"Hierarchical ML for Market Regime Detection and Trading Strategy Performance Prediction"

Sergio Vizcaino Ferrer

Motivation and Problem

Design

Results

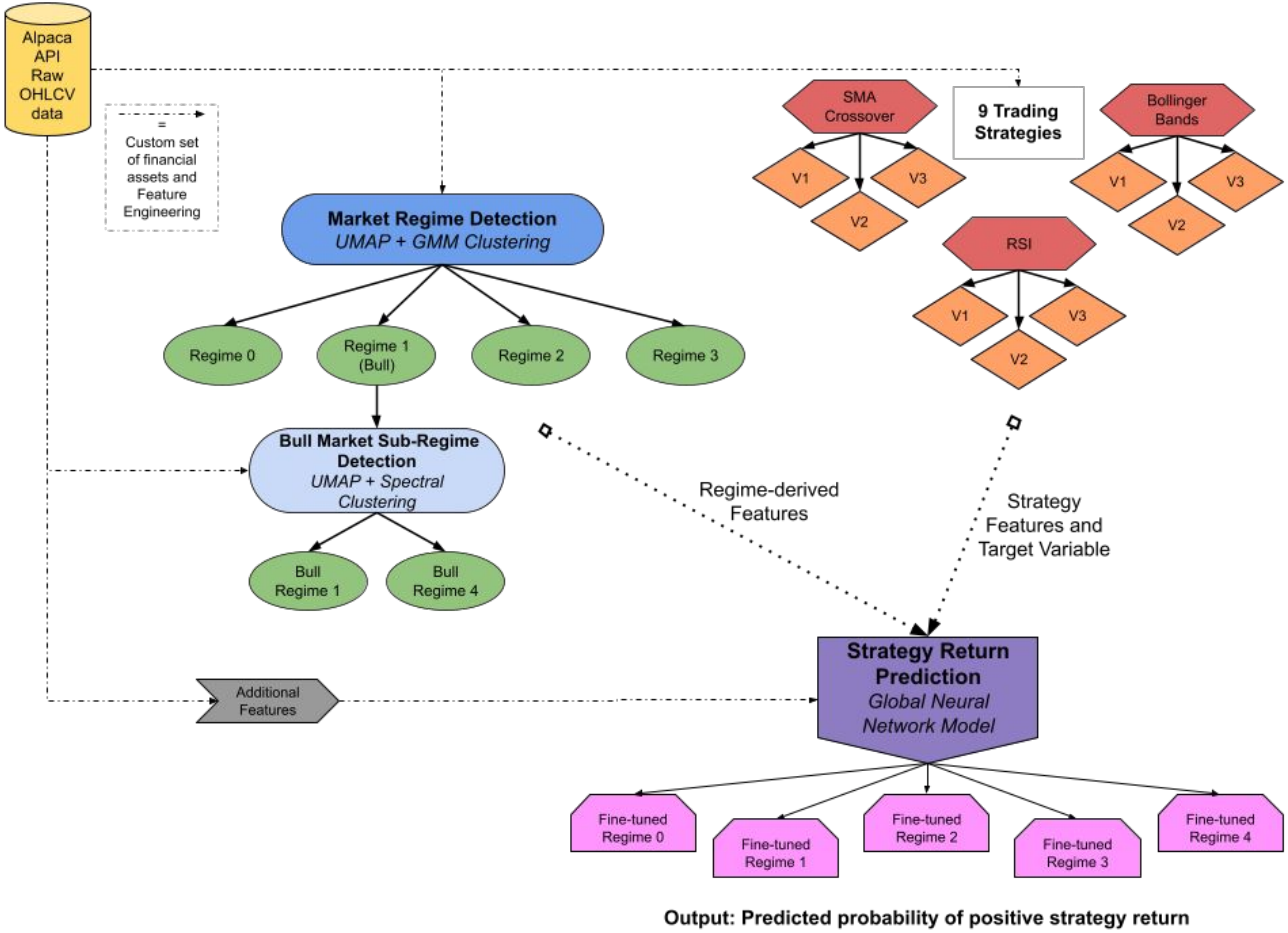
Motivation

- Intellectual **challenge** and **competitive** nature of financial markets as a testing ground for machine learning methods.
- Driven by the pursuit of strong, **above-average returns** despite the obstacles of transaction costs and non-stationary market behavior.

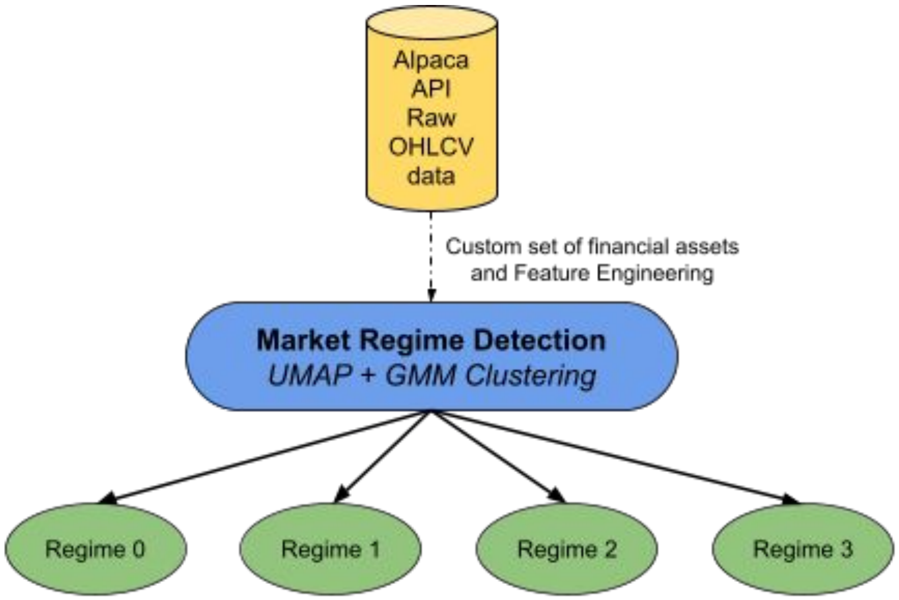
Problem

- **Traditional trading strategies** and machine learning models often **fail** under **regime shifts**, leading to fragile performance.
- There is a need for a **regime-aware framework** that integrates ML models with classical strategies to **improve strategy selection** and timing in real-world trading.

Design

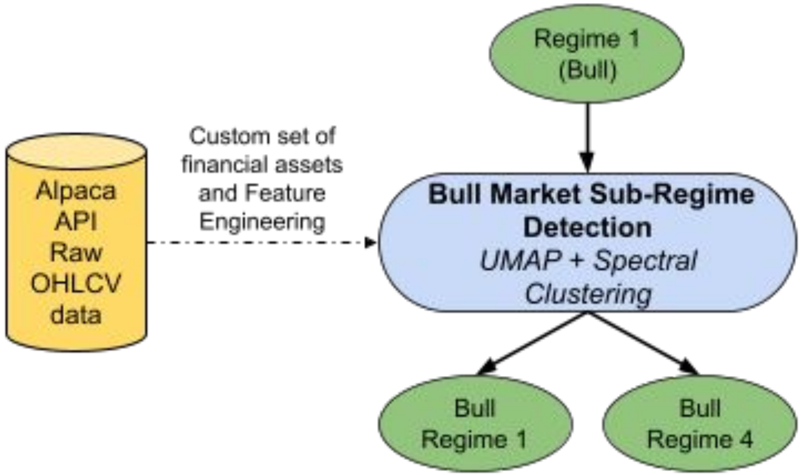

Alpaca API Raw OHLCV data
= Custom set of financial assets and Feature Engineering
Market Regime Detection
UMAP + GMM Clustering
Regime 0
Regime 1 (Bull)
Regime 2
Regime 3
Bull Market Sub-Regime Detection
UMAP + Spectral Clustering
Bull Regime 1
Bull Regime 4
9 Trading Strategies
SMA Crossover
Bollinger Bands
RSI
V1
V2
V3
Regime-derived Features
Strategy Features and Target Variable
Additional Features
Strategy Return Prediction
Global Neural Network Model
Fine-tuned Regime 0
Fine-tuned Regime 1
Fine-tuned Regime 2
Fine-tuned Regime 3
Fine-tuned Regime 4
Output: Predicted probability of positive strategy return

Regime Detection Model



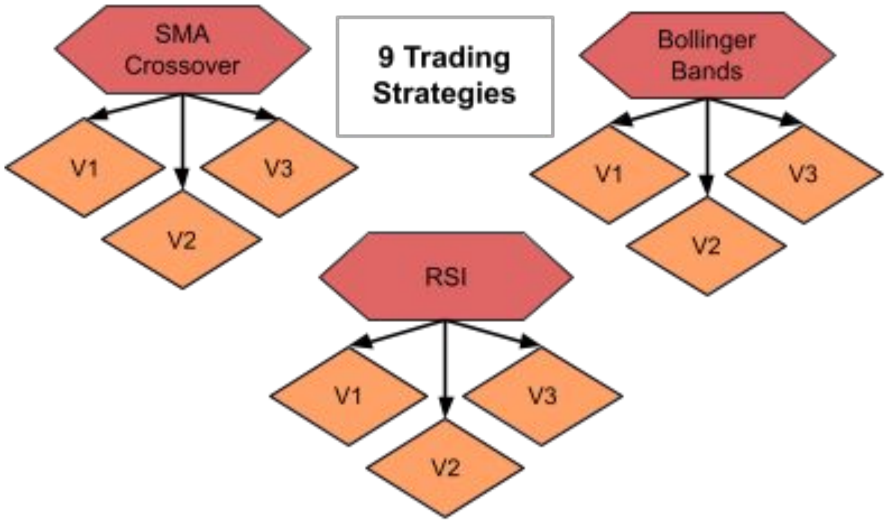
- ❑ Complex engineered **features** that capture short- and medium-term patterns, relative strengths, and trends, which **measure** the **investors risk appetite** and global **market state**
- ❑ Preprocessing, scaling and **dimensionality reduction** with **UMAP** (3 components and correlation distance as metric)
- ❑ **GMM** Clustering model, with K=4. **Post**-clustering **smoothing**, 5-day rolling majority vote

