Uncovering Hidden Market Regimes in S&P 500 Returns Using Gaussian Hidden Markov Models

Soham Naukudkar, Siddhikesh Gavit, Shreyash Borkar
Department of Computer Science and Engineering
Indian Institute of Information Technology, Vadodara
Emails: {202351135, 202351040, 202351132}@iiitvadodara.ac.in

***Abstract*—Financial time series, such as stock returns, often exhibit regime-switching behaviors characterized by periods of stability and turbulence. This tutorial provides a comprehensive guide to modeling these hidden regimes using Gaussian Hidden Markov Models (HMMs). We apply a 2-state Gaussian HMM to daily log returns of the S&P 500 index from January 1, 2015, to November 22, 2025, identifying low- and high-volatility states. The model reveals persistent regimes with low transition probabilities, offering insights into market dynamics. We detail data preprocessing, model fitting via the Expectation-Maximization algorithm, state decoding with Viterbi, and evaluation through log-likelihood and visualizations. An extension to 3 states captures finer granularity, while a bonus comparison with Apple (AAPL) stock highlights asset-specific volatility. All steps are implemented in Python with `hmmlearn` and `yfinance`, ensuring reproducibility. This approach aids in risk assessment and regime forecasting for financial decision-making.**

***Index Terms*—Gaussian Hidden Markov Model, Financial Time Series, Market Regimes, Volatility Clustering, Regime Switching**

I. Introduction

Financial markets are inherently non-stationary, transitioning between unobservable "regimes" such as bull markets (low volatility, positive drifts) and bear markets (high volatility, negative drifts). Traditional models like ARIMA assume constant parameters, failing to capture these shifts. Gaussian Hidden Markov Models (HMMs) address this by positing latent states that govern observable returns, modeled as Gaussian emissions.

This tutorial walks through the application of Gaussian HMMs to S&P 500 (GSPC) returns, covering data acquisition, preprocessing, fitting, inference, evaluation, and interpretation. We use a 10+ year dataset (2015–2025) to encompass events like the 2020 COVID-19 crash. The 2-state model identifies core regimes, while a 3-state extension and AAPL comparison (bonus) add depth.

Key HMM components:

- **Hidden States**: $Z_t \in \{1, \ldots, N\}$, evolving via Markov chain with transition matrix $\mathbf{A} = [a_{ij}]$, where $a_{ij} = P(Z_{t+1} = j \mid Z_t = i)$.
- **Emissions**: Observations $r_t \mid Z_t = i \sim \mathcal{N}(\mu_i, \sigma_i^2)$.
- **Initial Probabilities**: $\boldsymbol{\pi}$.

Parameters are estimated via Expectation-Maximization (EM), and states decoded using Viterbi algorithm.

This report is submitted for the Laboratory Problems course. Code repository: https://github.com/Siddhu-04/Lab5AI.

TABLE I: S&P 500 Log Returns Statistics (2015–2025)

Statistic	Value
Count	2739
Mean	0.000426
Std. Dev.	0.011349
Min	-0.127652
25%	-0.003799
Median	0.000687
75%	0.005739
Max	0.090895

Python code snippets are provided for each step; full implementation at [https://github.com/Siddhu-04/CS307$_LabCodes$].

II. Data Collection and Preprocessing

A. Acquisition

We fetch historical data using the Yahoo Finance API via `yfinance`:

```
import yfinance as yf
ticker = '^GSPC'
start_date = '2015-01-01'
end_date = '2025-11-22'
data = yf.download(ticker, start=start_date, end=e
```

This yields 2740 daily observations, including adjusted closing prices (noting `yfinance`'s default auto-adjustment, using the 'Close' column).

B. Preprocessing

Extract closing prices P_t and compute log returns for stationarity:

$$r_t = \log\left(\frac{P_t}{P_{t-1}}\right), \quad t = 2, \ldots, T. \tag{1}$$

Drop NaNs, resulting in 2739 returns. Summary statistics:

The positive mean indicates long-term growth, while the std. dev. reflects volatility clustering (e.g., 2020 spikes).

