

PREDICCIÓN DE PRECIOS DE LA ACCIÓN APPLE MEDIANTE MODELOS DE REGRESIÓN POLINOMIAL

Milca Correa, Camila Zapata
& Arturo Santillanes



OBJETIVOS A ALCANZAR

- **Recopilación de Datos:** Obtener un conjunto de datos históricos de los precios de una acción en la bolsa de valores, que incluya fechas y precios de cierre.
- **Análisis Exploratorio de Datos:** Realizar un análisis detallado de los datos para comprender su distribución, tendencias y posibles patrones temporales.



OBJETIVOS A ALCANZAR

- **Creación de Modelos de Regresión Polinomial:** Implementar modelos de regresión polinomial de grado 1 a 5 para ajustar las curvas a los datos históricos de precios de la acción.
- **Evaluación de Modelos:** Utilizar métricas de evaluación, como el error cuadrático medio (MSE) o el coeficiente de determinación (R^2), para determinar cuál de los modelos de regresión polinomial se ajusta mejor a los datos históricos.



OBJETIVOS A ALCANZAR

- **Predicción del Precio:** Utilizar el modelo seleccionado para predecir el precio de la acción para el día siguiente.
- **Evaluación de la Predicción:** Comparar las predicciones realizadas por el modelo con el precio real de la acción del día siguiente y calcular métricas de rendimiento, como el error absoluto medio (MAE) o el error porcentual absoluto medio (MAPE), para evaluar la precisión de las predicciones.



MODELO QUE REPRESENTA EL PROBLEMA

- El problema se puede representar mediante un modelo de regresión polinomial, que busca encontrar la mejor curva polinómica (de grado 1 a 5) que se ajuste a los datos históricos de precios de una acción en la bolsa de valores. En términos matemáticos, el modelo de regresión polinomial se expresa de la siguiente manera:

$$y = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$

- y representa el precio de cierre de la acción.
- x representa la fecha (o algún otro factor temporal relevante).
- $a_0, a_1, a_2, \dots, a_n$ son los coeficientes del polinomio que se ajustan para minimizar la diferencia entre las predicciones del modelo y los precios reales.