Bull Sub-Regime Model



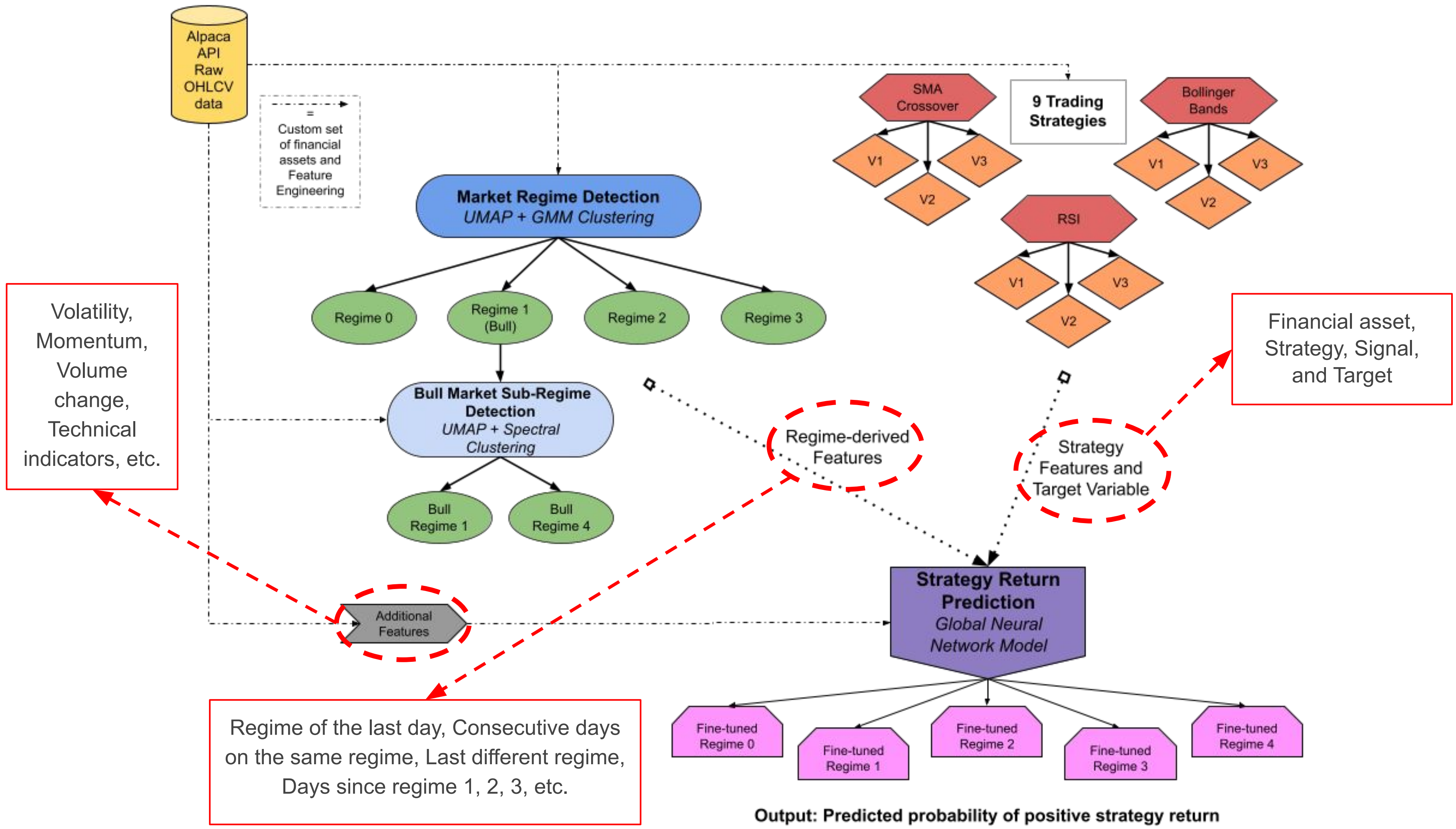
- **Why?** Because **Bull** regime is the **most frequent** regime with **40%** share, potential of more **granular separation**
- Different **feature set** designed to reflect the positive **trend consistency** and **strength**
- Preprocessing, scaling and **dimensionality reduction** with **UMAP** (3 components and correlation distance as metric)
- **Spectral Clustering** model, with K=2

Trading Strategies Computation

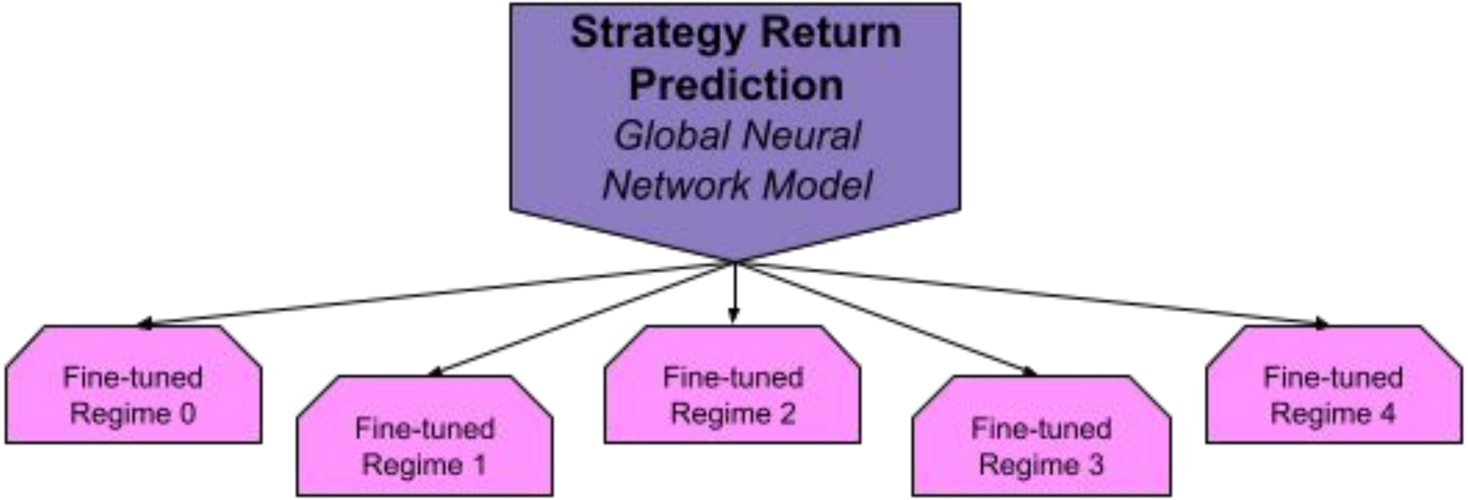


- 3 of the most **classical** and straight forward **trading strategies**, with 3 different version each. They produce **signals** of **buy** (long position) or **sell** (short position) financial assets

- **After 10 days** the profitability of the signal is computed. If the trade **earned money**: 1, if not: 0, that is our **binary target variable** to predict.



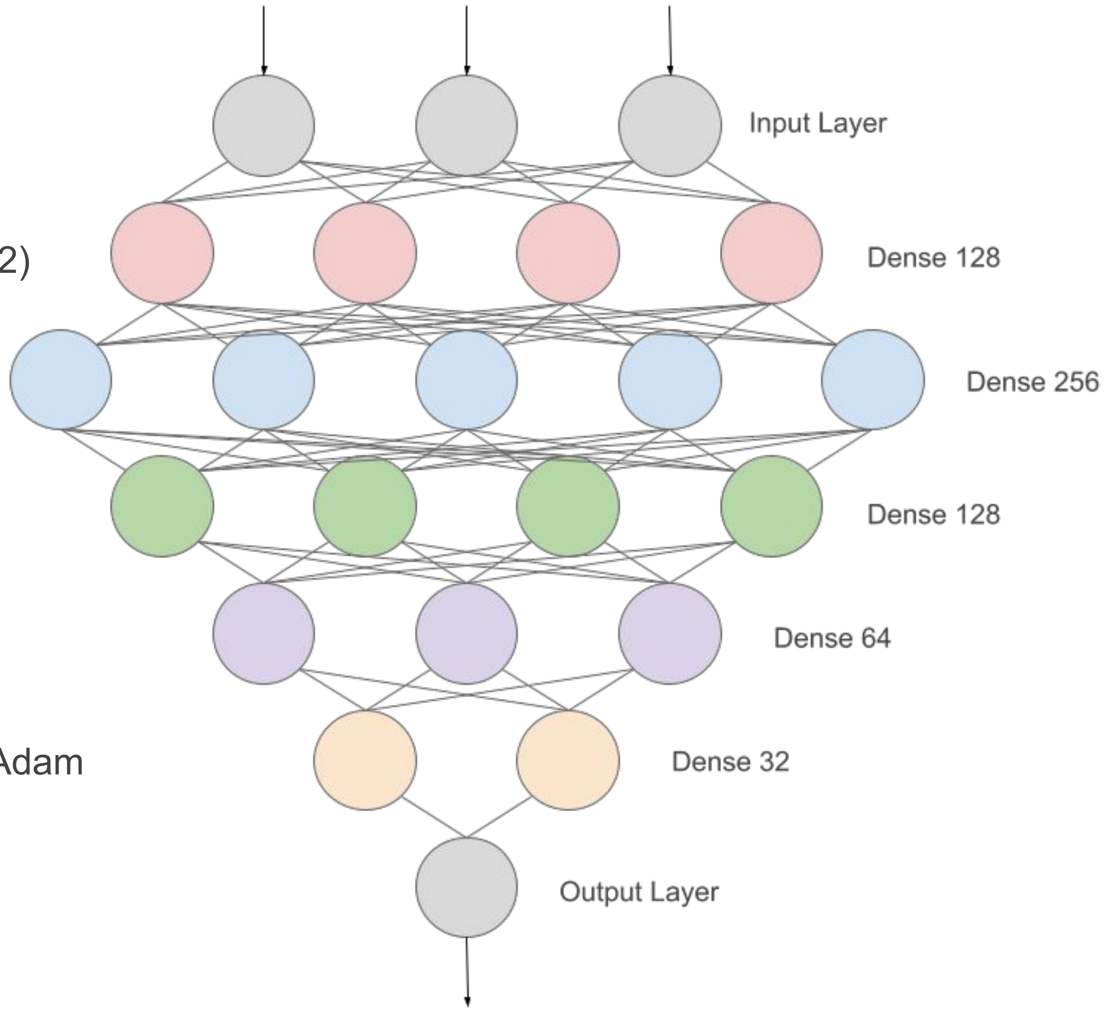
Alpaca API Raw OHLCV data
=
Custom set of financial assets and Feature Engineering
Market Regime Detection
UMAP + GMM Clustering
Regime 0
Regime 1 (Bull)
Regime 2
Regime 3
Bull Market Sub-Regime Detection
UMAP + Spectral Clustering
Bull Regime 1
Bull Regime 4
SMA Crossover
9 Trading Strategies
Bollinger Bands
RSI
V1
V2
V3
Volatility, Momentum, Volume change, Technical indicators, etc.
Financial asset, Strategy, Signal, and Target
Regime-derived Features
Strategy Features and Target Variable
Additional Features
Strategy Return Prediction
Global Neural Network Model
Fine-tuned Regime 0
Fine-tuned Regime 1
Fine-tuned Regime 2
Fine-tuned Regime 3
Fine-tuned Regime 4
Regime of the last day, Consecutive days on the same regime, Last different regime, Days since regime 1, 2, 3, etc.
Output: Predicted probability of positive strategy return



Output: Predicted probability of positive strategy return

Neural Network Architecture

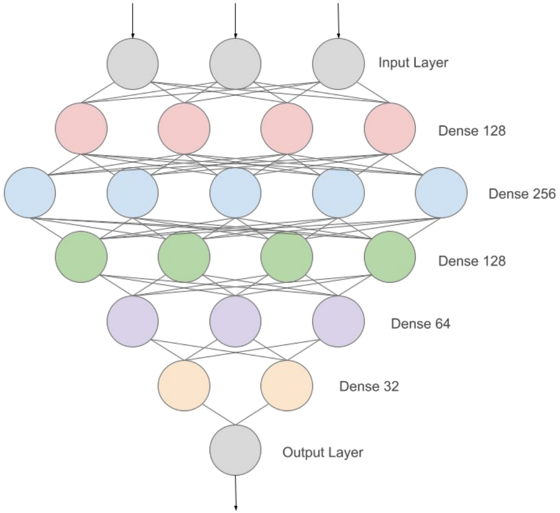
- **5 Dense Layers** (128, 256, 128, 64, 32)
- High **dropout** rate (40%) to avoid **overfitting**
- 50% **more weight** to the **positive class** (signals that return positive investment)
- Batch normalization, He initialization, Adam



Neural Network Training

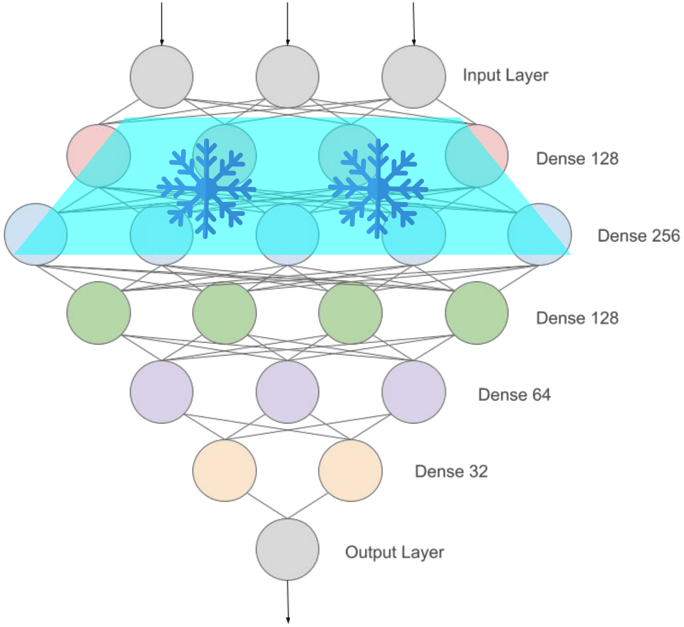
1. **Standardize** numerical variables (using only the fold's train set to fit the Scaler to avoid data leakage).