III. Gaussian HMM Fitting and Parameter Analysis

A. Fitting the Model

Reshape returns as a 2D array (for `hmmlearn`) and fit a 2-state Gaussian HMM:

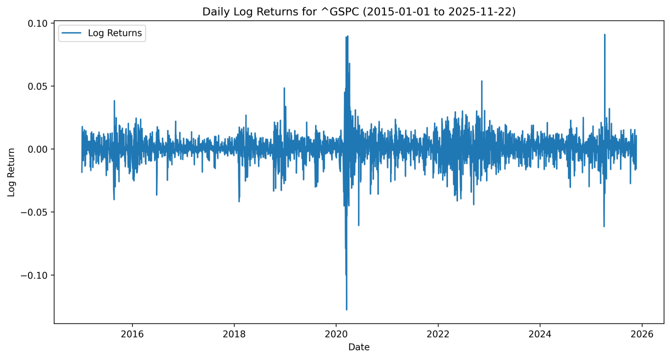

Fig. 1: Daily log returns of S&P 500 (2015–2025), showing volatility clusters.

TABLE II: 2-State Gaussian HMM Parameters

State	Label	Mean μ_i	Variance σ_i^2
0	Low Volatility	0.001003	0.000051
1	High Volatility	-0.001603	0.000430

TABLE III: 2-State Transition Matrix

From \To	Low Vol.	High Vol.
Low Vol.	0.986	0.014
High Vol.	0.052	0.948

```
from hmmlearn import hmm
X = log_returns.values.reshape(-1, 1)
n_states = 2
model_2 = hmm.GaussianHMM(n_components=n_sta
                          covariance_type="d
                          n_iter=100, random
model_2.fit(X)
hidden_states_2 = model_2.predict(X)
```

The EM algorithm iterates to maximize log-likelihood until convergence.

Fitted parameters:

State 0 (low volatility) has positive mean and low variance, aligning with growth phases. State 1 (high volatility) shows negative mean and high variance, indicative of downturns.

B. State Decoding

Viterbi path: $\hat{Z} = \arg\max_{\mathbf{z}} P(\mathbf{r}, \mathbf{z} \mid \boldsymbol{\theta})$.

IV. Interpretation and Inference

A. Inferred States and Transitions

States are overlaid on returns for visualization (Fig. 2). Low-volatility periods dominate pre-2020 and post-2022 recovery, while high-volatility clusters during 2020.

Transition matrix $\mathbf{A}$:

High diagonals (> 0.94) indicate regime persistence; switches are rare, modeling "sticky" markets.

Sample regime durations (days):

Longer low-vol runs (e.g., 154 days) reflect extended bulls.

TABLE IV: Sample State Durations

Run	Low Vol. (State 0)	High Vol. (State 1)
1	0	4
2	154	0
3	0	14
4	58	0
5	0	52

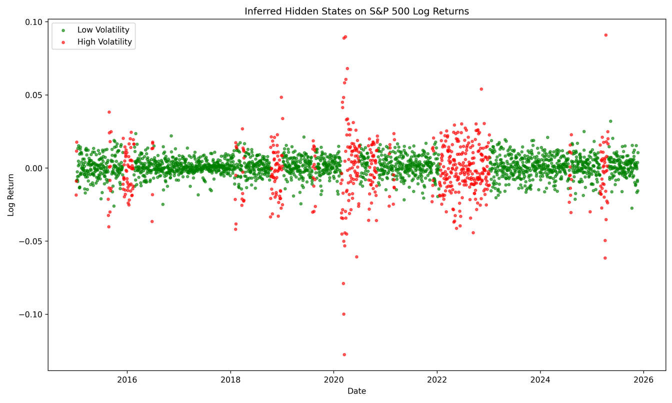

Fig. 2: Inferred states on S&P 500 log returns: Green (low vol.), red (high vol.).

