el script `data.py`.

- **Fuente de Datos:** Yahoo Finance (`yfinance`).
- **Activos Analizados:** El archivo obtiene el historial de las siguientes 4 acciones colombianas:
 - `ECOPETROL.CL`
 - `ISA.CL`
 - `GRUPOARGOS.CL`
 - `GEB.CL`
- **Ventana de Tiempo y Visualización:** El historial abarca desde el **2007 hasta la primera semana de diciembre de 2025**. Al ejecutar el código `data.py`, la información se estructura dentro de un *dataset* para posteriormente ser mostrada en una gráfica usando `matplotlib`. En este histórico se visualiza tanto el **precio de la acción** como el **volumen de cambio** (representado mediante gráficos de área apilados).
- **Variables del Estado Inicial:**
 - Fecha (`Date`)
 - Precio de Cierre Ajustado (`Close`)
 - Volumen de Transacción (`Volume`)

3 Definición de función sucesora

La función sucesora en este sistema de IA se implementa principalmente en el archivo `random_forest.py` y consta de dos etapas clave: la transformación de datos y la lógica del algoritmo de aprendizaje.

3.1 Transformación de Datos (Ingeniería de Características)

Antes de aplicar el algoritmo, el estado inicial se transforma para crear indicadores técnicos que sirven como *inputs* (predictores). Se generan variables derivadas como:

- Return_1d: Retorno diario.
- SMA_10: Media Móvil Simple de 10 días.
- **Volatilidad**: Desviación estándar móvil.
- RSI: Índice de Fuerza Relativa.

3.2 Algoritmo de Transición (Random Forest)

Utilizamos la librería `sklearn` para implementar un algoritmo **RandomForestRegressor** de aprendizaje supervisado.

- **Estructura de Bosque**: Este algoritmo genera una secuencia de árboles de decisión independientes, conjunto al cual se le denomina “bosque”.
- **Sistema de Decisiones (Consenso)**: Cada uno de estos árboles recibe la información de entrada (X_t) y llega a una respuesta individual. Para obtener el valor final, se llega a un **consenso**, analizando y agregando las respuestas dadas por los múltiples árboles (promediando sus salidas en el caso de regresión) para elegir la predicción final.
- **Entrenamiento e Inferencia**: Se realiza un entrenamiento estricto hasta la fecha de corte 2024-12-31. Usando los datos hasta esta fecha, el algoritmo aprende los patrones. Posteriormente, para los datos del año **2025**, el modelo debe **inferir** los resultados basándose únicamente en su entrenamiento previo; es decir, busca predecir el porcentaje de cambio día a día sin haber visto estos datos durante la fase de aprendizaje.

4 Prueba de Objetivo

La prueba de objetivo consiste en evaluar la capacidad del modelo para generalizar su aprendizaje en datos no vistos (el futuro).

- **Criterio de División (Split)**: Se utiliza una fecha de corte estricta:

$$\text{DEADLINE} = \text{"2024-12-31"}$$

- **Entrenamiento**: Datos ≤ 2024 .
- **Prueba (Objetivo)**: Datos > 2024 (Año 2025).
- **Métrica de Evaluación**: Se utiliza el **Error Absoluto Medio (MAE)** calculado sobre el precio.

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_{\text{real},i} - y_{\text{predicho},i}|$$

Donde y_{predicho} se reconstruye a partir del retorno inferido:

$$\text{Precio}_{\text{estimado}} \approx \text{Precio}_{\text{real},t} \times (1 + \text{Retorno}_{\text{predicho}})$$

El objetivo se considera “logrado” si el modelo es capaz de seguir la tendencia general del precio en 2025 manteniendo un MAE bajo.

5 Resultados

5.1 Configuración del Algoritmo

Para este experimento, se configuró el algoritmo `RandomForestRegressor` con los siguientes hiperparámetros para balancear la complejidad y el rendimiento:

- **Número de nodos (estimadores):** `n_estimators=150` (150 árboles en el bosque).
- **Profundidad máxima:** `max_depth=5`.
- **Muestras mínimas por hoja:** `min_samples_leaf=5`.

5.2 Comparación y Logro de Objetivo

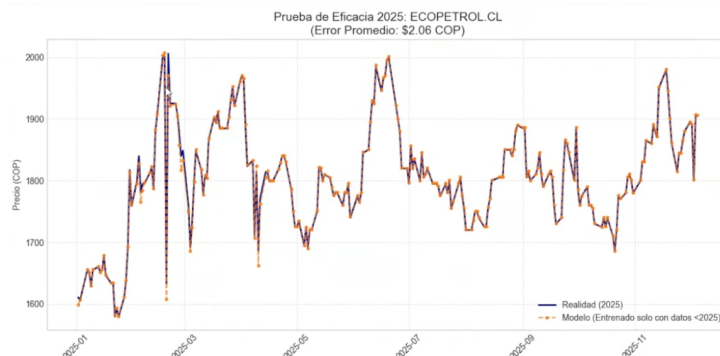
Al ejecutar el código, se genera un gráfico comparativo entre el **precio real de la acción** y el **predicho por el algoritmo**:

1. Visualización:

- **Azul (Realidad):** Representa el precio real de mercado en 2025.
- **Naranja (Modelo):** Representa la predicción del algoritmo basada exclusivamente en el aprendizaje previo a 2025.

2. Análisis de Desempeño:

- El modelo logra capturar la dirección general del mercado.
- Debido a la naturaleza de consenso del Random Forest, las predicciones tienden a ser menos volátiles que el precio real (suavizado).
- El **MAE** resultante indica la desviación promedio en pesos colombianos (COP), permitiendo cuantificar la precisión financiera del modelo.



6 Conclusiones y enlace al repositorio

1. **Eficacia de la Segmentación Temporal:** La separación estricta de datos por fecha (entrenar hasta 2024 y probar en 2025) proporciona una evaluación realista, evitando el sesgo de “mirar al futuro” (*data leakage*).
2. **Robustez del Random Forest:** El uso de un ensamblaje de 150 árboles permite reducir el riesgo de memorizar ruido excesivo del mercado, logrando una generalización aceptable.

3. **Utilidad de las Variables Técnicas:** La inclusión de indicadores como el RSI y la distancia a la media móvil (SMA) demostró ser fundamental para que el modelo entendiera el contexto del precio.

Enlace al repositorio:

El código fuente y la documentación completa se encuentran disponibles en:

<https://github.com/FabianEspitia-it/random-forest-algorithm.git>

7 Referencias

- **L. Khaidem, S. Saha y S. R. Dey (2016):** "Predicting the direction of stock market prices using random forest," *Applied Mathematical Finance*, vol. 00, no. 00, pp. 1–20.
Disponible en: <https://arxiv.org/pdf/1605.00003>
- **Librería Scikit-Learn:** *Ensemble methods and RandomForestRegressor documentation.*
<https://scikit-learn.org/stable/modules/ensemble.html>
- **Yahoo Finance API (yfinance):** *Documentation and usage.*
<https://pypi.org/project/yfinance/>