Project Report

Project Title: Crypto-Index Construction via large-Scale correlation analysis and machine learning techniques, including gradient boosted trees, k-means clustering and principal component analysis

Course: *0842 Data Processing 2: Scalable Data Processing, Legal & Ethical Foundations of Data Science* | Date: *13.02.2025*

Team 1: *Andreas Oberdörfer, Fedor Samorokov, Alireza Ismaili*

1 Project Overview and Goal

1.1 Context and Motivation

Most retail cryptocurrency investors gain exposure through simple market-capitalisation-weighted portfolios, which tend to concentrate on a small number of highly correlated assets [1]. Highly correlated constituents reduce diversification benefits [2], so a naive "buy the biggest coins" approach may deliver lower diversification per unit of risk than a systematically constructed basket. Our project targets a *risk-adjusted* index construction workflow that aims to select a set of comparatively distinct assets and weight them in a stable way, with the long-term objective of providing an automated, transparent alternative to discretionary portfolio selection.

From a business perspective, the project can be viewed as a prototype for an index-construction engine that could power an investable product and "democratise" portfolio design. From a societal perspective, the project provides a reproducible study of how diversification and concentration arise in crypto markets and which methodological choices can reduce redundancy.

An important remark should be made here. The results of the project under any circumstances do not constitute an investment advice. Readers should bear in mind that in financial markets, historic returns do not mean anything about future returns.

1.2 Goal and Research/Business Questions

- **Primary goal:** Construct a monthly rebalanced cryptocurrency index using large-scale correlation analysis and ML (machine learning) techniques and compare it against simple baselines, such as Bitcoin returns.
- **Key questions:** (1) Does correlation-based pruning reduce redundancy while preserving performance? (2) Which Machine Learning algorithm yields the most robust out-of-sample risk-adjusted results?
- **Success criteria:** Out-of-sample evaluation using (at minimum) annualised return, annualised volatility, Sharpe ratio, maximum drawdown, and turnover; comparison against baselines such as Bitcoin.

1.3 Domain Terminology

We define acronyms and terminology at first use:

- **K-line (candlestick):** OHLCV time series (open, high, low, close, volume) over a fixed interval.
- **Volatility:** standard deviation of returns over a window; used as a risk proxy.

- **Sharpe ratio:** risk-adjusted performance proxy defined as $(\mu - r_f)/\sigma$; we use $r_f \approx 0$ for crypto unless otherwise stated.
- **Ragged time series:** assets have different start/end dates and missing timestamps due to listings, delistings, outages, or illiquidity; this affects the number of overlapping observations used for statistics.
- **Survivorship bias:** bias introduced when the universe includes only assets that "survived" to the end of the sample (e.g., currently listed pairs), thereby overstating historical performance by excluding delisted/failed assets.
- **The 5 Vs:** **V**olume, **V**elocity, **V**ariety, **V**eracity, and **V**alue.
- **Inverse-volatility weighting.** For the final set of constituents, we assign weights proportional to inverse volatility:

$$w_i = \frac{\sigma_i^{-1}}{\sum_{j=1}^{N} \sigma_j^{-1}},$$

  where σ_i is the estimated volatility of asset i over the weighting window, and $\sum_i w_i = 1$. Intuitively, more volatile assets receive smaller weights, which helps reduce concentration in extremely volatile coins.
- **Sharpe-based filtering.** For each asset, we compute a rolling mean return μ and volatility σ and rank assets by a Sharpe-like score $(\mu - r_f)/\sigma$. We set the risk-free rate r_f to approximately zero for crypto markets unless otherwise stated. This step prioritises assets with high return per unit risk.
- **Correlation pruning.** We compute the Pearson correlation matrix of asset returns over a fixed lookback window. If two assets exceed a correlation threshold $\rho^\star$ (e.g., $\rho^\star = 0.85$), we treat them as redundant and keep only one representative (the one with the higher risk-adjusted score).

2 Project Data Sources and Architecture

2.1 Data Sources

Our primary data source is the *Binance Public Data* repository [3], which provides downloadable historical market data. We use monthly K-line (candlestick) data in CSV format for a large universe of trading pairs (coins against USDT), which enables scalable backtesting and cross-asset correlation analysis.

We treat the repository as the source of truth for prices/volumes within the project scope and document any limitations (e.g., missing months, listing changes, stablecoin peculiarities). We also maintain a derived "universe table" that records each asset's first/last observed timestamp to support time-aware evaluation and avoid leakage.

