



# UM Hackathon 2025

## Team Error 404

- Members:**
- Lim Beng Rhui
  - Ng Xuan Jack
  - Chan Chun Ming
  - Eu Jun Hong

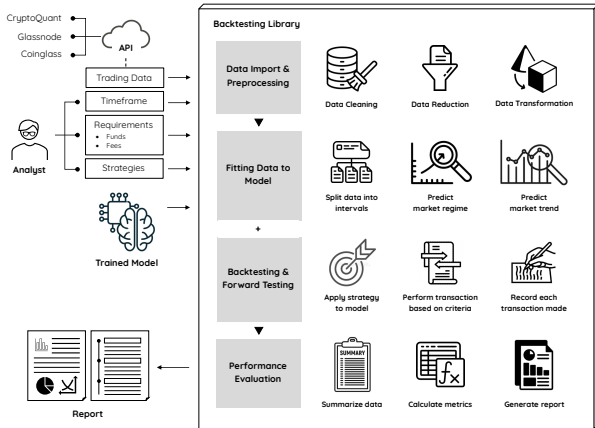
# Conceptual Diagram

Three components:

- Input
- Backtest library
- Output

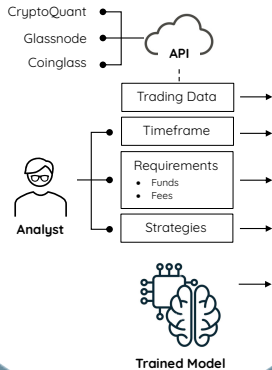
Additional library:

- `pandas`
- `backtrader`
- `scikit-learn`



# Input

- Trading data
- Timeframe
- Requirements (e.g. funds and fees)
- Strategies
- Trained model



# Trained Model

Three components:

- Identify and label market regime
- Predict future market regime
- Predict market trend based on regime



Requirements

Strategies



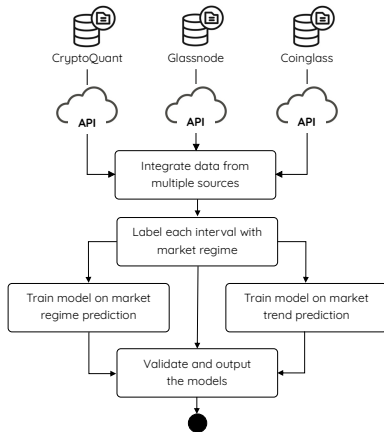
Trained Model

Fitting

Back  
Forward

# Model Training Flow

This is how we train our ML models.



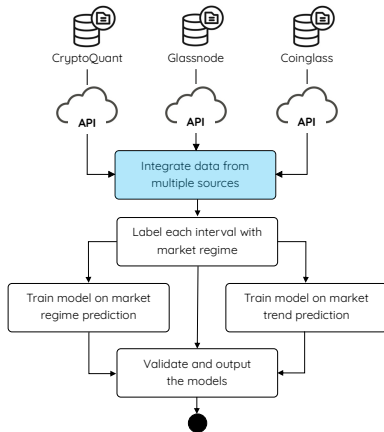
# Model Training Flow

Data source:

- CryptoQuant
- Glassnode
- Coinglass

Actions:

- Combine datasets
- Remove unused attributes
- Preprocessing



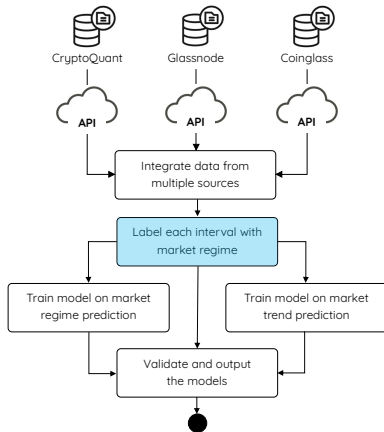
# Model Training Flow

Types of market regime:

- Bull
- Bear
- Neutral

Algorithms:

- HMM models
- Clustering models (e.g. k-means algorithm)



