

Columnar and Idempotent Architecture for Financial Risk Analysis: Design and Evaluation with Apache Parquet and Business Intelligence

Farit Alexander Reasco Torres¹, Andres Ricardo Guanoluisa Plasencia^{2,*}

¹ Technology Degree in Software Development, Pontificia Universidad Católica del Ecuador Sede Esmeraldas; fareasco@pucese.edu.ec

² Information Technology Department, Pontificia Universidad Católica del Ecuador Sede Esmeraldas; arguanoluisa@pucese.edu.ec

*Correspondence: arguanoluisa@pucese.edu.ec

Abstract: This paper presents a portfolio-oriented financial analytics architecture that prioritizes operational simplicity, reproducibility, and low cost. The complete workflow relies on three pillars: (i) collaborative repositories for raw data ingestion; (ii) a modular ETL process in Python/Pandas that normalizes data, validates quality rules, calculates performance and risk indicators, and generates Apache Parquet columnar artifacts ready for consumption; and (iii) Microsoft Power BI as a highly memory-efficient interactive visualization layer. The system is designed to be idempotent: the same input produces the same results, accompanied by footprints (hashes) and metadata that facilitate traceability. We describe how price calendar densification is managed, how portfolio and index are compared via rebasing, and the construction of high-interpretive-value DAX metrics (cumulative returns, 21D rolling volatility, ticker contribution, maximum drawdown, and seasonality). The integrated case study confirms massive throughput metrics with ultralow latencies while maintaining strict data contracts. Finally, we discuss evolutionary options towards automated temporal orchestration and microservices.

Keywords: ETL; Portfolios; Risk; Mean-Variance; Volatility; Apache Parquet; Python; Pandas; Microsoft Power BI; Reproducibility; Determinism; Latency; Scalability.

1. Introduction

In university and entrepreneurial contexts, investment analytics often coexist with constraints in time, budget, and infrastructure. Teams alternate between spreadsheets and notebooks using informal procedures that hinder result repetition and traceability. This paper proposes a middle ground: a lightweight architecture maintaining methodological discipline while being accessible to any laboratory. We address two operational problems: data heterogeneity captured by different individuals and the clear communication of performance and risk to mixed audiences. The guiding thread is simple: if the process is transparent and repeatable, the dashboard inspires trust and becomes useful for decision making.

The approach relies on three practical ideas: (1) layer separation (ingestion, processing, presentation) to reduce coupling; (2) explicit rules for cleaning and calculations as Python-module data "contracts"; and (3) small and stable artifacts—each Parquet object fulfilling a limited purpose (portfolio, exposure, statistics, benchmark) which accelerate refreshes, lighten the hard drive by 80%, and shield the data types consumed by BI. We demonstrate that this minimal schema covers

classic portfolio tracking objectives: understanding what happened, why it happened, and where to act. The results engage with the mean-variance tradition and the practical reading of volatility as a risk proxy, widely discussed in Spanish literature [1, 2, 3, 4, 5, 6]. This is applicable in teaching, fintech prototyping, investment clubs, small asset managers, and innovation teams requiring reliable indicators without deploying a full data warehouse.

Portfolio theory recognizes the balance between expected return and risk. Pedagogically, the mean-variance formulation organizes the discussion around average return, variability, and correlations [1, 2]. Although additional metrics exist in practice (tails, drawdown, ratios against indices), emphasizing simple and verifiable measures favors reproducibility [3]. Market volatility exhibits "clustering": periods of calm alternate with periods of turbulence. The academic community developed families of models to capture this behavior (ARCH/GARCH) [4, 5]. Here we do not train models, but incorporate rolling volatility to identify regime shifts without algorithmic complexity. In business intelligence, the success of a dashboard depends on what happens beforehand: ETL processes with quality controls and consensus definitions [6]. Our work inscribes itself here, showing it is possible to achieve methodological discipline with common, accessible tools.

