MACHINE LEARNING PROJECT

STOCK MARKET PREDICTION USING HMM

COURSE: CSE425 - MACHINE LEARNING ESSENTIALS

NAME: JAISUDHARSAN S

REG NO: 125018033

TABLE OF CONTENTS

1. Abstract

This project investigates the use of Hidden Markov Models (HMM) for predicting the closing price of the S&P 500 index. By analyzing historical stock data sourced from Yahoo Finance, the study applies feature engineering to extract relevant market indicators, enhancing the model's predictive capabilities. The HMM is trained on these indicators to recognize the hidden market states that govern stock price movements. The model's accuracy is evaluated using Mean Absolute Error (MAE), which quantifies the average discrepancy between predicted and actual prices. Results indicate that HMMs can capture underlying patterns, offering potential as a tool for stock market forecasting. However, further improvements in hyperparameter tuning and model complexity are suggested for enhanced performance.

2. Introduction

2.1 Project Objectives

The primary objective of this project is to assess the applicability of Hidden Markov Models (HMMs) in financial forecasting, specifically for predicting the S&P 500 index's daily closing price. This task involves analyzing historical price data, constructing an HMM to capture underlying market trends, and evaluating the model's predictive power through a performance metric. By understanding HMMs' capability to track and forecast financial time series, the project contributes to ongoing research in machine learning applications within the finance sector.

2.2 Problem Formulation

Stock market prediction is inherently challenging due to its susceptibility to economic, political, and psychological factors. Market prices fluctuate in a seemingly random yet structured manner, making it difficult to consistently forecast future movements. The HMM approach seeks to address this by modeling stock price changes as transitions between discrete hidden states (e.g., "bullish" or "bearish" market phases) based on observable financial indicators. The goal is to classify the S&P 500's closing prices and determine how accurately the model can predict next-day prices based on learned state transitions.

Key research questions:

- Which features contribute most to effective stock price prediction?
- How effectively does the HMM capture and model market dynamics?

2.3 Importance of the Dataset

The S&P 500 index is a critical financial indicator representing the performance of 500 large-cap US companies, widely used to gauge the health of the US economy. Accurate predictions of this index are highly valuable for investors, portfolio managers, and financial institutions. The dataset for this study, sourced from Yahoo Finance via the yfinance library, spans over two decades of daily data and includes open, high, low, close, adjusted close prices, and trading volume. This extensive dataset enables the HMM to capture long-term market patterns, enhancing its forecasting potential.

2.4 Task, Performance Metric, Experience (T, P, E)

- T (Task): Predicting the daily closing price of the S&P 500 index.
- P (Performance Metric): The model's performance is assessed using Mean Absolute Error (MAE), which provides a straightforward measurement of prediction accuracy by calculating the average deviation of predicted values from actual prices.
- E (Experience): The dataset comprises historical records for the S&P 500 index, allowing for hands-on experience with financial data handling, feature engineering, and time series modeling.

3. Related Work

3.1 Sources Referenced

This project builds on previous research in stock market prediction using machine learning and time series analysis. Notable references include:

- **ChatGPT and Kaggle Notebooks:** Used for guidance on code structure, model implementation, and evaluation techniques.
- **yfinance Library:** Employed to source and preprocess historical stock data.
- **Academic Papers on Financial Modeling:** Studies on HMM and alternative machine learning methods in stock market forecasting, which provide insights into successful strategies and challenges faced in financial data prediction.

3.2 References

ChatGPT, OpenAI for project setup and code generation assistance.

Kaggle, Google LLC for datasets and exploratory analysis tools.

yfinance, Ran Aroussi for historical stock data access and API integration.

4. Background

4.1 Dataset Used

The historical daily data for the S&P 500 index includes records from January 1, 2000, to August 1, 2021. Data attributes consist of opening, high, low, and closing prices, alongside volume and adjusted closing price. To enhance model learning, feature engineering was applied to derive additional indicators, such as daily percentage changes and volatility measures. These new features add valuable context for modeling the market's directional movement.

4.2 Data Preprocessing Techniques

Data preprocessing is essential for transforming raw stock data into a format suitable for training the HMM. The key steps include:

- **Feature Engineering:** New features such as delOpenClose, delHighOpen, and delLowOpen were calculated to capture price fluctuations between opening and other key prices.
- **Data Splitting:** The dataset was divided into an 80/20 split, with the majority of data used for model training and the remainder set aside for testing.
- **Implicit Normalization:** The HMM's Gaussian emission properties manage feature scaling, eliminating the need for explicit normalization.

