# Machine Learning and Stochastic Filtering Techniques for Statistical Arbitrage

## Master’s Thesis - BADS 2024/25

Giovanni Piva
Riccardo Perego

# Purpose

The aim of this thesis is to develop a rigorous methodological framework for the design, estimation, and validation of a quantitative trading strategy (**pairs trading**), with a particular focus on the application of machine learning models and the avoidance of statistical and methodological biases.

The goal is indeed not to propose a profitable trading strategy per se, but rather to highlight the importance of sound methodology and robust statistical validation, which often reveal the inherent limitations of such models and strategies.

# Mean-Reversion processes

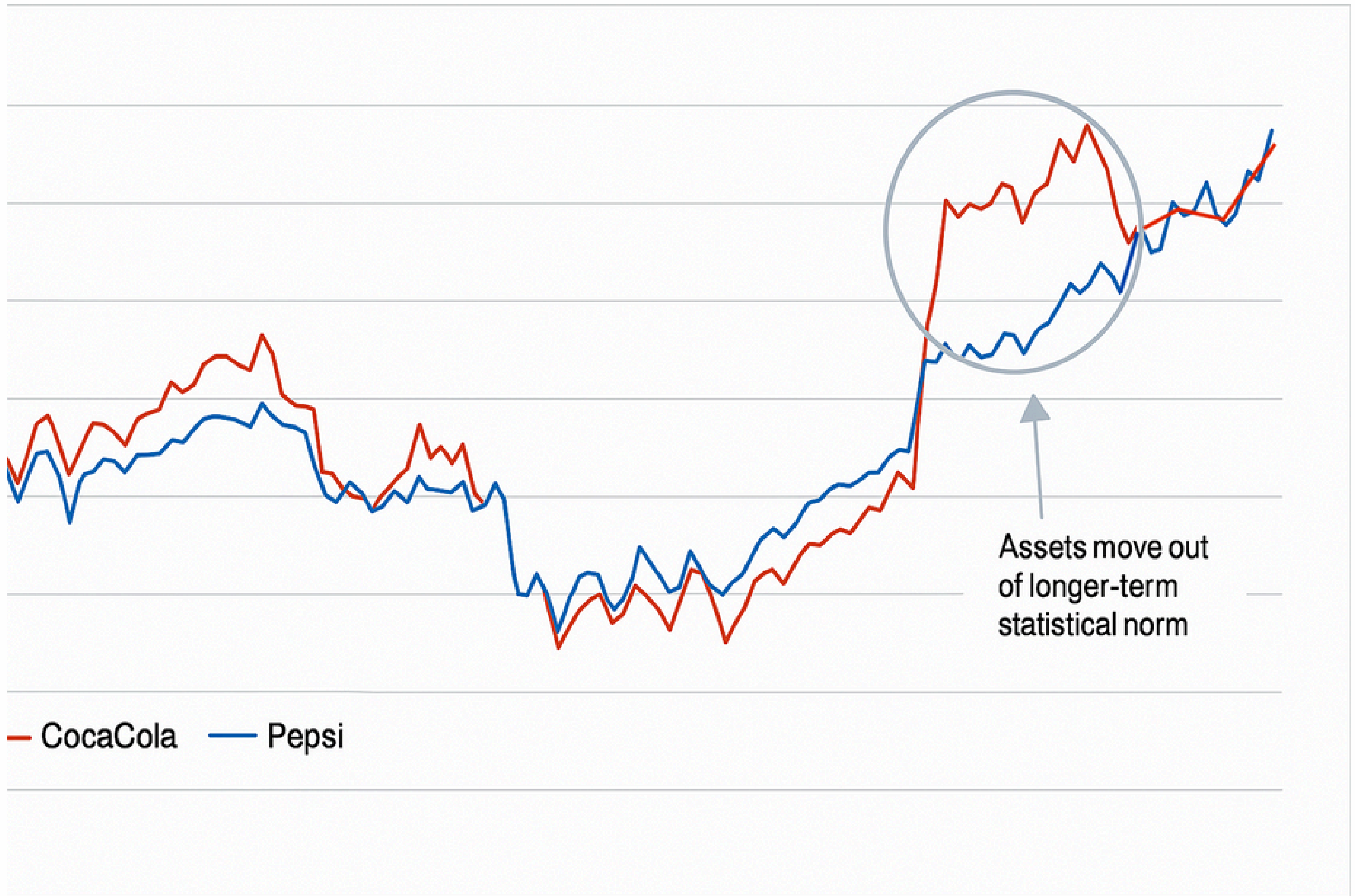
The pairs-trading strategies rely on the mean-reversion phenomenon, according to which **a given variable tends to fluctuate over time around a long-term average value**.
In this context, the variable of interest is the **spread** between the two assets, that is, the residual of their linear relationship:

$$\epsilon_t = y_t - (\alpha_t + \beta_t x_t)$$

According to the Ornstein-Uhlenbeck process, which describes this phenomenon, the stationary distribution of this process is normal:

$$\epsilon_t \sim \mathcal{N}\left(\mu, \frac{\sigma^2}{2\kappa}\right)$$

Example:



