



AVANCE DEL PROYECTO

Modelo Predictivo de Criptomonedas a corto plazo



ASIGNATURA: METODOS NUMERICOS

INTEGRANTES:
ANTHONY CONTRERAS
KEVIN GARCIA

FECHA DE ENTREGA:
03/08/2025

Contenido

OBJETIVOS	2
DESARROLLO	2
CONCLUSIONES.....	9
REFERENCIAS.....	9

OBJETIVOS

- Avanzar en el proceso de creación del proyecto de manera coherente.
- Desarrollar las fases del proyecto.
- Mostrar los avances del proyecto hasta el momento.

DESARROLLO

Vamos a crear un modelo el cual tendrá como principal objetivo el predecir el precio de las criptomonedas a futuro, teniendo como antecedentes un grupo de datos. Estos datos nos muestran de cómo la moneda se ha comportado en el mercado. Todo esto pensando en que el modelo sea aplicable en un corto plazo de tiempo.

El problema con el mercado de las criptomonedas es respecto a que no conocemos como actuará en el futuro y realizar prácticas de trading como el scalping se vuelve complicado. Teniendo como principal enemigo a la probabilidad o riesgo vs beneficio. Esto lleva a que minimizar los riesgos sea lo más importante para poder tener ganancias al realizar trading. Una forma de minimizar los riesgos es poder usar o crear herramientas que nos ayuden al análisis del mercado cripto, siendo esta la razón más importante en la realidad para tener la necesidad de crear un modelo preventivo como el de este informe.

Teniendo en cuenta lo antes mencionado, el principal objetivo será reducir dicho riesgo y facilitar realizar ese tipo de actividades dentro del mundo cripto.

Dentro de este proyecto, se dispone de datos extraídos por un datasheet.

El conjunto de datos incluye un archivo CSV para cada moneda. El historial de precios está disponible diariamente desde el 28 de abril de 2013. Este conjunto de datos contiene información histórica sobre los precios de algunas de las principales criptomonedas por capitalización de mercado.

Vale recordar que nuestro proyecto este compuesto de 3 fases, las cuales son:

- **Análisis:** Es la parte más importante ya que si se omiten consideraciones importantes o si esta fase es mal realizada, las siguientes fases presentaran fallas que necesitaran de un tiempo considerable para poder corregirlas.
- **Simulación y Ajuste:** Dentro de esta fase se crea el modelo basado en todas las pautas y métodos considerados para la creación de este. Dentro de esta fase se debe aplicar directamente al lenguaje de programación Python. Se toma consideraciones para la creación del formato visual, ingreso de datos y procesos matemáticos correspondientes. además de ir ajustando el modelo según las necesidades o dificultades que se presenten en esta fase.

También se identifican problemas y fallas en el mismo, estos se irán corrigiendo poco a poco hasta tener un modelo confiable basado en los objetivos generales del proyecto.

- **Predicción y aplicación:** Es la última fase del proyecto, aquí se busca aplicar el modelo creado a diferentes criptomonedas y verificar su exactitud y precisión. El objetivo principal es poder identificar fallas en tiempo real del mismo y sus limitaciones al momento de aplicarlo a diferentes mercados con diferentes tendencias y volatilidades.

Seguidamente se realizan tablas y graficas para comprender como está trabajando el modelo creado. Además de poder obtener las conclusiones del proyecto junto a una comparación del modelo creado VS el análisis independiente de una persona con conocimientos en el área de las criptomonedas y su mercado.

FASE 1: Análisis

En esta fase se va a considerar filtrar los datos de la mejor manera posible. Se va a utilizar a una moneda modelo a Ethereum y con ella vamos a verificar que los datos no sean nulos o corruptos. También se va a analizar todas las posibles alteraciones o corrupciones que pueden afectar al modelo en la fase 2. Esto como objetivo de tener mayor visión del área que se va a trabajar y cubrir todos los posibles problemas que se tendrán que superar.

SNo	Name	Symbol	Date	High	Low	Open	Close	Volume	Marketcap
1	Aave	AAVE	2020-10-05 23:59:59	55,11236	49,7879	52,67503	53,21924	0	89128129
2	Aave	AAVE	2020-10-06 23:59:59	53,40227	40,73458	53,29197	42,4016	583091,5	71011441

Disponemos de 2160 datos diarios que se dividen en 5 columnas útiles que son:

- Sno = Es el número de día.
- Nombre
- Fecha
- High
- Low

- Precio de Apertura
- Precio de Cierre
- Volumen
- Capital de Mercado

De todos estos datos, se tomó los que nos importan para el modelo o función principal. Estas variables van a ser:

- Día
- Precio de Cierre
- Precio de Apertura (El que se va a obtener)
- Low (El que se va a obtener)
- High (El que se va a obtener)

Todo depende de que queremos predecir en nuestro modelo.