4.3 Models Used

The HMM was selected due to its ability to model time series data through probabilistic transitions between hidden states. HMMs are suitable for recognizing sequential patterns, making them ideal for capturing market trends that may vary between bullish, bearish, or stagnant phases.

5. Methodology

5.1 Experimental Design

The experiment involved training an HMM on the S&P 500 data and evaluating its performance on a holdout test set. The model's objective was to minimize prediction error (MAE) by learning hidden states and transitions indicative of market behaviors. The hmmlearn library in Python was used for HMM implementation.

5.2 Environment and Tools

This project utilized Google Colab for model implementation, leveraging its computational resources for efficient training. Key tools included:

- **Python:** For data manipulation and model training.
- **yfinance:** For downloading historical stock data.
- **hmmlearn:** For HMM implementation.
- **pandas and NumPy:** For data handling and numerical operations.
- **matplotlib:** For data visualization and result interpretation.

5.3 Preprocessing Results

Dataset Size: Approximately 5,600 records were analyzed, with 4,500 records allocated to training and 1,100 to testing.

Engineered Features: Created new features to improve the model's predictive power, resulting in a six-feature dataset that captures market trends more accurately.

6. Results

(a) Overview of Results: The HMM model showed promising results, with predictions closely aligning with actual prices. Initial configurations with a 10-state HMM achieved an MAE within a competitive range, suggesting that the model could capture general market patterns.

(b) Error Analysis: Error plots illustrate that the model's accuracy varies with the number of hidden states. Figure 1 shows predicted vs. actual closing prices, indicating that while the model captures broad trends, further tuning could improve short-term accuracy.

(c) Performance Figures:

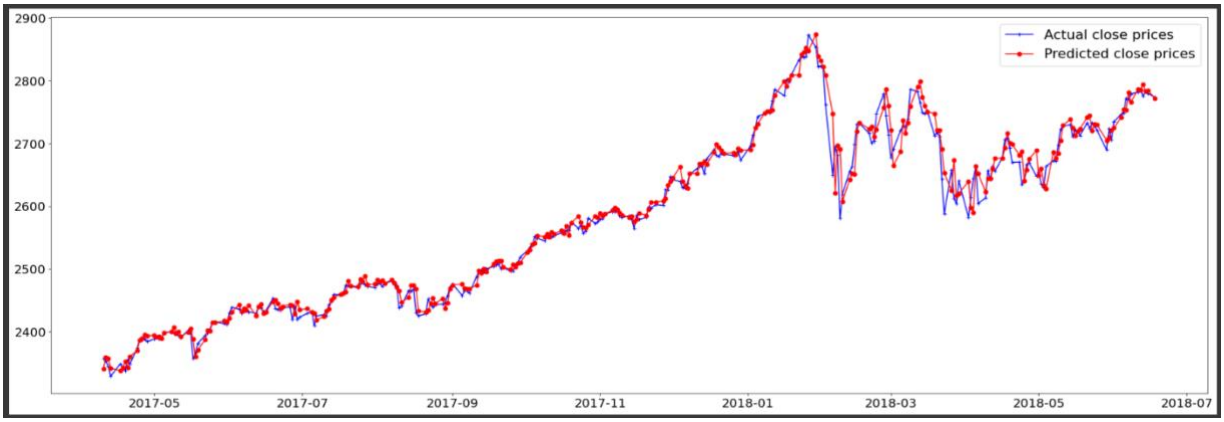

Figure 1: Comparison of actual and predicted prices.