# Steps

- **Pair(s) selection** based on economic similarity, correlation, and cointegration tests.
- **Spread estimation** using the *Kalman Filter* to dynamically model the relationship between the assets.
- **Spread standardization** through a rolling-window procedure.
- **Regime identification** with *Hidden Markov Models*, to distinguish mean-reverting phases from volatile or unreliable periods.
- **Trading rule definition** based on the standardized spread (z-score) and regime classification.
- **Backtesting** and performance evaluation on historical data.

# Mastercard - Visa pair

To determine whether a pair is suitable for the strategy, we must verify that the two assets are correlated and that their relationship is stationary, as assessed through a cointegration test.

These conditions ensure that the spread between the two assets exhibits a **stable long-term equilibrium** and **mean-reverting behavior**.

*But how can we estimate the spread using only past data?*

| Test | Results |
|---|---|
| Correlation | 0.9835 |
| ADF Statistic | -4.687 |
| Cointegration p-value | $8.9 \times 10^{-5}$ |
| Critical Values | 1%: $-3.44$; 5%: $-2.86$; 10%: $-2.57$ |

# Spread Estimation with Kalman Filter

To dynamically estimate the relationship between the two assets, $s_t = \begin{bmatrix} \alpha_t \\ \beta_t \end{bmatrix}$,we employ the Kalman filter, a recursive algorithm that provides optimal estimates of time-varying parameters in the presence of noise.

Intuitively, the filter answers the question: "*How much should I trust my estimate compared to the actual measurement*?"

- **State equation:** $s_t = As_{t-1} + w_t, \quad w_t \sim \mathcal{N}(0, Q)$ $\quad A = I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$
- **Observation equation:** $y_t = H_t s_t + v_t, \quad v_t \sim \mathcal{N}(0, R)$ $\quad H_t = \begin{bmatrix} 1 & x_t \end{bmatrix}$

The operation of the Kalman Filter is structured around two main iterative phases:

- **Prediction**: estimates the next state based on the previous one.
- **Update**: the new observation is incorporated to correct the prediction, reducing uncertainty and refining the state estimate.

# Kalman Filter | Estimating Q and R

## *The importance of Q and R:*

- If $Q$ is small, the parameters α_t and β_t change little over time, making the model similar to a *static* OLS regression.
- If $Q$ is large, the parameters become more flexile, autocorrelation decreases, and the model is more *dynamic*.

- If $R$ is small, the filter strongly relies on the observed data, adapting *quickly* to new information.
- If $R$ is large, the observations are treated as highly noisy, so the filter relies more on the model forecast, leading to *slower* updates.

## *Estimation:*

The estimation aims to maximize the log-likelihood of observing the time series $y_t$:

$$\ell(\theta) = \sum_{t=1}^{T} \log f(y_t \mid y_{1:t-1}, \theta)$$

It is performed through the Expectation–Maximization (EM) algorithm, an extension of MLE which iteratively updates the parameters $\theta = \{Q, R\}$ until convergence by alternating between two recursive steps:

- **E-step**: computes the expected value of the latent variables given the current parameter estimates.
- **M-step**: maximizes the likelihood function with respect to *Q* and *R* based on the expectations from the E-step.