Como variables de complemento para el análisis del grafico o función obtenida serán:

- Volumen
- Capital

Como vamos a buscar predecir el siguiente valor, entonces trabajamos con el tiempo y con los precios (pueden ser High, Low, Close o Open). Predecir de Open a Open es lo mas optimo para el proyecto, pero también podemos usar los otros 4 valores, dependería de lo que queremos predecir. Un Low un High o un Open.

Con todo lo antes mencionado, se consiguió un total de 15120 datos individuales. Dado a la cantidad de datos, se procedió a realizar la verificación de cada uno por medio de uso de herramientas que nos ayudan a tener mayor control de dichos datos.

Aplicando funciones sencillas de Excel, se pudo verificar la integridad de los datos, además se creó un diagrama grafico de los datos disponibles para comprobar la existencia de datos atípicos y sus posibles complicaciones en nuestro modelo.

Las ecuaciones lógicas utilizadas fueron:

=SI(CONTAR.SI(E2:J2161;0)>0;"Hay ceros"; "No hay ceros ")

=SI(CONTAR.SI(E2:J2161; "<0")>0; "Hay negativos"; "No hay negativos")

=SI(CONTAR(E2:J2161)=CONTARA(E2:J2161); "Todo es numérico"; "Hay texto o errores")

además de graficar el comportamiento de la moneda.

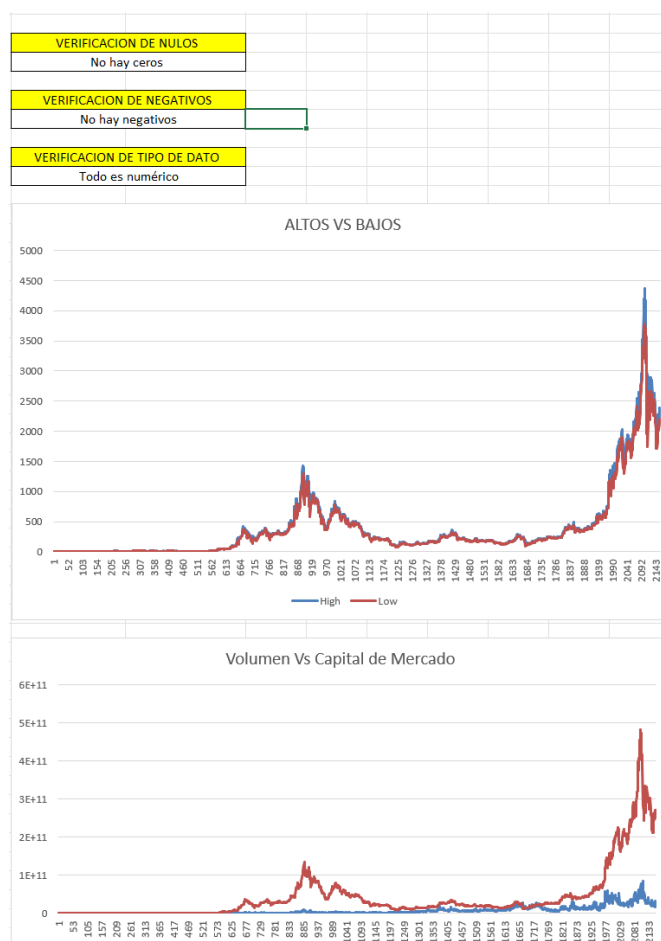


Ilustración 1-Análisis y segmentación de los Datos

Seguido se pasó a revisar a detalle las gráficas en búsqueda de manipulación del mercado. Estas son normales dentro del mercado de las criptomonedas. Sin embargo, para los datos obtenidos, no tenemos una manipulación considerable que requiera eliminar o rechazar ciertos datos diarios. Al comparar las gráficas de capital de mercado y la gráfica de altos y bajos, tenemos los siguientes puntos importantes a considerar para el modelo predictivo, además de incluir consideraciones importantes para la toma de datos:

- 1- Que los movimientos dados de la moneda son coherentes con el movimiento del capital y volumen de esta. Por lo que no son manipulaciones o valores atípicos relevantes.
- 2- Que su movimiento se mantiene bastante suave y es continuo. Lo cual nos genera más garantía de obtener un modelo preciso y exacto.
- 3- Al tener conciencia de la volatilidad del mercado crypto, se consideró aplicar filtros extras a los datos que se toman en tiempo real. Algunos de ellos son:
 - a. Poner una tolerancia para el movimiento (Al alza o a la baja) del valor o precio de la moneda (Apoyado con la ayuda del punto B).