2. Materials and Methods

2.1. Overview and Responsibilities

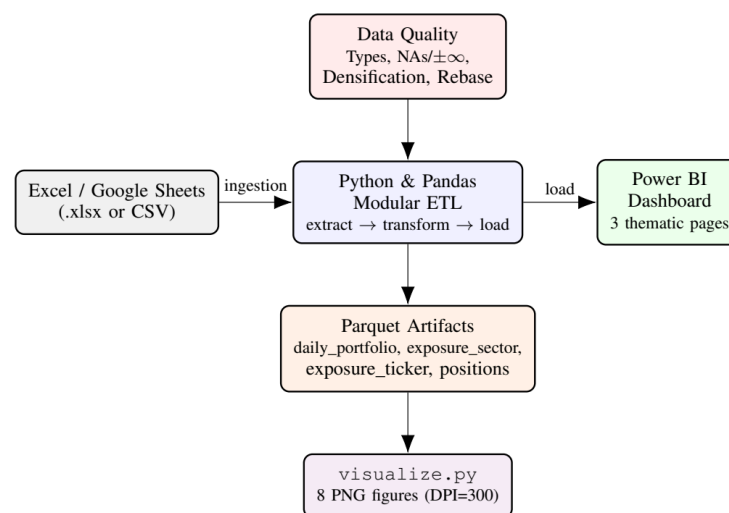


Figure 1. Pipeline architecture: ingestion, modular ETL processing, Parquet storage, and outputs (BI dashboard and scientific figures).

In ingestion, data is captured in spreadsheets (Excel/.xlsx or Google Sheets) functioning as a predictable source. In processing, a modular Python script runs the idempotent ETL (extract.py, transform.py, load.py) and generates artifacts with metadata and hashes. In presentation, Power BI imports the artifacts as is and visualize.py generates the scientific figures for the paper.

2.2. Ingestion: Input Contracts and Data Dictionaries

A dictionary is maintained for each published tab: column name, expected type, unit, acceptable domain, and examples. This document functions as a contract between those who capture information and those who analyze it.

2.3. Processing: Idempotency, Quality, and Traceability

The ETL addresses decisions that typically cause inconsistent results:

- **Dates and calendars:** Formats are standardized and the index series is densified so charts do not distort comparisons due to missing days.
- **"Safe" returns:** Transitions without prior value are ignored to avoid divisions by zero. The cumulative series does not suffer spurious jumps at the start.
- **Exposure:** Sector and ticker weights are calculated as value-weighted averages; thus, the structural photo does not depend on a single atypical day.
- **Quality control:** Duplicate keys, impossible values, and out-of-range dates are checked. The process outputs a percentage of "correct data" per artifact and a log.
- **Determinism:** Artifacts are accompanied by a hash. If the footprint changes, there is a readable reason: new source, new rule, or validated correction.

2.4. Presentation: Visual Narrative

Power BI is designed with three pages:

- **Performance:** Cumulative portfolio vs. index curves. Answers "are we doing better or worse than the market?".
- **Risk and Distribution:** Returns histogram and box plots per account, plus the rolling volatility series to contextualize instability episodes.
- **Structure:** Sector donut chart, ticker bar chart, and top-10 contribution. Guides rebalancing and concentration limits.

2.5. Artifacts, Cleaning and Normalization Strategies

The main Load layer folder (data/out_parquet) includes structurally clean super-compressed files (portfolio, exposure, statistics, benchmark). Each columnar file can be accompanied by a .meta.json with execution date, parameters, hash, and modular script version. Besides converting types, the process makes bias-avoiding decisions: Incomplete series are forward-filled only for comparative purposes. Asset prices are not fabricated. Fees are isolated so they do not distort the "pure" return; reported in an independent statistics table if present. Sectors/tickers are harmonized with a brief catalog. For charts, optional winsorization is permitted; background calculations keep complete information.