2. **Fit** the **Neural Network** with the entire fold's train set (**Global** Model)



Neural Network Training

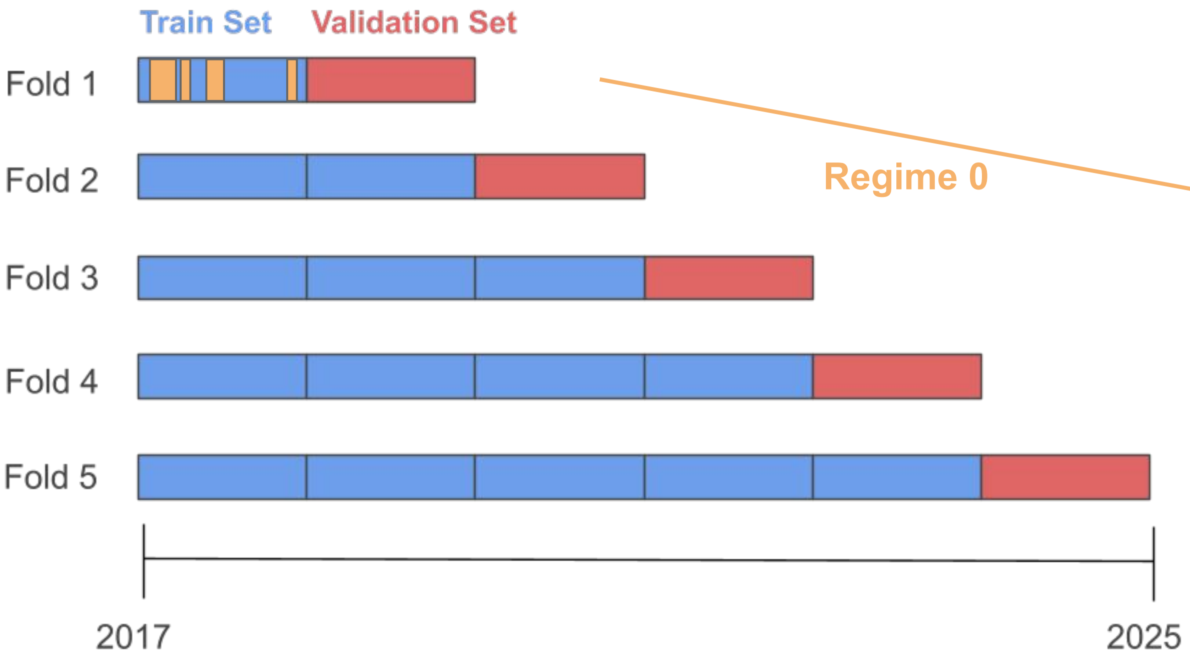
3. **Freeze** the first 2 layers

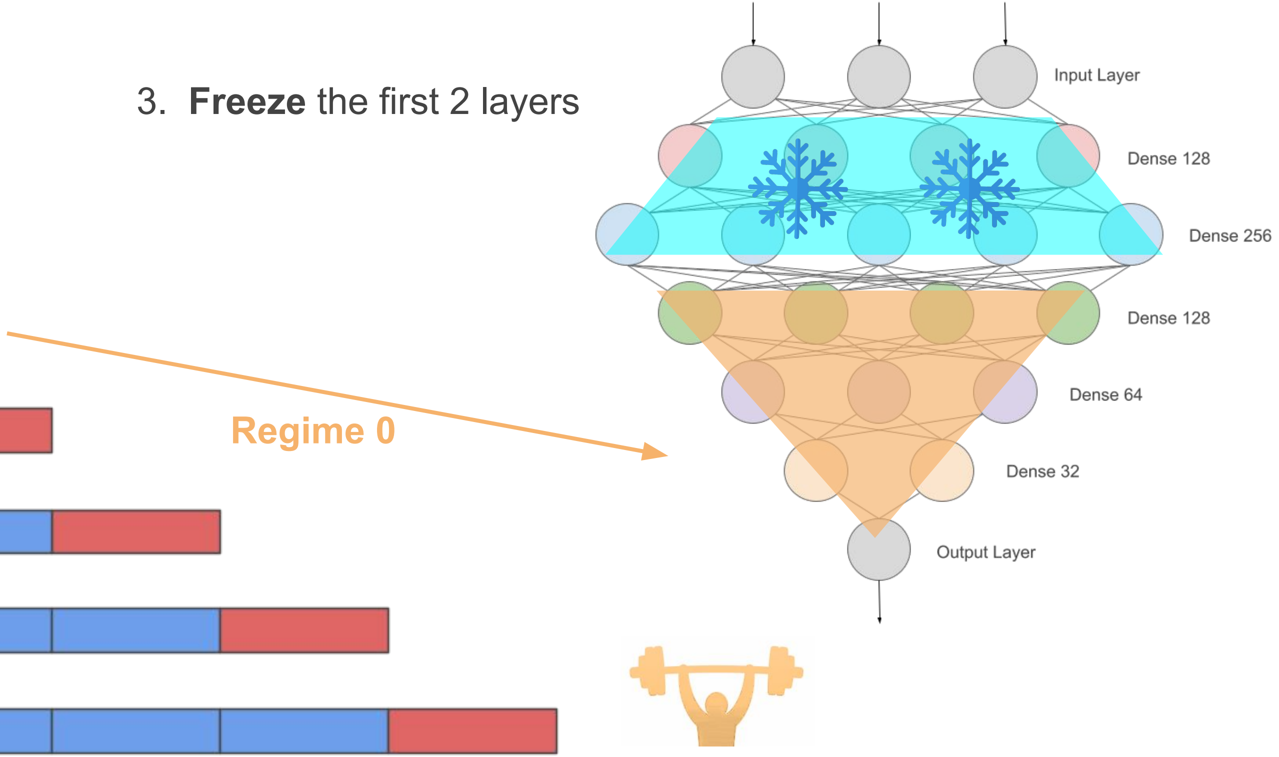


4. **Fine-tune** per Regime

Neural Network Training

3. **Freeze** the first 2 layers

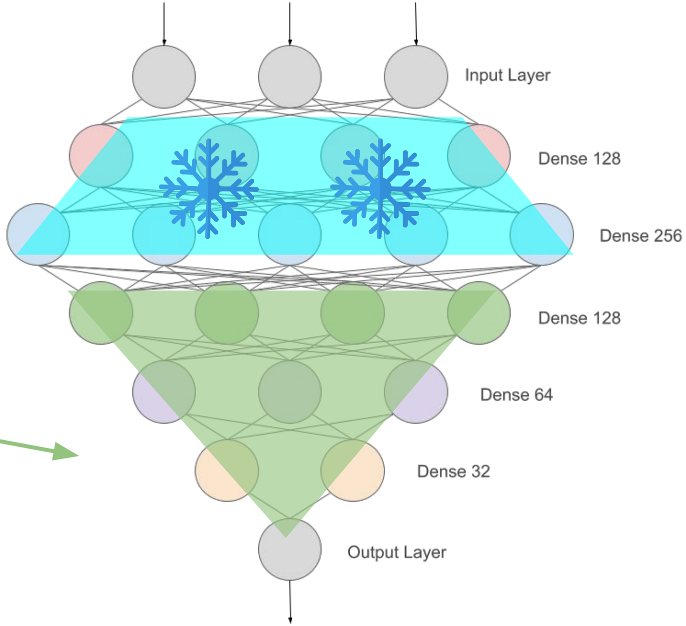




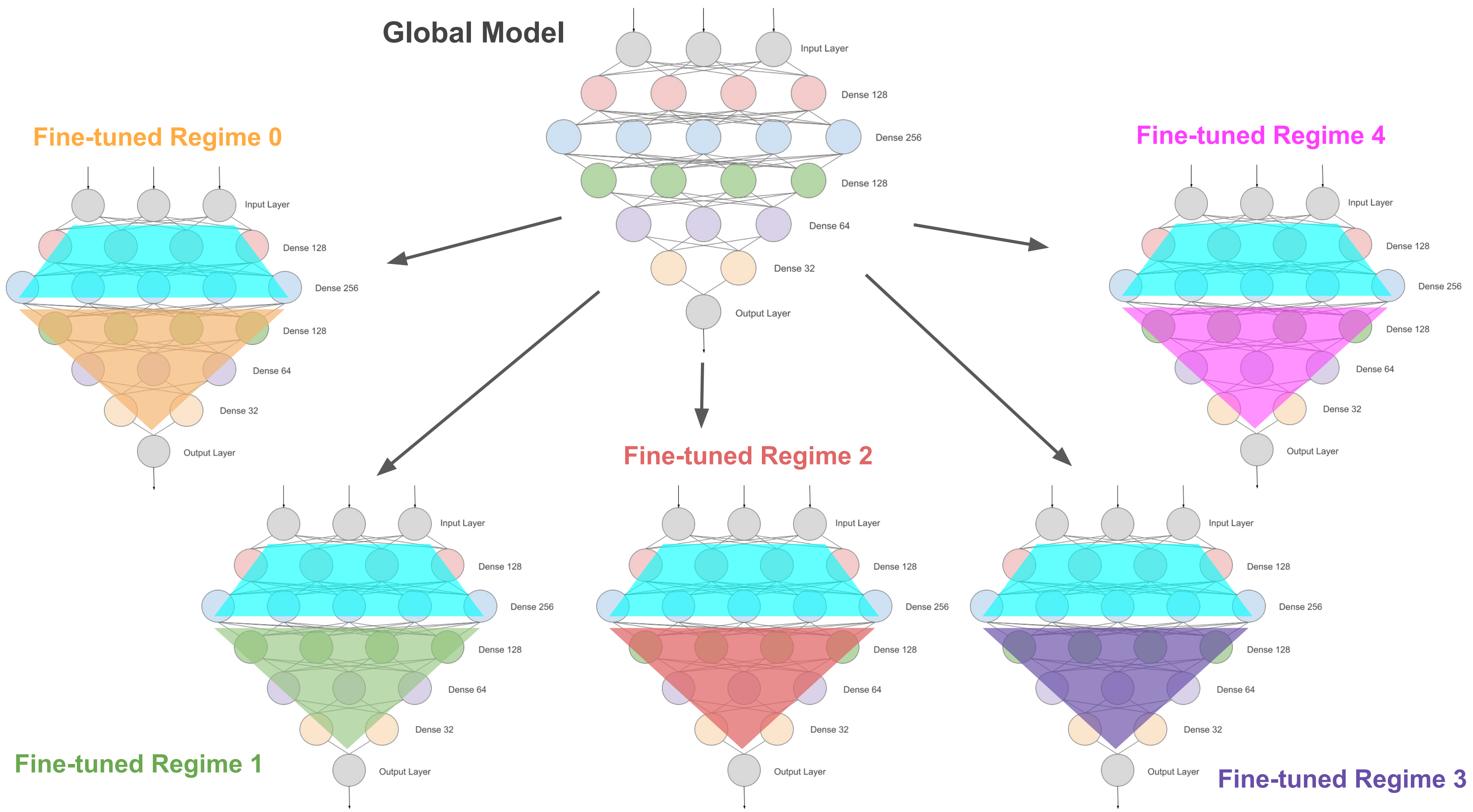
4. **Fine-tune** per Regime. Train a network for each regime using only the days labelled previously as the regime