- b. Agregar más herramientas externas usadas en el mundo del estudio cripto para mejorar la toma de los datos al momento de aplicar el modelo al mercado en tiempo real. (Estas serán nombradas y explicadas en la fase 2 y 3 del proyecto).
 - c. Considerar y tomar en cuenta las alteraciones mundiales que se aplican al mercado cripto (Como las pandemias, guerras, tratos internacionales de comercio y cierre o apertura de bancos internaciones o regionales. Todo lo antes mencionado será más profundizado en la fase 3 de este proyecto)
- 4- Los movimientos y volatilidad de las criptomonedas dependen del tipo de moneda. En este caso se utiliza a Ethereum. Una moneda reconocida por su poca volatilidad y manipulación en el mercado. Esto debido a que se posiciono como una de las monedas mas famosas y confiables del mercado cripto. Pero si el modelo se aplica a otras monedas del mercado se debe considerar lo siguiente:
- a. Criptomonedas con precios o valores muy pequeños son más fáciles de manipular. Por eso los modelos predictivos se vuelven muy pocos precisos y a veces inútiles.
 - b. Criptomonedas con poco tiempo de nacimiento o con poco tiempo de existencia en el mercado, estas son fáciles de manipular por sus costos iniciales y las descompensaciones en el volumen y capital del mercado. Aplicar un modelo predictivo a este tipo de monedas resulta inútil en la mayoría de los casos, ya que estas monedas aun no crean una tendencia o aun no se normalizan en el mercado.

Después de mencionar los anteriores puntos a considerar antes de tomar los datos o una moneda como objetivo predictivo, pasamos con la preparación final de nuestros datos a formato .CSV para poder trabajar de mejor manera dentro del lenguaje de programación Python.

Herramientas técnicas importantes que considerar

Dentro de este segmento de la fase de análisis, se considero conocer de herramientas actuales que se utilizan para el análisis de las criptomonedas en el mercado cripto.

Estas herramientas son parte fundamental para el trading y la predicción, aunque solo buscan ayudar a obtener una aproximación futura del precio.

Las siguientes herramientas serán utilizadas como un apoyo:

1. **Soporte y resistencias:** Son líneas importantes que son identificadas para comprender donde el precio rebota o se detiene.
2. **Tendencias:** Normalmente son líneas diagonales que marcan la dirección del mercado, sea al alta o a la baja.
3. **Canales:** Se usan para tener una marca de rangos dentro del precio de una tendencia.
4. **Medias móviles:** Son indicadores super utiles, las más importantes para seguir el precio y su movimiento son:
 - a. Media móvil exponencial (EMA).
 - b. Media móvil simple (SMA).
5. **Indicadores técnicos:** Sirven para analizar el comportamiento del movimiento y como este va a actuar en el futuro. además de que nos ayudan a entender como esta fluctuando la moneda. Algunos de ellos son:
 - a. Indicé de fuerza relativa (RSI), mide la fuerza del movimiento.
 - b. Movimiento promedio de convergencia y divergencia (MACD).
 - c. Bandas de Bollinger (Importantes para entender la volatilidad).
 - d. Perfiles de Volumen (Graficas que muestran la cantidad de volumen en el tiempo).
6. **Niveles Fibonacci:** Usados para detectar las entradas y salidas de zonas donde el precio se corrige. Esto en búsqueda de normalizarse.

Regresión Polinomial por Mínimos Cuadrados

Es una técnica de ajuste de curvas que permite modelar relaciones no lineales entre una variable independiente **X** y una dependiente **Y**. El método de mínimos cuadrados consiste en encontrar los coeficientes del polinomio que minimizan la suma de los cuadrados de los errores entre los valores reales y los valores predichos por el modelo.

Para un polinomio de grado n , la forma general es:

$$P(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$$

El sistema de ecuaciones resultante se resuelve mediante álgebra matricial bajo la idea de $Ax=b$ usando la transpuesta y la matriz cofactores.

Método de Euler hacia adelante

El método de Euler es una técnica de aproximación numérica para resolver ecuaciones diferenciales ordinarias. Se utiliza para predecir el valor futuro de una variable, dadas su tasa de cambio (derivada) y su valor actual.

La fórmula básica es:

$$y_{n+1} = y_n + h * f(x_n, y_n)$$

$$P_{n+1} = P_n + h * f(t_n)$$

Donde:

- y_n valor actual
- h : paso (en este caso, 1 día)
- $f(x_n, y_n)$: derivada o tasa de cambio en x_n

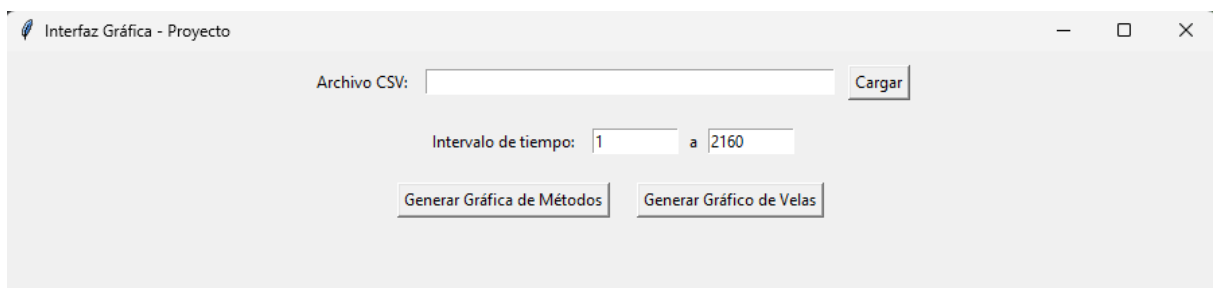
En este proyecto, se aplica para estimar el precio de cierre del día siguiente usando la derivada del polinomio ajustado.