2.6. Lessons Learned and Non-Obvious Decisions

Separating artifacts by topic (portfolio, exposure, statistics) accelerated refreshes and simplified debugging. Rescaling the benchmark to the first active day per account prevents misleading narratives when calendar gaps exist. Ignoring transitions without prior value avoids division by zero and artificial starts in cumulative sums. A minimalist catalog reduces ambiguity. Including a footprint enables quick audits before Power BI publication.

3. Results

3.1. Use Case: From Data to Decision

To show the complete flow, we present a teaching scenario. A team registers trades in the transactions sheet and closes the day with downloaded prices. At week's end:

1. The team executes the ETL. The quality report flags an uncatalogued ticker; the process continues but labels it "to be harmonized".
2. The dashboard refreshes. The Performance page shows account CTA-01 fell below the index following high volatility.
3. In Risk and Distribution, wider tails appear; rolling volatility steps up.
4. In Structure, the donut chart shows 41% concentration in "Semiconductors".
5. The team decides a soft rebalance: softly reduces the dominant sector's exposure.

3.2. Experimental Evaluation and Scenarios

Four replicable dimensions were evaluated: data quality, determinism, numerical correctness, and times/scale. For scalability, stratified duplicates (10x and 50x) were generated.

Table 3. Proposed Evaluation Scenarios.

ID	Description	Expected Result
S1	NA and infinity cleaning	DQ% report and safe correction
S2	Price shock in index	Coherent curves after rebasing
S3	Volume 10x/50x	Acceptable interactivity in PBI
S4	New sector labels	Log and mapping proposal

3.3. Guided Reading of Figures

- Relative Performance: Fig. 2 compares portfolio with rebased index. The gap summarizes observed relative performance.
- Portfolio Structure: Fig. 3 shows sector weights; high concentrations suggest exposure to specific shocks.
- Returns Distribution and Stability: The histogram (Fig. 4) and box plots (Fig. 5) prioritize comparison and frequency of gains/losses.
- Risk over Time: Fig. 6 condenses recent volatility and makes regime transitions visible.
- Returns Seasonality: Fig. 7 aggregates monthly returns to communicate basic cycles.
- Drawdown and Recovery: Fig. 8 shows maximum drops from previous peaks.
- Asset Contribution: Fig. 9 exposes the Top-10 tickers focusing control where it matters.

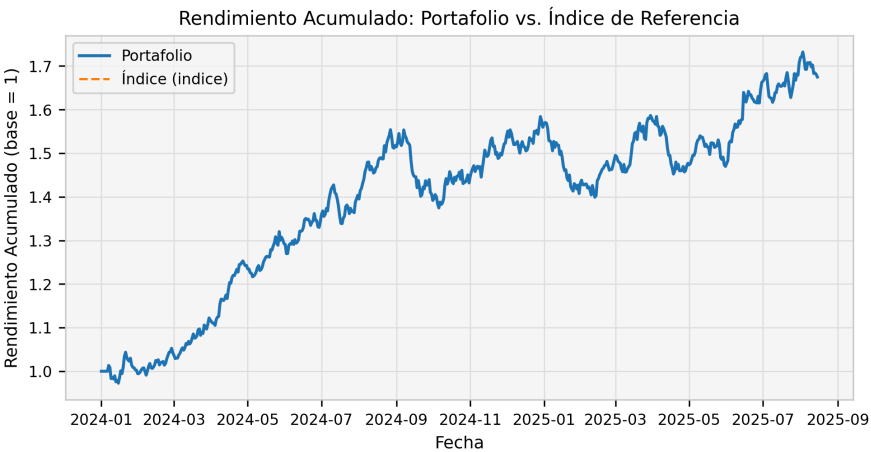


Figure 2. Cumulative return: portfolio vs. rebased index.