CÓDIGO

```
m = len(data)
x = np.linspace(0, m-1, m)
y = data["Adj Close"].to_numpy()

# y = b0 + b1*x

m = len(y)
beta = np.random.rand(2)
ones = np.ones([m, 1])
x_factor = np.reshape(x, [m, 1])
X = np.hstack( (ones, x_factor) )
y_vector = np.reshape(y, [m, 1])

# y = b0 + b1*x + b2*x**2
beta2 = np.random.rand(3)
x2_factor = np.reshape(x**2, [m, 1])
X2 = np.hstack( (ones, x_factor, x2_factor) )

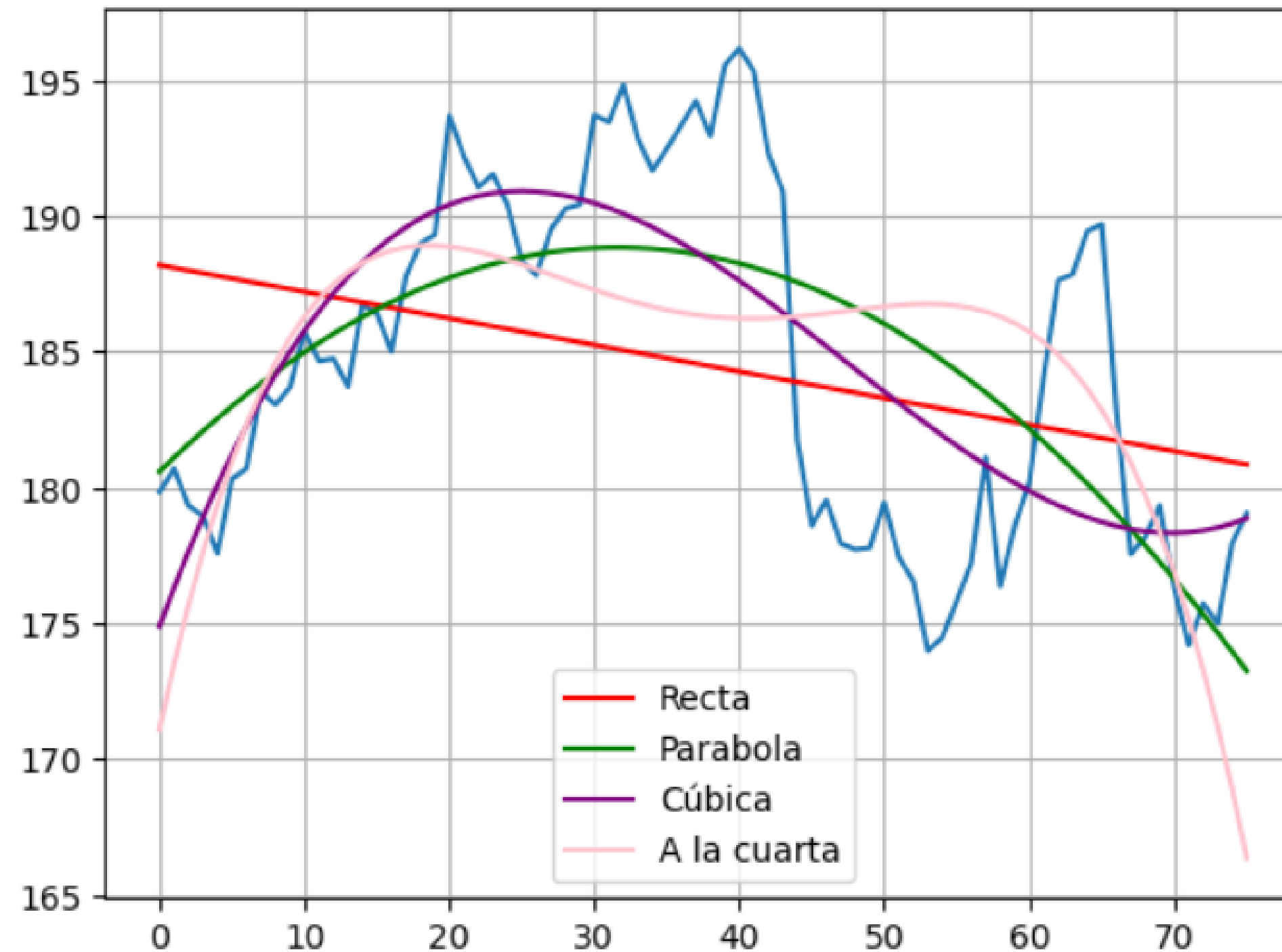
# y = b0 + b1*x + b2*x**2 + b3*x**3
beta3 = np.random.rand(4)
x3_factor = np.reshape(x**3, [m, 1])
X3 = np.hstack( (ones, x_factor, x2_factor, x3_factor) )