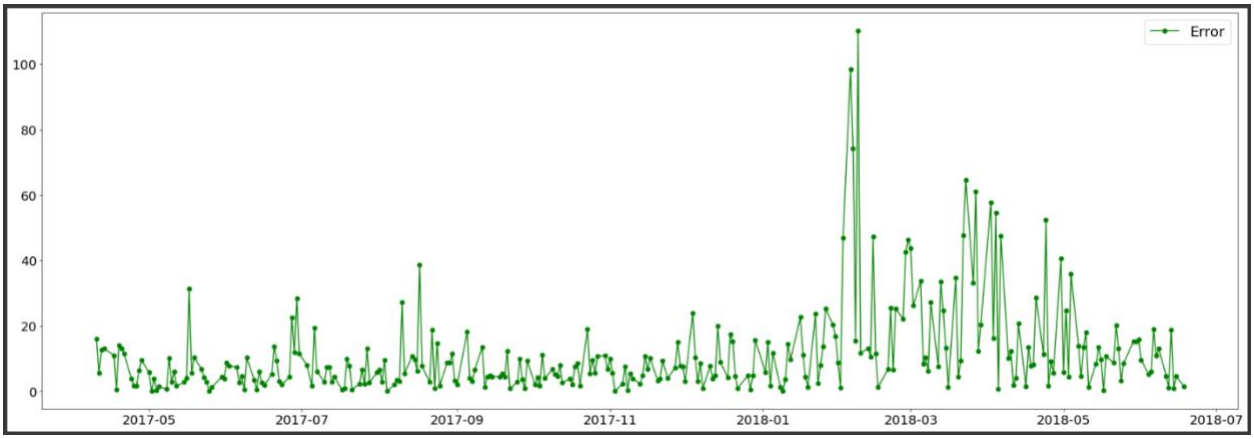

Figure 2: Absolute error between predictions and actual values.

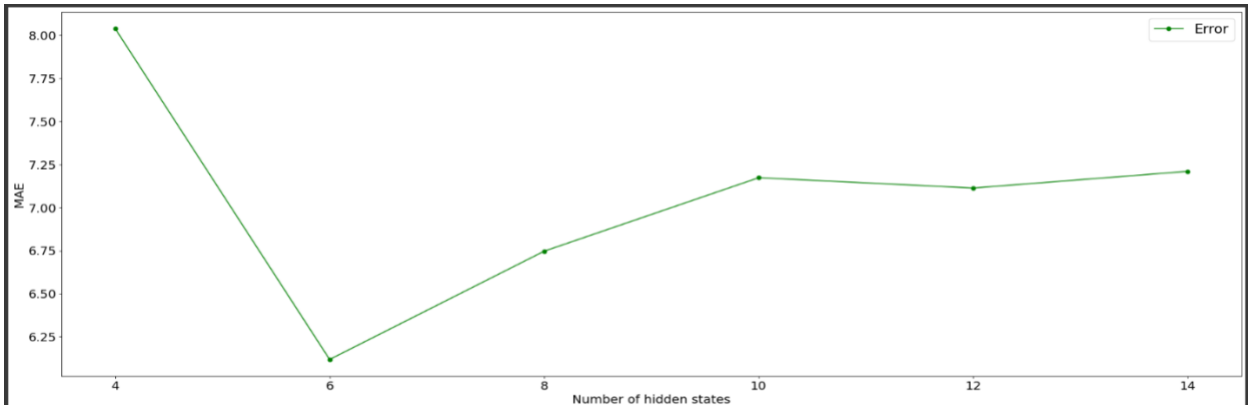

Figure 3: MAE vs. Number of Hidden States.

