UM Hackathon 2025

UNIVERSITI TENAGA NASIONAL
The Energy University

SolidTech

Alpha Strategies Using HMM or ML

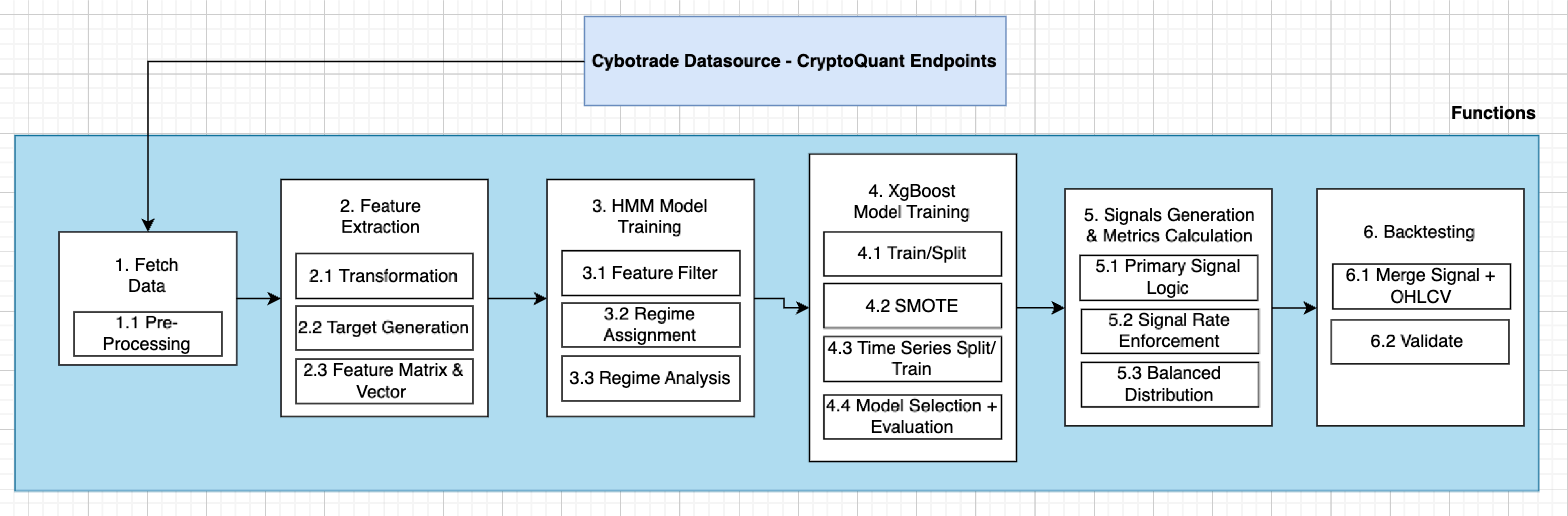
Speaker
Shawn Garcia

Date
13 April 2025

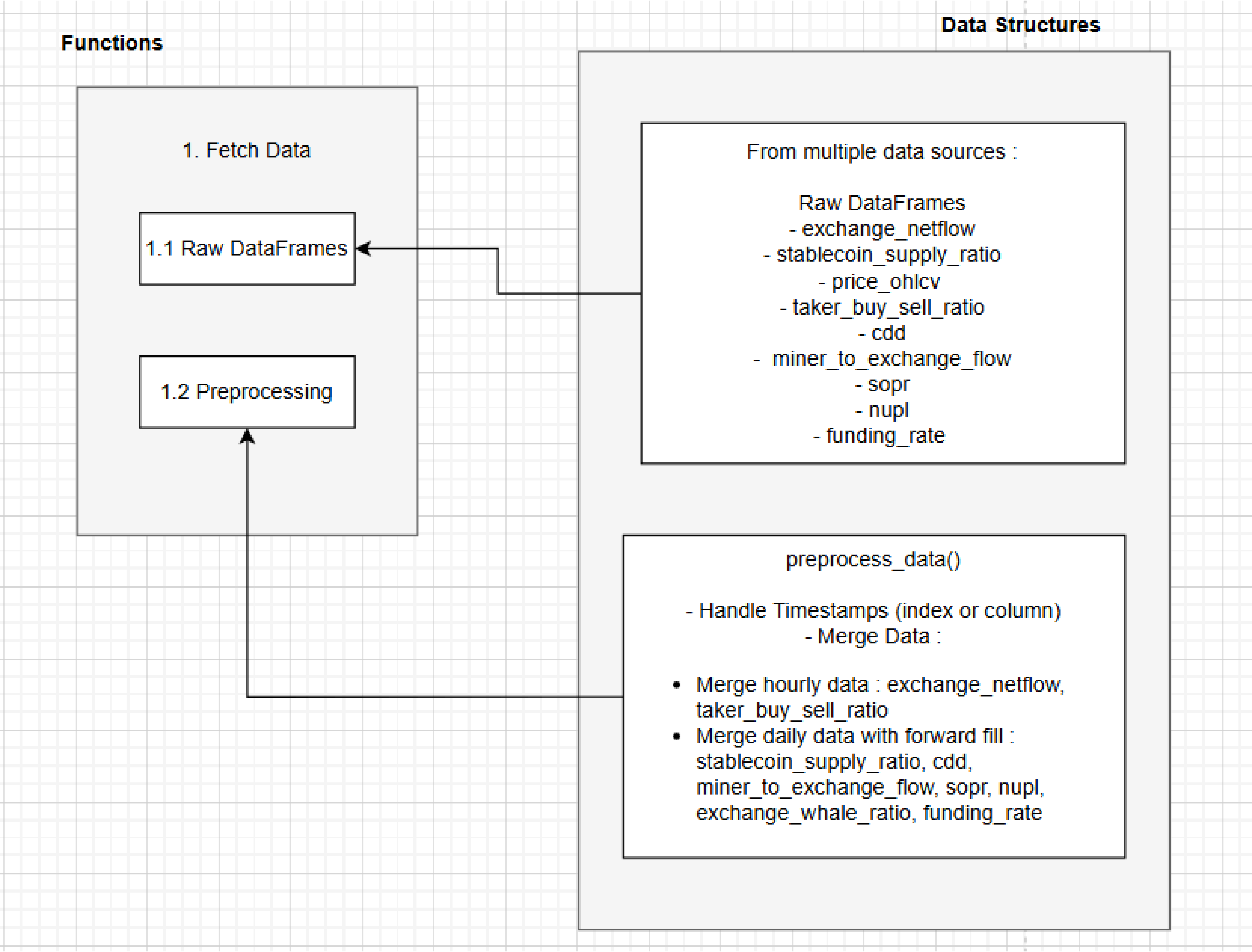
Problem Statement

Develop a Machine Learning (ML) model that analyzes on-chain data from various sources (e.g., CryptoQuant, Glassnode, Coinglass) to generate an alpha trading strategy that maximizes profit.

Architectural Overview of the ML-based Alpha Trading Strategy



Fetch Data & Pre Processing



Fetch Data & Pre Processing

Data Fetching

Data Sources Configuration:

- Defines data points with customizable parameters (time window, exchange type).
- Supports hourly or daily data fetching.

Fetching Mechanism:

- **Asynchronous Fetching:** Ensures multiple datasets are fetched in parallel for efficiency.

Data Output:

- Stored in a dictionary of Pandas DataFrames.
- **Filtering:** Some sources are excluded if incompatible (ex: mvrv for hourly data).

