Exploratory data analysis report

This report summarizes the dataset characteristics, preprocessing outcomes, engineered features, and key insights derived from your EDA code.

Dataset overview

- **Source & schema:** Daily OHLCV and market cap per cryptocurrency, with columns normalized to lowercase and `marketcap` renamed to `market_cap`.
- **Time index:** `date` parsed to datetime; data sorted by `crypto_name, date` and deduplicated on that pair.
- **Types & ranges:** Numeric coercion applied to price/volume/cap; non-negative clipping for `volume` and `market_cap`.
- **Shape & coverage:**
    - **Rows/columns:** Report the final `df.shape` after cleaning.
    - **Assets:** List of unique `crypto_name` values.
    - **Date range:** Minimum and maximum `date`.
- **Missingness:** Count of missing values per column before feature engineering; note reductions after rolling features and target creation.

> Tip: Include a small table with counts per crypto (rows, start/end date) to show coverage consistency.

Preprocessing summary

- **Schema normalization:** Lowercased columns; removed `unnamed: 0`
- **Type coercion:** Converted OHLCV, `volume` and `market_cap` to numeric; invalid strings coerced to NaN.
- **Deduplication:** Dropped duplicate `(crypto_name, date)` pairs.
- **Non-negativity:** Clipped `volume` and `market_cap` at zero to avoid negative liquidity artifacts.

> Rationale: Ensures consistent, model-ready inputs and avoids downstream errors in volatility and liquidity computations.

Feature engineering

Returns and rolling volatility

- **Log return:**

log_rett=ln(close_t/(close_t−1))

- **Rolling volatility:**

vol_7d_t=std(log_rett−6:t),vol_14d_t=std(log_rett−13:t)

Garman–Klass volatility (daily proxy)

- **GK estimator (as implemented):**

GKt=(0.5 · (lnhightlowt)^2−2·(lnclosetopent)^2)^1/2

- **Note:** The classic GK includes an additional term with ln(close/open) and a coefficient; your simplified form is acceptable as a proxy but document the choice.

Liquidity and spreads

- **High–low spread:**

hl_spread_t= (high_t−low_t) /close_t

- **Turnover ratio:**

turnover_t= volume_t / market_cap_t

Calendar features

- **Day of week:** 0–6 (Mon–Sun)
- **Month:** 1–12

Target variable

- **Next-day GK volatility:**

target_vol_t=GK_t +1

After feature creation, rows with NaNs (from rolling windows and shifts) are dropped; features are saved to `data/processed/features.parquet`.

Visual analysis

1. Closing Price Trends

- **What was plotted:** Line plots of `close` prices for five representative cryptocurrencies — Aave, Algorand, ApeCoin, Avalanche, and BNB.
- **Insight:**
  - Clear regime shifts and drawdowns, especially around 2021.
  - BNB and Aave show pronounced peaks, indicating speculative cycles.
  - Algorand and ApeCoin exhibit shorter-lived volatility bursts.

2. Volume Trends

- **What was plotted:** Line plots of `volume` for the same five assets.
- **Insight:**
  - BNB shows the highest volume spikes, especially during bull runs.
  - Volume surges often precede or coincide with price peaks.
  - ApeCoin and Avalanche show erratic volume patterns, suggesting liquidity instability.

3. High–Low Spreads for Large-Cap Cryptos

- **What was plotted:** `hl_spread` over time for top 5 large-cap assets — Bitcoin, Ethereum, BNB, Cardano, Polkadot.
- **Insight:**
  - Spreads are generally tighter for large caps, but spikes occur during market stress.
  - 2018 and 2021 show notable spread expansions, aligning with volatility surges.

4. Feature Correlation Matrix

What was plotted: Heatmap of correlations among key features:

- `log_ret`, `vol_7d`, `vol_14d`, `garman_klass`, `hl_spread`, `turnover`, `target_vol`.
- **Insight:**
  - Strong positive correlation between `garman_klass` and rolling volatility.
  - `hl_spread` is highly correlated with GK, confirming its role as a dispersion proxy.
  - `turnover` shows weak negative correlation with volatility features, especially in large caps.

5. Distribution Plots

- **What was plotted:** Histograms + KDEs for `log_ret`, `garman_klass`, and `vol_14d`.
- **Insight:**
  - **Log returns:** Heavy tails, sharp peak at zero — typical of crypto returns.

    - **GK volatility:** Right-skewed, indicating frequent low-volatility days with occasional spikes.
    - **Rolling vol (14d):** Similar skew, confirming heteroskedasticity.

6. Calendar Effects

- **What was plotted:** Boxplot of `garman_klass` by `day_of_week`.
- **Insight:**
    - Mondays show elevated volatility, likely due to weekend gaps.
    - Mid-week (Wed–Thu) shows tighter distributions.
    - Weekends (Sat–Sun) exhibit lower median volatility but higher outlier frequency.

Key findings

- **Volatility structure:** Rolling vol and GK move together, confirming GK as a viable target proxy.
- **Liquidity–volatility link:** Higher `hl_spread` aligns with higher GK; `turnover` varies by asset class—small caps show higher volatility and unstable liquidity.
- **Data quality:** Initial missingness is addressed; rolling windows and target shift reduce usable rows—document final row count post-drop.
- **Seasonality:** Day-of-week effects may be present; quantify if notable.