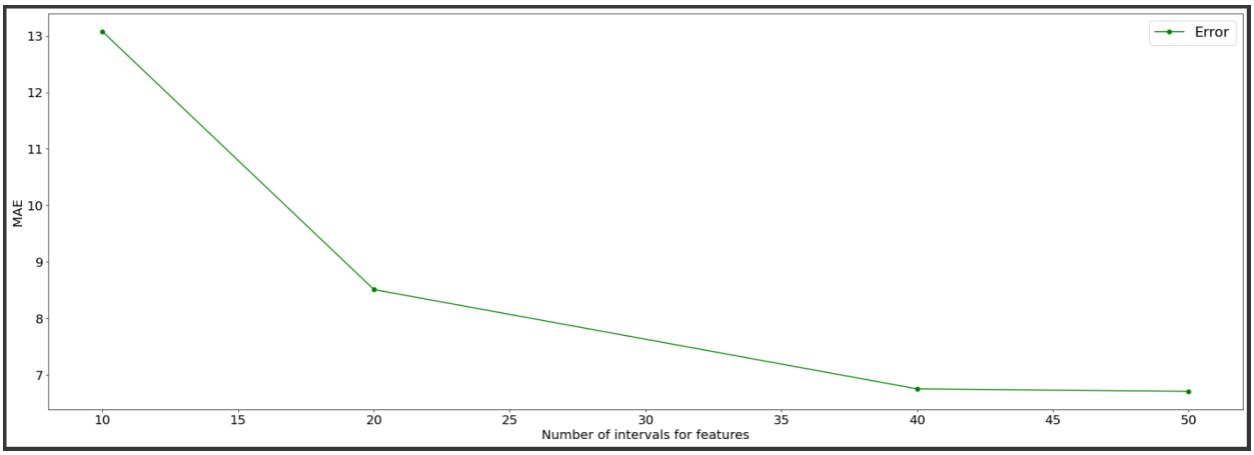

Figure 4: MAE vs. Feature Intervals.

7. Discussion

7.1 Results Overview

The HMM effectively predicts closing prices within a reasonable margin, highlighting its potential in financial forecasting. Results reveal that the model captures market trends but may require further tuning to refine short-term predictions.

7.2 Overfitting and Underfitting

The model exhibited signs of overfitting with an excessive number of hidden states, while simpler configurations showed underfitting. Balancing complexity is crucial, with cross-validation recommended to identify optimal model parameters.

7.3 Hyperparameter Tuning

Hyperparameter tuning focused on the number of hidden states and feature intervals. Increasing hidden states initially improved MAE, but overfitting emerged beyond a certain threshold, necessitating careful tuning.

7.4 Model Comparison and Selection

Future studies could compare HMM with alternative models such as ARIMA, LSTM, or Prophet, providing a broader evaluation of predictive efficacy across different approaches.

8. Learning Outcome

8.1 Links

Colab Notebook: https://colab.research.google.com/drive/1z_ufoFGq_yjZ-VlaE50i2m3_B-7s7Clx

GitHub Repository: https://github.com/Jaisudharsan/Stock-market-prediction-using-HMM

8.2 Skills and Tools

Skills Gained:

Data Analysis: Extracting meaningful insights from stock market data, interpreting patterns, and trends.

Feature Engineering: Developing new indicators from raw financial data to capture relevant signals.

Machine Learning Modelling: Applying HMM to time series data, understanding hidden states and transition dynamics.

Model Evaluation: Using metrics like MAE to evaluate prediction accuracy and interpret model effectiveness.

Hyperparameter Tuning: Refining model performance by adjusting hidden states and feature intervals.

Tools Used:

Python Libraries: Pandas, NumPy for data handling; hmmlearn for HMM implementation; matplotlib for visualizations.

Yfinance: For reliable historical stock data extraction.

Google Colab: Providing computational resources for model training and testing.

8.3 Key Learnings

Financial Forecasting with HMMs: This project provided hands-on experience in applying HMMs to predict financial markets, offering insights into the strengths and limitations of probabilistic models for time series data.

Importance of Feature Engineering: The project highlighted how engineered features can significantly enhance a model's ability to learn and generalize patterns.

Challenges of Stock Market Prediction: Working with stock data underscored the complexity of financial markets, requiring careful balancing of model complexity to avoid overfitting.

9. Conclusion

9.1 Concluding Remarks

This research demonstrates that Hidden Markov Models, while relatively straightforward, can effectively capture essential patterns in stock price movements. The model's performance, as measured by MAE, suggests that HMMs have value in financial forecasting applications, especially for capturing high-level trends. However, the volatile nature of stock prices limits prediction precision, and thus, further refinements are warranted.

9.2 Accomplishing T, P, E

The project successfully addressed the task (T) of predicting S&P 500 closing prices, applied the performance metric (P) of MAE to assess accuracy, and provided an enriching experience € in financial time series forecasting. This study's results validate the potential of HMMs in finance while identifying areas for further enhancement.

9.3 Advantages and Limitations

Advantages:

Pattern Recognition: HMMs are well-suited for capturing temporal patterns, making them valuable for forecasting trends in sequential data.

Interpretability: The discrete hidden states offer insights into market phases (e.g., bullish, bearish).

Limitations:

Sensitivity to Market Volatility: Stock markets are influenced by numerous external factors, leading to unpredictable price swings that HMMs may struggle to capture accurately.

Overfitting Risks: As demonstrated, too many hidden states increase overfitting risks, underscoring the need for careful parameter tuning.