Pre Processing

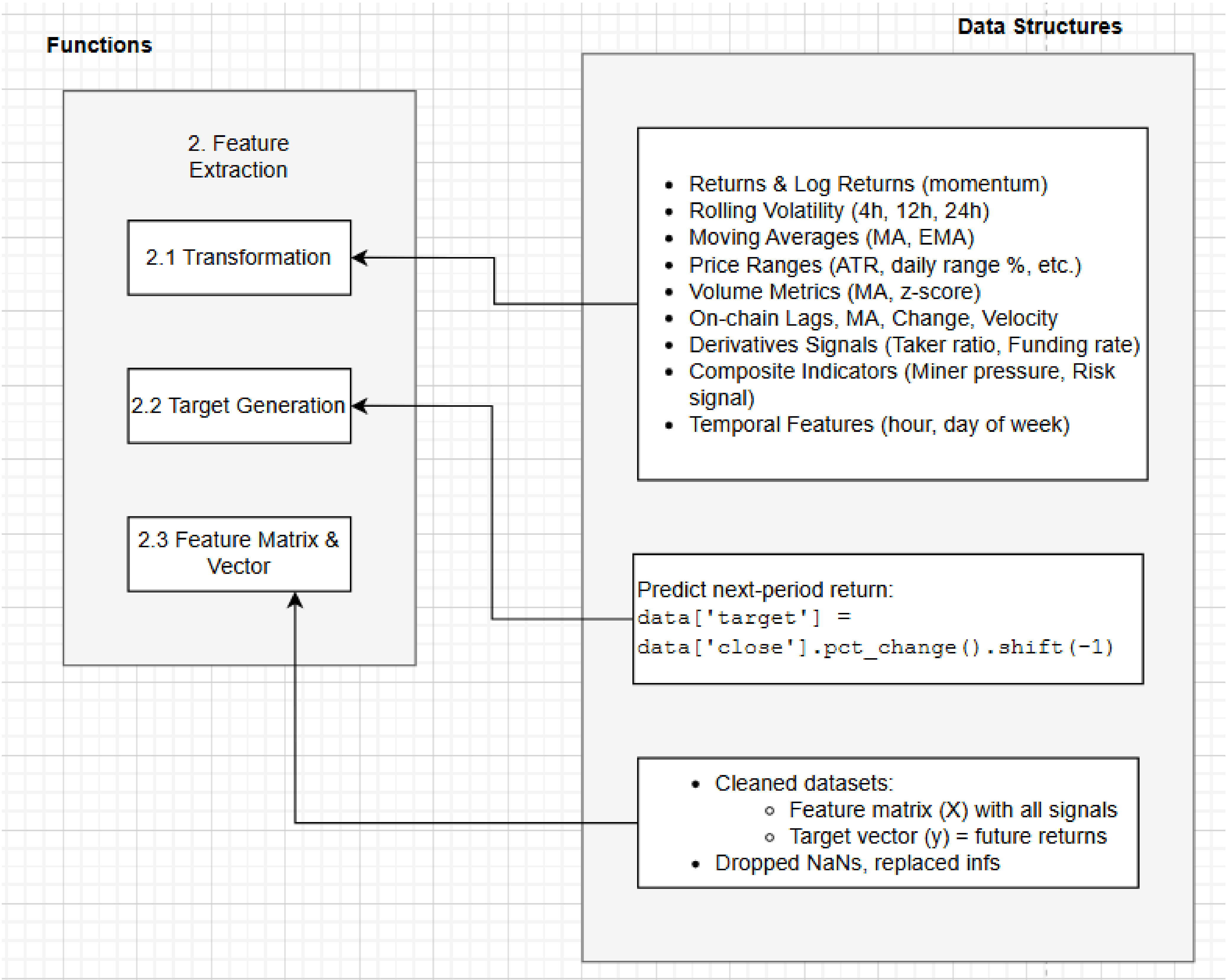
Data Cleaning:

- Checks any data frames that are empty or none & skips them.
- Standardized timestamps.
- Rename the column to match the source name.

Data Merging:

- **Hourly data sources** (e.g., exchange_netflow, taker_buy_sell_ratio) are merged directly into the primary dataset (price_ohlcv).
- **Daily data sources** (e.g., stablecoin_supply_ratio, sopr): **Forward filling** is applied to carry the last known value across the rest of the day to ensure there are no missing data points for those intervals.

Features Engineering



Features Engineering

1

Price-Based Features

- **Returns:** *returns = close.pct_change()*
  - → Captures price momentum.
- **Log Returns:** *log_returns = log(close).diff()*
  - → Stabilizes variance for modeling.
- **Rolling Volatility:** *rolling(std of returns)*
  - → Measures recent price fluctuation over 4h, 12h, 24h.
- **Moving Averages (MA, EMA):** *rolling(mean of close)* and *exponential moving average*
  - → Identifies short- and medium-term trends.
- **Price Range:** *ATR = high.max - low.min, range_pct = (high - low) / close*
  - → Detects volatility spikes and intraday dynamics.

2

Volume & Activity-Based Features

Compute liquidity and interest signals:

- *volume_ma = volume.rolling(10).mean()*
- *volume_zscore = (volume - mean) / std*
  - → These indicate volume surges or anomalies.

3

On-Chain Metrics Transformation

For each on-chain metric (like netflow, sopr, nupl, etc.):

- **Lagged Values:** *feature_lagN = feature.shift(N)*
  - → Captures delayed reactions in investor behavior.
- **Moving Averages:** *feature_maN = feature.rolling(N).mean()*
  - → Smooths out noise in indicators.
- **Changes & Velocity:** *feature.diff(N) and feature.diff / std*
  - → Measures how fast the signal is changing.

Features Engineering

4

Derivatives Market Signals

For *taker_buy_sell_ratio*, we compute:

- **Short-Term Sentiment Smoothing:** *rolling mean, z-score,* and extreme condition flags
  - → Helps detect aggressive buying or selling pressure.
- **Funding Rate:** *deviation from mean*
  - → Measures trader bias in perpetual futures.