# y = b0 + b1*x + b2*x**2 + b3*x**3 + b4*x**4
beta4 = np.random.rand(5)
x4_factor = np.reshape(x**4, [m, 1])
X4 = np.hstack( (ones, x_factor, x2_factor, x3_factor, x4_factor) )

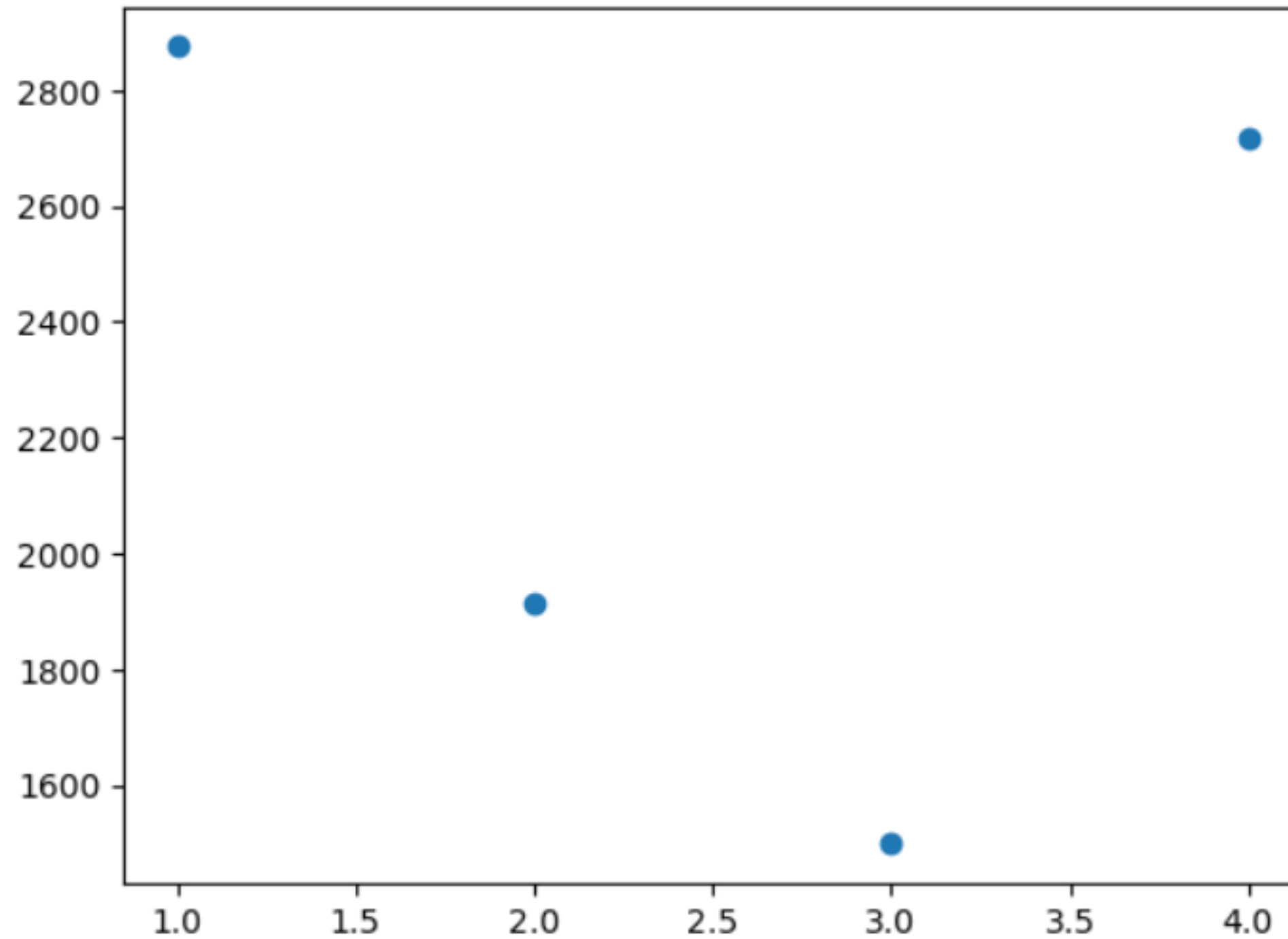
# y = b0 + b1*x + b2*x**2 + b3*x**3 + b4*x**4 + b5*x**5
#beta5 = np.random.rand(6)
#x5_factor = np.reshape(x**5, [m, 1])
#X5 = np.hstack( (ones, x_factor, x2_factor, x3_factor, x4_factor, x5_factor) )

def funcion_de_costo(beta, X, y):
    m, n = X.shape
    beta = np.reshape(beta, [n, 1])
    y_pred = X @ beta
    error = y_pred - y
    return (error.T @ error)[0][0]
```