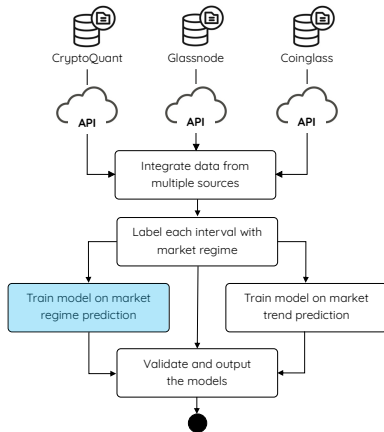
# Model Training Flow

Actions:

- Split dataset into training and testing sets
- Apply classification algorithm
- Evaluate model and improvise

Algorithms:

- Random forest
- Neural network





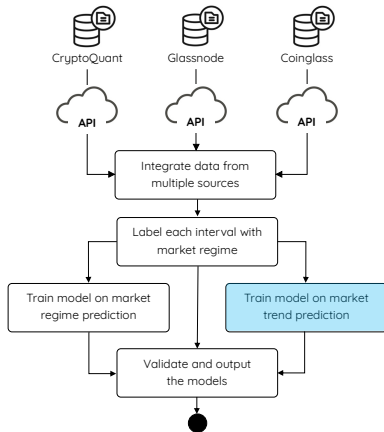
# Model Training Flow

Actions:

- Split dataset into training and testing sets
- Apply regression or time-series algorithm
- Evaluate model and improvise

Algorithms:

- Polynomial regression
- LSTM model



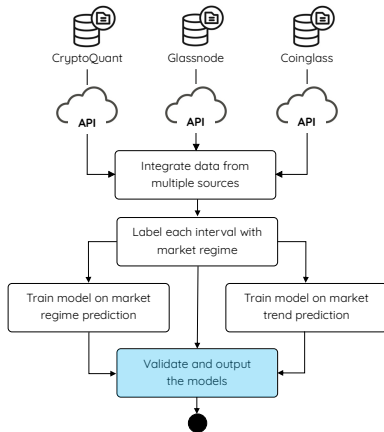
# Model Training Flow

Actions:

- Perform final evaluation
- Save models to be used in backtest library

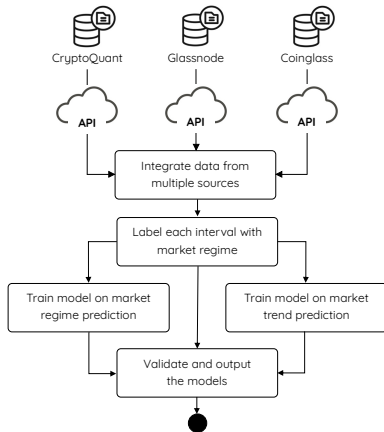
Approach:

- Hybrid model
- Output models separately



# Model Training Flow

This is how we train our ML models.



# Trained Model

Three components:

- Identify and label market regime
- Predict future market regime
- Predict market trend based on regime



Requirements

Strategies



Trained Model

Fitting

Back  
Forward

# Backtest Library

Four components:

- Data import and preprocessing
- Fitting data to model
- Backtesting and forward testing
- Performance evaluation

CryptoQuant  
Glassnode  
Coinglass



Analyst

Inputs

Strategies



Model



Report

## Backtesting Library

Data Import & Preprocessing



Data Cleaning



Data Reduction



Data Transformation

Fitting Data to Model



Split data into intervals



Predict market regime



Predict market trend

Backtesting & Forward Testing



Apply strategy to model



Perform transaction based on criteria



Record each transaction made

Performance Evaluation



Summarize data



Calculate metrics



Generate report

# Backtest Library

## Data Import and Preprocessing

Actions:

- Remove null or missing data
- Filter out unused data
- Feature extraction
- Correct data format
- Normalization (if needed)

