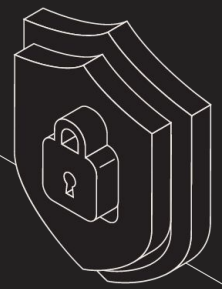
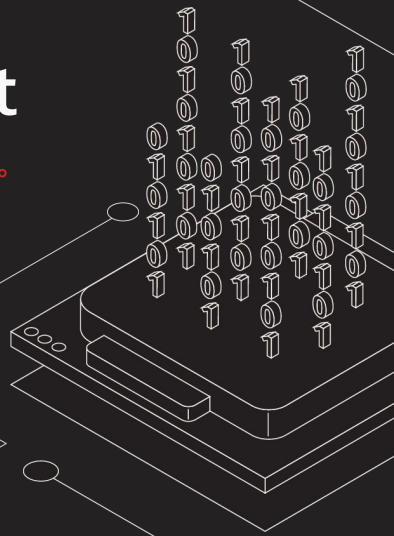
Adaptive Risk Management

for Dynamic Portfolio Management

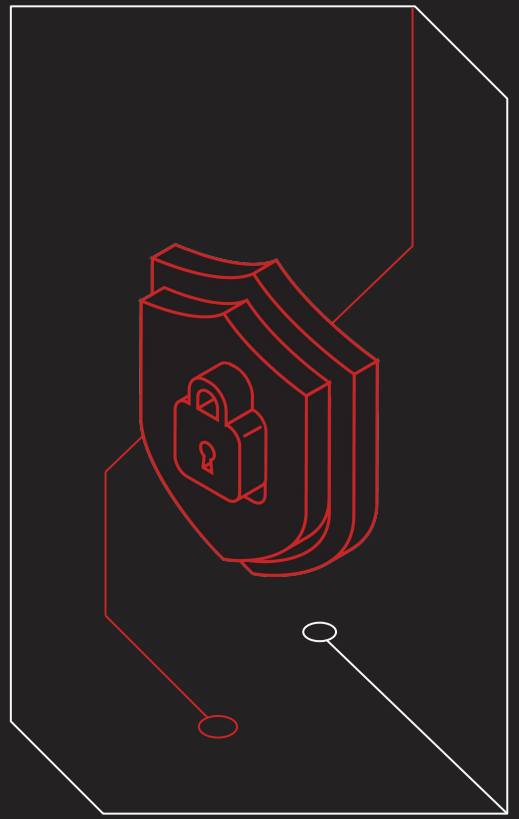
Team (PyCh)ARM





Purpose

- Risk can either hinder or enhance potential rewards.
- Market volatility and changing conditions increase uncertainty and risk.
- Static investment strategies can lead to unnecessary risk and missed opportunities.
- Traditional portfolios often lack the ability to adapt to evolving market conditions.



Goals & Objectives

Regime Switching

Use HMM to identify market regimes (bullish/bearish) based on historical returns.

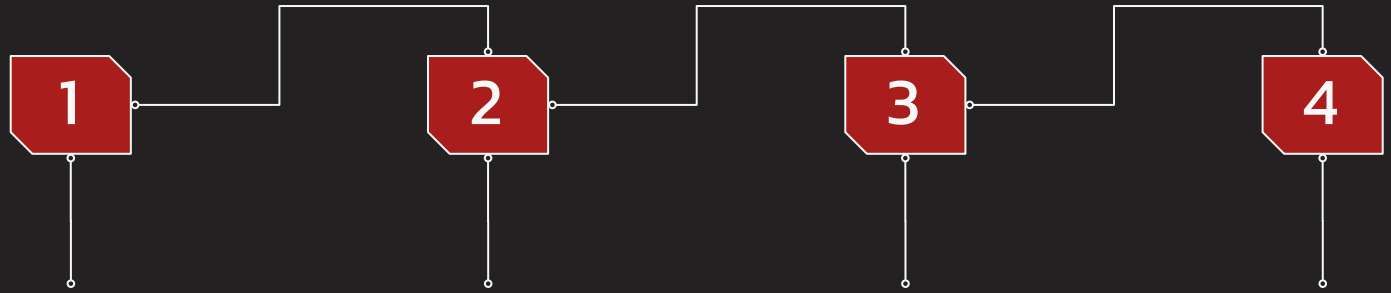
ML Modeling

Train an LSTM model to predict future annualized returns based on historical data sequences.

Markets Trends

Implement technical indicators such as SMA.