Below is a short raw-data excerpt 1 from a Binance K-line CSV (e.g., `BTCUSDT-4h-2025-12.csv`), showing the schema and typical values:

open_time	open	high	low	close	volume	close_time	quote_volume	trades	taker_buy_base	taker_buy_quote	ignore
1.76455E+15	90360.01	90417	86161.61	86346.13	10249.65966	1.76456E+15	897849123.3	1722973	4191.17162	367273730.3	0
1.76456E+15	86346.13	86578.06	85604	86550.3	4937.16707	1.76458E+15	425115018	893445	2255.05761	194226852.2	0
1.76458E+15	86550.3	86938.01	86131.01	86149.15	3005.76787	1.76459E+15	260281847.2	691575	1246.98554	108038831.5	0
1.76459E+15	86146.59	86674	83822.76	84677.87	8124.37241	1.7646E+15	694074542.7	1860732	3634.31812	310708906.6	0
1.7646E+15	84677.87	85555	84030.95	85024.5	4598.10295	1.76462E+15	390099606.5	1456458	2365.71048	200745256.4	0
1.76462E+15	85024.5	86860	85007.69	86286.01	3593.94231	1.76463E+15	309603909.5	1084502	1831.09009	157725217	0
1.76463E+15	86286.01	87350	86184.39	86976.99	3245.81604	1.76465E+15	281278192.6	988788	1587.16677	137592391.8	0
1.76465E+15	86977	87288.48	86818.14	87012.65	1733.23534	1.76466E+15	150818966.7	565292	966.30676	84088679.12	0
1.76466E+15	87012.64	87628.78	86296.21	87368.92	3361.03567	1.76468E+15	292051091.3	691900	1527.03312	132822108.3	0
1.76468E+15	87368.92	91200	87032.75	90850.01	9300.50019	1.76469E+15	828829075.4	1777835	4675.2058	416512652.3	0

Figure 1: Screenshot of Binance K-line (candlestick) CSV schema and sample rows (BTCUSDT, 4h).

2.2 Architecture

Figure 2 visualizes this end-to-end pipeline. A downloader pulls monthly K-line archives from Binance Public Data and stores them as raw ZIP files; these are extracted and normalised into a CSV archive. A schema-enforcement step converts CSV into a columnar Parquet "data lake" layout that is efficient for Spark scans and rolling-window computations.

Within the Spark engine, the Parquet data are ingested into Spark DataFrames. We then iterate through a rolling monthly rebalance loop: for each rebalance date, we compute features over a training window (e.g., the last 90 days), estimate returns and volatility, apply a correlation filter to remove highly redundant assets, select the top-N assets by risk-adjusted score (Sharpe-like), and assign portfolio weights via inverse-volatility weighting. The resulting portfolio is held over the next test window (e.g., t+1 month), where we compute daily returns and aggregate cumulative performance metrics.

Finally, we export artefacts for reporting: (i) performance plots versus a Bitcoin baseline and (ii) monthly returns as a CSV for further analysis.