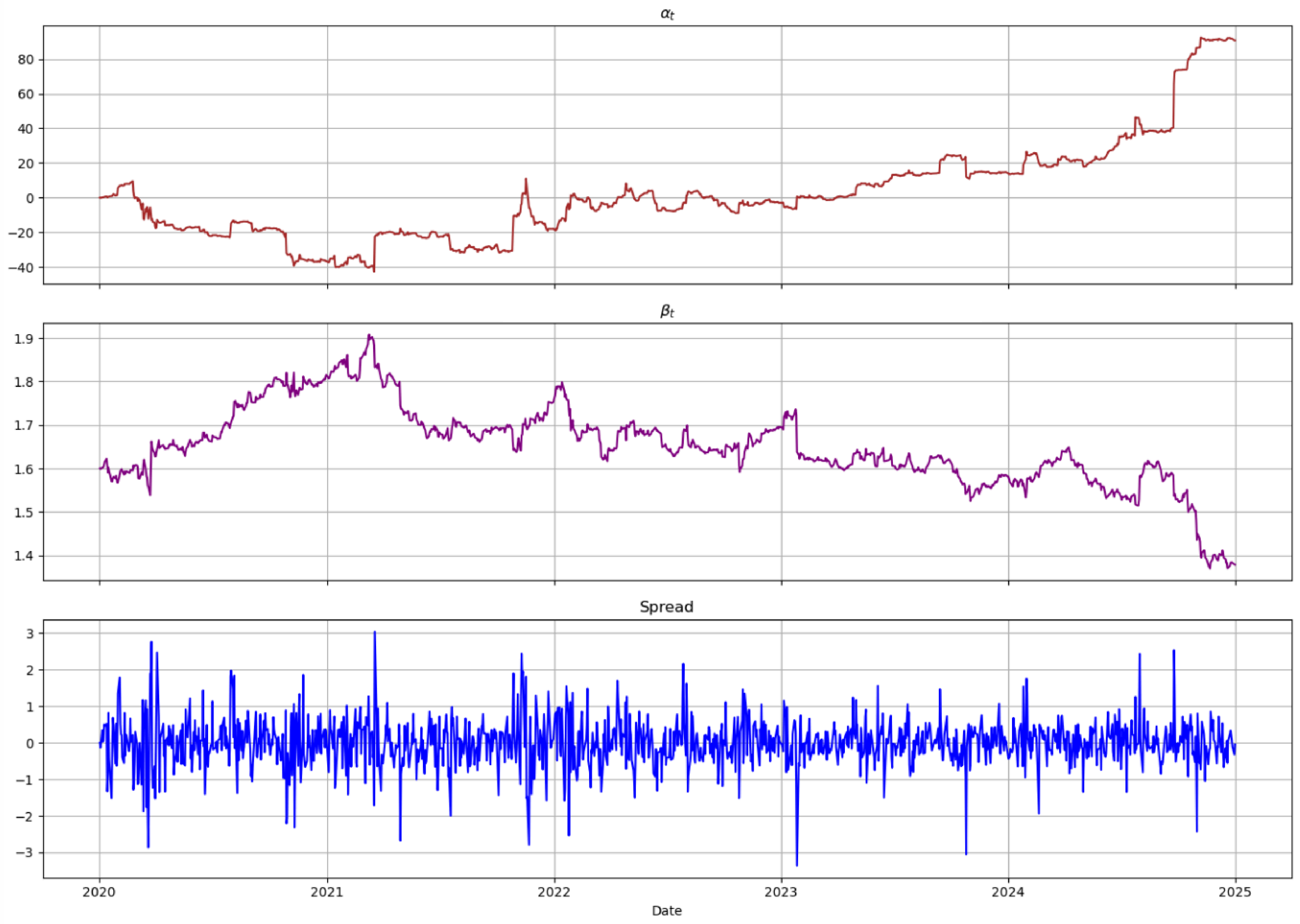
# Kalman Filter | Results

With a 315 days training window:

$$Q = \begin{bmatrix} \mathrm{Var}(\alpha_t) & \mathrm{Cov}(\alpha_t, \beta_t) \\ \mathrm{Cov}(\beta_t, \alpha_t) & \mathrm{Var}(\beta_t) \end{bmatrix}, \qquad R = [\sigma^2_{\mathrm{obs}}]$$

$\big\downarrow$ **EM ALGORITHM**

$$Q_{\text{MA-V}} = \begin{bmatrix} 1.6279 & -0.0019 \\ -0.0019 & 0.0001 \end{bmatrix}, \qquad R_{\text{MA-V}} = 2.28$$




After standardizing the spread using a rolling window, we are ready to define the two market regimes.

# Hidden Markov Model | Markov Chains & Market Regimes

- A **Markov chain** is a stochastic process where the next state depends only on the current state, not on the sequence of past states.

$$\mathbb{P}(X_{t+1} = j \mid X_0, X_1, \ldots, X_t) = \mathbb{P}(X_{t+1} = j \mid X_t)$$
$$\forall j \in \{0, 1, \ldots, N-1\}$$

- A **transition probability matrix** defines how likely the system is to move from one state to another, with each row representing probabilities that sum to one. In our context, states represent market regimes.

$$P = \begin{bmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{bmatrix} \quad \text{with} \quad \begin{cases} p_{00} + p_{01} = 1 \\ p_{10} + p_{11} = 1 \end{cases}$$

A **Hidden Markov Model** extends this framework to situations where the underlying state (*stable* vs. *volatile*) is not directly observable. Indeed, we only observe quantities such as returns or spreads, whose behavior depends on that hidden state. The HMM links these unobserved regimes to observable data through state-dependent probability distributions, allowing us to infer which regime the market is likely in at any point in time and how it transitions over time.

$$\mathbb{P}(X_{t+1} = j \mid X_t = i) = p_{ij}, \quad \forall i, j \in \mathcal{S}, \quad \sum_{j=0}^{N-1} p_{ij} = 1.$$

$$Z_t \mid X_t = i \sim \mathcal{N}(\mu_i, \sigma_i^2)$$

where each hidden state $i$ is associated with its own mean $\mu_i$ and variance $\sigma_i^2$.