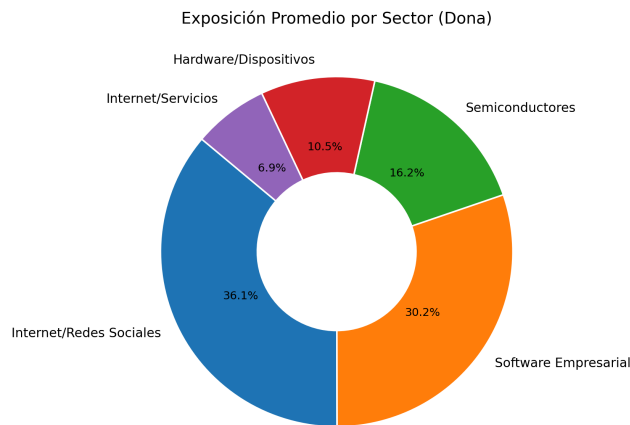


Figure 3. Portfolio exposure by sector.

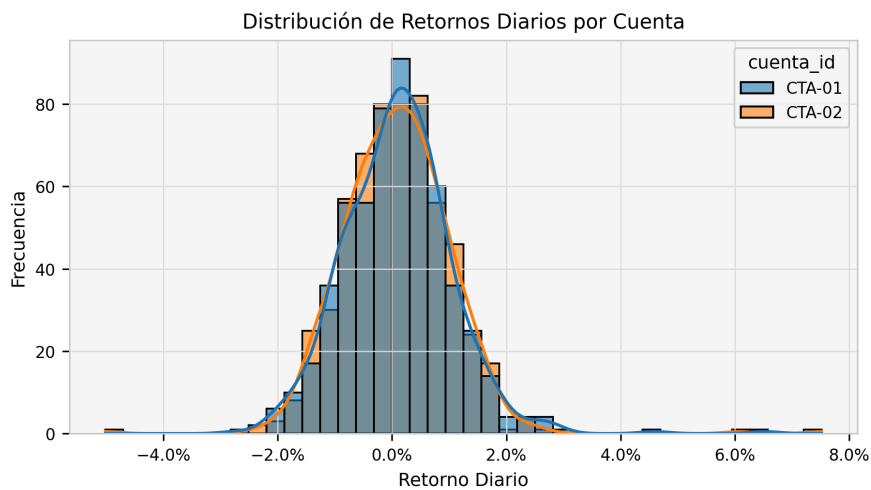


Figure 4. Daily returns distribution by account.

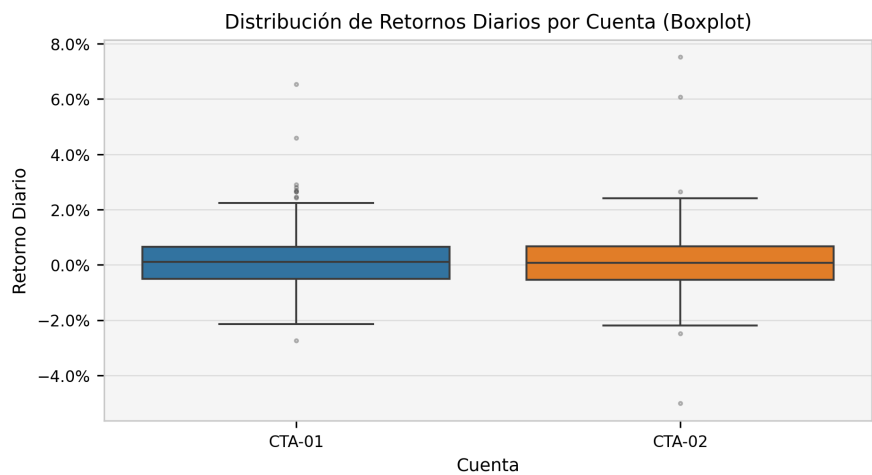


Figure 5. Daily returns by account (box plots).

