Methodology and Reproducibility for Participant-Level Order Flow Forecasting: A Pre-Registered Pipeline

Joerg Osterrieder

2026-04-25

Abstract

We present a pre-registered, reproducibility-first methodology stack for daily-frequency forecasting research with participant-decomposed Xetra order flow data. The pipeline encodes a strict synthetic-first development contract, a walk-forward purge-and-embargo cross-validation design, a pre-committed estimator universe (Majority, Persistence, Momentum, Logistic, RandomForest with `RandomizedSearchCV(n_iter=50)`), paired hypothesis tests (DeLong for ROC-AUC, Diebold-Mariano with HAC standard errors and the Harvey-Leybourne-Newbold small-sample correction for predictive-accuracy differentials), Benjamini-Hochberg false-discovery-rate control over a closed 36-name hypothesis family, a serial-correlation-aware block bootstrap with locked parameters, and a fixed five-basis-point half-spread cost model with a Corwin-Schultz sensitivity. We define a three-tier claim-to-evidence taxonomy that ties every numerical claim in the paper to either a deterministic-block manifest hash (Tier alpha), a bootstrap computation seeded against that hash (Tier alpha-derived), or a Docker-only figure rendering (Tier beta, prohibited from in-line text). On synthetic data the deterministic block reproduces bit-identically across the reference run with manifest SHA `7be9ecaadf707a0bd948b5b83574c89d729ceca4893f53a84f9d6b413ca6fe6d`. An imposed-signal power demonstration confirms the methodology recovers daily information ratios in the literature-reported range. The contribution targets the methods gap that Harvey (2017) and Welch (2019) identify in empirical finance: when reproducibility primitives are missing, null findings are unpublishable and positive findings are unverifiable.

1. Introduction

Empirical finance has acknowledged a reproducibility crisis. Harvey (2017), in his AFA presidential address, documented that a substantial fraction of published cross-sectional asset-pricing factors fail out-of-sample, and attributed the gap to under-reported methodology choices, multiple-testing inflation, and the absence of pre-registration. Welch (2019) extended the critique by enumerating concrete reproducibility primitives that empirical finance papers routinely omit: pre-committed model universes, code and data archives, deterministic random-seed management, walk-forward cross-validation with explicit purge and embargo, and explicit multiple-comparisons correction over a closed hypothesis family. The emergence of granular participant-decomposed order flow data, available to academic researchers via venue-licensed feeds, both increases the research opportunity and raises the methodology bar: with hundreds of candidate features per firm-day and several plausible forecast horizons, the family of testable hypotheses inflates rapidly. A methodology that does not control this inflation produces results that no reviewer can defend.

This paper presents a pre-registered, reproducibility-first methodology stack for participant-level order flow forecasting. The empirical headline on Xetra data is reserved for a companion paper, submitted after the methodology is independently reviewed and the pre-registration tag (`pre-reg-v1`) is cut. The argument for addressing methodology before results is practical: without publicly auditable primitives, neither a positive finding nor a subsequent replication attempt has a defensible foundation.

The methodology documentation is precise enough for a second team to reproduce every primitive on their own data feed. A synthetic-with- imposed-signal power study demonstrates that the pipeline recovers information ratios at magnitudes consistent with the granular order-flow literature, providing the calibration check that methods-only papers are routinely challenged for omitting. The full pipeline is released as open-source code with a deterministic reference run whose manifest hash matches bit-for-bit across a multi-platform continuous-integration matrix, making the verification floor explicit for any downstream empirical claim.

1.1 Contributions

The contributions below are each pre-registered and tied to a specific code artefact in the open-source release.

(1) Synthetic-data manifest hashes match bit-for-bit across Linux-local, Linux-Docker, and macOS-local continuous-integration runners; the reference SHA is committed at `tests/fixtures/synthetic_reference_full_manifest.sha256` (current value `7be9ecaadf707a0bd948b5b83574c89d729ceca4893f53a84f9d6b413ca6fe6d`). The hash covers configuration, code commit, per-input-CSV SHA-256, fold edge hashes, model parameters, and the pipeline-step list. Provenance fields (wallclock, hostname, user, platform) are recorded but not hashed, so collaborators on different machines see the same hash whenever the deterministic inputs match.

(2) A three-tier claim-to-evidence taxonomy that disciplines paper text. Every numerical claim in the published manuscript carries a row in `docs/claim-to-evidence.md` with its tier and provenance. Tier alpha claims trace to a single 64-character manifest hash. Tier alpha-derived claims (bootstrap confidence intervals, paired-test p-values) trace to a tuple of `(manifest_sha, bootstrap_script_sha, bootstrap_seed)`. Tier beta claims, which depend on figure-pixel rendering and therefore on platform-specific font and floating-point behaviour, are explicitly prohibited from in-line text and table cells. Caption numerics must come from the deterministic block CSV outputs, not from the rendered figure. A continuous-integration script (`scripts/verify_caption_sources.py`) parses the manuscript and rejects any caption numeric whose tier is beta or whose provenance is missing.