FASE 2: Simulación y Ajuste

Esta fase del proyecto ya incluye la aplicación de todo lo visto en la Fase 1, pero aplicado directamente a código Python por medio del programa.

CLASE interfaz

La interfaz se ve la siguiente manera y tiene como entrada el archivo CSV y el intervalo de trabajo.



Dentro de esta interfaz utilizamos

```
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
import matplotlib.pyplot as plt
```

Estas librerías permiten mostrar el modelo gráfico y la predicción de este. Para esto tenemos las siguientes funciones encargadas de unificar el proyecto:

```
def __init__(self, root):
def cargar_archivo(self):
def graficar_metodos(self):
def graficar_velas(self):
def mostrar_grafica(self, fig):
```

La función cargar_archivo (). Tiene que usar el path del archivo CSV y lo deja listo para ser usado en otras funciones. Este tiene relación con el apartado usado en la interfaz mostrada donde se debe escoger el archivo CSV. (Usa la clase OpenFile)

La graficar_metodos (). Primero detecta que este cargado correctamente el archivo y seguido llama al método de mínimos cuadrados para poder extraer los datos y almacenarlos en variables.

Después se llama al método de Euler y también se almacenan los datos. Lo demás de la función es ajustar todo usando las librerías graficas antes mencionadas a partir de todo lo obtenido de los métodos numéricos usados.

La Graficar_velas(). Nuevamente verifica que se encuentre cargado el archivo y seguido usa los datos referentes al intervalo de análisis ingresado por el usuario dentro de la interfaz.

La mostrar_grafica (): Aquí con el uso de CANVAS de nuestras librerías mencionadas, dibujamos el modelo respecto al tiempo (en el eje x).

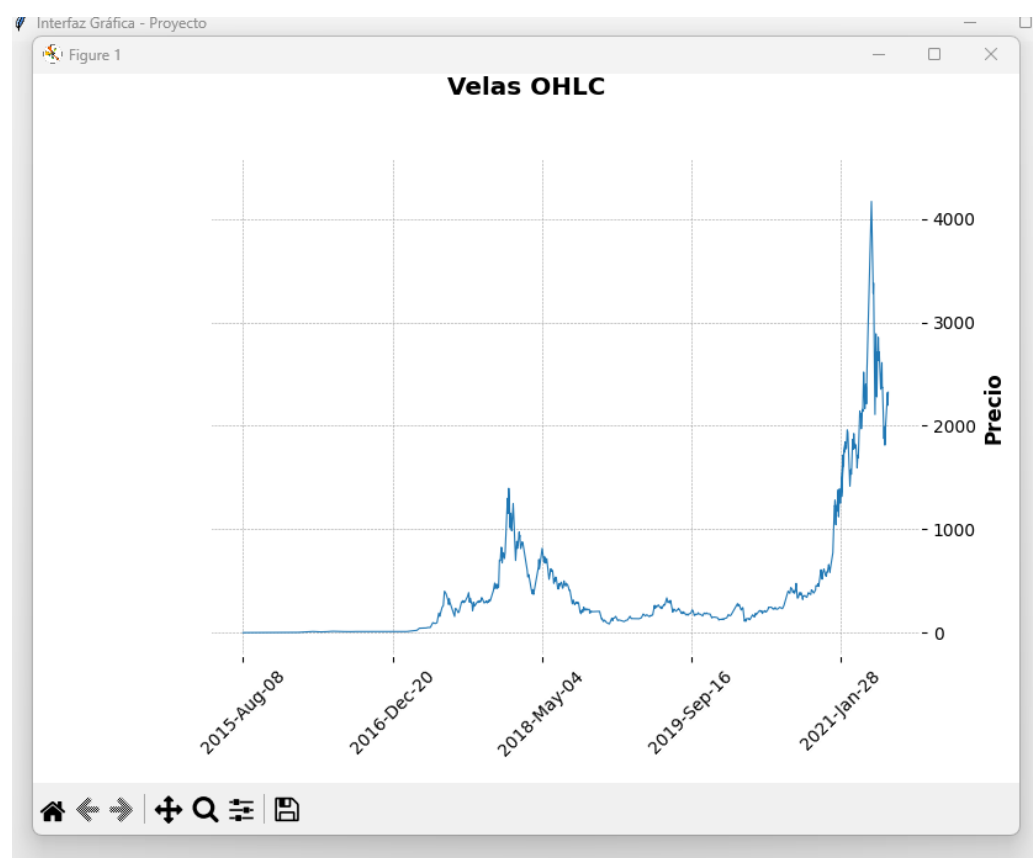
CLASE graficos

Utilizamos las siguientes librerías para poder usarlas como APOYO y AYUDA.

```
import pandas as pd
import mplfinance as mpf
```

Es una clase la creamos solo como apoyo que nos ayuda a verificar como se vería el modelo respecto graficado, nuevamente esto respecto al tiempo.