Neural Network Training

3. **Freeze** the first 2 layers



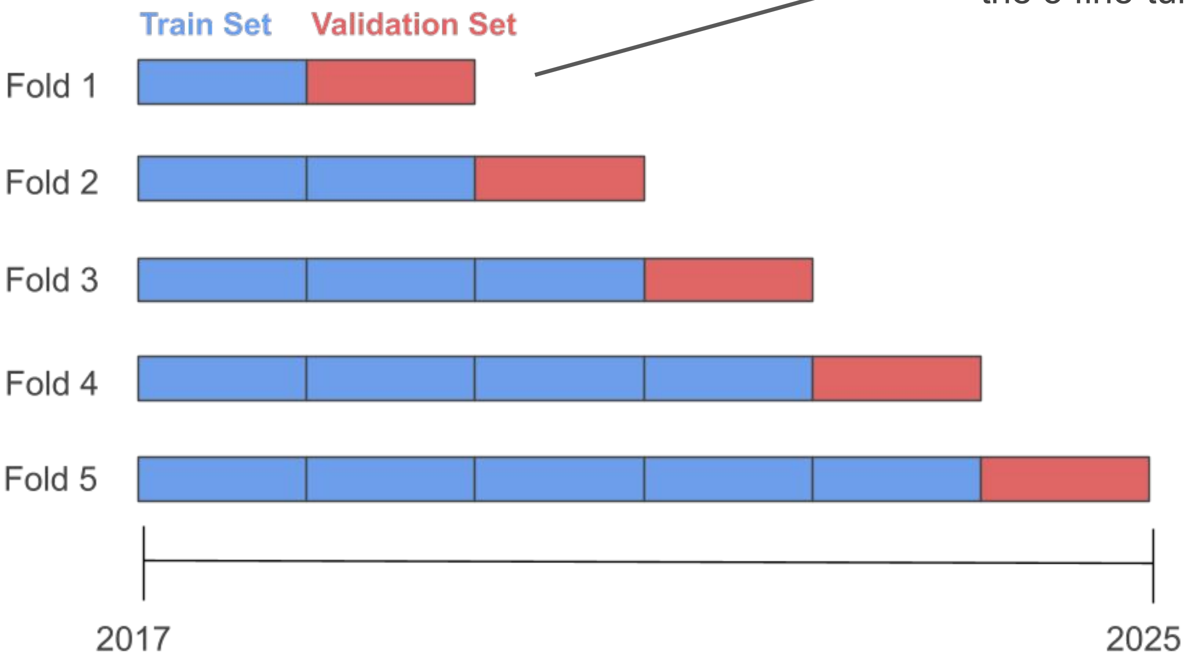
4. **Fine-tune** per Regime. Train a network for each regime using only the days labelled previously as the regime



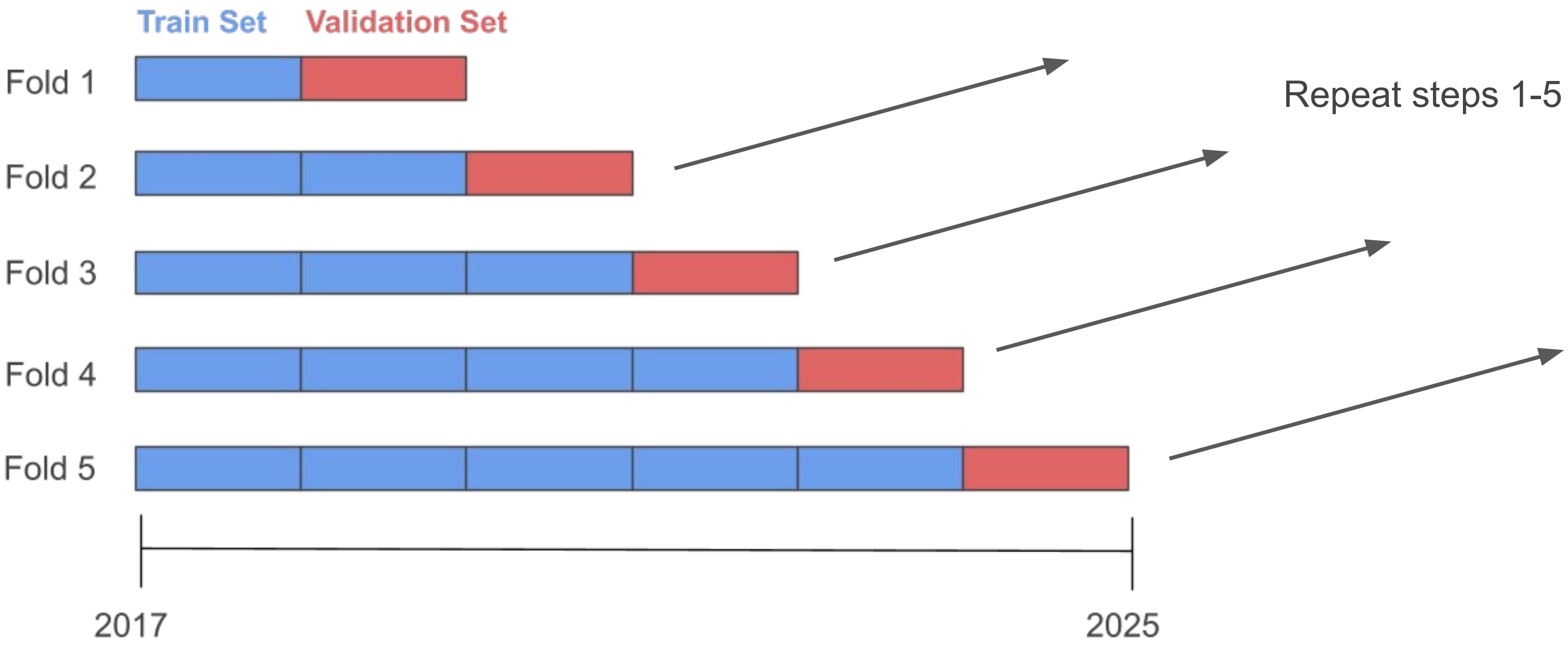
Global Model
Input Layer
Dense 128
Dense 256
Dense 128
Dense 64
Dense 32
Output Layer
Fine-tuned Regime 0
Fine-tuned Regime 4
Fine-tuned Regime 2
Fine-tuned Regime 1
Fine-tuned Regime 3

Neural Network Training

5. **Evaluate** and compute metrics, returns, performance of all 6 models, the Global and the 5 fine-tuned ones.