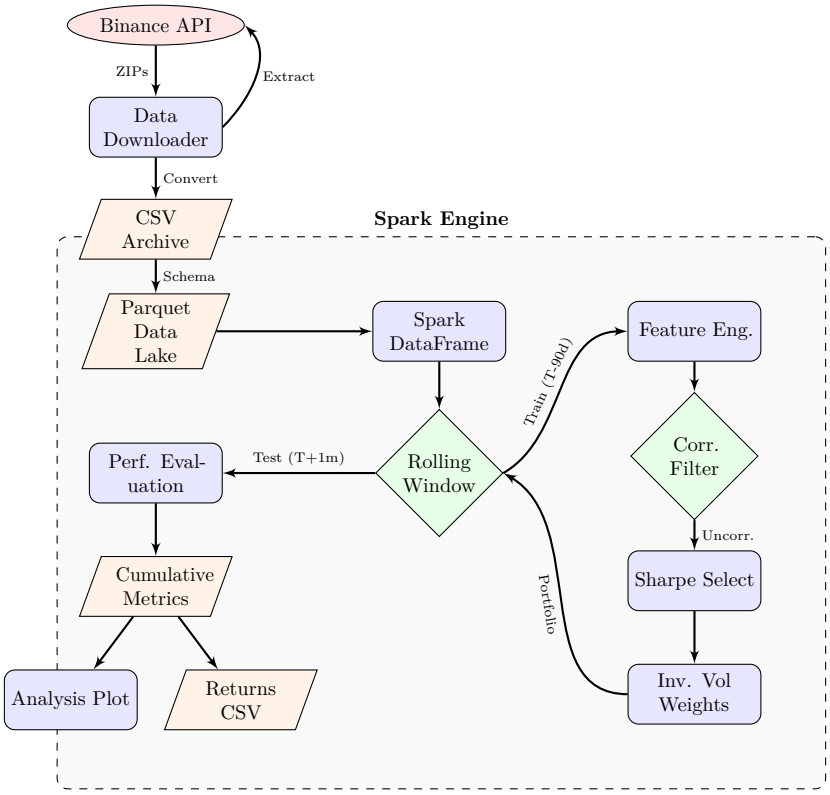


Figure 2: System Architecture: Data flows from Binance through a Parquet data lake into the Spark engine, where a rolling window backtester trains on historical data (right loop) and evaluates on future periods (left path).

2.3 Components Description

- **Data Downloader:** A multi-threaded scraper that fetches historical 4-hour candlestick data for all USDT trading pairs from Binance, eliminating survivorship bias by scanning the entire market rather than a fixed list.
- **Spark Engine:** The core processing unit that ingests partitioned Parquet files and executes the backtesting logic. It handles large-scale data transformation and aggregation using PySpark.

- **Rolling Backtester:** A custom module that simulates active management. It constructs a portfolio at the end of each month using only past data (metrics from $T - 90$ days) to predict performance for the next month $(T + 1)$, strictly avoiding look-ahead bias.
- **Portfolio Logic:** Implements a three-stage filter: liquidity check, correlation pruning (tournament to remove redundant assets), and Sharpe Ratio ranking, followed by Inverse Volatility Weighting to minimize risk.

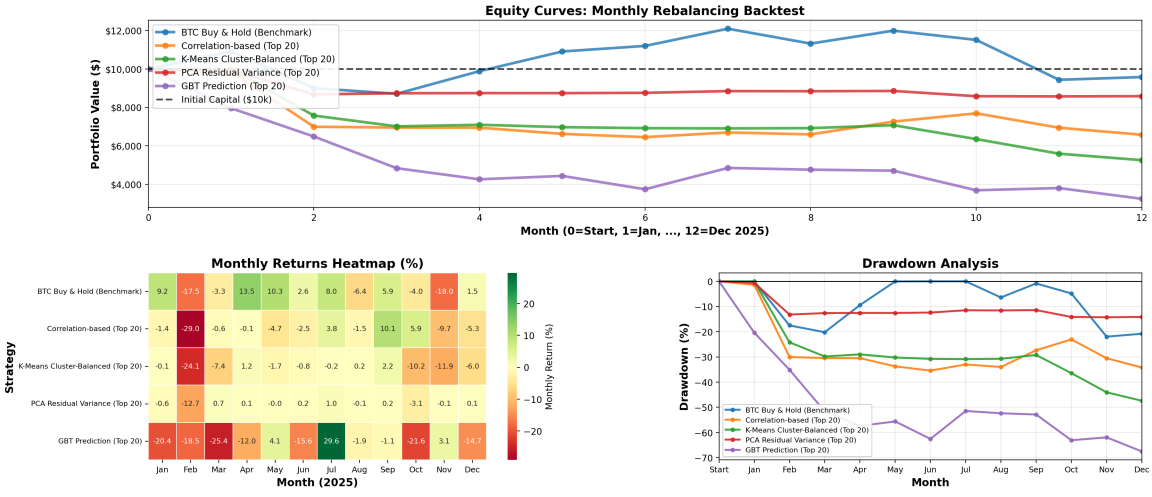
3 Analysis Steps and Results

3.1 Analysis Steps

To validate the "Smart Index" strategy, we performed a walk-forward analysis (rolling window backtest) over a 12-month period. At the end of each month, the engine dynamically scanned the entire available universe of cryptocurrencies. In the training phase, metrics such as annualised volatility and correlation were calculated based on the preceding window. The asset-selection logic then identified the top 20 uncorrelated assets with the highest Sharpe ratios. These assets were weighted by inverse volatility to construct a diversified portfolio, which was held for the subsequent month. This process repeated monthly, compounding returns to generate a continuous equity curve benchmarked against a passive Bitcoin (BTC) buy-and-hold strategy.