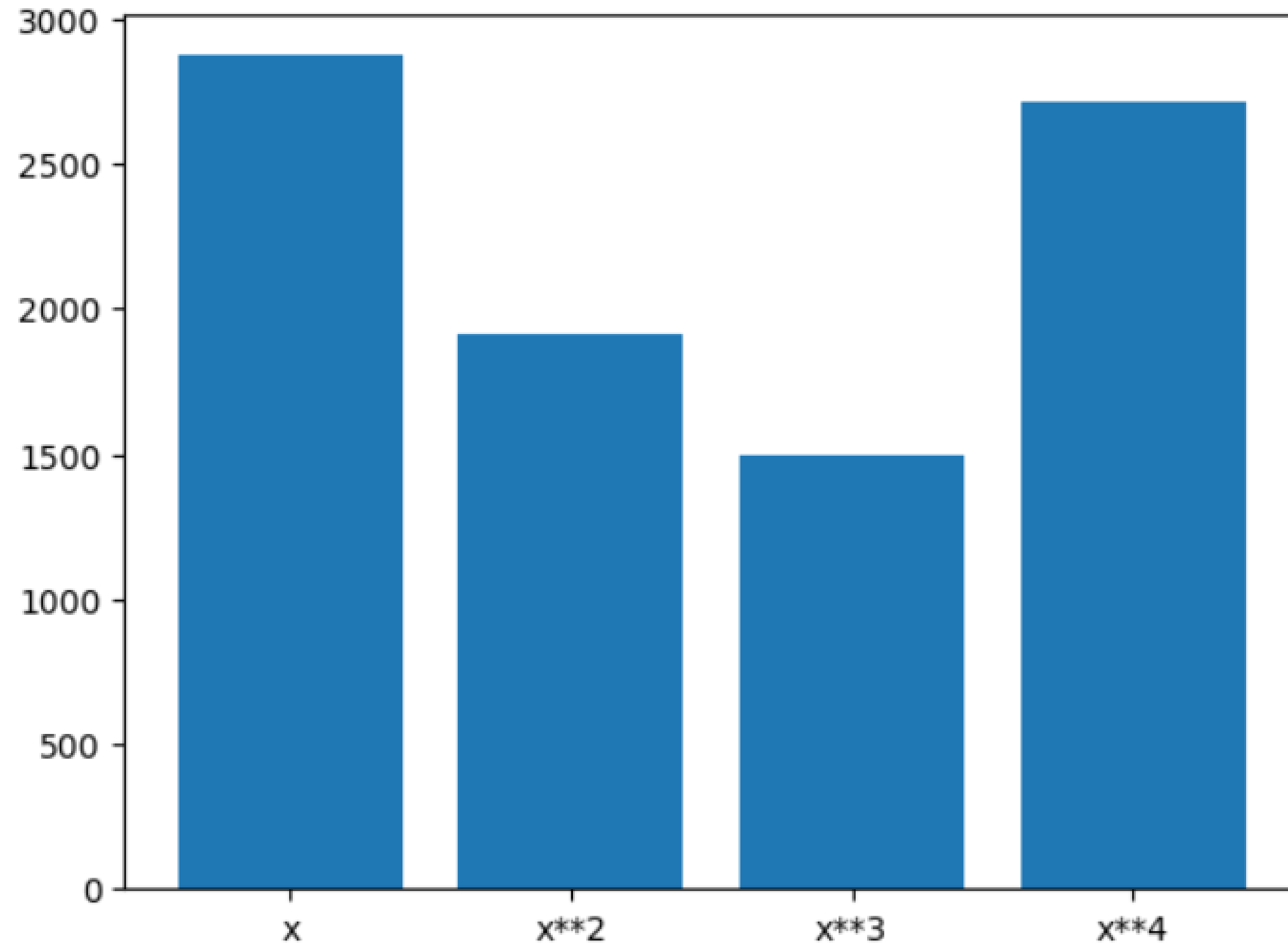
VISUALIZACIÓN POLINOMIOS



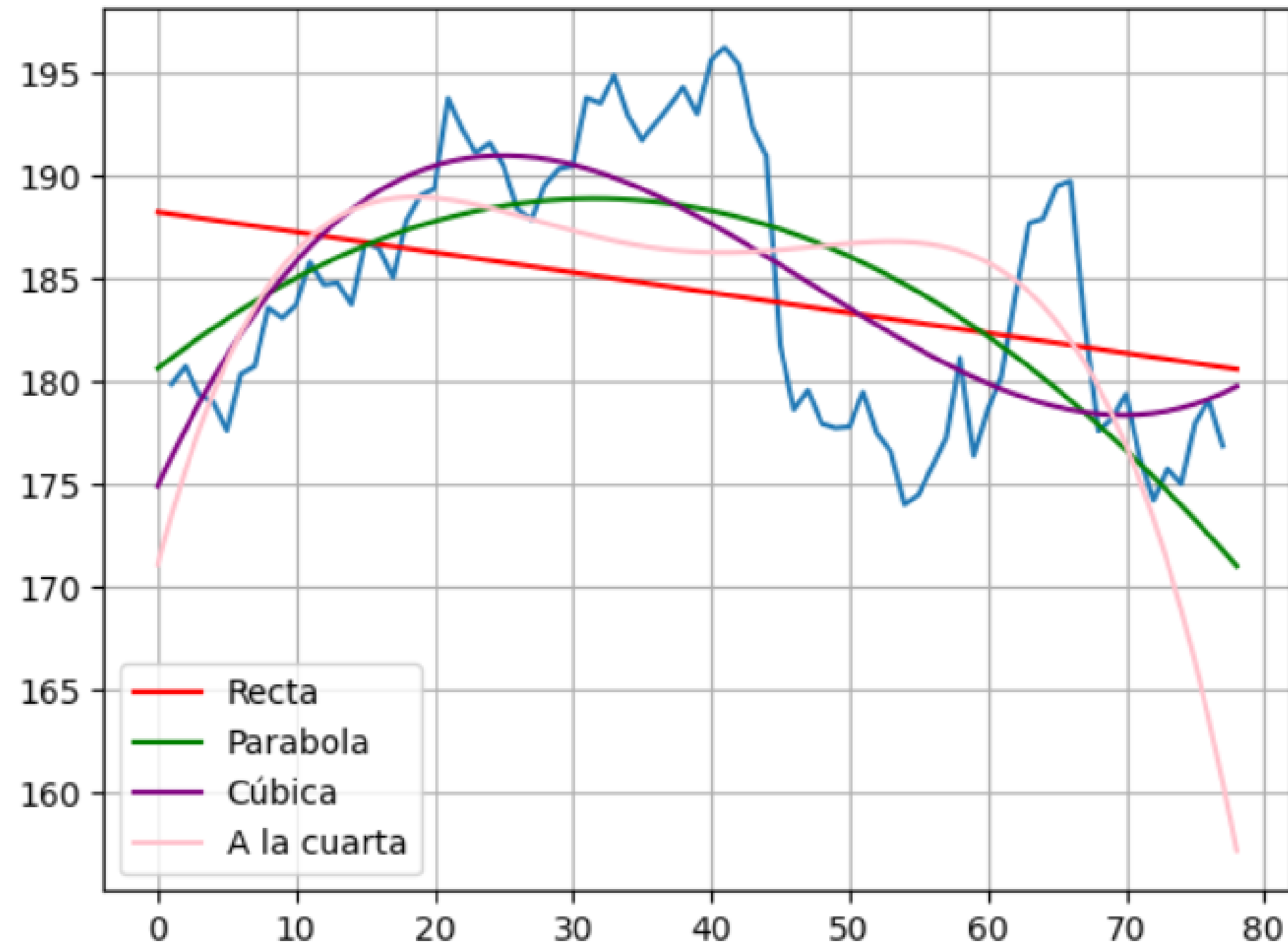
VISUALIZACIÓN ERROR CUADRÁTICO



VISUALIZACIÓN NÚMERO DE TERMINOS



VISUALIZACIÓN DE LA PREDICCIÓN DEL PRECIO DEL DÍA SIGUIENTE



OBJETIVO Y CONCLUSIÓN

Encontrar el mejor conjunto de coeficientes $a_0, a_1, a_2, a_3, a_5, \dots, a_N$ para el grado de polinomio seleccionado (1 a 5) de modo que el modelo pueda predecir el precio de la acción para el día siguiente.

La solución se obtuvo al generar el modelo matemático e intentar de ajustar la curva a los datos obtenidos, no se obtuvo una solución factible porque, aunque la cubica se haya ajustado bien el resultado reflejado fue incorrecto.

Según los resultados obtenidos el polinomio al cubo es el que mejor se ajusta a los datos de la acción de Apple en estos últimos 3 meses. Por lo cual es en el que nos basaríamos para tomar decisiones.

Se observó que no hay una tendencia definida, es una acción muy volátil y el precio depende de la incertidumbre que haya en el momento.

Al elevar los polinomios de la primera a la cuarta potencia no hubo problemas, pero al elevarla a la quinta daba valores muy grandes y era imposible para la computadora localizar mínimos y máximos.