5

Composite Features (Cross-Interacted Signals)

- *netflow_stablecoin_ratio = netflow / stablecoin_ratio*
- *miner_whale_pressure = miner_to_exchange_flow * exchange_whale_ratio*
- *risk_signal = (nupl & sopr conditions)*
  - → These synthesize multiple signals into strategic alpha indicators.

6

Target Generation (Formulation for Supervised Learning)

- *data['target'] = data['close'].pct_change().shift(-1)*
  - → You predict the next-hour return, aligning features with future labels.

Assumption & Hypothesis

1

Buy Signal (📈🟢 Bullish Regime):

Hypothesis: If the **stablecoin ratio is decreasing (💰↓)**, and technical indicators (like a rising RSI out of oversold levels or a bullish MACD crossover) confirm momentum, this suggests that **investors are moving away from safety toward growth**—potentially signaling an emerging **bull market**.

Action: Consider **buying.**

2

Sell Signal (📉🔴 Bearish Regime):

Hypothesis: As mentioned earlier, **if the stablecoin ratio is rising sharply (💰 ↑)** indicating higher "stablecoin velocity," investors might be **shifting to cash.** Combined with bearish technical signals (such as declining MACD or RSI in overbought territory), this could mean the market is **entering or already in a bear mode.**

Action: Consider **selling.**

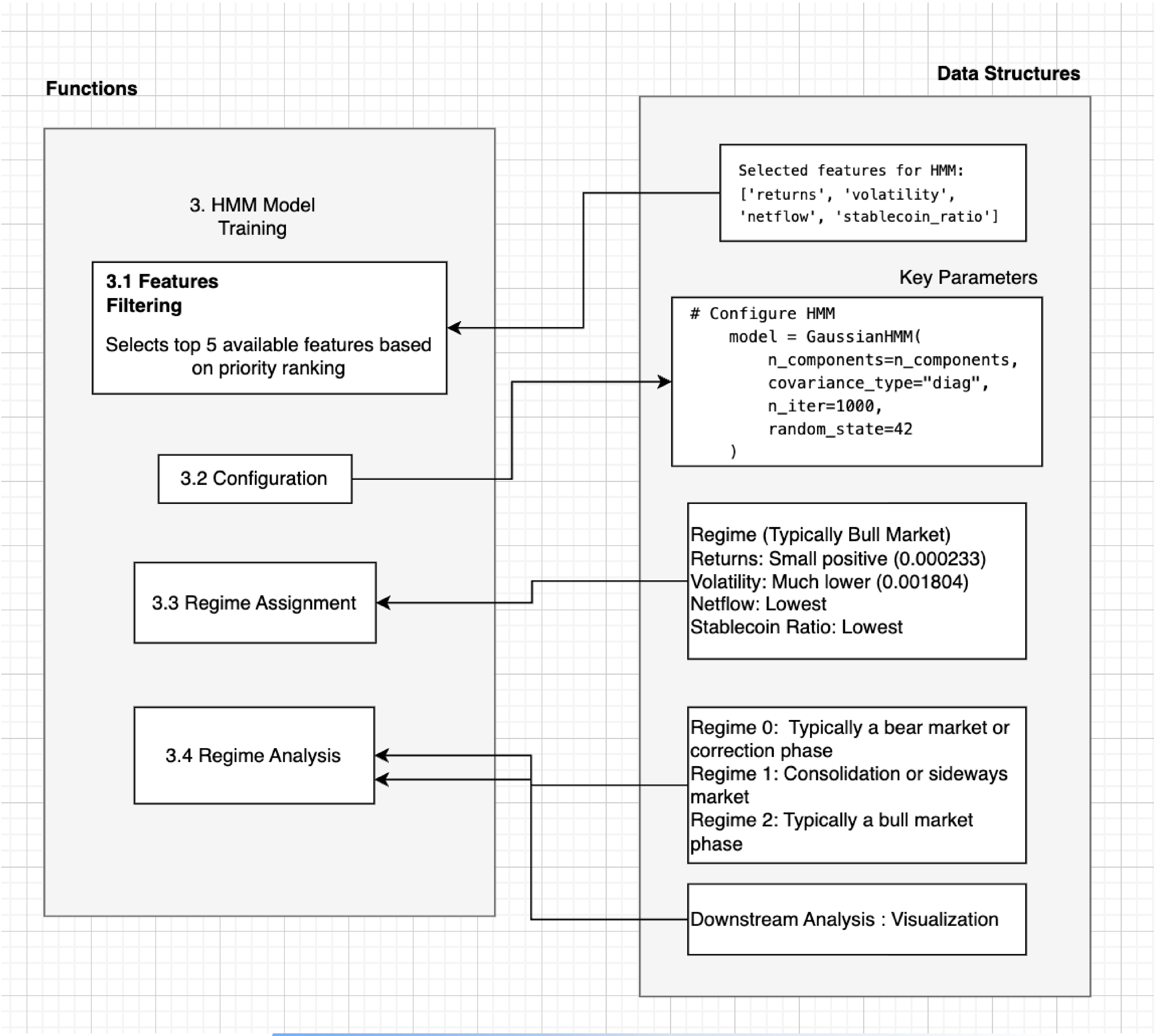
3

Hold Signal (🤔⏸️ Neutral Regime):

