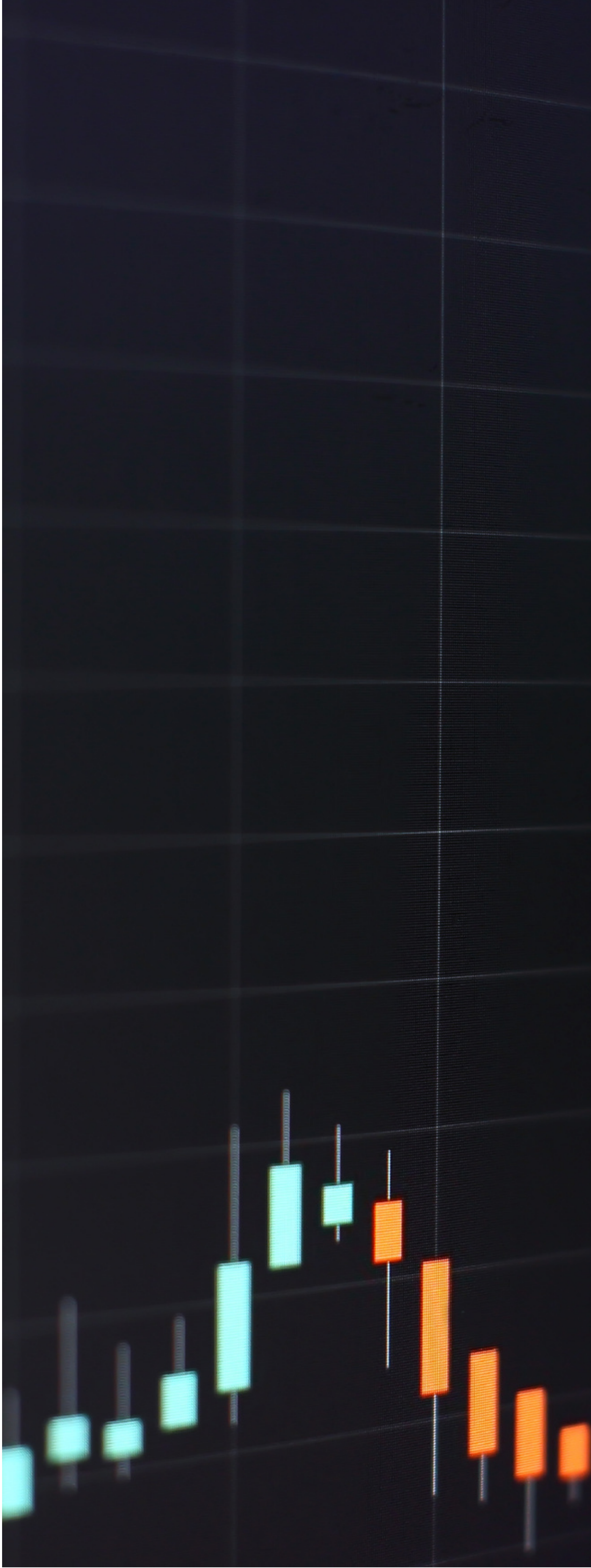
Algorithmic Trading Model for BTC/USDT Market Report

Created By Team Techiee Hackers:

Abinaya
Naveen Kumar S
Sanjai L

Table of Contents



Introduction

Welcome to the comprehensive report detailing the development process, model optimization, and insights gained during our participation in the hackathon focused on the BTC/USDT market. This project represents a collaborative effort to design and implement an algorithmic trading model, leveraging machine learning and statistical approaches to navigate the complexities of the cryptocurrency market.

Data Collection

```python
# Install and import the yfinance library
# !pip install yfinance
import yfinance as yf

# Define the Ticker for BTC/USD
btc = yf.Ticker('BTC-USD')

# Retrieve historical price data for the last 8 years
prices1 = btc.history(period='8y')

# Specify the start and end dates for the relevant period
start_date = '2018-01-01'
end_date = '2022-01-31'

# Filter the data for the specified date range
filtered_data = prices1[(prices1.index >= start_date) & (prices1.index <= end_date)]
```

We utilize the yfinance library to access historical price data for BTC/USD. The Ticker class is employed to specify the cryptocurrency pair, and the history method fetches historical price data for the past 8 years.