Esta clase recibe directamente la clase MetodosNumericos y saca los datos para poder graficarlos. Obtenemos lo siguiente:



CLASE OpenFile

Esta clase es la mas importante ya que se encarga de recibir los datos de nuestro CSV según sus columnas y las filtra como una lista. Elimina los a valores NaN y verifica el formato de este para asegurar su correcta aplicación dentro del modelo.

Tiene la función Open que recibe el archivo y lo filtra y luego una función Data que se encarga de crear una lista de datos de cada columna del DataFrame, retornando una Lista de listas.

Si existen fallas se mostrarán directamente en pantalla gracias al control de errores.

CLASE main

Es la que une todas las clases además de juntar los datos y permitir correr la aplicación. Aunque también tiene una función run () que nos ayudara a tener el tiempo de ejecución total de nuestro programa.

CLASE metodosNumericos

Esta clase tiene todos los procesos matemáticos y aplicaciones de los métodos explicados en la Fase 1.

La __init__ () Se encarga de pasar los datos usando el OpenFile y asignarlos en datos que usaremos dentro de esta clase.

La solution () Es la que nos permite facilitar el envío de los resultados de la función minimosCuadrados a otras clases. Inyecta los datos respecto a los datos de la clase obtenidos por la función anterior.

La solucionEuler (Recibe X0). Es la que nos permite facilitar el envio de los resultados de Euler a las otras clases. Inyecta los datos necesarios a la funcion solución_euler según la funcion solution () y la init(). Donde X0 es el valor final que conocemos.

La resolver_Ecuaciones (A, b): Es una de las mas importantes ya que es la usada para poder resolver el sistema de ecuaciones mediante matrices utilizando eliminación Gaussiana para el método mínimos cuadrados. Donde A es la matriz con los coeficientes y b los términos independientes.

La función mínimos_cuadrados(X, Y, grado): Método para calcular la regresión lineal por el método de mínimos cuadrados se calcula el comportamiento de low dado high. Utilizamos grado 5 por que fue el que mejor se ajustó a la moneda de análisis Ethereum.

x: Los valores independientes (High).

y: Los valores dependientes (Low).

grado: Grado con el que se espera trabajar.

Coefficientes, y_ajustada y error cuadrático medio.

Aquí se realiza los cálculos mediante el uso de las matrices para poder crear el modelo que vamos a graficar.

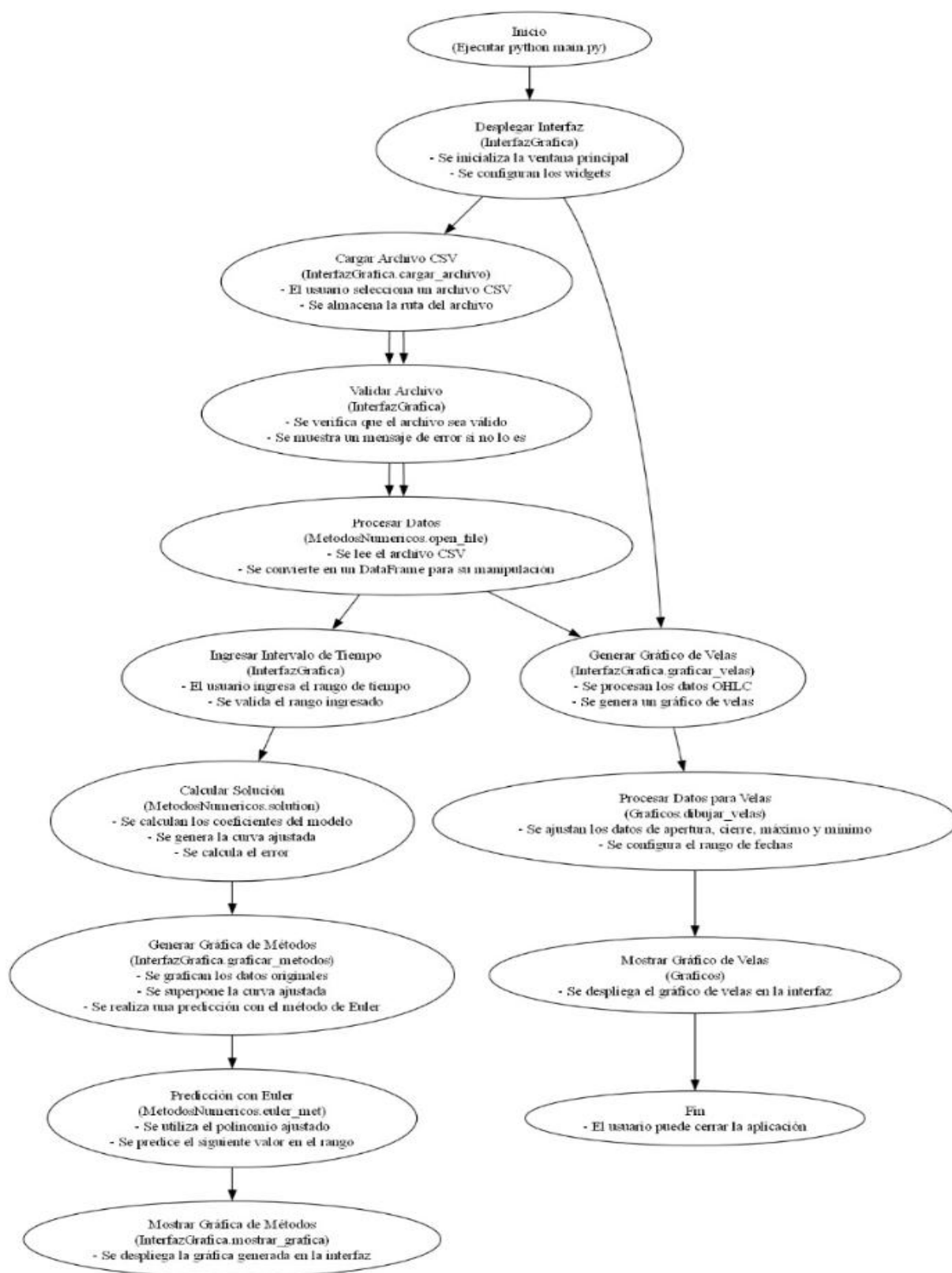
La funcion derivar(coe): Simplemente devuelve la derivada que es necesaria para aplicarse en el Euler_met().