Hypothesis: When indicators are **mixed or neutral**—for example, if the stablecoin ratio is relatively stable and technical indicators show **no clear trend**—the market might **not be committing to a bullish or bearish regime**.

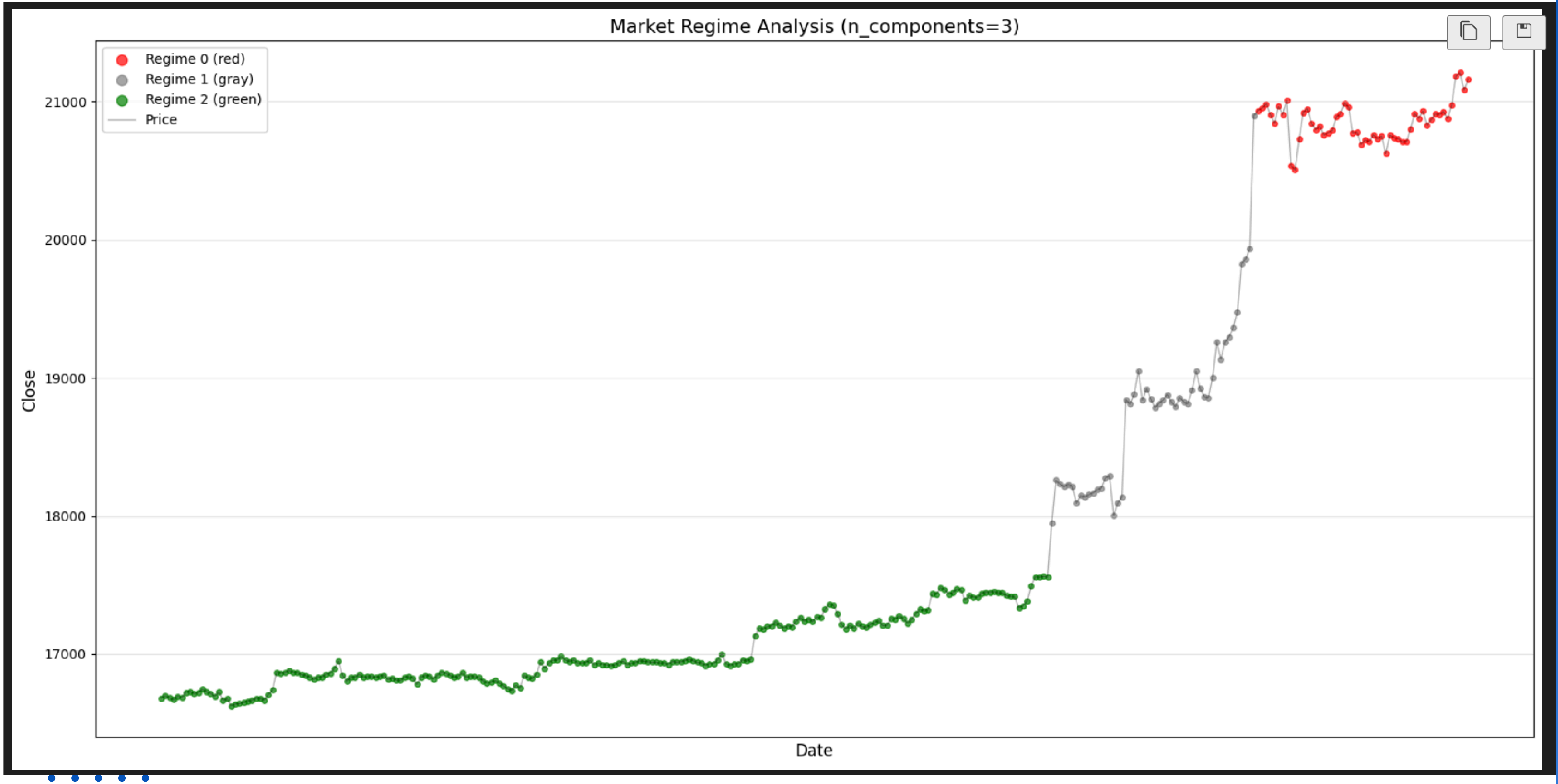
Action: Hold your position until **clearer signals** emerge.