Technologies



Data Collection

yfinance

Technical Indicators

Pandas, numpy

ML Modelling

PyTorch, sklearn

Regime Switching

hmmlearn

Market Trends: SMA

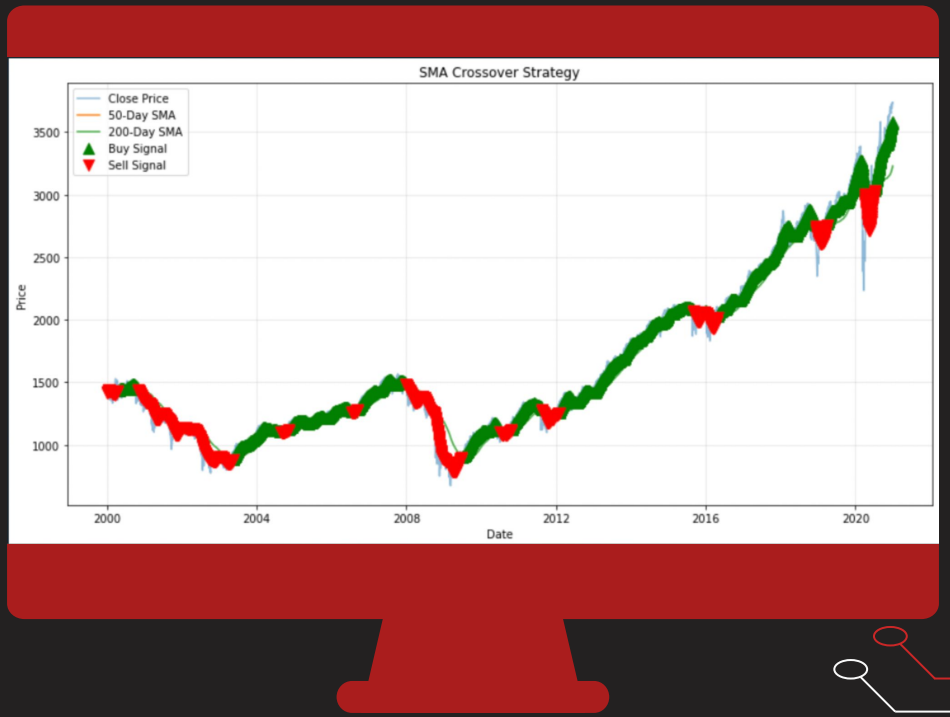
Steps to Create the SMA:

- Fetch data from yfinance (closing prices)
- Calculate averages over both windows
- Create buy or sell signals

SMA Average Calculation:

The short and long-term SMAs are calculated using rolling windows:

- 50-day window for the short-term trend
- 200-day window for the long-term trend



- **Buy** (1): when short-term moving average crosses above long-term moving average (indicating potential buying opportunities -> short-term trend turning bullish)
- **Sell** (-1): when short-term moving average crosses below a long-term moving average (indicating potential selling opportunities -> short-term trend turning bearish)

ML Modeling: LSTM

Steps to Create the LSTM:

- Fetch historical stock data
- Preprocess and calculate returns
- Train an LSTM model on the data
- Generate predictions and signals
- Evaluate performance with metrics and visualizations

Model Structure:

- 2 LSTM layers with 60 hidden units
- Dropout layer for regularization
- Fully connected layer for output

Framework: PyTorch
Input: 120 days of annualized returns
Output: Predicted return for the next day

Buy (1): Predicted return > Previous actual
Sell (−1): Predicted return <= Previous actual

Regime Switching: HMM

Steps to Create the HMM:

- Fetch historical stock data
- Calculate & scale Log returns, volatility, momentum
- Train an HMM on the preprocessed data
- Predict market regimes (bull & bear)
- Visualize regimes & evaluate predictions

Model Structure:

- Log returns, volatility, momentum
- 2 Hidden States (Bull & Bear)
- Covariance Type: Full covariance matrix

Framework: hmmlearn
Input: Historical stock data with features
Output: Predicted market regimes

Bull Market (State 0): Positive mean return
Bear Market (State 1): Negative mean return

Results (Combined/Integrated Model)

^GSPC - Mode 1 (Low Volatility)

Strategy Metrics:
Total Return: 174.49
Annual Return: 4.70
Annual Volatility: 5.01
Sharpe Ratio: 0.94
Max Drawdown: -9.92

PG - Mode 2

Strategy Metrics:
Total Return: 155.62
Annual Return: 4.37
Annual Volatility: 5.92
Sharpe Ratio: 0.74
Max Drawdown: -11.83

NVDA - Mode 3

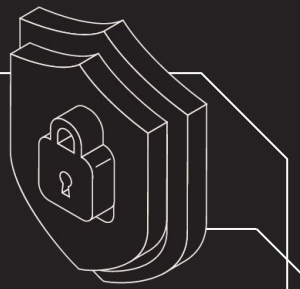
Strategy Metrics:
Total Return: 35222.21
Annual Return: 30.71
Annual Volatility: 28.32
Sharpe Ratio: 1.08
Max Drawdown: -45.61

Future Improvements

Technical Indicators:
1) Add more complex technical indicators for diverse inputs
Regime Switching:
1) Using more than 2 regimes & a larger hidden size to capture more complex patterns
2) Training on a holistic set of diverse stocks to encompass all market regimes instead of using past data of a stock
ML Modelling:
1) Add more features (like trading volume) for the LSTM
2) Experiment with different model architectures or try model ensembling
3) More hyperparameter tuning (ex. number of layers)

Combined:

1) Neural Network for optimizing weights for each model
2) Automatic mode switching based on the stock
3) Ability to work with multiple stocks at once



Thank you!

Questions?