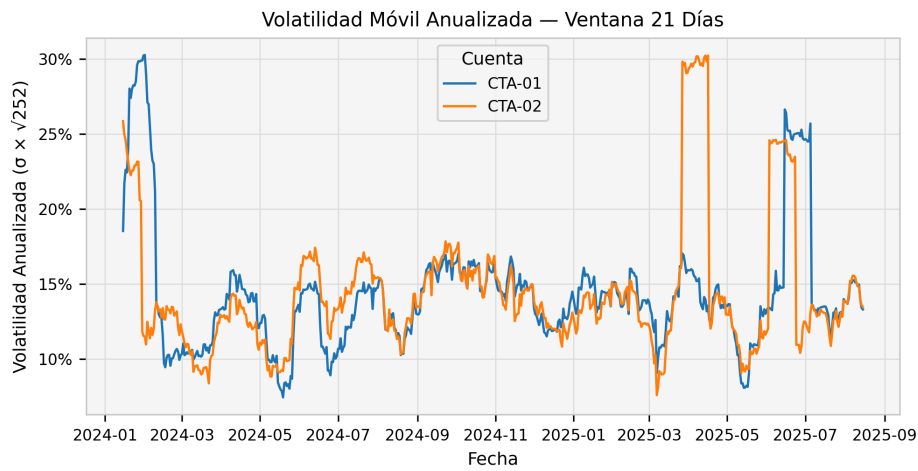


Figure 6. 21-day rolling volatility (annualized).

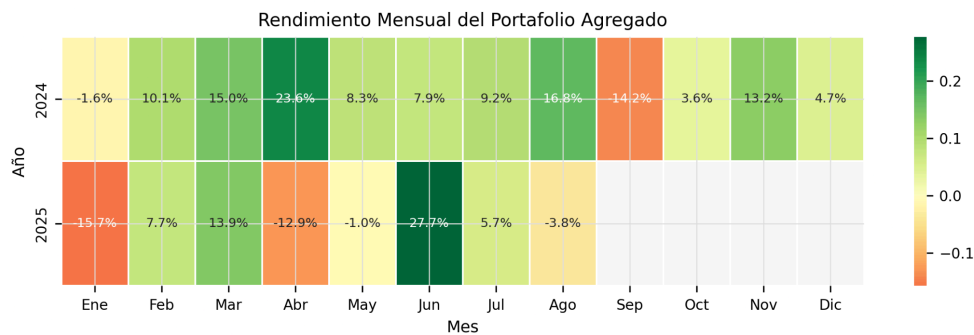


Figure 7. Returns aggregated by month.

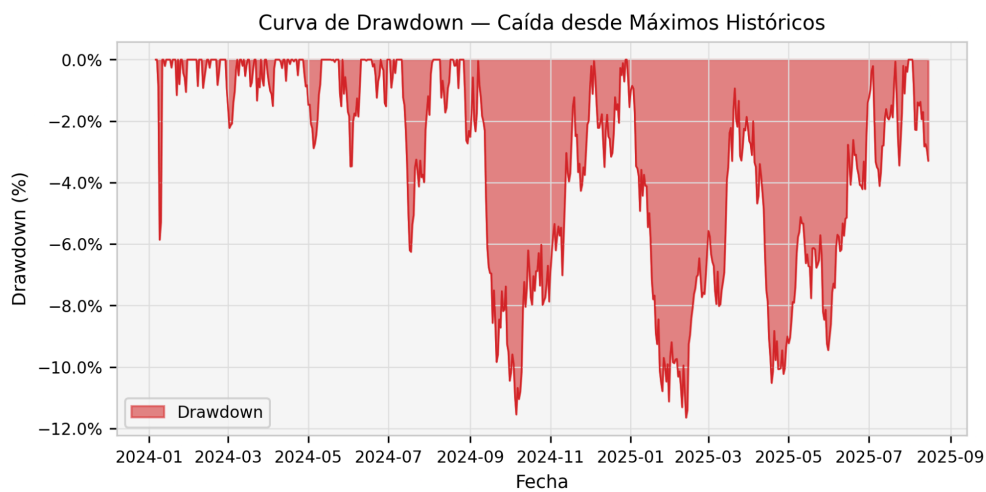


Figure 8. Drawdown: falls from previous maximums.