# Hidden Markov Model | The Algorithm & Regimes Identification

The **Viterbi algorithm** efficiently computes the most likely sequence of hidden states given the observed data, avoiding the need to evaluate all possible state paths. In our setting, it allows us to infer the most probable market regime at each point in time based on the observed z-scores: $\arg\max_{X_1, X_2, \ldots, X_T} \prod_{t=1}^{T} p(Z_t \mid X_t)\, p(X_t \mid X_{t-1})$

**Model Training**

The HMM is estimated on a 315-day training window using the EM algorithm, which learns both the *transition probabilities* and the *Gaussian emission parameters* $(\mu_i, \sigma_i^2)$ for each hidden state. The state with lower variance is labeled as the stable regime, and the state with higher variance as the risky regime; these parameters remain fixed for the entire out-of-sample period.

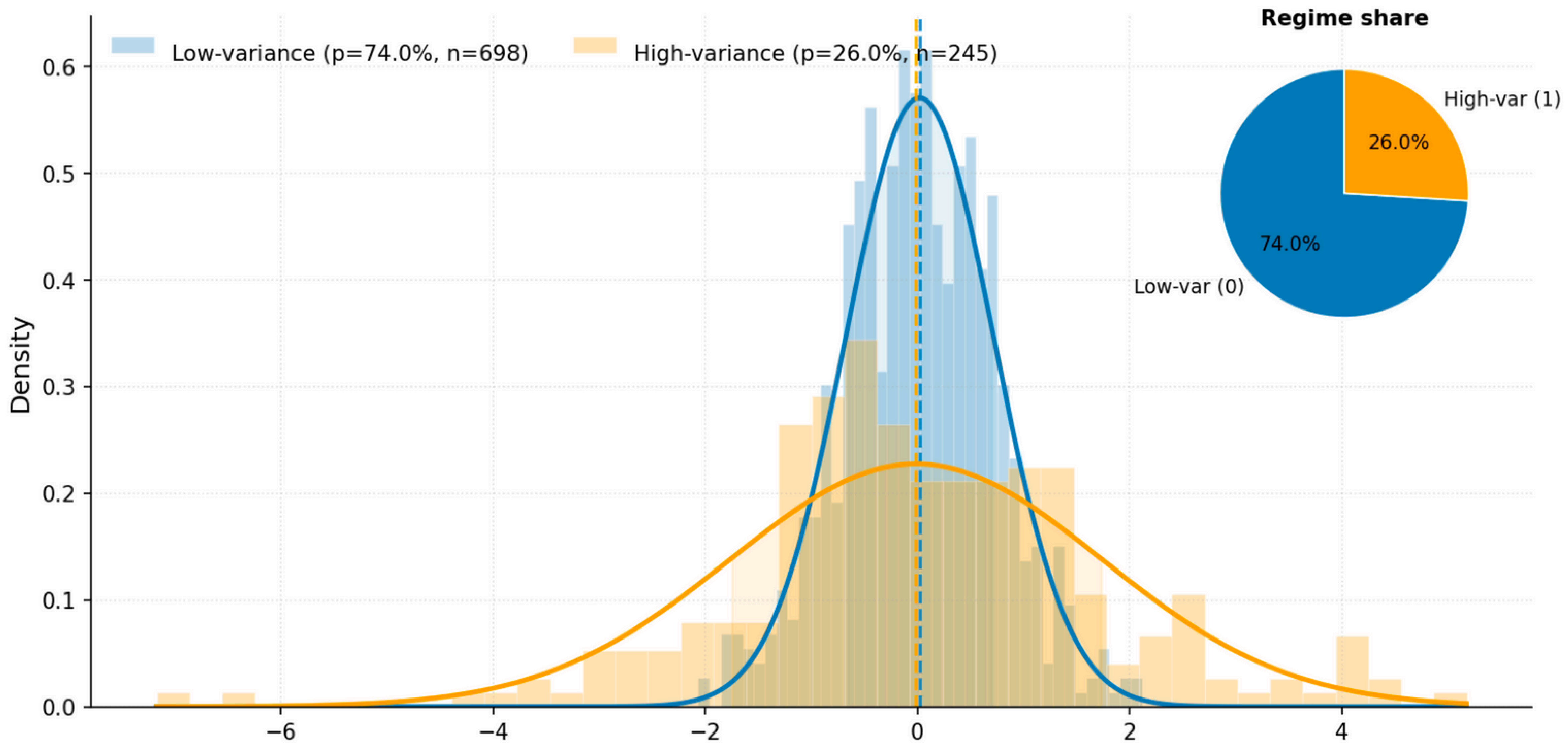
**Real-Time Regime Inference**

At each time $t$, the model takes the observed z-scores up to $t$ and applies the Viterbi algorithm to determine the most likely hidden state path, assigning today's regime $R_t$ based solely on past and current information, thereby avoiding look-ahead bias.