La funcion evaluarP(coe, x): Evalúa al polinomio respecto a un punto que también es importante para poder aplicarse en Euler_met(). Recibe los coeficientes del polinomio y la X inicial o la que vamos a evaluar.

La funcion Euler_met(coe, x0, h): Es donde se aplica el metodo de Euler usando como ayuda a los cálculos las 2 funciones anteriores. Devuelve el valor de y_predecido o y1 como tal. Donde h el número de día a predecir, x0 el valor inicial de ultimo valor del intervalo.

La funcion Calcular_Variacion (Open_Values, Close_values): Simplemente calcula la variación diaria del precio entre la apertura y el cierre. Retorna la lista de variación diaria.

Modelo de relaciones entre funciones:



FASE 3: Predicción y aplicación

En esta fase empezamos a realizar las simulaciones y comparar los valores obtenidos con un análisis independiente al modelo.

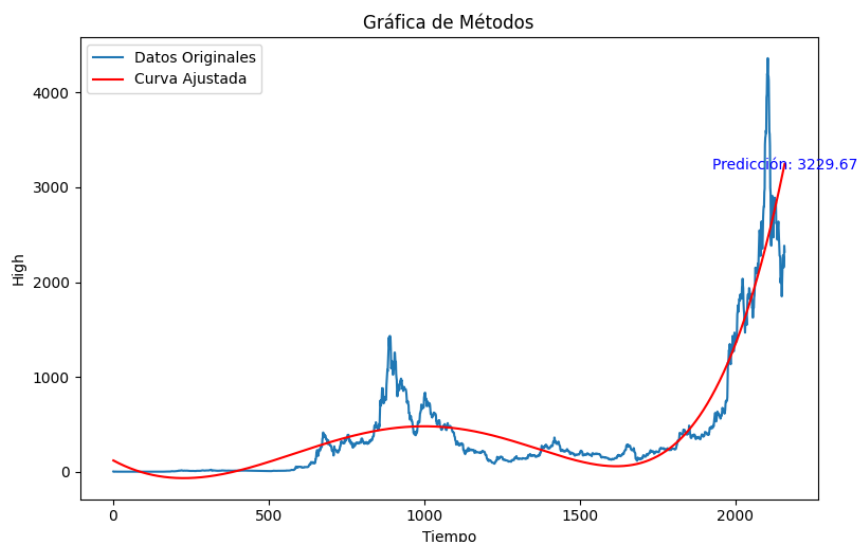
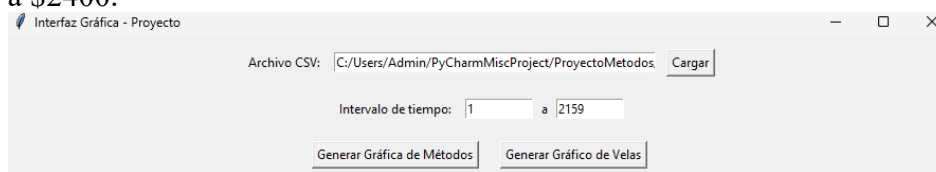
También se verificará el error del mismo modelo. Vale recordar que las monedas escogidas deben considerarse bajo lo mencionado dentro de la FASE 1.

1- ETHERUM (Ejemplo de 1º tipo- Consideraciones)

Usando un modelo de High Vs Tiempo (días) con un rango del día 1 al día 2159, tenemos una predicción para el día 2160 de \$3229.67.

El valor real del día 2160 es: \$2346,294.