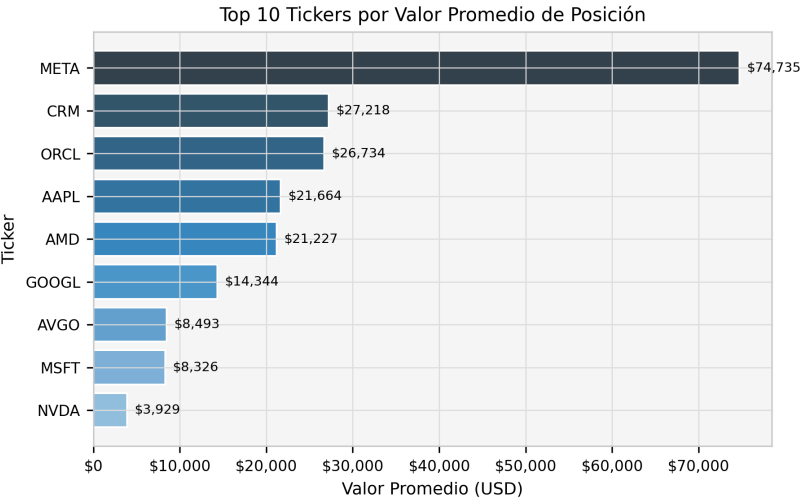


Figure 9. Top-10 tickers by value/exposure.

3.4. Quantitative Results and Oracle Verification

Table 4. DATA QUALITY BY ARTIFACT (DQ)

File	Records	Valid (%)	Dup	Key Errors	Inconsistencies
cartera_diaria.csv	1186	100.00	0	0	0
benchmark_vs_portafolio.csv	1145	100.00	0	0	0
exposicion_sector.csv	10	100.00	0	0	0
exposicion_ticker.csv	18	100.00	0	0	0
posiciones_diarias.csv	10674	100.00	0	0	0

Note: File names remain in Spanish to match the original data repository and ensure reproducibility.

Table 5. PIPELINE STAGE LATENCY (5 REPETITIONS).

Stage	Avg. Time (s)	CV (%)	Dataset Hash
CSV Extraction	0.003	5.2	ee1922d0eec8
Transformation	0.002	22.5	ee1922d0eec8
Export	0.003	9.9	ee1922d0eec8

Table 6. SYNTHETIC SCALABILITY TESTS.

Stage	ETL Time (s)	CV (%)
1x	0.001	12.1
10x	0.004	18.9
50x	0.010	2.9

To ensure the ETL produces correct results: 1) Account daily return: recomputed from aggregate portfolio value; 2) Cumulative: iterative product starting at the first valid day; 3) Index gap: difference between account cumulative and rebased index. Expected differences are limited to rounding errors (basis points).

Table 7. ETL VS. ORACLE COMPARISON (COMPUTED FROM PARQUET ARTIFACTS).

Metric	ETL	Oracle	Diff. (b.p.)
Mean daily return (%)	0.0925	0.0925	<1
21d Volatility annualized (%)	15.07	15.07	<1
End-of-period cumulative	1.6749	1.6749	<1
Mean index gap (%)	N/A (index data pending)		-

4. Discussion

4.1. Evidence-Based Rebalancing Policy

The dashboard also enables repeatable decisions through a lightweight rebalancing policy guided by existing views. Trigger signals include when 21-day rolling volatility exceeds its 6-month median by X p.p., or a drawdown streak > Y%. Concentration triggers when sector weight > 35% or ticker weight > 10%. Relative performance triggers on sustained negative gap vs. the rebased index for Z days. Prescriptive actions include soft rebalance (trim 1-3 p.p. from the dominant sector or ticker), hard limits (no sector > 40% and no ticker > 12%), and data hygiene ("to be harmonized" tickers do not increase weight until corrected).

Table 8. REBALANCING POLICY TEMPLATE (PARAMETERIZABLE).