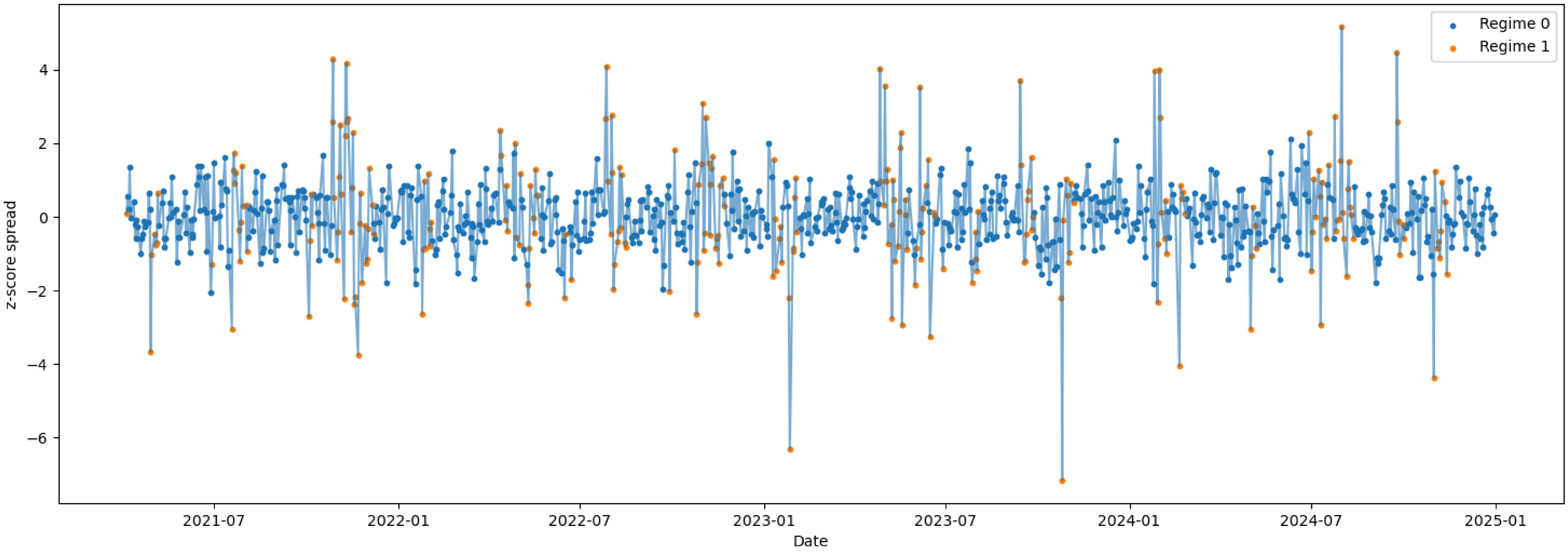
# Hidden Markov Model | Results

$$\text{Estimated State 0: } \mu = 0.02, \quad \sigma = 0.699$$
$$\text{Estimated State 1: } \mu = -0.01, \quad \sigma = 1.745$$