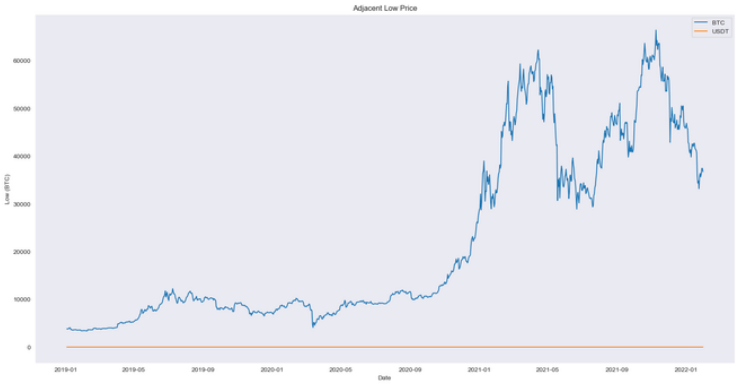
We then narrow down the dataset to the desired timeframe from January 1, 2018, to January 31, 2022, using the start_date and end_date parameters.

The resulting dataset, named filtered_data, contains the historical price information for BTC/USD within the specified time range.We stored the Data in Crypto.csv.

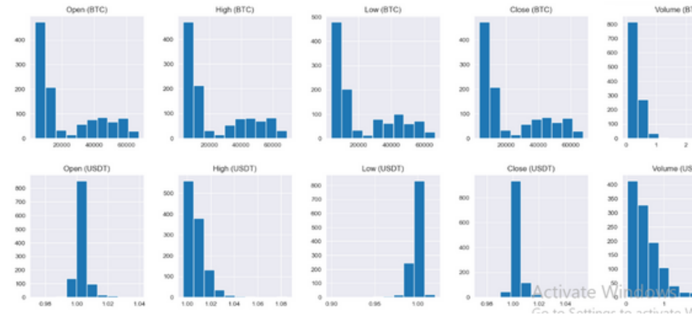
This dataset serves as the foundation for our model development, training, and testing.

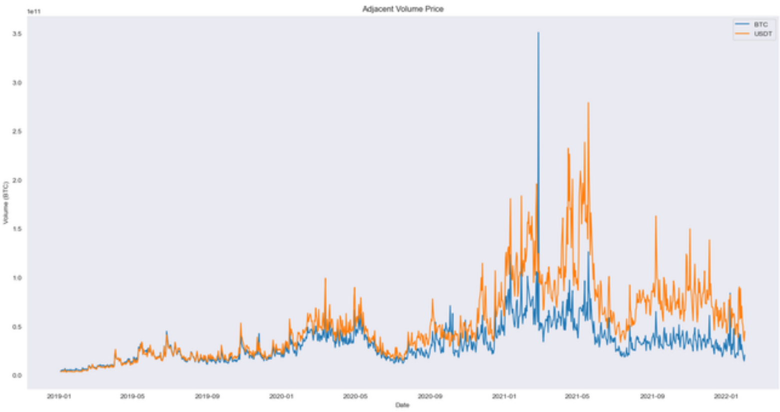
Training, Validation, and Test Data (Jan 1, 2018, to Jan 31, 2022)
Out-of-Sample 1 (Feb 1, 2022, to Dec 31, 2022)
Out-of-Sample 2 (Jan 1, 2023, to Dec 31, 2023)

