

Objective

Predict next-day cryptocurrency volatility using engineered features from OHLCV + market cap data, enabling risk management and trading strategy optimization.

System Architecture

- **Data Layer:**
 - `data.py` → load, clean, fill missing, winsorize outliers.
- **Feature Layer:**
 - `features.py` → rolling volatility, GK/Parkinson, ATR, liquidity ratios, calendar features.
- **Model Layer:**
 - `models.py` → preprocessing (RobustScaler + OneHotEncoder), XGBoost regression pipeline.
- **Training Pipeline:**
 - `train.py` → orchestrates data load → feature build → train/eval → save model → export metrics.
- **Serving Layer:**
 - `serve.py` → Streamlit app for interactive predictions (auto latest data or manual input).

Data Flow

Raw CSV → load_data → fill_missing → clean_outliers → build_features → train_and_eval → save_model → serve.py (Streamlit UI)

Non-Functional Requirements

- **Reproducibility:** fixed random seed, saved artifacts.
- **Scalability:** modular pipeline, parquet storage.
- **Deployment:** Streamlit app with cached model/data loading.
- **Monitoring:** metrics.csv for performance tracking.