$$P = \begin{bmatrix} 0.97 & 0.03 \\ 0.06 & 0.94 \end{bmatrix}$$



# Hidden Markov Model | Results



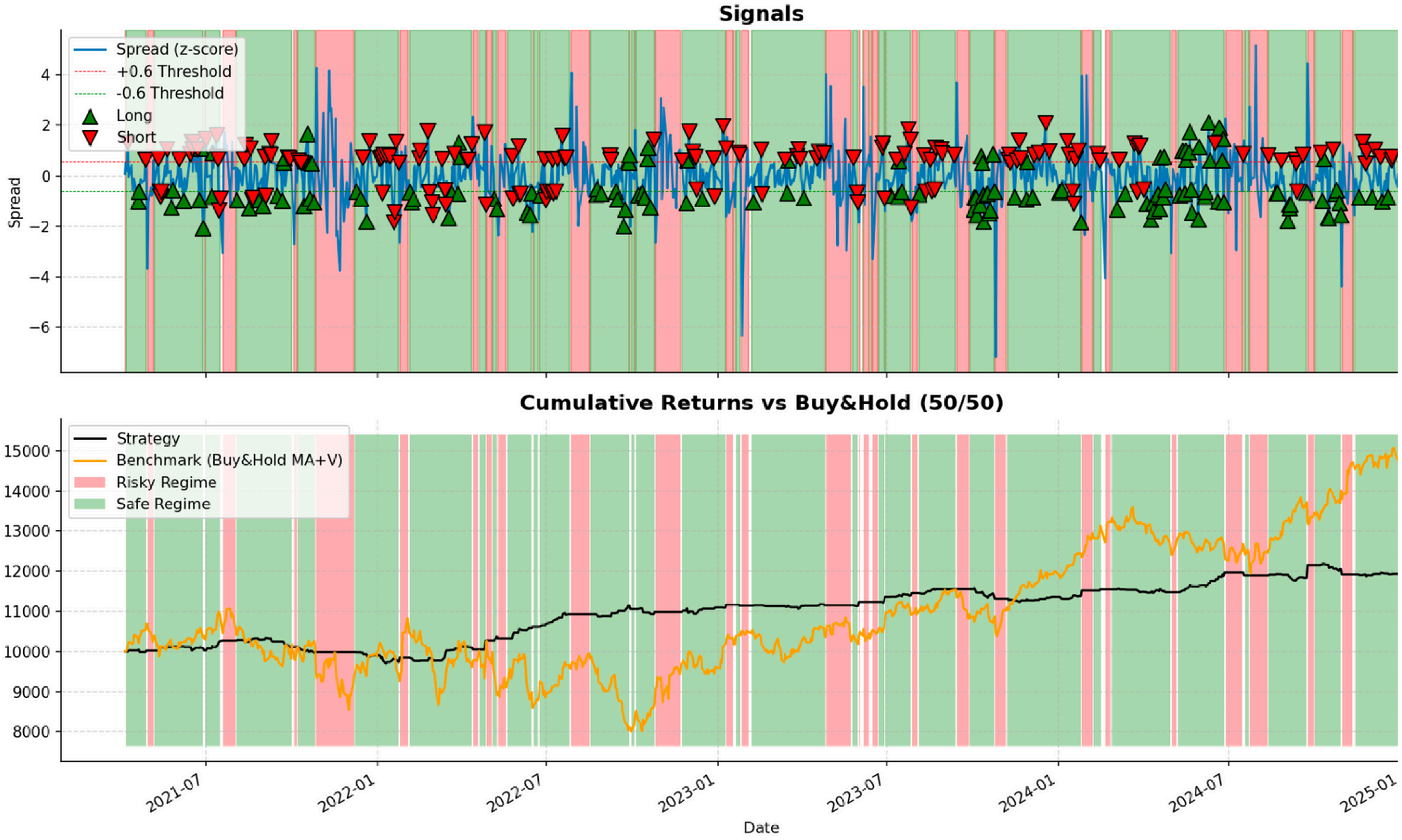
## The Strategy | Signals & Rules

The strategy requires a 315-day burn-in period, meaning that the **training and initialization** phase is necessary for the Kalman Filter to estimate $Q$ and $R$ and for the HMM to learn how the z-score maps to different market regimes; signals are produced only after this period, starting on April 5, 2021.

A signal is generated only when the z-score crosses one of the predefined entry or exit thresholds and the HMM classifies the market as mean-reverting, so **the signal generation conditions** depend jointly on statistical deviation and the inferred regime.
There are three kind of signals: long, short, hold/no trade.

Trades are executed on the day after the signal is triggered to avoid look-ahead bias, and **the execution rules and threshold choices** influence how sensitive the strategy is: the thresholds used are fixed and manually selected, but in principle could be adapted dynamically.

# The Strategy | Backtest & Performance Evaluation



# The Strategy | Backtest & Performance Evaluation

| Metric | MA–V | IBM–NVDA | SHEL–MPC | SHEL–VLO | BP–COP | AMZN–DLR |
|---|---|---|---|---|---|---|
| Correlation | 98.35% | 93.30% | 96.18% | 96.00% | 92.40% | 87.40% |
| Coint. $p$-value | 8.9e-05 | 0.0026 | 0.0007 | 0.0012 | 0.0034 | 0.0004 |
| Entry / Exit | 0.6–0.5 | 0.8–0.2 | 0.6–0.1 | 1.1–0.1 | 1.1–0.3 | 0.8–0.6 |
| Sharpe ratio | 1.51 | 0.78 | 1.08 | 1.51 | 1.24 | 1.59 |
| Sortino ratio | 2.73 | 1.21 | 1.61 | 3.22 | 1.86 | 2.89 |
| Total Returns | 19.16% | 39.86% | 52.03% | 26.01% | 35.02% | 115.8% |
| Ann. Volatility | 3.80 % | 15.71% | 13.50% | 5.05% | 8.12% | 16.52% |
| CAGR | 7.03% | 13.80% | 17.54% | 9.36% | 12.32% | 34.58% |
| Max Drawdown | -6.22% | -24.21% | -13.92% | -3.43% | -9.81% | -19.21% |
| Calmar | 1.13 | 0.57 | 1.26 | 2.73 | 1.25 | 1.80 |
| Kurtosis | 24.96 | 17.8 | 6.18 | 6.19 | 27.60 | 25.91 |
| Trades | 459 | 433 | 766 | 128 | 230 | 570 |
| Activity ratio | 32.45% | 37.33% | 72.85% | 10.92% | 18.98% | 41.14% |
| Win rate | 53.61% | 46.65% | 47.12% | 61.72% | 50.43% | 52.63% |
| HMM regimes | ON | ON | ON | ON | ON | OFF |