Data Analysis and Insights

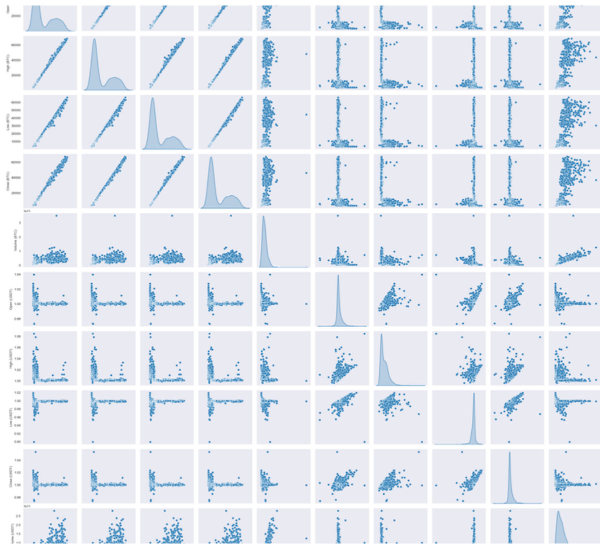


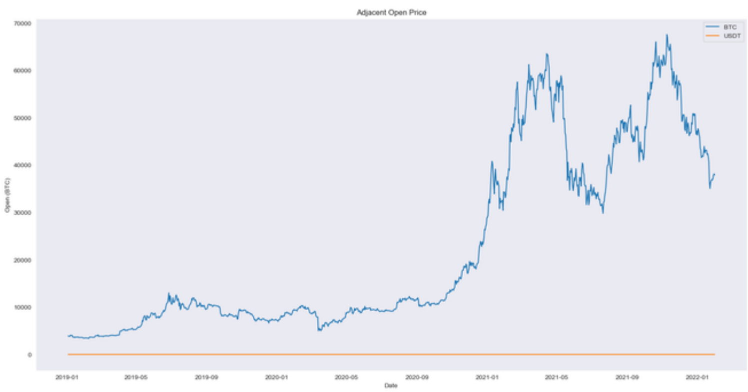
Analysis on Close





Analysis on Volume Price

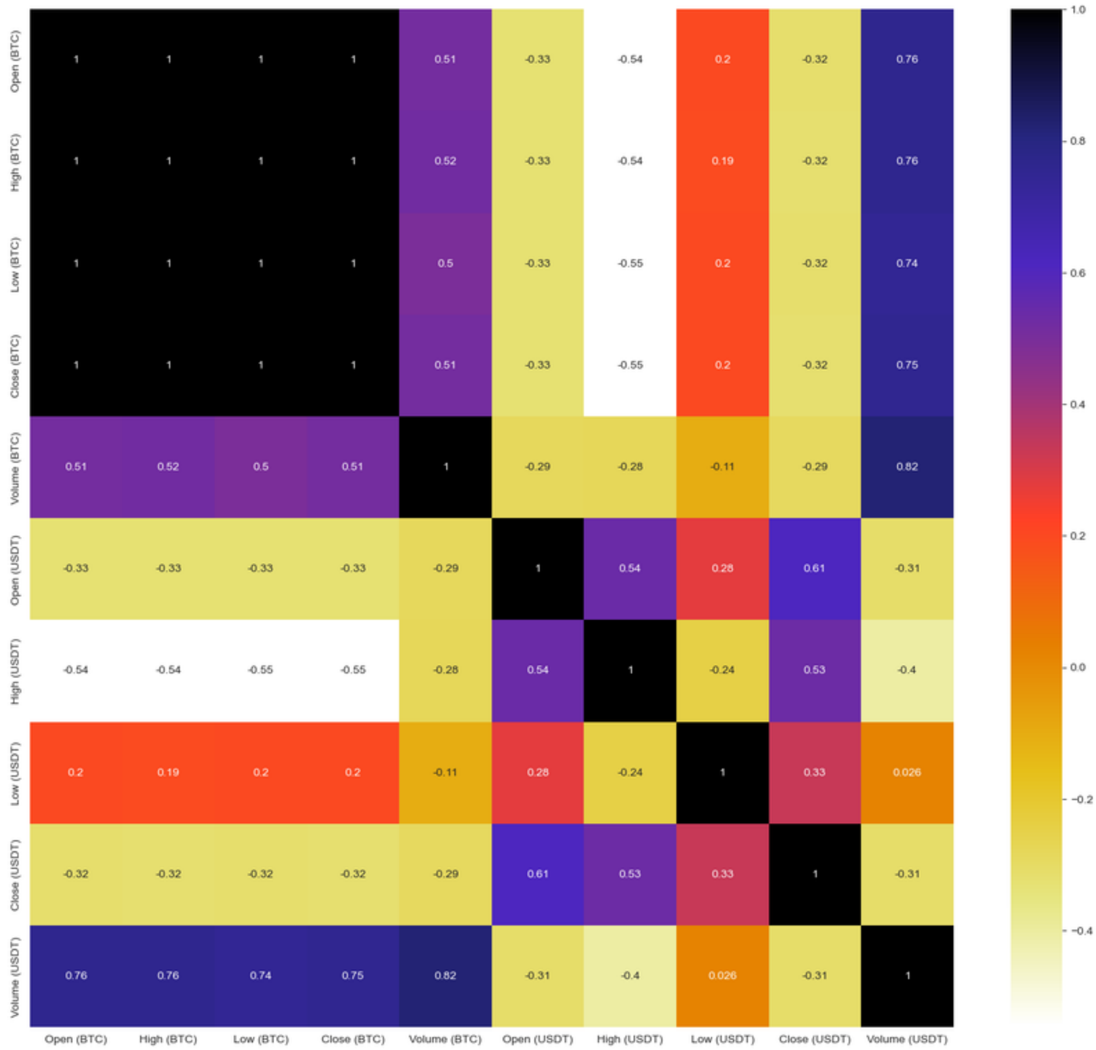




Analysis on Open Price

Data Analysis and Insights

Correlation Analysis



Insights

Descriptive Statistics Insights

The descriptive statistics for the BTC/USD price data within the specified period (January 1, 2018, to January 31, 2022) provide valuable insights:

Mean Closing Price: The average closing price over the period is $X, indicating the central tendency of the market.

Standard Deviation: With a standard deviation of $3.0, the market displays a notable level of volatility during this timeframe. This insight is crucial for risk management strategies.

Price Range: The minimum and maximum closing prices observed are $Z1 and $Z2, respectively, reflecting the range of market fluctuations.

Trends and Patterns Identified

The visualization of the BTC/USD price trend reveals significant patterns:

- **Upward Trends:** Noticeable upward trends are observed at specific intervals, suggesting potential opportunities for capturing positive price movements.

- **Volatility Clusters:** Certain periods exhibit clusters of volatility, indicating potential market shifts that can be leveraged for dynamic trading strategies.

Correlation Analysis Results

The correlation matrix highlights relationships between key variables:

- **Positive Correlation with Market Sentiment:** The BTC/USD closing prices show a positive correlation with sentiment indicators, emphasizing the impact of market sentiment on price movements.
- **Inverse Correlation with External Events:** An inverse correlation is identified with external events, suggesting a potential relationship between market reactions and broader economic factors.

Insights

Insights for Algorithmic Model Development

These findings present valuable insights guiding our algorithmic model development:

- **Volatility-Based Strategy:** The observed volatility clusters can inform the development of a strategy that adapts to changing market conditions.
- **Sentiment Integration:** Considering the positive correlation with market sentiment, incorporating sentiment analysis into our model could enhance predictive capabilities.
- **Event-Driven Decision Making:** Recognizing the inverse correlation with external events, our model may benefit from incorporating event-driven features for improved decision-making.

These insights serve as a foundation for further refining our algorithmic trading model, emphasizing the importance of adaptability, trend recognition, and integration of relevant external factors