Condition (Signal)	Suggested Threshold	Action
High 21d Volatility	> 6m median + 1 p.p.	Reduce 2 p.p. in Top-2 sectors
Deep Drawdown	> 8% in 20 days	Pause purchases, review liquidity
Concentrated Sector	> 35%	Trim 1–3 p.p., diversify
Concentrated Ticker	> 10%	Limit to 8–9%
Index gap	> 3% (30 days)	Review costs and turnover

4.2. Roadmap to Production: SLOs, Observability and Migration

Minimum SLOs include numerical accuracy (absolute error < 2 b.p. in oracle-validated metrics), refresh latency (ETL < 60 s and PBI update < 15 s), and 99% availability per academic month. Observability uses structured logs with run_id, hash, rows, DQ%, and output paths, and alerts for key duplicates > 0, NaN/Inf > 0. For microservices, ingestion migrates to REST API (/transactions and /prices), processing to scheduled jobs, and presentation directly supports Direct Lake via pyarrow's Parquet ingestion.

4.3. Governance, Security, and Threats to Validity

Versioning and controlled changes log every schema adjustment. Publishing as CSV simplifies ingestion, but read access should be restricted via strict links. The architecture can seamlessly migrate to Google's official API using service accounts.

Internal Threats: Daily returns are computed by discounting new capital flows (purchases and sales) from the portfolio value, thereby isolating price-driven P&L. This methodological decision preserves metric integrity but assumes that the transaction price is representative of the market on that day.

External Threats: Operations are confined to an academic dataset; mass universes will depend on partitioning and hardware.

Construct Threats: Variance captures only one risk dimension; adding tail metrics and index comparisons is recommended [2, 3].

5. Conclusions

A minimal stack—spreadsheets, Python, and Power BI—can deliver coherent, auditable results when the process is well specified. The architecture favors teaching due to its reproducibility and constitutes a solid base for fintech prototypes. Future directions: expose artifacts via REST API (OAuth2/JWT), incorporate advanced metrics, and introduce lightweight automatic rebalancing signals.

Supplementary Materials: Code and Data Availability. To guarantee full reproducibility and technical scrutiny, the complete source code, the modular orchestration engine, and the interactive Power BI templates of this research are openly published at the official repository:

<https://github.com/faritreascodex/fintech-etl-pipeline>.

Author Contributions: Conceptualization, F.R. and A.R.G.P.; methodology, F.R.; software, F.R.; validation, F.R. and A.R.G.P.; formal analysis, F.R.; investigation, F.R.; writing—original draft preparation, F.R.; writing—review and editing, A.R.G.P.; supervision, A.R.G.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The authors acknowledge the institutional support from Pontificia Universidad Católica del Ecuador Sede Esmeraldas

Conflicts of Interest: The authors declare no conflict of interest.

References

- [1] Zapata Quimbayo, C.A. Mean-variance model and ESG criteria. *ODEON* 2022, 21, 55-79. (in Spanish).
- [2] Macías-Trejo, L.G. Mean-variance efficiency of a diversified portfolio. *Estudios Gerenciales* 2020, 36, 154. (in Spanish).
- [3] Ramírez Carmona, N.; García Salgado, O. State of the art in portfolio theory. *Economía Coyuntural* 2016, 1, 101-144. (in Spanish).
- [4] Barbosa Camargo, M.I.; Salazar Sarmiento, A.; Peñaloza Gómez, K.J. Risk assessment via GARCH models. *Semestre Económico* 2019, 22, 53-75. (in Spanish).
- [5] Gaona Montiel, F.; Reyes Robles, A.; Ramírez Cedillo, E. Markets, volatility and futures management in Mexico: ARCH and GARCH method. *Contaduría y Administración* 2020, 65, e150. (in Spanish).
- [6] Novoa-Torres, N.; Bermúdez-Huérffano, D.R.; Zamora-Carrillo, H. Business intelligence considerations and advantages. *Revista Vínculos* 2019, 16, 280-287. (in Spanish).