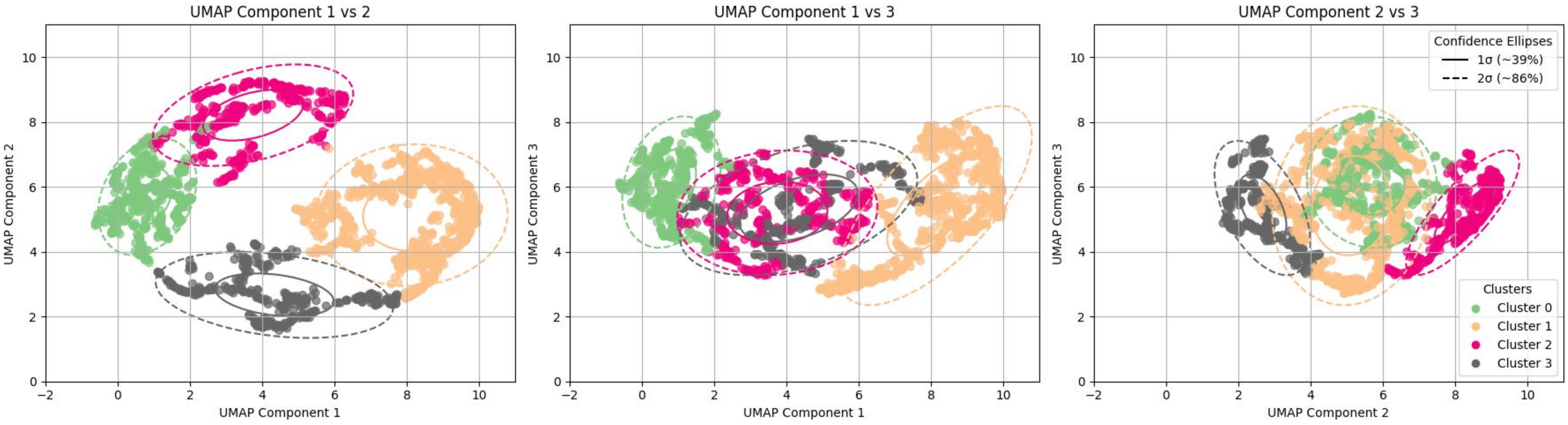
Neural Network Training



Repeat steps 1-5 per Fold

Results

Regime Detection GMM



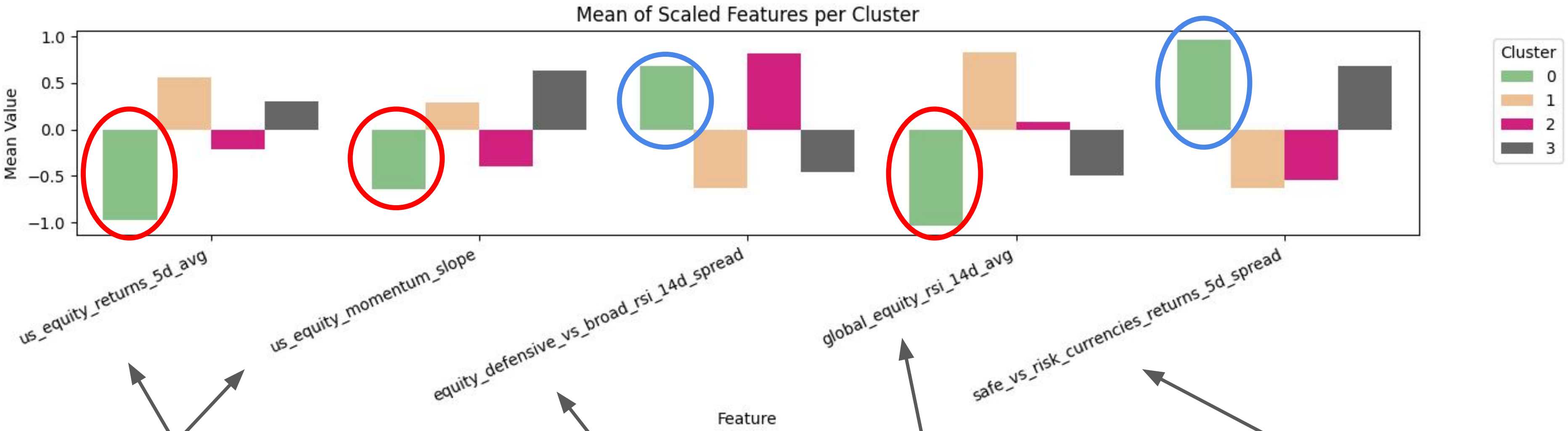
Interpretation

Low values = pessimistic environment

High values = investors looking for protection

Regime 0 = Bear

Market going down, Crisis periods, …



Measure **U.S.** stocks trends

Strength spread between defensive companies vs normal companies

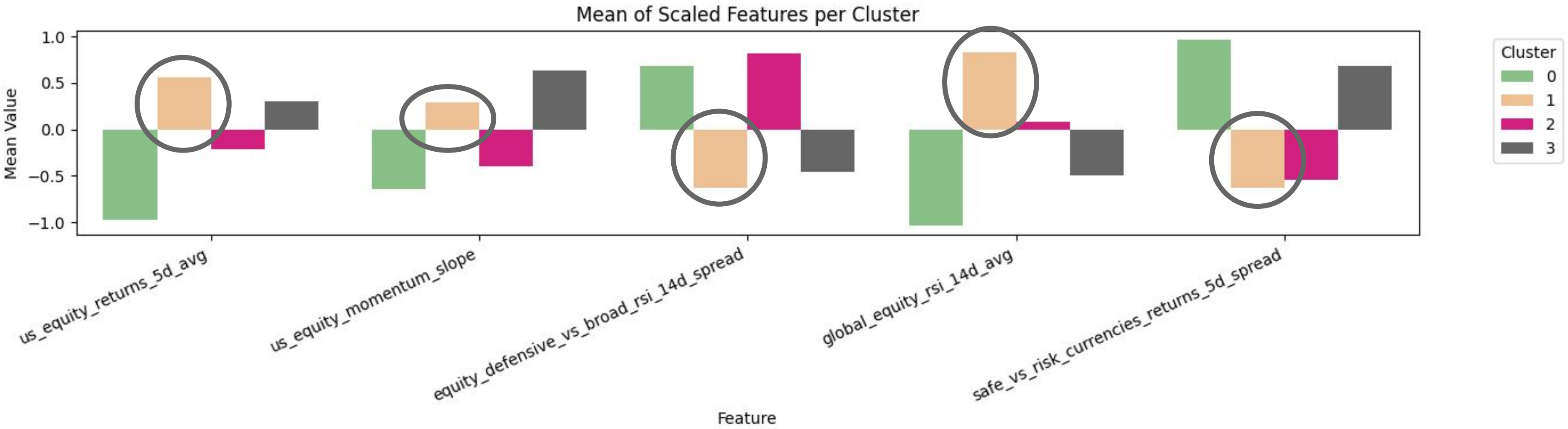
Strength of China, Brazil, Europe markets, etc (Excluding US)

Returns **spread** (difference) between **safer** currencies vs **riskier**

Interpretation

Regime 0 = Bear

Regime 1 = Bull



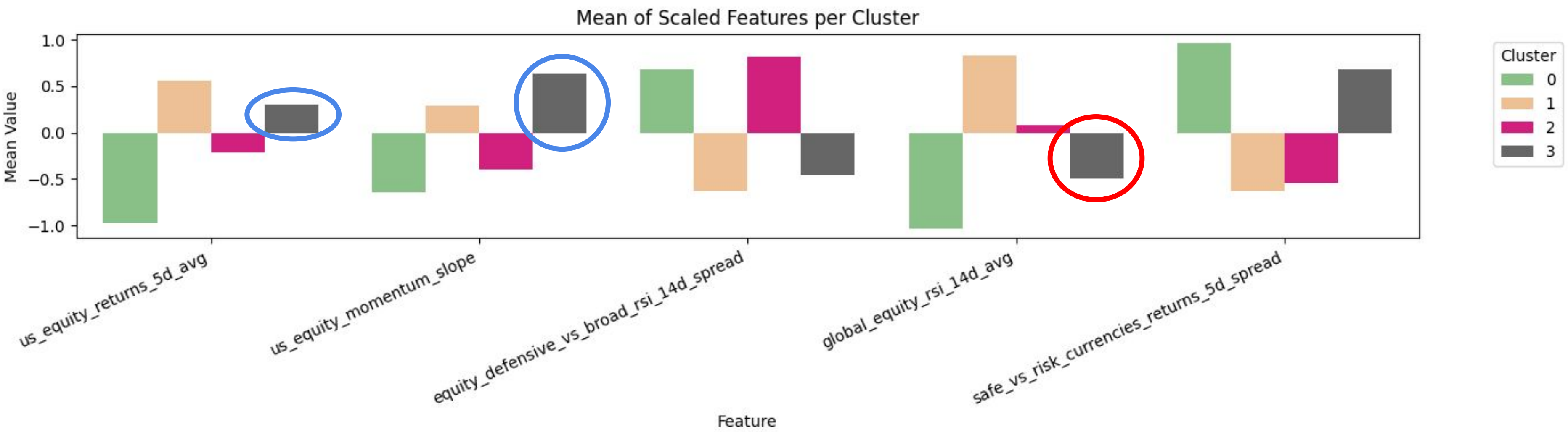
Completely the **opposite** of Bear regime, optimistic, prices up, investors risking money outside defensive assets. That is Bull

Regime 0 = Bear

Interpretation

Regime 3 = U.S. Bull Only

Regime 1 = Bull



Also positive values, sames as Bull but only for U.S. variables

Negative for markets outside U.S.