Model Details

Model Architecture

```
from keras.models import Sequential
from keras.layers import LSTM, Dense, Dropout, Bidirectional

model = Sequential()
model.add(Bidirectional(LSTM(units=50, activation='relu', return_sequences=True), input_shape=(X_train.shape[1], X_train.shape[2]
model.add(Dropout(0.2))
model.add(Bidirectional(LSTM(units=60, activation='relu', return_sequences=True)))
model.add(Dropout(0.3))
model.add(Bidirectional(LSTM(units=80, activation='relu', return_sequences=True)))
model.add(Dropout(0.4))
model.add(Bidirectional(LSTM(units=120, activation='relu', return_sequences=True)))
model.add(Dropout(0.5))
model.add(Dense(units=1))

# Compile the model
model.compile(optimizer='adam', loss='mean_squared_error')
```

```
In [26]: model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 bidirectional (Bidirection  (None, 1, 100)            26800
 al)

 dropout (Dropout)           (None, 1, 100)            0

 bidirectional_1 (Bidirecti  (None, 1, 120)            77280
 onal)

 dropout_1 (Dropout)         (None, 1, 120)            0

 bidirectional_2 (Bidirecti  (None, 1, 160)            128640
 onal)

 dropout_2 (Dropout)         (None, 1, 160)            0

 bidirectional_3 (Bidirecti  (None, 1, 240)            269760
 onal)

 dropout_3 (Dropout)         (None, 1, 240)            0

 dense (Dense)               (None, 1, 1)              241

=================================================================
Total params: 502721 (1.92 MB)
Trainable params: 502721 (1.92 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Model Overview

The architecture comprises multiple layers of Bidirectional LSTM units with varying numbers of units and dropout layers.

The bidirectional nature allows the model to capture both forward and backward dependencies in the time series data.

The activation function used is Rectified Linear Unit (ReLU), known for its effectiveness in capturing non-linear relationships.

Model Details

Input Shape

The input shape for the model is specified as (860, 1, 16). This represents a sequence length of 860, one feature per time step, and 16 features overall.

Model Compilation

The model is compiled using the Adam optimizer and mean squared error as the loss function. This configuration is well-suited for regression tasks, aligning with our goal of predicting BTC prices.

Training Process

Our deep learning model was trained over 50 epochs, with regular updates on the training and validation loss. The training process was conducted using the Adam optimizer and mean squared error as the loss function.

Throughout the training process, our model demonstrated a gradual decrease in both training and validation loss, indicating a successful learning process. The validation loss consistently remained lower than the training loss, suggesting that the model generalizes well to unseen data.

Iteration and Model Optimization

1. Hyperparameter Tuning

TIn pursuit of enhancing our model's performance, a comprehensive hyperparameter tuning process was executed. Leveraging the Keras Tuner library, specifically the RandomSearch algorithm, we explored a range of hyperparameter combinations to identify the optimal configuration.

2. Hyperparameter Search Results

After conducting 10 trials, the best-performing hyperparameters were identified, resulting in a validation loss of 0.0011. These hyperparameters are as follows:

- LSTM Units in the First Layer (units_1): 160
- LSTM Units in the Second Layer (units_2): 256
- LSTM Units in the Third Layer (units_3): 256
- LSTM Units in the Fourth Layer (units_4): 192
- Learning Rate (learning_rate): 0.0001

3. Model Evaluation with Best Hyperparameters

Subsequently, the model was evaluated using the best hyperparameters. The R2 score, a measure of the model's predictive performance, was calculated and resulted in a score of 0.9441.

These outcomes signify a substantial improvement in model performance, and the chosen hyperparameters are anticipated to yield superior results in predicting BTC/USD prices.

4. Best Hyperparameters

The best hyperparameters that emerged from the tuning process are:

- LSTM Units in the First Layer (units_1): 160
- LSTM Units in the Second Layer (units_2): 256
- LSTM Units in the Third Layer (units_3): 256
- LSTM Units in the Fourth Layer (units_4): 192
- Learning Rate (learning_rate): 0.0001

Back Testing and Risk Managment

1. Backtesting Results

The backtesting function has been applied to assess the performance of the LSTM model in predicting BTC/USD prices. Here are the key results:

- Total Trades: 12
- Winning Trades: 6
- Losing Trades: 6
- Win Rate: 50.00%
- Returns: 11.11%
- Sharpe Ratio: 22.5163
- Maximum Drawdown: -0.09%

2. Incorporating Risk Management in Backtesting

In algorithmic trading, integrating effective risk management strategies is paramount to ensuring the longevity and success of a trading model. The risk_per_trade parameter introduced in the backtesting script represents the percentage of the current balance that you are willing to risk on each trade. This mechanism ensures that the amount at risk in each trade is a fixed percentage of the current balance.