3.2 Results Summary and Interpretation

The results can be summarised in Figure 3.



Performance Summary (12-Month Backtest)

Strategy	Final Value	Total Return	Ann. Return	Ann. Vol	Sharpe	Max DD	Win Rate
BTC Buy & Hold (Benchmark)	**$9,581**	**-4.19%**	**-4.19%**	**34.36%**	**-0.12**	**-21.97%**	**58.3%**
Correlation-based (Top 20)	$6,581	-34.19%	-34.19%	32.36%	-1.06	-35.43%	25.0%
K-Means Cluster-Balanced (Top 20)	$5,259	-47.41%	-47.41%	25.31%	-1.87	-47.41%	25.0%
PCA Residual Variance (Top 20)	$8,587	-14.13%	-14.13%	12.46%	-1.13	-14.25%	50.0%
GBT Prediction (Top 20)	$3,248	-67.52%	-67.52%	51.39%	-1.31	-67.52%	25.0%

Figure 3: Backtest comparison of strategies versus a Bitcoin (BTC) buy-and-hold benchmark over the evaluation period.

The 12-month backtest reveals a challenging market environment where the benchmark (Bitcoin) itself ended

slightly negative (−4.19%).

Performance interpretation.

- **PCA Residual Variance Strategy:** This was the most effective active strategy. While it posted a negative return of −14.13%, it demonstrated superior risk management. Its annualised volatility was low at 12.46% (compared to BTC's 34.36%), and its maximum drawdown was limited to −14.25% (vs. BTC's −21.97%). This supports the thesis of capital preservation and volatility damping, making it a "Smart Index" candidate for risk-averse allocation.
- **K-Means Cluster-Balanced Strategy::** This approach attempted to diversify across different market "clusters" identified by K-Means, but ended with a -47.41% return and a maximum drawdown of -47.41%. Despite moderate volatility (25.31%), the strategy failed to capture positive returns, suggesting that clustering based on price patterns alone may group assets that share similar downward trends rather than providing true diversification benefits.
- **Correlation-based Strategy:** This strategy underperformed, ending with a −34.19% return and higher drawdowns (−35.43%). This suggests that selecting "uncorrelated" assets without a robust quality filter can lead to persistent exposure to losing assets.
- **GBT Prediction Strategy:** The ML-based prediction performed worst in this regime, with a maximum drawdown of −67.52% and the highest volatility (51.39%), consistent with overfitting to past data and/or poor adaptation to the market regime during the test period.

Conclusion. The core hypothesis regarding volatility reduction was validated by the PCA strategy, which reduced market risk substantially. However, alpha generation (beating the benchmark in absolute terms) remained elusive in this downtrending market, highlighting the difficulty of long-only strategies during bearish phases characterised by market pessimism and fallig prices.

4 Legal and Ethical Issues

4.1 Potential Issues and Mitigations

This project processes public cryptocurrency market data to produce index-like portfolio outputs. The main legal and ethical risks relate to (i) data usage rights and compliance with provider terms, (ii) the possibility that results are interpreted as investment advice, and (iii) methodological pitfalls that can mislead readers (e.g., look-ahead bias, survivorship bias, and overfitting).

Data usage, licensing uncertainty, and terms of use. We source historical K-line (OHLCV) data from Binance Public Data. The repository does not clearly provide an explicit open-data licence, so we treat the dataset as *restricted-use* for this course project: we download only what is necessary, retain the original attribution, and do not redistribute raw data dumps. We publish only derived aggregates/figures. Where legal terms are ambiguous, we follow a conservative interpretation: non-commercial academic use and citation. This is in line with Article 3 of the DSM Directive (2019/790), which mandates that research organizations can perform data mining for scientific research without rightsholder permission, provided they have lawful access [4].

Investment-advice risk and responsible communication. An index construction pipeline can produce outputs that look "investable". This creates a risk that readers interpret the report as financial advice. We mitigate this by (1) presenting all results as *historical backtests* rather than forecasts, (2) explicitly stating that the work is educational and not intended for live trading, (3) reporting adverse outcomes and limitations (including cases where the strategies underperform the benchmark), and (4) avoiding prescriptive language such as "buy" or "guaranteed." We also avoid claims that are not supported by evidence and include baseline comparisons to reduce cherry-picking.

