

# Tutorial: Predicción de Mercados Bursátiles y Criptomonedas con Python

## 1. Configuración Inicial

### ### Instalación de Bibliotecas

Asegúrate de instalar las siguientes bibliotecas:

```
""  
pip install pandas numpy matplotlib seaborn scikit-learn tensorflow yfinance  
""
```

`yfinance` nos permitirá descargar datos históricos de acciones y criptomonedas directamente desde Yahoo Finance.

### ### Crear el Entorno de Trabajo

Abre Jupyter Lab desde tu terminal:

```
""  
jupyter lab  
""
```

## 2. Descarga de Datos

Usaremos `yfinance` para obtener datos históricos. Como ejemplo, descargaremos datos de Apple (AAPL) y Bitcoin (BTC-USD).

```
""python  
import yfinance as yf  
import pandas as pd  
  
# Descargar datos de Apple y Bitcoin  
apple_data = yf.download('AAPL', start='2015-01-01', end='2023-12-31')  
bitcoin_data = yf.download('BTC-USD', start='2015-01-01', end='2023-12-31')  
  
# Guardar los datos en archivos CSV  
apple_data.to_csv('apple_data.csv')  
bitcoin_data.to_csv('bitcoin_data.csv')  
  
print("Datos descargados y guardados correctamente.")  
""
```

### 3. Exploración y Preparación de Datos

### Análisis Exploratorio

Visualicemos las series de tiempo:

```
python
import matplotlib.pyplot as plt

# Visualizar los precios de cierre
plt.figure(figsize=(14, 7))
plt.plot(apple_data['Close'], label='Apple (AAPL)')
plt.plot(bitcoin_data['Close'], label='Bitcoin (BTC-USD)')
plt.title('Precios Históricos')
plt.xlabel('Fecha')
plt.ylabel('Precio de Cierre')
plt.legend()
plt.show()
```

### 4. Preparación para Predicción

Creamos secuencias de datos para entrenamiento y validación:

```
python
import numpy as np

def create_sequences(data, sequence_length):

    sequences = []

    targets = []

    for i in range(len(data) - sequence_length):

        seq = data[i:i + sequence_length]

        target = data[i + sequence_length]

        sequences.append(seq)

        targets.append(target)

    return np.array(sequences), np.array(targets)

# Longitud de la secuencia
```

```

time_steps = 60

apple_close = apple_data[('Close', 'AAPL')]

btc_close = bitcoin_data[('Close', 'BTC-USD')]

# Crear secuencias para Apple

apple_sequences, apple_targets = create_sequences(apple_close.values, time_steps)


# Crear secuencias para Bitcoin

bitcoin_sequences, bitcoin_targets = create_sequences(btc_close.values, time_steps)

```

## 5. Construcción y Entrenamiento del Modelo

Usaremos una Red Neuronal Recurrente con LSTM:

```

python
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense, Dropout

# Construcción del modelo
model = Sequential([
    LSTM(50, return_sequences=True, input_shape=(time_steps, 1)),
    Dropout(0.2),
    LSTM(50, return_sequences=False),
    Dropout(0.2),
    Dense(1)
])

model.compile(optimizer='adam', loss='mean_squared_error')

# Entrenamiento del modelo para Apple
history = model.fit(
    apple_sequences[:, :, np.newaxis],
    apple_targets,
    epochs=20,
    batch_size=32,
    validation_split=0.2
)

```

## 6. Evaluación y Predicción

Evaluamos el modelo y realizamos predicciones:

```
python
# Predicción
apple_predictions = model.predict(apple_sequences[:, :, np.newaxis])

# Desnormalizar los datos
apple_predictions = scaler.inverse_transform(apple_predictions)

# Visualizar las predicciones
plt.figure(figsize=(14, 7))
plt.plot(apple_data['Close'][time_steps:].values, label='Real')
plt.plot(apple_predictions, label='Predicción')
plt.title('Predicción de Precios - Apple')
plt.legend()
plt.show()
```

## 7. Extensión al Mercado de Criptomonedas

Repita los mismos pasos para criptomonedas, ajustando los datos y el modelo según sea necesario. También podrías explorar arquitecturas avanzadas como Transformers o modelos con series temporales multivariadas.

## 8. Conclusiones

Has aprendido a:

1. Descargar y explorar datos históricos.
2. Preparar datos para modelos predictivos.
3. Entrenar un modelo LSTM para predecir precios.
4. Aplicar la técnica tanto a mercados bursátiles como a criptomonedas.

A continuación expandiremos este análisis incorporando indicadores técnicos, datos macroeconómicos o métricas específicas del mercado de criptomonedas.