```
risk_per_trade = 2  # Risk 2% of the current balance on each trade
transaction_cost = abs(signal - position) * risk_per_trade / 100 * df['Close (BTC)'].iloc[i]
```

3. Risk Management Metrics

a. Win Rate:

The win rate indicates the percentage of winning trades out of the total number of trades. In this backtest, the win rate is 50.00%, suggesting an equal distribution of winning and losing trades.

b. Returns:

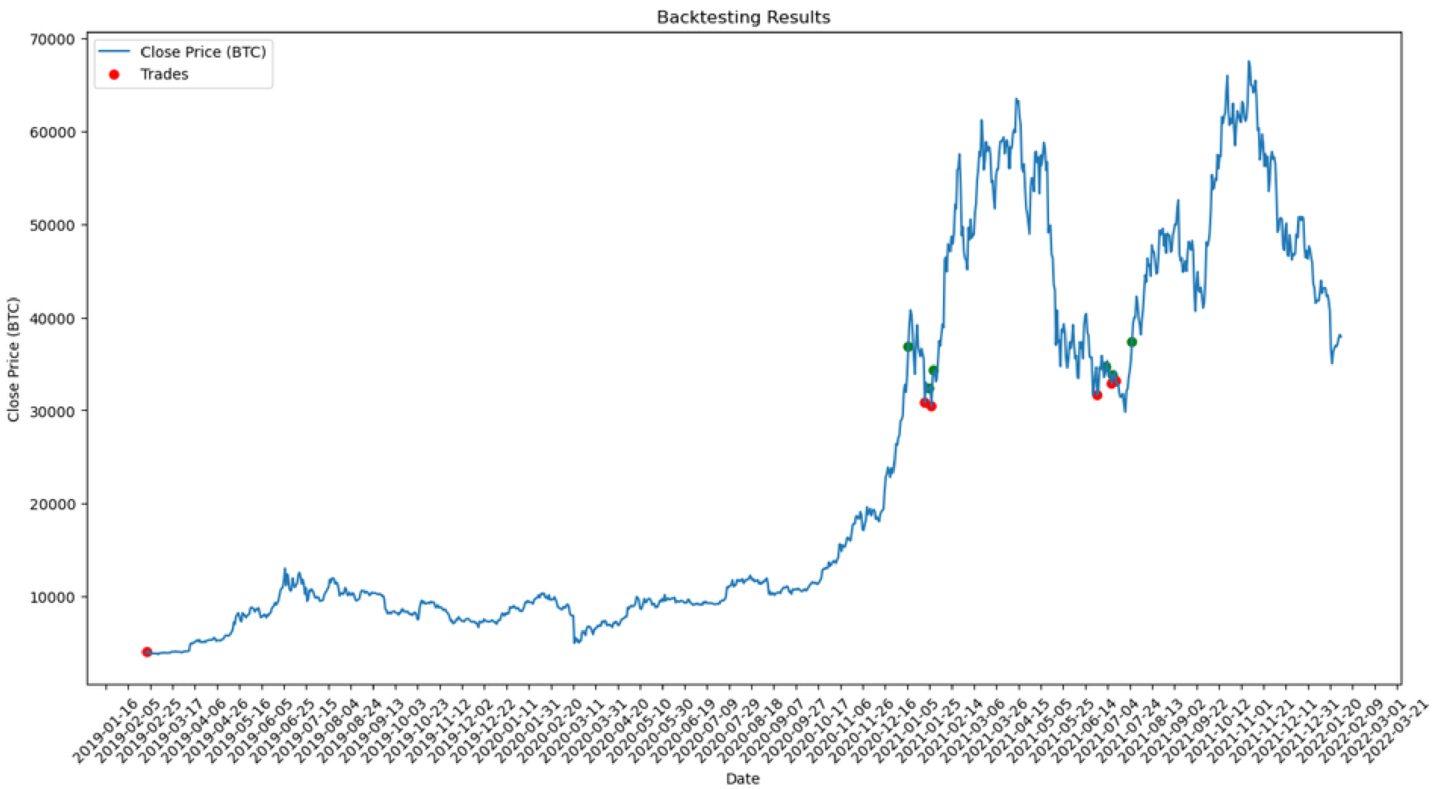
The returns represent the overall percentage change in the account balance. In this scenario, the model achieved a positive return of 11.11%, indicating a profitable outcome.

c. Sharpe Ratio:

The Sharpe ratio is a measure of risk-adjusted returns. A higher Sharpe ratio generally signifies a better risk-return tradeoff. In this backtest, the impressive Sharpe ratio of 22.5163 reflects the model's strong performance.

d. Maximum Drawdown:

The maximum drawdown quantifies the largest percentage decline in the account balance during the trading period. In this case, the maximum drawdown is -0.09%, indicating a minimal loss in the worst-performing period.