Esta diferencia fue causada por como actuó el volumen en el día 2159, donde tuvimos desde el día 2157 una pérdida de volumen considerable. Esto provocó una caída de precio en la moneda muy grave dentro de un rango de 60 a 90 días. Pasando de \$4100 a \$2400.



Todo esto fue provocado ya que en estas fechas recién empezó la pandemia por COVID 19 y llevo a una caída económica en el mercado cripto de manera global.

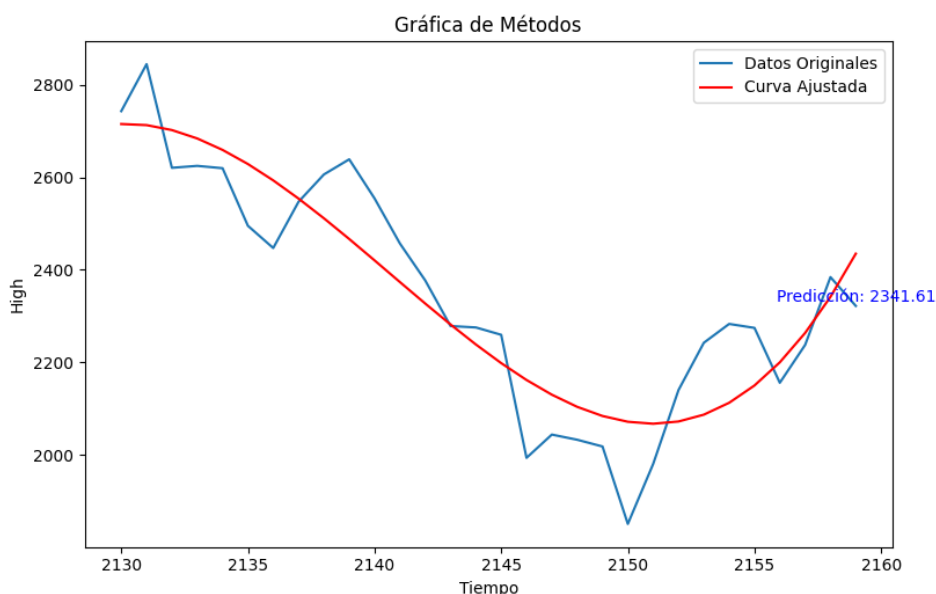
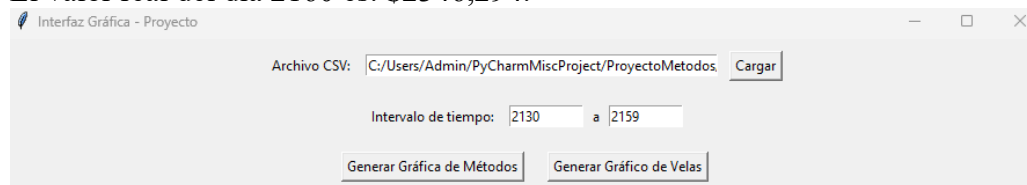
Siendo este un ejemplo claro de que se debe considerar los puntos mencionados en el análisis para poder aplicar el modelo predictivo y a la vez nos deja claro que la economía internacional es muy sensible a todo tipo de cambios relacionados a la mismo alrededor del mundo.

Entonces aplicamos un intervalo de estudio mas pequeño para aumentar le eficiencia de los métodos. Ahora tomamos un intervalo del día 2100 al 2159, tenemos una predicción para el día 2160 de \$2164.28
El valor real del día 2160 es: \$2346,294.

Ahora la diferencia de error fue mucho menor, así que finalmente tomamos un intervalo mas pequeño para reducir el error, del día 2130 al día 2159. El resultado fue increíble.

Predicción para el día 2160 de \$2341.61

El valor real del día 2160 es: \$2346,294.



El error es prácticamente de \$5 en el HIGH lo cual representa un error de 0.0019986 o equivalente a un error de 0.2%!.

Finalmente hacemos el análisis con herramientas de estudio para las criptomonedas (Mencionadas en el análisis).

Lo cual nos permite verificar que el modelo creado es muy efectivo, ya que realizando el análisis uno mismo, podemos ver que tenemos un techo en el precio, justo en \$2348. Lo cual es un límite de precio al máximo valor de la moneda (Que la moneda topo y luego regreso) para el siguiente día y nuestro modelo fue capaz de predecir con éxito el valor High.



Para terminar, luego de realizar 3 predicciones con diferentes rangos de trabajo (días) y en una fecha donde se debe considerar la situación económica mundial (COVID 19) que altero la tendencia del mercado junto a la perdida de Volumen y capital de mercado. El modelo fue sorprendentemente efectivo aun teniendo casi todas las dificultades mencionadas en la Fase 1.

Terminamos mostrando la siguiente tabla de valores predichos vs los reales en diferentes monedas y su error, que se realizaron de la misma manera como el anterior ejemplo.