El error cuadrático lo utilizamos para saber que polinomio tiene menor error al ajustarse a los datos y así determinamos que la que esta elevada al cubo es la que mejor se ajusta.

Se utilizo el modelo visto en clase, en este se utilizaron betas.

Al analizar nuestros resultados con lo ocurrido se puede observar, que el modelo predijo que el precio iba a subir cuando en realidad bajo.

Como parámetros elegimos solamente, el día y el precio en el que cerro ese día.

REFERENCIAS

Ingenierías, I. E. I. E. (n.d.). Aplicación de la regresión polinomial para la caracterización de la curva del COVID-19, mediante técnicas de machine learning | Investigación e Innovación en Ingenierías. Investigación E Innovación En Ingenierías - Universidad Simón Bolívar. Retrieved September 21, 2023, from <https://revistas.unisimon.edu.co/index.php/innovacioning/article/view/4103/5218#:~:text=La%20regresi%C3%B3n%20polinomial%20es%20un,grado%2022%2C23%2C24>.

Aparicio, A. (n.d.). Regresión polinómica. <https://www.excelavanzado.com/2018/03/regresion-polinomica.html#:~:text=Podemos%20ajustar%20una%20nube%20de,ax2%2Bbx%2Bc>.

Yahoo is part of the Yahoo family of brands. (n.d.). <https://finance.yahoo.com/quote/AAPL?p=AAPL&.tsrc=fin-srch>

Datos históricos Apple (AAPL) - Investing.com México. (n.d.). Investing.com México. <https://mx.investing.com/equities/apple-computer-inc-historical-data>