# The Strategy | The impact of the HMM

| | MA-V | IBM-NVDA | SHEL-MPC | SHEL-VLO | BP-COP | AMZN-DLR |
|---|---|---|---|---|---|---|
| Sharpe ratio with HMM | **1.51** | 0.78 | 1.08 | **1.51** | **1.24** | 1.46 |
| Sharpe ratio without HMM | 1.34 | 0.50 | 0.66 | 1.09 | 0.85 | **1.59** |
| Sharpe ratio buy&hold | 0.70 | **1.89** | **1.21** | 0.92 | 0.85 | 0.62 |

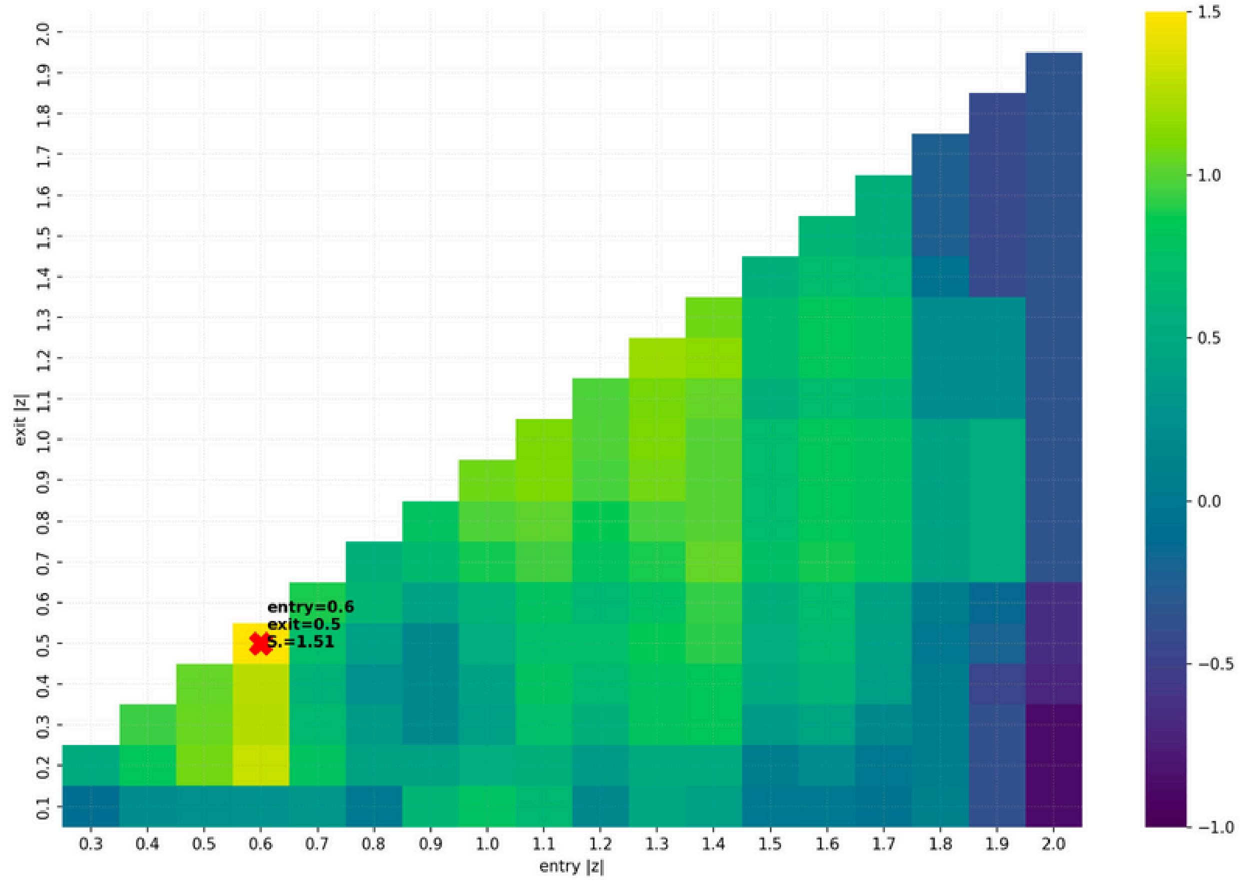



Figure 4.1: Grid search with HMM

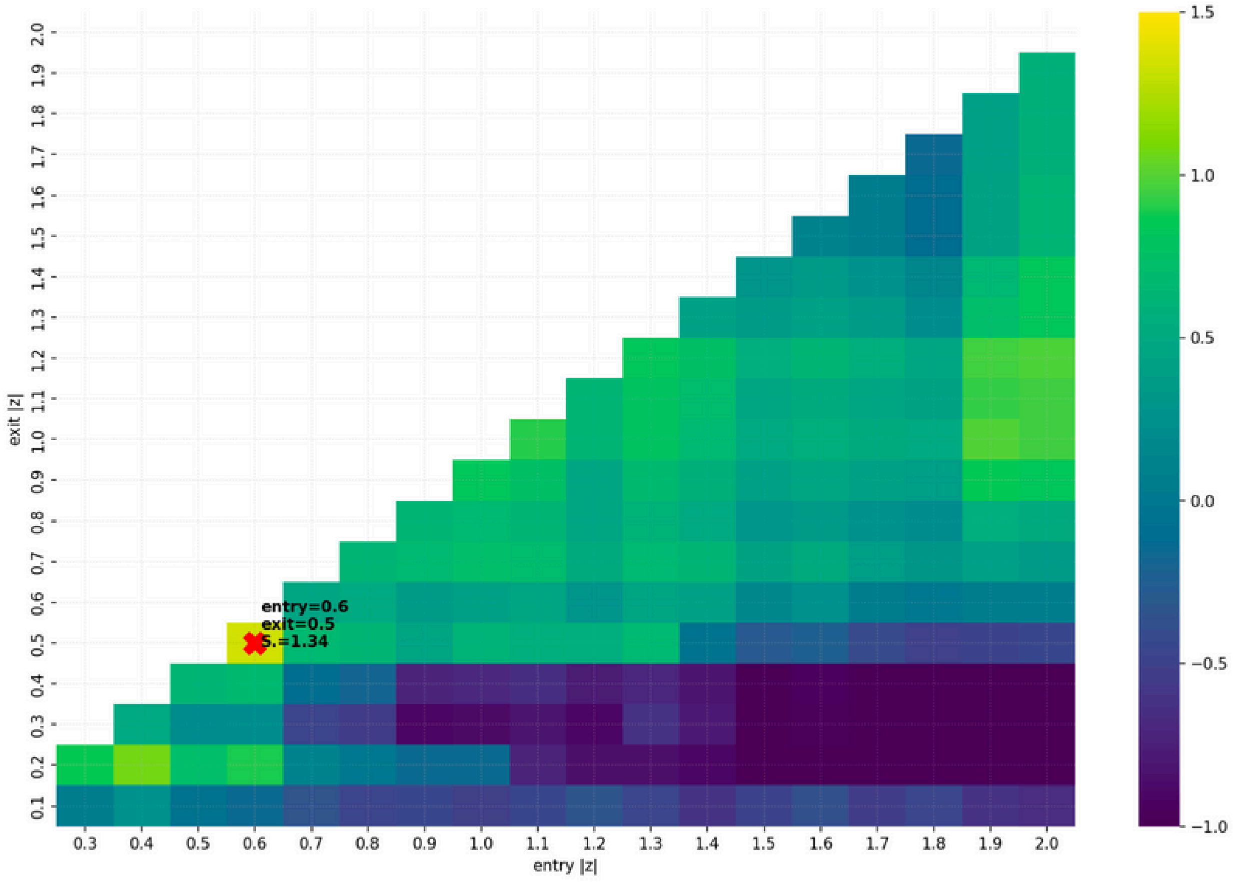



Figure 4.2: Grid search without HMM

# Conclusion

The project demonstrated how the combined use of the Kalman Filter and the Hidden Markov Model enables a strategy that adapts to changing market conditions while incorporating regime-based risk awareness.

The work emphasized the importance of rigorous methodology in quantitative strategy design, particularly in avoiding biases, managing parameter uncertainty, and ensuring that the model remains grounded in realistic trading conditions.

The findings suggest that future developments of the strategy may focus on dynamic threshold selection, refined parameter optimization, and enhanced risk controls, offering a path toward more adaptive and robust quantitative trading frameworks.

# Thank you