CryptoQuant

Thorsnode

Coinglass

API

Trading D

Timefram

Requirement

- Funds
- Fees

Strategies

Analyst

Backtesting Library

Data Import &  
Preprocessing

Data C

Fitting Data to

Sp  
inter

# Backtest Library

## Fitting Data to Model

Purpose:

- Prepare data for backtesting and forward testing

Steps:

- Divide data into equal intervals (e.g. years)
- Feed data into the models:
  - Identify market regime
  - Predict market regime
  - Predict market trend

Glassnode

Coinglass

API

Trading Data

Timeframe

Requirement

- Funds
- Fees

Strategy

Analyst

Trained Model

Fitting Data to Model

+

Backtesting &



Cleaning



Split data into

Apply to

# Backtest Library

## Backtesting and Forward Testing

Actions:

- Use existing data for backtesting
- Use newly predicted market trend for forward testing
- Apply the strategy to be tested
- Record each simulated transaction made



• Funds  
• Fees

Strategies



Trained M



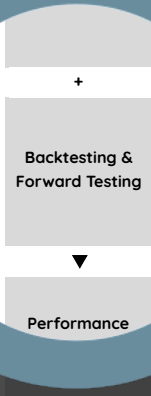
Data into  
als



Apply s  
to n



Report

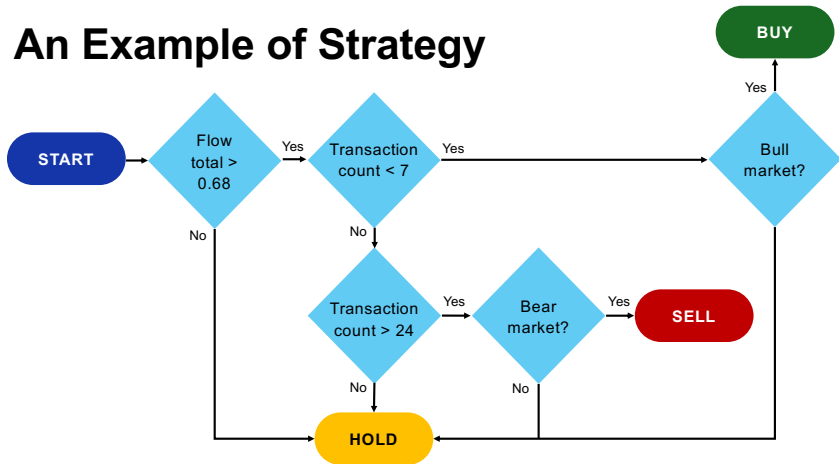


Performance

Summ



# An Example of Strategy



# Backtest Library

## Backtesting and Forward Testing

Actions:

- Use existing data for backtesting
- Use newly predicted market trend for forward testing
- Apply the strategy to be tested
- Record each simulated transaction made



• Funds  
• Fees

Strategies



Trained Model



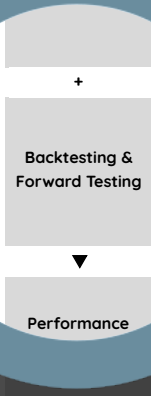
Data into  
models



Apply strategy  
to model



Report



Backtesting &  
Forward Testing

Performance

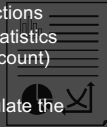
Summary

# Backtest Library

## Performance Evaluation

Actions:

- Summarize all transactions made and calculate statistics (e.g. total transaction count)
- Apply formula to calculate the associated metrics
- Generate report



Report

Trained Model

Forward Testing

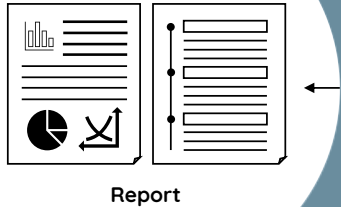
Performance  
Evaluation

Summary

# Output

Prepare report with details like:

- Performance metrics:
  - Sharpe ratio (SR)
  - Maximum drawdown (MDD)
  - Profit and Loss (PnL)
- Graphs and charts
  - Equity curve
  - Trade signal timeline
- Full list of trade records / transactions





thank  
you