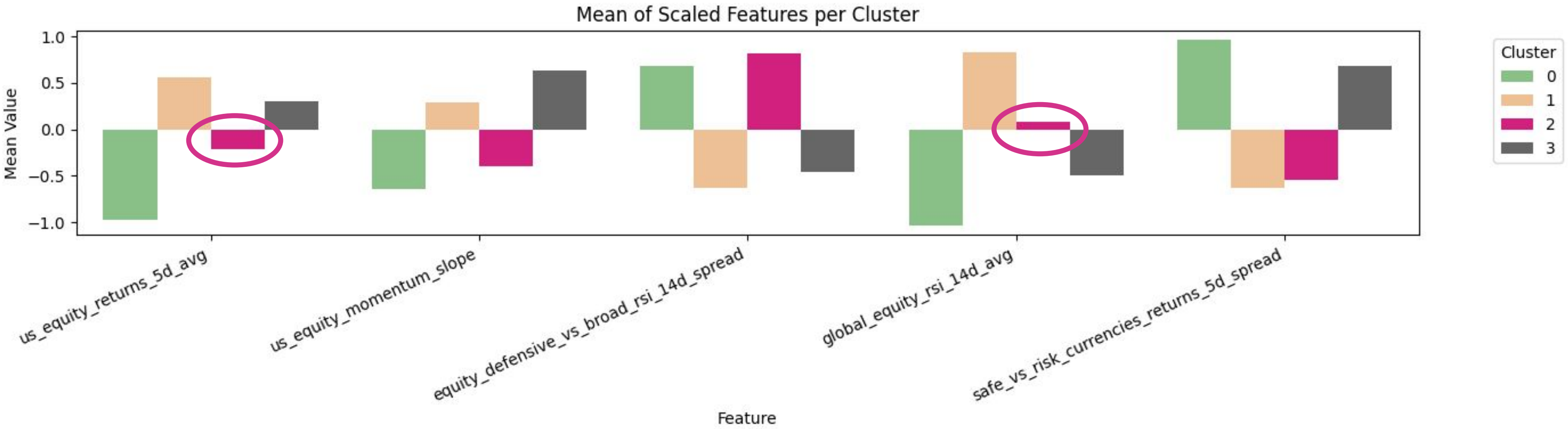
Regime 2 = Neutral

Regime 0 = Bear

Interpretation

Regime 3 = U.S. Bull Only

Regime 1 = Bull



Neutral values along U.S. and Europe, China, Brazil Markets

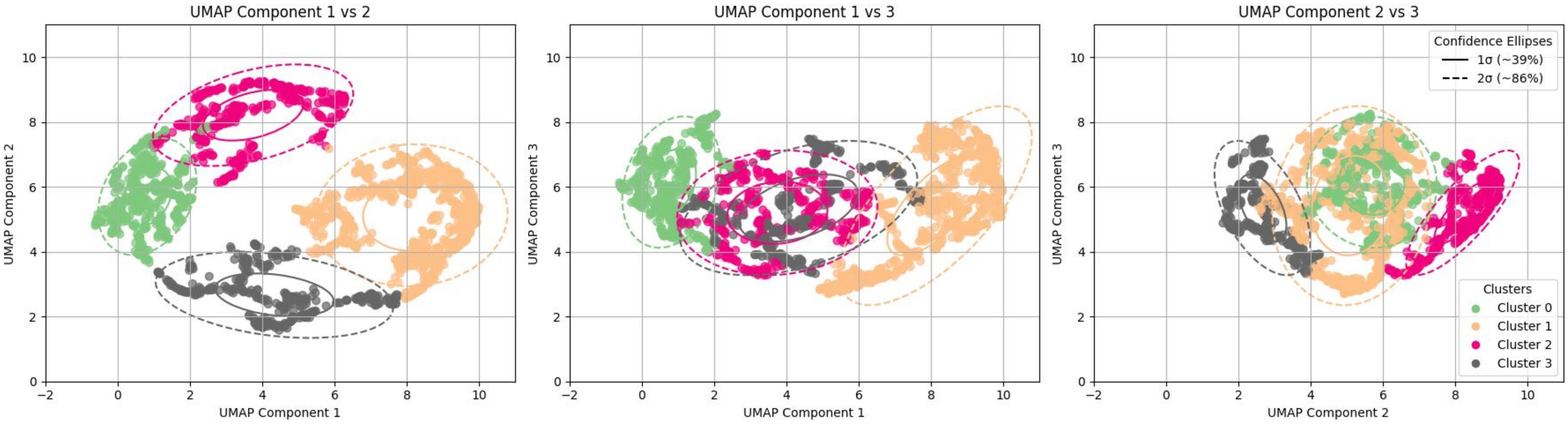
Regime Detection GMM

Regime 2 = Neutral

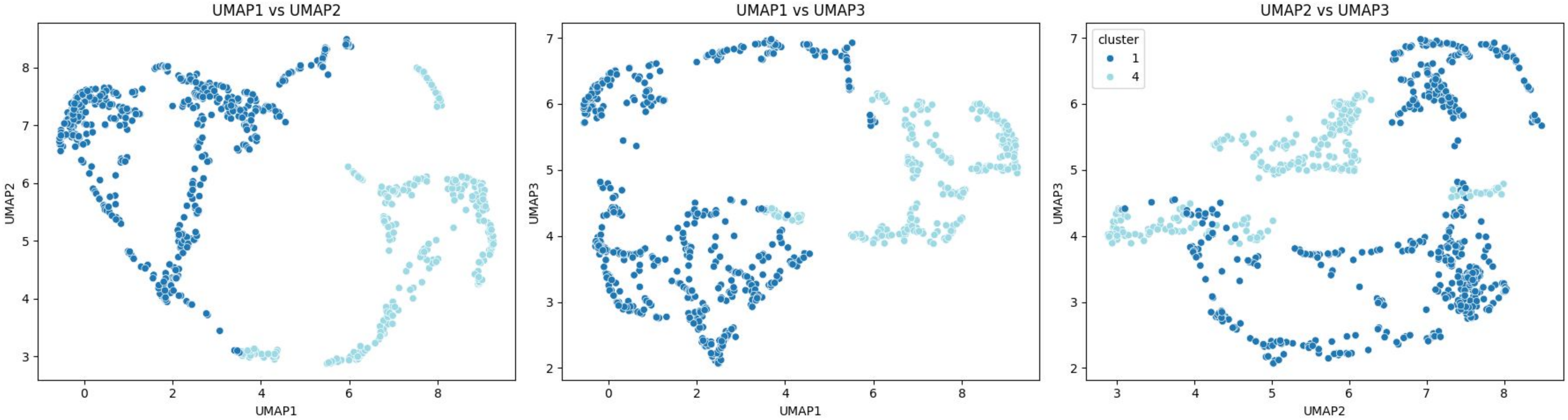
Regime 0 = Bear

Regime 3 = U.S. Bull Only

Regime 1 = Bull



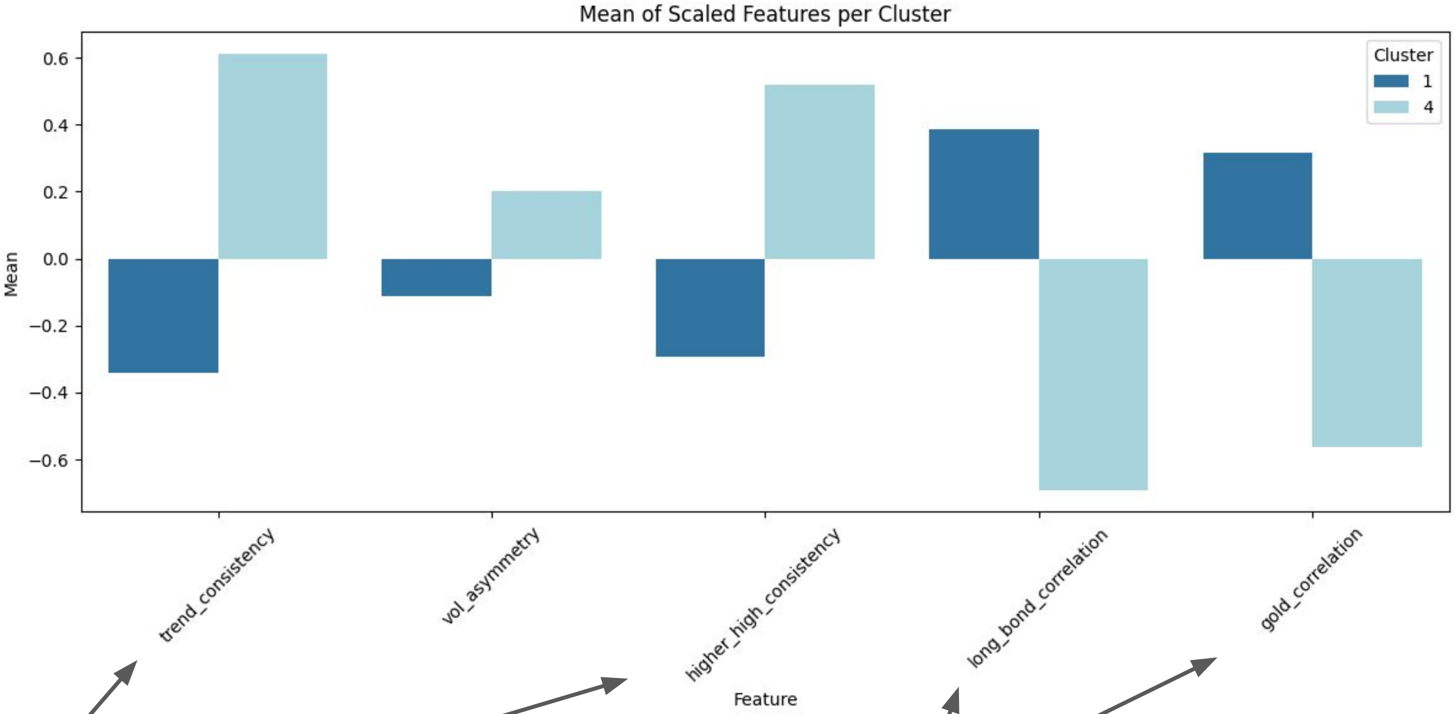
Bull Sub-Regime Spectral Clustering



Interpretation

Regime 1 = Defensive / Consolidating Bull

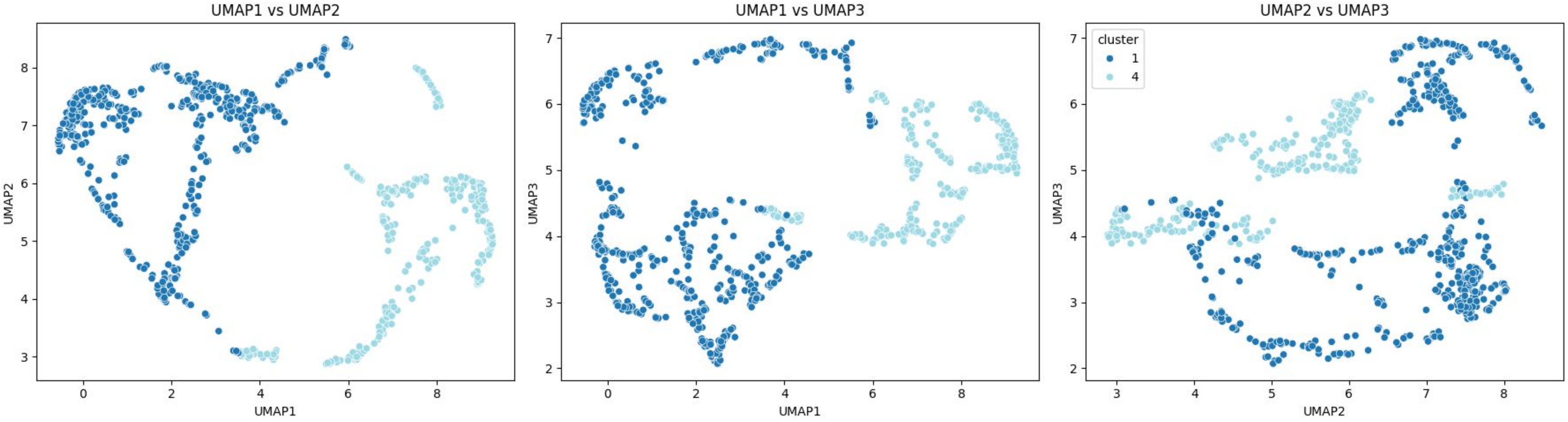
Regime 4 = Strong / Aggressive Bull



Variables that measure the positive **Trend Consistency**

Correlation with **haven/protective** assets like Gold or Bonds. Positive values = investors are still buying those defensive assets at the same time. Negative = investors selling those negative assets