HMM Modelling - Identify Potential Market Regimes



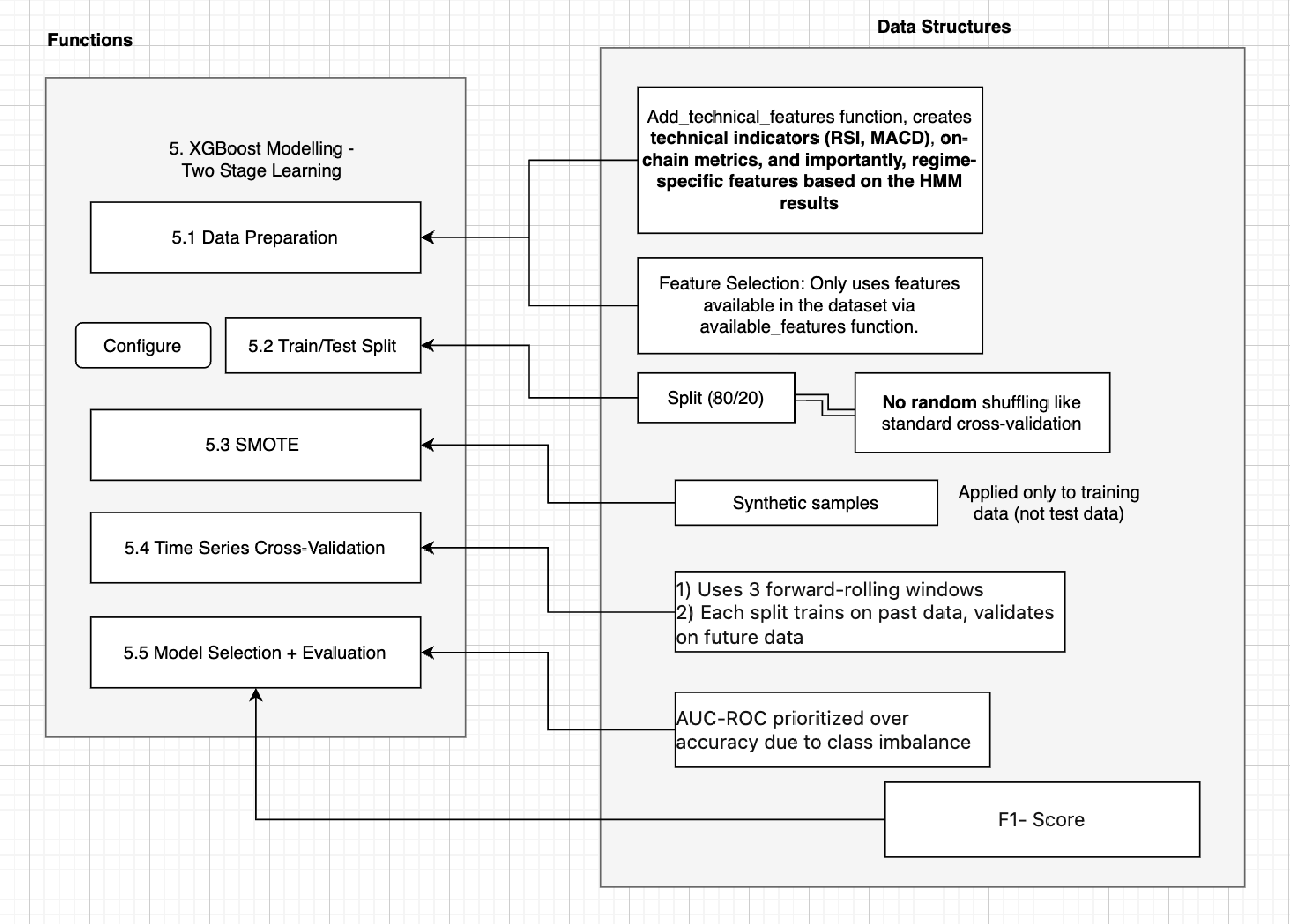
HMM Modelling - Identify Potential Market Regimes

Downstream Visualization



TWO- STAGE LEARNING

XGBoost - Predict Future Price Movement



=== Feature Engineering ===
Selected features: ['returns', 'volatility', 'rsi_14', 'macd', 'macd_hist', 'netflow_ma_ratio', 'stablecoin_velocity', 'regime', 'returns_regime_0', 'returns_regime_1', 'returns_regime_2']