MONEDA	VALOR DEL METODO	VALOR REAL	ERROR %
Ethereum (Estable)	2341.61	2346.29	0.2%
Bitcoin (Super Estable)	12564.87	12568.41	0.03%
DogeCoin (Normal)	5.62	5.70	1.4%
Matic (Movimiento Brusco)	3.44	3.59	4.30%
Luna (Movimiento fluctuante)	7.26	7.51	3.86%

Con los tiempos completos de ejecución del programa fueron:

```
PS C:\Users\Admin\PyCharmMiscProject\ProyectoMetodos> python main.py
Tiempo de Ejecución 5.08 seconds
PS C:\Users\Admin\PyCharmMiscProject\ProyectoMetodos> python main.py
Tiempo de Ejecución 8.16 seconds
PS C:\Users\Admin\PyCharmMiscProject\ProyectoMetodos> python main.py
Tiempo de Ejecución 22.13 seconds
PS C:\Users\Admin\PyCharmMiscProject\ProyectoMetodos> python main.py
```

Fueron altos debido a que la modularidad del modelo y que estamos midiendo la respuesta directa con la MAIN, pero cada función trabajó con los tiempos de:

Método	Descripción	Complejidad Aproximada	Tiempo estimado (para 2000 datos)
minimos_cuadrados	Armas matrices y resuelve sistema de ecuaciones	$O(n^2)$ o más (matricial)	Medio-Alto (0.1 a 0.5 s)
resolver_ecuaciones	Eliminación gaussiana	$O(n^3)$	Medio (0.01 a 0.2 s)
solution	Llama a minimos_cuadrados	Lo que tarde esa función	\approx lo mismo que minimos_cuadrados
solution_euler	Usa solution y luego Euler	Depende del polinomio	Bajo-Medio (0.01 a 0.1 s)
euler_met	Evalúa polinomio y su derivada	$O(n)$	Bajo (< 0.01 s)
evaluarP	Evalúa polinomio en un punto	$O(n)$	Muy Bajo (< 0.001 s)
derivar	Deriva el polinomio	$O(n)$	Muy Bajo (< 0.001 s)
calcular_variacion	Resta dos listas	$O(n)$	Muy Bajo (< 0.001 s)

Por lo tanto, podemos dar como exitoso y cumplido el objetivo de este proyecto dado que se hizo las simulaciones con las condiciones menos favorables para el modelo y su resultado fue excelente.

Enlace actual de Github

<https://github.com/7heAnsw3r/ProyectoMetodos>

Estadísticas de contribuciones del repositorio.

CONCLUSIONES

REFERENCIAS

- [1] *Pandas Documentation - User Guide*, [En línea]. Disponible en: https://pandas.pydata.org/docs/user_guide/index.html#user-guide. [Accedido: 8-jul-2025].
- [2] ESPOL, “Mínimos Cuadrados con Python,” *Análisis Numérico*, [En línea]. Disponible en: <https://blog.espol.edu.ec/analisisnumerico/minimos-cuadrados-con-python/>. [Accedido: 7-jul-2025].
- [3] GATE.io, “The Top 10 Trading Tools in Crypto,” *Gate Learn*, [En línea]. Disponible en: <https://www.gate.com/es/learn/articles/the-top-10-trading-tools-in-crypto/4883>. [Accedido: 7-jul-2025].

- [4] B2BinPay, “10 mejores herramientas de análisis blockchain para su negocio en 2024,” *B2BinPay News*, [En línea]. Disponible en: <https://b2binpay.com/es/news/10-best-blockchain-analysis-tools-for-your-business-in-2024>. [Accedido: 9-jul-2025].
- [5] Underdog Trading, “Que son las criptomonedas,” *YouTube*, 5 de abr. de 2024. [En línea]. Disponible en: <https://www.youtube.com/watch?v=8ZQ7-qCWTrE&t=35s>. [Accedido: 7-jul-2025].
- [6] Alfonso Santiago, “Matemático explica cómo funciona BLOCKCHAIN y las criptomonedas, y enumera sus virtudes y DEFECTOS” *YouTube*, 17 de oct. de 2022. [En línea]. Disponible en: https://www.youtube.com/watch?v=2fRYPDS_jrM. [Accedido: 11-jun-2025].
- [7] Emprender Simple, “Cómo Analizar Criptomonedas y Saber Las Que Subirán ,” *YouTube*, 7 de feb. de 2023. [En línea]. Disponible en: <https://www.youtube.com/watch?v=nheLwt4kyzY>. [Accedido: 24-jun-2025].
- [8] Fabián Bernal, “Bitcoin: 4 claves para entender la más grande de las criptomonedas y qué riesgos tiene | BBC Mundo,” *YouTube*, 9 de feb. de 2022. [En línea]. Disponible en: <https://www.youtube.com/watch?v=C-3aYnhF6Io>. [Accedido: 24-jun-2025].
- [9] Alex Ruiz, “Cómo Invertir En Criptomonedas ,” *YouTube*, 27 de mar. de 2023. [En línea]. Disponible en: https://www.youtube.com/watch?v=-GOaO_e_xA8. [Accedido: 30-jun-2025].