Bull Sub-Regime Spectral Clustering



Historic Validation

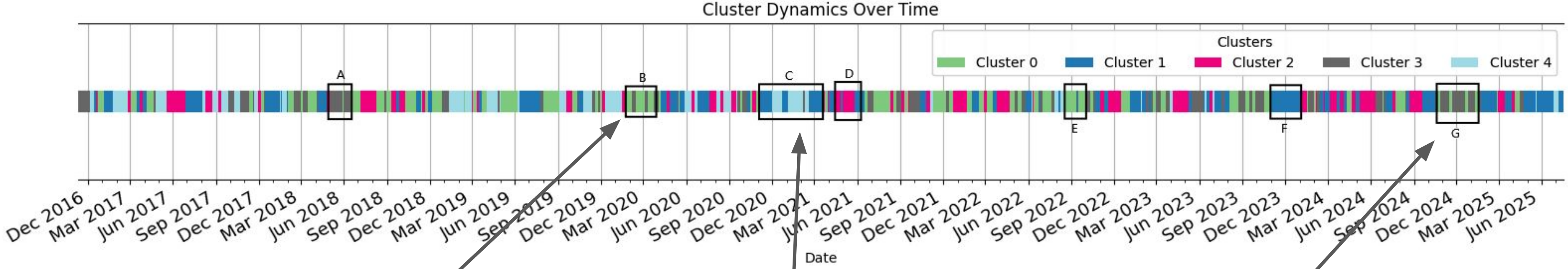
Regime 2 = Neutral

Regime 3 = U.S. Bull Only

Regime 4 = Strong / Aggressive Bull

Regime 0 = Bear

Regime 1 = Defensive / Consolidating Bull



Feb–Mar 2020 – COVID-19 Crash

Nov 2024–Jan 2025 – Trump Election

Nov 2020–Mar 2021 – Post-COVID Reopening

Neural Network Performance

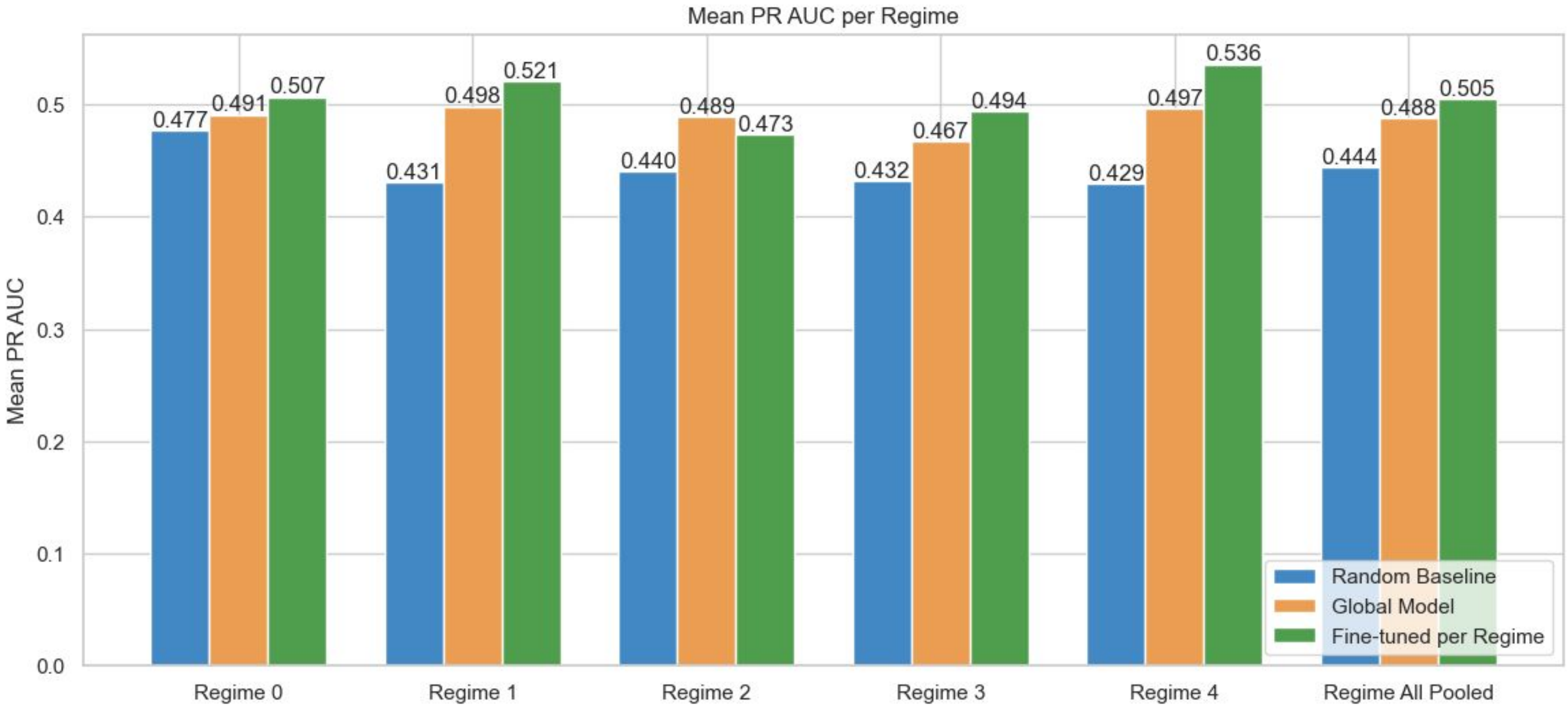
Regime 2 = Neutral

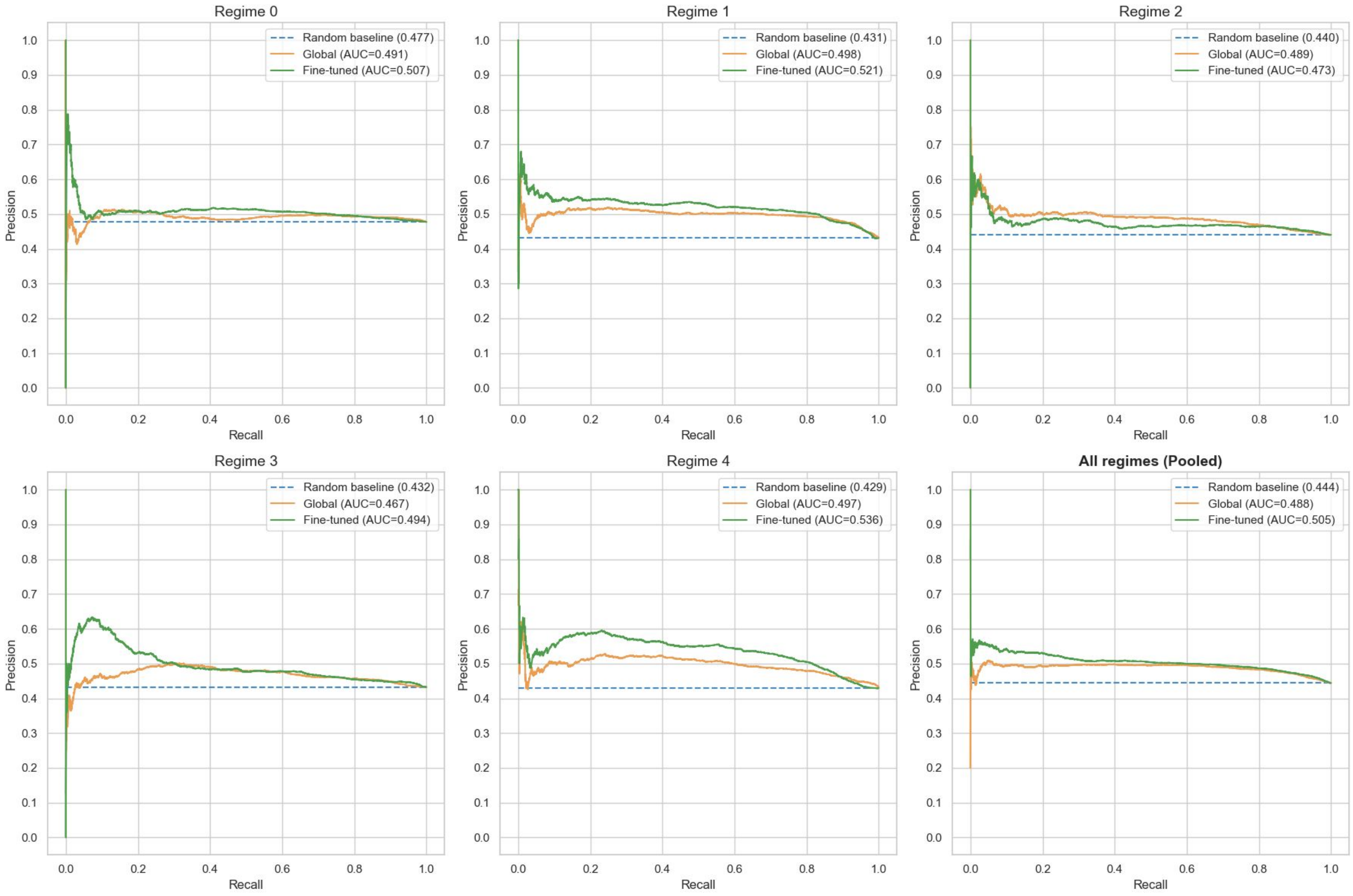
Regime 3 = U.S. Bull Only

Regime 4 = Strong / Aggressive Bull

Regime 0 = Bear

Regime 1 = Defensive / Consolidating Bull





Regime 0 = Bear

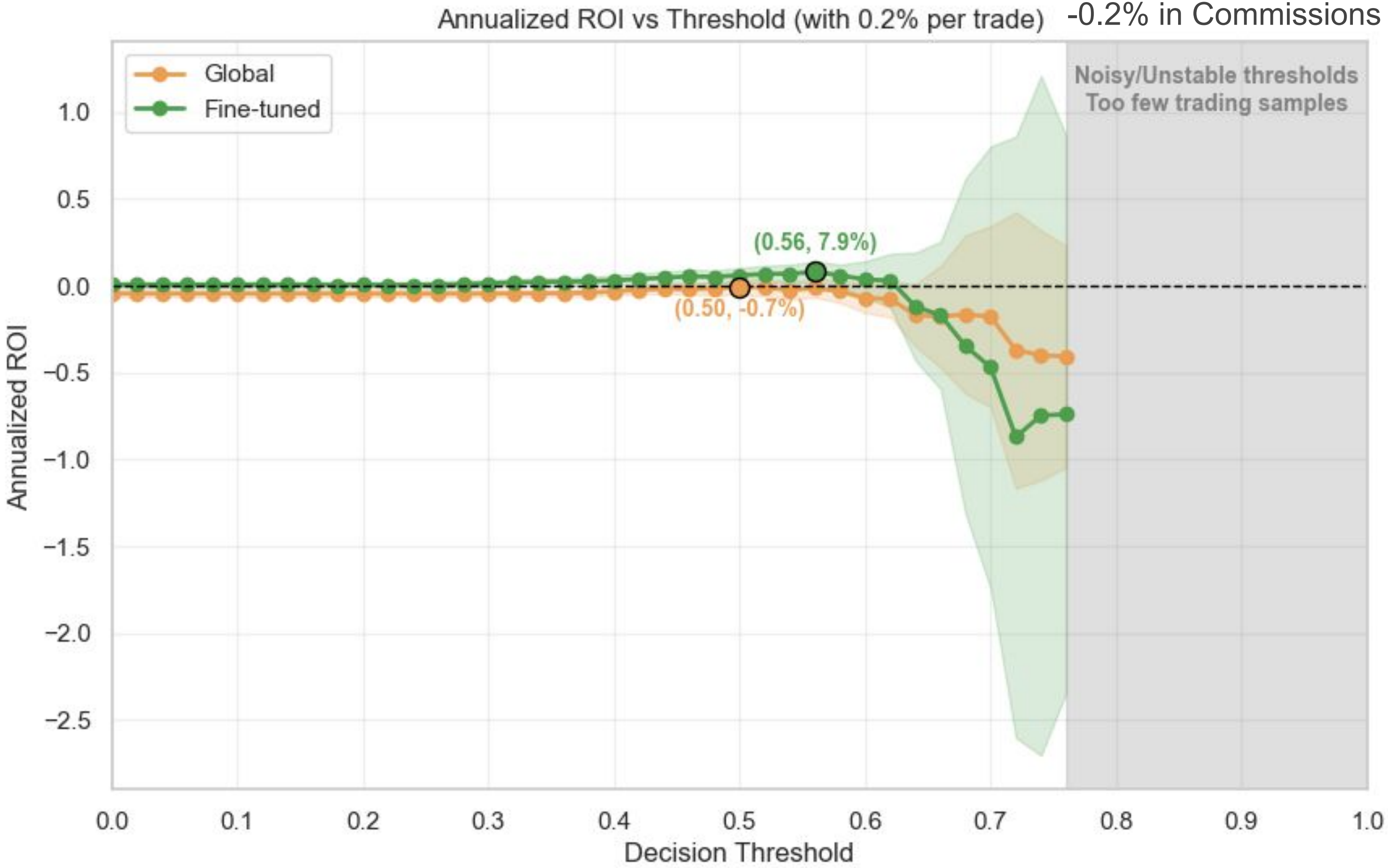
Regime 1 = Defensive / Consolidating Bull