=== Model Training ===

=== Model Performance ===
Best Validation AUC: 0.5962
Test AUC: 0.5562

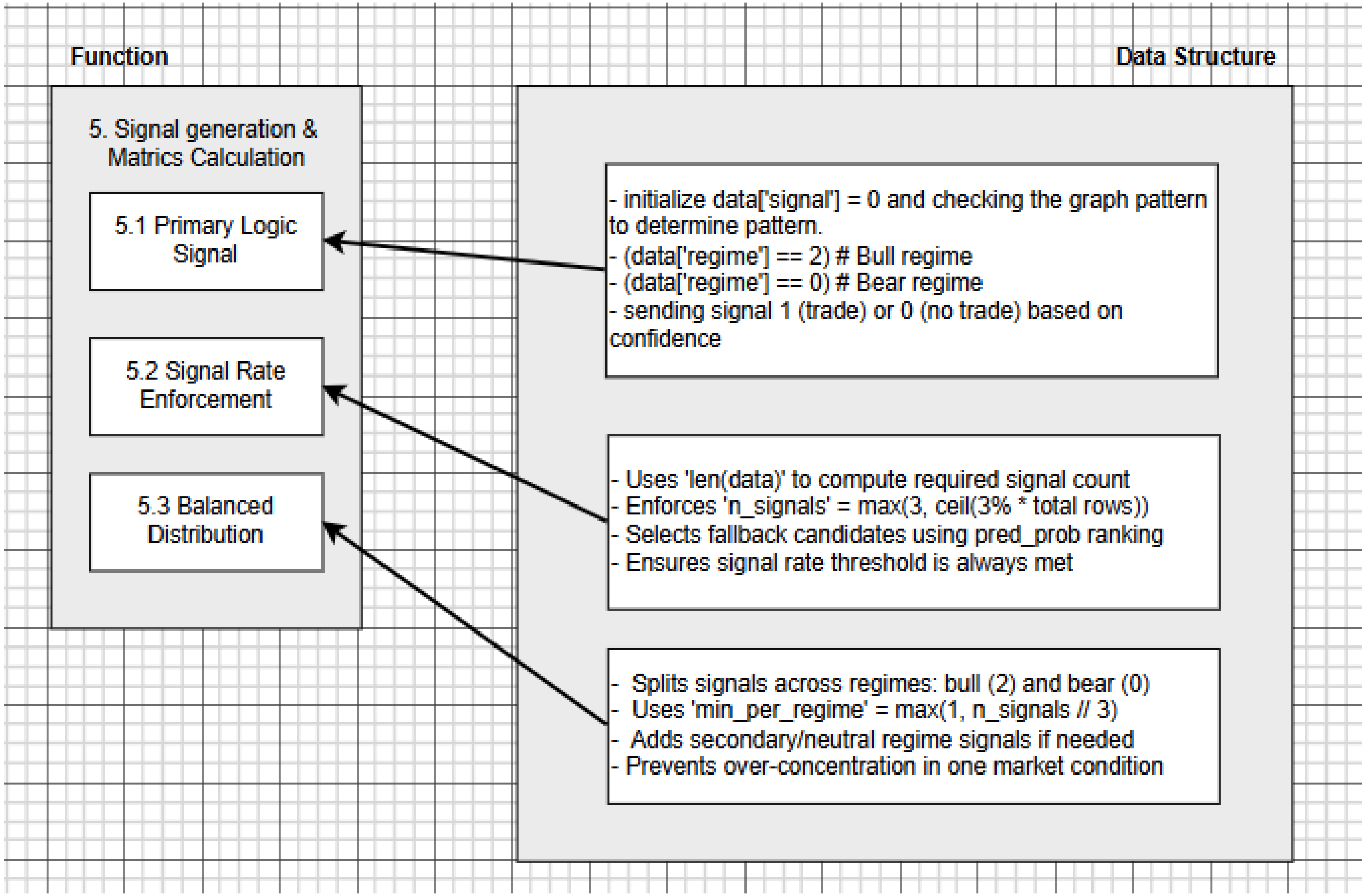
Classification Report:

	precision	recall	f1-score	support
0	0.43	0.87	0.57	23
1	0.75	0.25	0.38	36
accuracy			0.49	59
macro avg	0.59	0.56	0.47	59
weighted avg	0.62	0.49	0.45	59