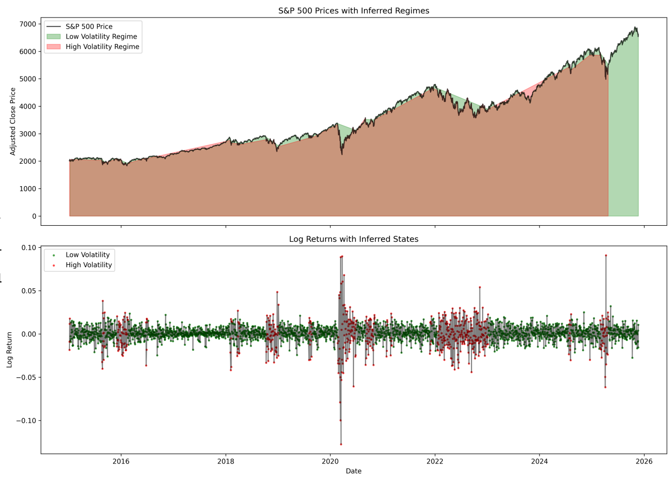

Fig. 3: S&P 500 prices (top) and log returns (bottom) with regime overlays.

B. Regime Visualization on Prices

Overlay states on prices (Fig. 3). High-vol shading highlights drawdowns (e.g., 2020 drop from ~5000 to ~2200).

V. Evaluation and Future Inference

Model fit via log-likelihood: 2-state = 8977.16 (strong evidence of regime structure).

For $N = 3$ states (bonus preview):

- Log-Likelihood: 8953.85 (marginal improvement; note: lower value may reflect overfitting or scaling—use BIC for selection in practice).
- Parameters: Stable Growth ($\mu = 0.00086$, $\sigma^2 = 6.80 \times 10^{-5}$), Moderate Vol ($\mu = 0.00108$, $\sigma^2 = 5.40 \times 10^{-5}$), High Vol ($\mu = -0.00216$, $\sigma^2 = 5.05 \times 10^{-4}$).

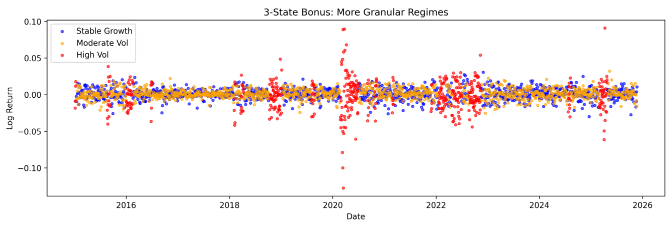

Fig. 4: 3-State bonus: Granular regimes on log returns.

Increasing states adds nuance (e.g., distinguishing moderate from stable) but risks complexity.

A. Future State Prediction

From recent data (last state: Low Volatility), transition probs: Low = 0.986, High = 0.014. Predicted next: Low Volatility (prob 0.986), suggesting short-term stability for risk-off strategies.

VI. Conclusions and Insights

The Gaussian HMM uncovers two regimes: low-volatility growth (State 0: positive drift, low risk) and high-volatility stress (State 1: negative drift, elevated risk). Persistence (0.986/0.948) implies infrequent shifts, useful for portfolio rebalancing—e.g., increase equities in low-vol periods.

Insights: The model captures 2020's high-vol regime amid COVID, informing hedging. For decision-making, monitor transitions for early warnings.

A. Bonus: 3-State Extension and AAPL Comparison

A 3-state model refines regimes (Fig. 4): Stable (blue), Moderate (orange), High Vol (red). It better fits tails but requires validation (e.g., AIC= -2 $\times LL + 2p$).

Comparing with AAPL (2015–2025):

- 2-State Variances: [0.000175, 0.001538].
- AAPL exhibits higher volatility (max var ∼30x S&P's), due to firm-specific risks vs. index diversification.

This underscores HMM's utility across assets.

References

[1] Yahoo Finance API Documentation, *yfinance Python Library*, 2023.
[2] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.
[3] M. V. J.-L. hmmlearn Developers, "hmmlearn: Hidden Markov Models in Python," 2023. [Online]. Available: https://hmmlearn.readthedocs.io/.