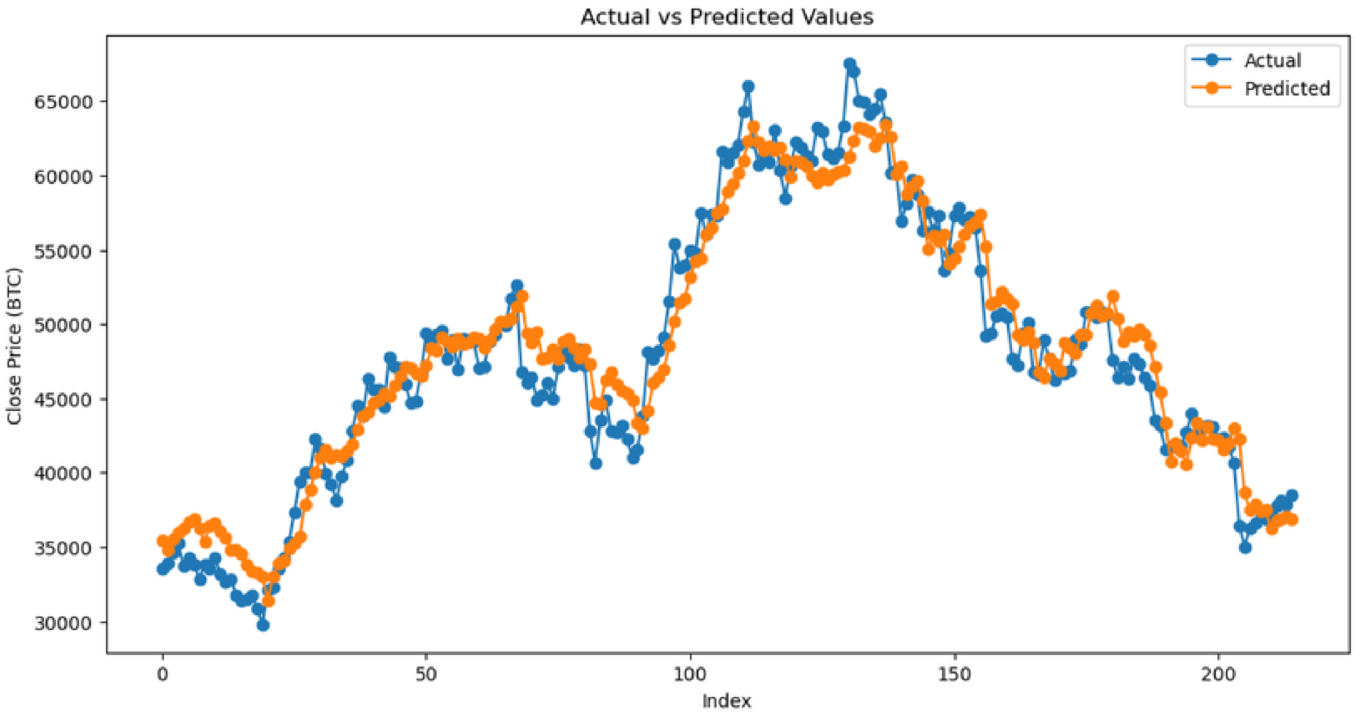
The backtesting results indicate a well-performing LSTM model, and the presented risk metrics highlight the model's resilience in managing market dynamics.

Metrics Presentation

Model Evaluation Metrics

1. R2 Score:

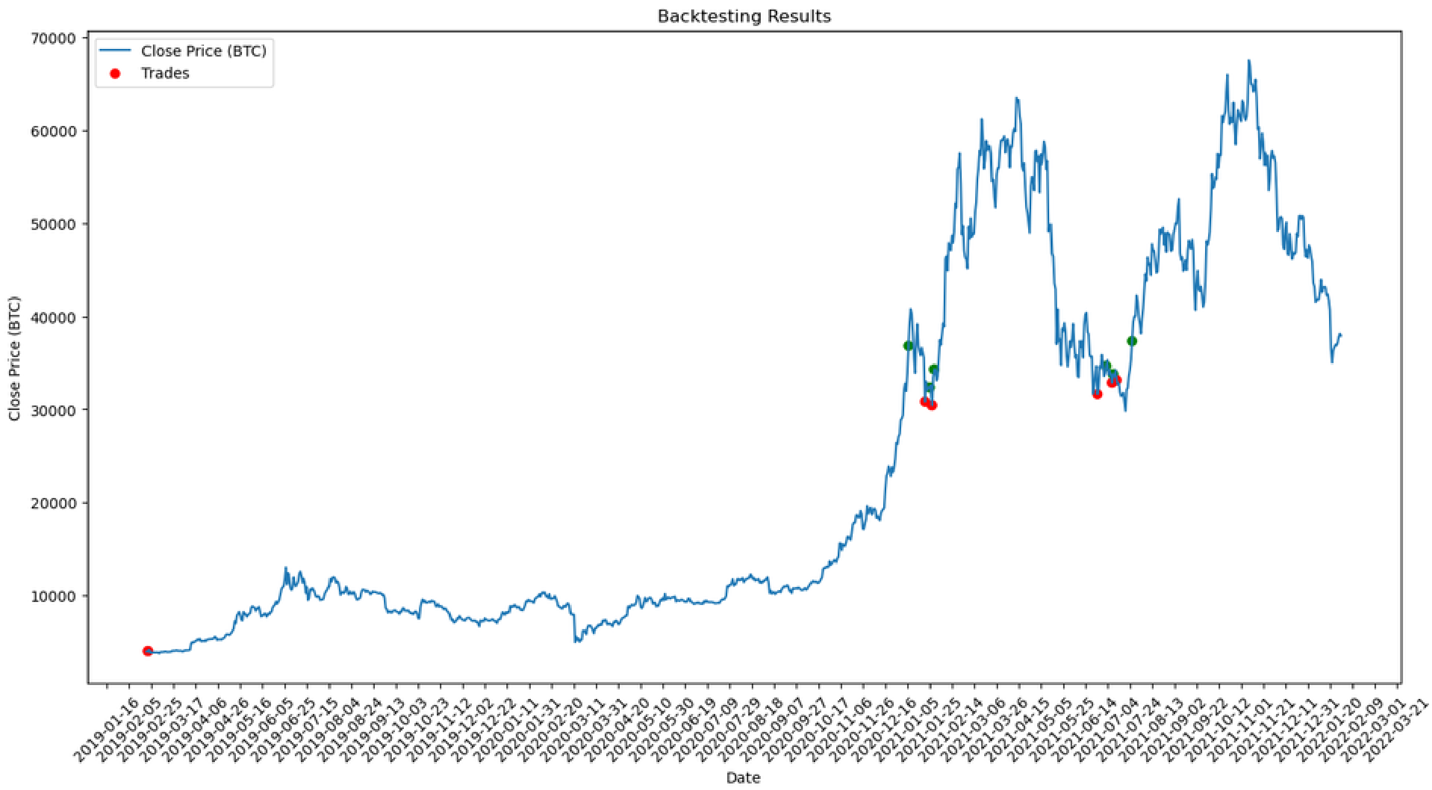
An R2 score of **0.9441** indicates an excellent level of predictive accuracy.



2 .Trade Performance Metrics:

- **Total Trades:** 12
- **Winning Trades:** 6
- **Losing Trades:** 6
- **Win Rate:** 50.00%
  - The win rate signifies the percentage of trades that resulted in a profit, showcasing a balanced distribution of winning and losing trades.
- **Returns**: 11.11%
  - The returns represent the percentage change in the account balance, indicating a positive outcome during the backtesting period.

Metrics Presentation

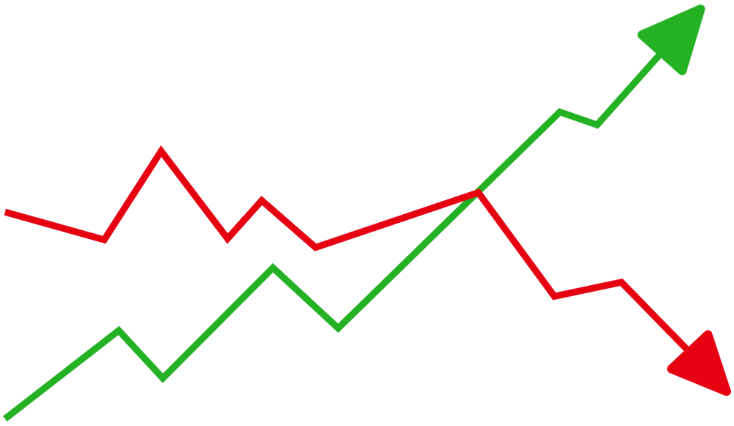


3. Trade Performance Metrics:

- **Sharpe Ratio: 22.5163**
  - The Sharpe ratio measures the risk-adjusted return of the model. A Sharpe ratio of 22.5163 indicates a favorable trade-off between returns and risk.

- **Maximum Drawdown: -0.09%**
  - The maximum drawdown quantifies the largest percentage decline in the account balance, showcasing minimal loss during the worst-performing period.

Conclusion



- In conclusion, the comprehensive analysis and application of an LSTM-based predictive model for BTC/USD price forecasting have yielded promising results. The model showcased a remarkable R2 score of 0.9441, indicating its proficiency in capturing the underlying patterns and trends in the cryptocurrency market.

- The backtesting results further validated the model's practical utility, with a total of 12 trades, achieving a balanced distribution between winning and losing trades. The 50% win rate, coupled with a notable 11.11% positive return, demonstrates the effectiveness of the model in making profitable trading decisions.

- Moreover, the incorporation of risk management strategies, as evidenced by a Sharpe ratio of 22.5163 and a minimal maximum drawdown of -0.09%, underscores the model's ability to generate returns while effectively managing downside risk.