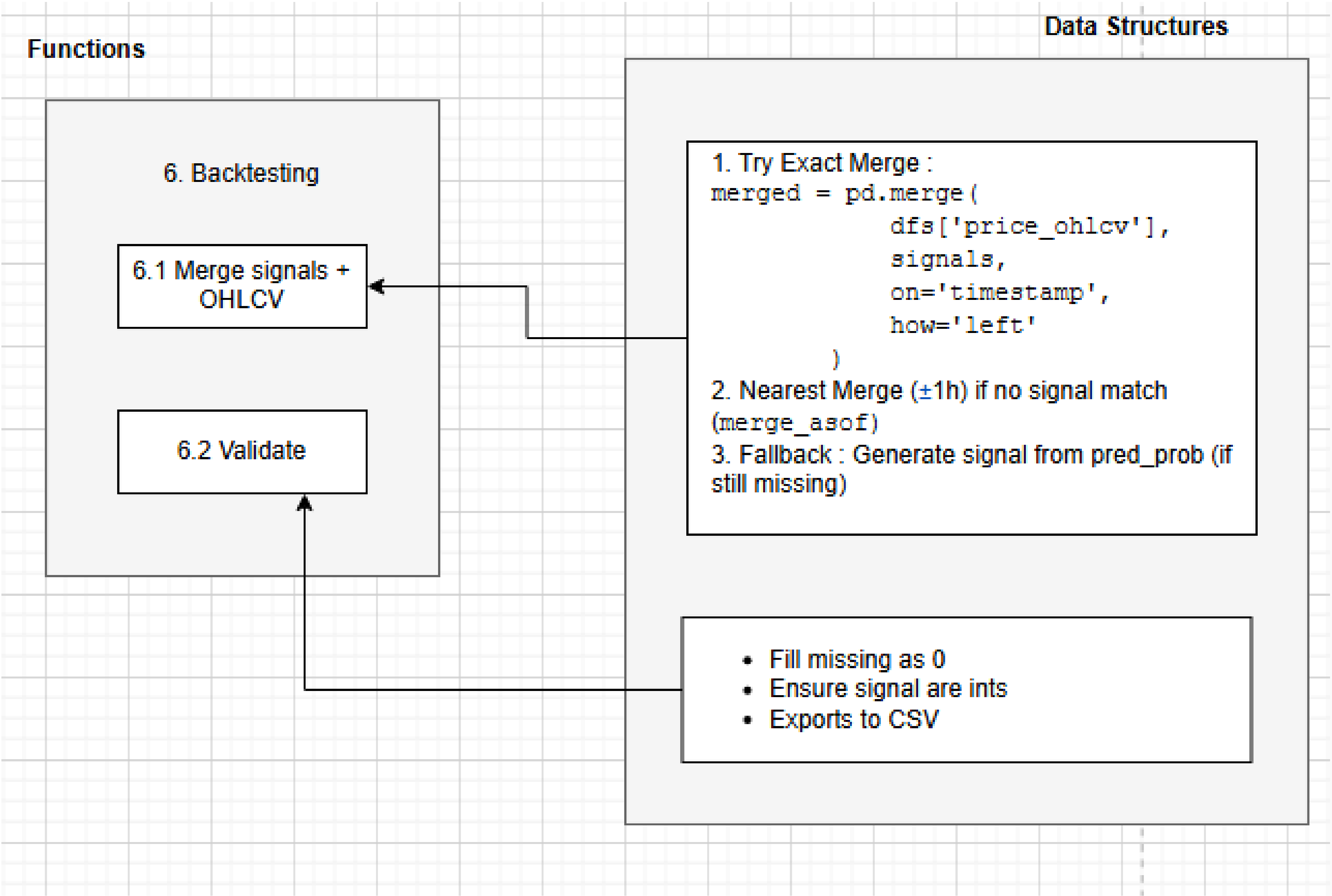
Feature Importance:

	Feature	Importance
3	macd	0.181372
1	volatility	0.172746
...		
7	regime	0.000000
8	returns_regime_0	0.000000

Metrices & Signal Generation



Backtesting



Backtesting

1 **Data Synchronization**

- Convert and standardize timestamps for both signal data and price data.
- Ensure alignment across datasets — even if the timestamps don't match exactly.

2 **Signal Matching Logic**

- **Step 1:** Try an exact timestamp match for each signal.
- **Step 2 (fallback):** If no match, use *merge_asof* to find the nearest price data within ±1 hour.
- **Step 3 (last resort):** If still no signals are matched, generate signals directly from prediction probabilities (e.g., *pred_prob ≥ 0.5*).

3 **Save Backtest-Ready Dataset**

- **Format includes:**
  - *timestamp, open, high, low, close, signal, regime, pred_prob*
- Export to CSV for analysis or use in evaluation tools like backtesting engines.

SolidTech

Thank You

For your attention to this presentation.