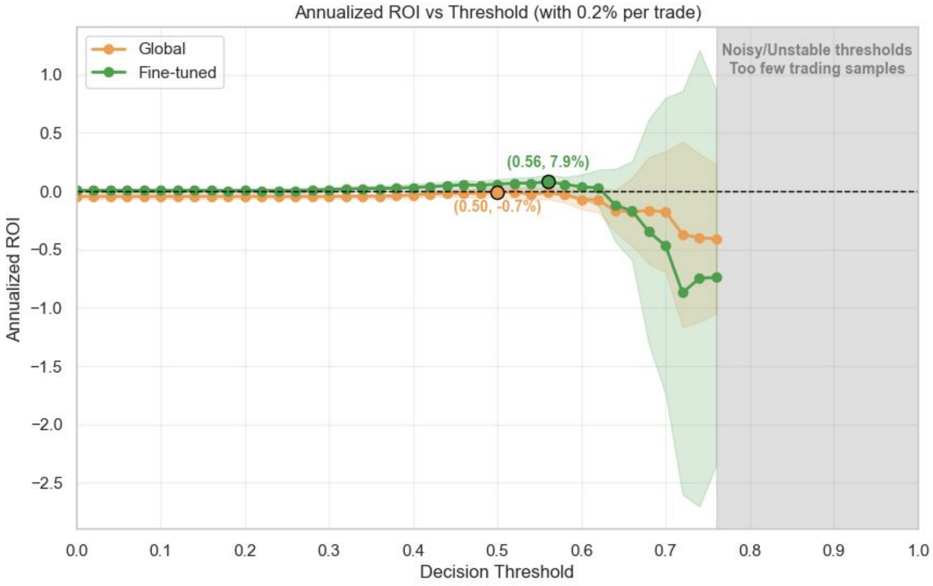
Regime 2 = Neutral

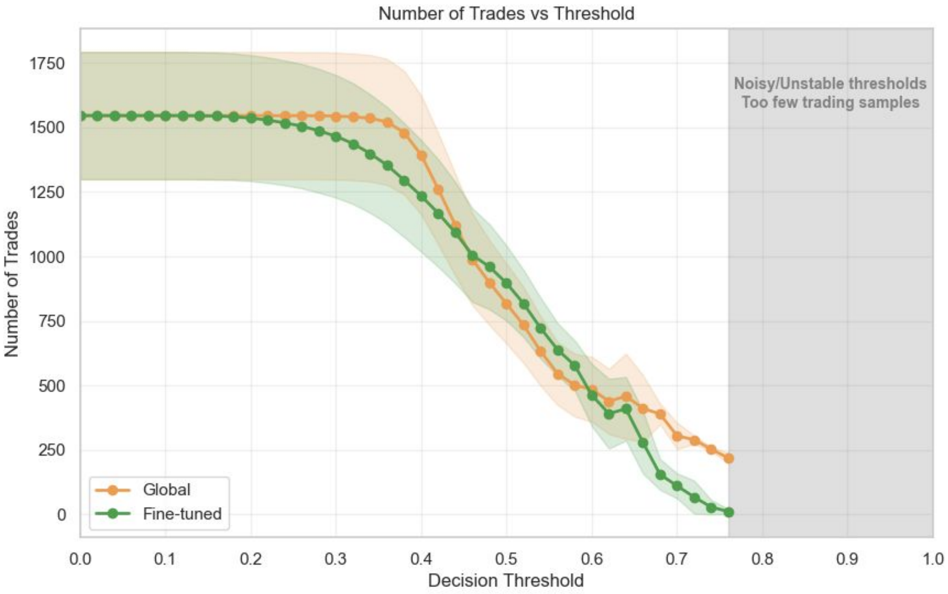
Regime 3 = U.S. Bull Only

Regime 4 = Strong / Aggressive Bull

-0.2% in Commissions

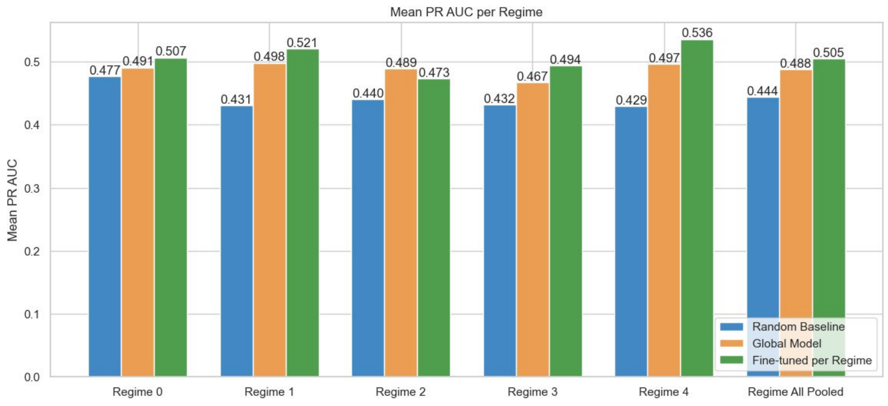


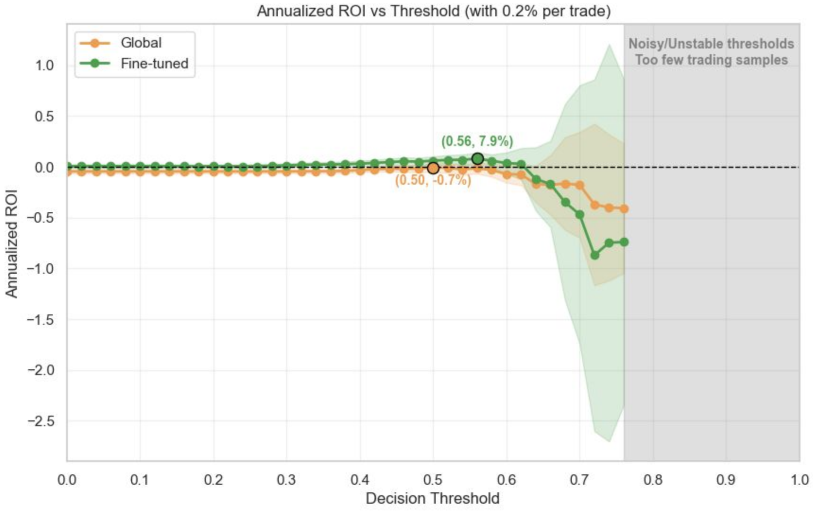

Annualized ROI vs Threshold (with 0.2% per trade)
Global
Fine-tuned
Noisy/Unstable thresholds
Too few trading samples
(0.56, 7.9%)
(0.50, -0.7%)
Annualized ROI
Decision Threshold


Number of Trades vs Threshold
Noisy/Unstable thresholds
Too few trading samples
Global
Fine-tuned
Number of Trades
Decision Threshold

Conclusions

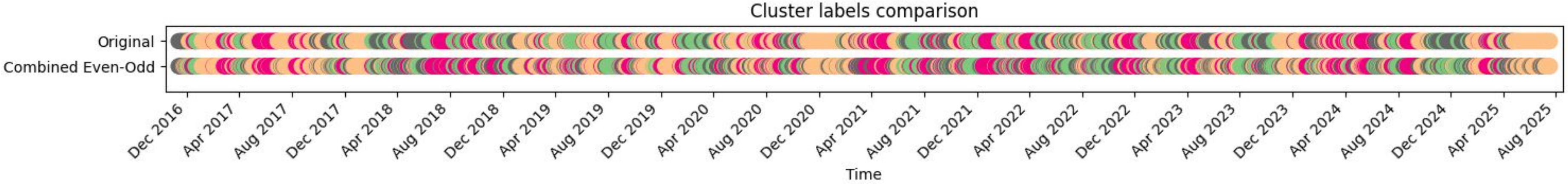
This thesis **proved** that a **regime-aware** framework clearly **improves** the **prediction** of trading strategies' performance. By integrating regime detection models with fine-tuned neural networks delivering more reliable and actionable results under realistic trading conditions.



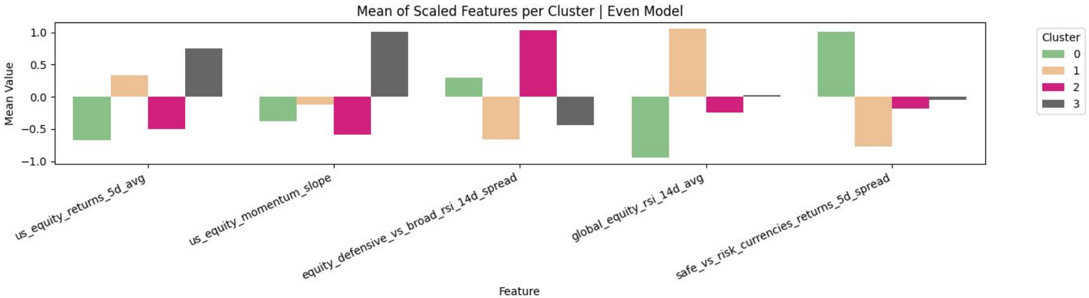


Experiment: UMAP + GMM robustness. Even vs Odd model

70% coincidence



Even days



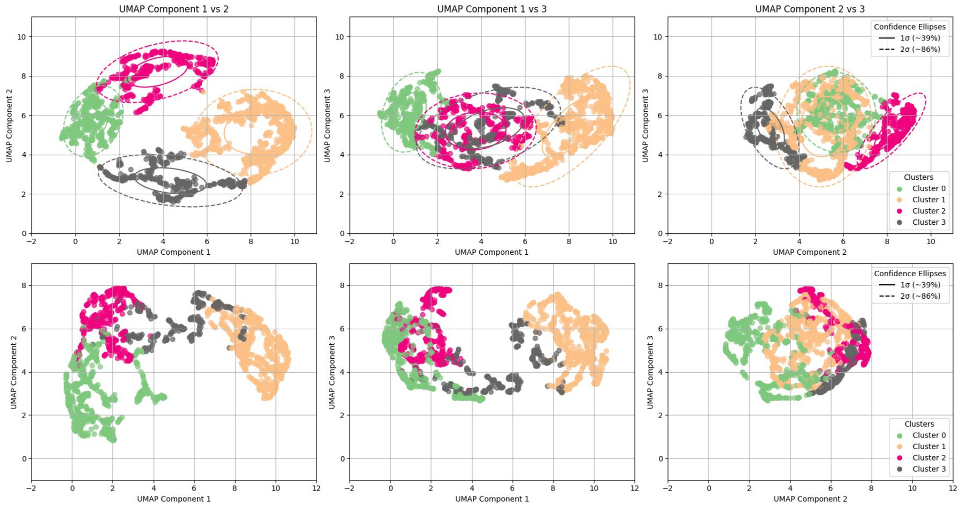
Odd days

HMM vs GMM runs regime detection

Same conditions of environment, data and processing

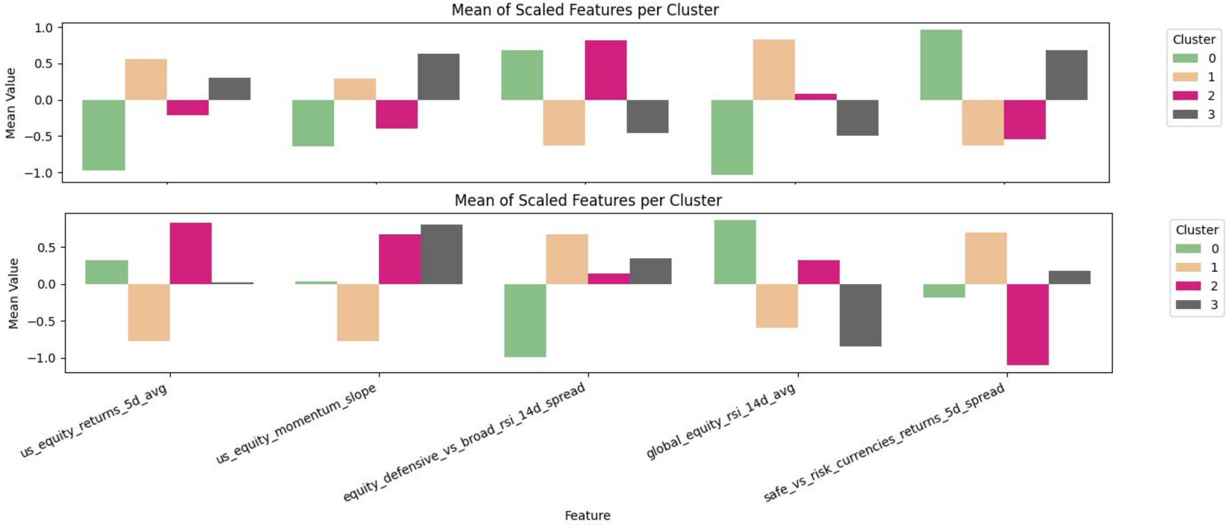
GMM

HMM



GMM

HMM



Spectral vs GMM runs sub-regime

Same conditions of environment, data and processing

GMM

Spectral