4.2 Guiding Principles

We guided the project using the following principles:

- **Compliance and attribution:** follow provider terms, cite data sources, and avoid redistributing raw data when licensing is unclear.
- **Transparency:** document assumptions, parameters (windows, thresholds), and all preprocessing decisions affecting results.
- **Bias control:** use time-aware evaluation, explicitly address survivorship bias, and avoid look-ahead leakage.
- **Responsible communication:** present outputs as retrospective simulations, avoid investment-advice framing, and report limitations and negative findings.

5 Experience Gained

5.1 Challenges Encountered

Data Quality & Survivorship Bias: One of the biggest hurdles was ensuring the backtest was realistic. Early attempts used a hardcoded list of today's top coins, which artificially inflated results. We had to engineer a dynamic downloader that reconstructs the market state as it existed in the past.

Spark Integration: Transitioning from a simple Pandas loop to a distributed Spark environment for feature engineering required significant refactoring.

Look-Ahead Bias: Strictly enforcing the separation between "Training" (past) and "Testing" (future) windows was conceptually simple but implementation-heavy, requiring a custom rolling-window class to manage the time-series splitting correctly.

5.2 Team Learning (per Member)

- **Fedor Samorokov:** Focused on the project's conceptual framework and verified that our methodology aligned with ethical data practices. Gained significant experience in synthesizing technical results into a coherent academic report using LaTeX, ensuring the legal implications of automated trading strategies were considered.
- **Alireza Ismaili:** Took charge of code quality and documentation. Gained experience into Python best practices code documentation for complex algorithms, ensuring the codebase is maintainable for future researchers.
- **Andreas Oberdörfer:** Driving force behind the core strategy and the backtesting engine. Gained extensive practical experience with PySpark for financial time-series analysis and the implementation of advanced portfolio construction techniques in a production-like environment.

5.3 Recommendations and Future Work

- **Regime Detection:** The current strategy struggles in bear markets (with falling prices). Implementing a "Market Regime" filter (e.g., using Hidden Markov Models) to switch to stablecoins (USDT) during downtrends could significantly reduce drawdowns.

- **Transaction Costs:** The current backtest assumes zero fees. Future iterations must incorporate exchange fees and slippage models to provide a more realistic net-return estimate.
- **Alternative Data:** Enhancing the model with on-chain metrics (e.g., wallet activity, hash rate) or sentiment analysis could improve the asset selection process beyond simple price-volume technicals.

6 Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used Antigravity in order to generate and refactor the codebase, ChatGPT in order to perform conceptual research and topic generation, Gemini in order to search for references and conduct legal research, and Prism in order to draft the report. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

7 Licensing

We publish **code** under the MIT License to maximise reusability of implementation components while retaining attribution and a clear disclaimer of warranty [5]. We publish the **report text and figures** under CC BY 4.0 to enable sharing and reuse of the written/visual material with attribution, which is common for academic writing [6]. This separation matches the different nature of the artefacts (software vs. written content).

References

[1] Bybit Learn. *Navigating Bull and Bear Markets: A Dive Into Users' Asset Allocation (Q2 2024)*. 2024. Accessed February 11, 2026. `https://drive.google.com/file/d/1gd2c08F5Onz1v3WLunC4WGKPGqZfTUCa/view`.

[2] Binance Research. *Annual Crypto Correlations 2019*. `https://www.binance.com/en/research/analysis/annual-crypto-correlations-2019`. Accessed February 13, 2026.

[3] Binance. *Binance Public Data*. Accessed February 11, 2026. `https://data.binance.vision`.

[4] European Union. *Directive (EU) 2019/790 of the European Parliament and of the Council of 17 April 2019 on copyright and related rights in the Digital Single Market*. Official Journal of the European Union. `https://eur-lex.europa.eu/eli/dir/2019/790/oj/eng`. Accessed February 13, 2026.

[5] MIT Technology Licensing Office. *Exploring the MIT Open Source License: A Comprehensive Guide*. `https://tlo.mit.edu/understand-ip/exploring-mit-open-source-license-comprehensive-guide`. Accessed February 13, 2026.

[6] Creative Commons. *Creative Commons Attribution 4.0 International (CC BY 4.0)*. `https://creativecommons.org/licenses/by/4.0/deed.en`. Accessed February 13, 2026.