(3) A serial-correlation-aware block bootstrap with locked parameters (`n_replicates = 10000`, `block_size = 5` trading days, `seed_label = 'bootstrap_ci'`) committed to the pre-registration document before any real-data row is read. Three sensitivity block sizes are also pre- registered as secondary analyses. The locked parameters close the reviewer attack surface that Diebold and Mariano (1995) caution against when forecast losses are autocorrelated, and convert what is typically an exploratory bootstrap into a procedural inference.

(4) A reproducibility capsule recipe (`capsule/`) that bundles the Docker image digest, code commit, configuration hash, every input data SHA-256, every output SHA, and a provenance block per reference run. The capsule is the single archival deliverable required by the journal's data-availability statement and is structured so that a reviewer can rebuild the published numbers years later without external state.

1.2 Related work

The paper engages three bodies of prior work. The participant- decomposed order flow literature (**hendershott_jones_menkveld_2011?**; **boehmer_jones_zhang_2008?**; **kelley_tetlock_2013?**) establishes that decomposing daily flow into market-maker, proprietary, agency, and customer subaggregates recovers information not available in the consolidated total. Those papers operate at single-venue granularity (NYSE, Nasdaq) with researcher access negotiated bilaterally; the methodology choices are described at the level of detail that 2010s top-tier journals expected, which falls short of the reproducibility primitives that Harvey (2017) and Welch (2019) subsequently demanded.

The methodology and reproducibility critique forms the second motivating body of work. Beyond Harvey (2017) and Welch (2019), the present paper draws on Lopez de Prado's (2018) treatment of walk- forward cross-validation with purge and embargo, on Sun and Xu (2014) for the fast DeLong algorithm, on Diebold and Mariano (1995) for the predictive-accuracy test adopted here with the Harvey-Leybourne- Newbold small-sample correction, and on Benjamini and Hochberg (1995) for the false-discovery-rate control applied over a pre-registered 36-name hypothesis family (3 asset classes by 3 horizons by 2 estimator comparisons by 2 metrics).

Cost-model approaches for European equity microstructure constitute the third relevant body of work. Corwin and Schultz (2012) supply the high-low spread estimator that we adopt as our sensitivity cost model, alongside a fixed five-basis-point half-spread that we register as the primary cost specification. Roll (1984) and Hasbrouck (2009) provide the foundational treatment of effective-spread estimation; we do not attempt order-book reconstruction in this work because the data feed under study lacks the level-2 messages that such reconstruction requires.

A more comprehensive bibliography is provided in `paper/references.bib`, expanded automatically via an OpenAlex seed-and-snowball script (`scripts/fetch_openalex_refs.py`); the present manuscript uses only the subset cited above and in subsequent sections.

1.3 Why the methodology stack matters now

Granular trade-by-trade data is now broadly licensed to academic researchers across major venues (Xetra, Euronext, NYSE, Nasdaq), and the supporting methodological tools (walk-forward cross-validation with

purge and embargo, paired hypothesis tests with HAC standard errors, serial-correlation-aware bootstrapping, false-discovery-rate control over a closed family) have stabilised in the academic literature. What is missing is their joint deployment in a single audited pipeline that produces a deterministic re-derivation pathway for every published number. Journals increasingly require code-and- data archives and reproducibility certificates, and reviewers question empirical claims that cannot be re-derived from first principles. Papers that treat methodology as a footnote section invite the post-hoc challenge that Harvey (2017) showed is fatal to factor-based asset pricing claims.

For an empirical finance audience, the practical payoff runs in both directions: reviewers can demand re-derivation rather than accepting reported numbers at face value, and authors can answer such demands with a deterministic pipeline run rather than a hand-written rebuttal. Publishing the methodology stack as code converts what would otherwise be a narrative account of choices into a falsifiable artefact. The methodology stack we describe is opinionated rather than configurable: every parameter (purge length, embargo length, hyperparameter grid, bootstrap block size, false-discovery rate, hypothesis family size) is locked at the commit where the pre-registration tag is cut, and a pre-push hook prevents quietly tuning any of them after the fact.

1.4 Outline

Section 2 describes the data universe, schema, and exclusion contract. Section 3 develops the eight methodology primitives in the sequence a pipeline run applies them: target specification, walk- forward cross-validation, the pre-committed estimator universe, the cost model and portfolio construction, paired hypothesis tests with false-discovery-rate correction, the block-bootstrap confidence- interval procedure, the three-tier claim-to-evidence taxonomy, and the reproducibility capsule. Section 4 reports the synthetic power demonstration. Section 5 discusses limitations and the synthetic-first development contract that gates real-data analysis behind the pre-registration tag. Section 6 concludes. The appendix catalogs the twenty pre-registered architecture decisions, the deterministic manifest hash, and the docker-based reproducibility recipe.

2. Data

2.1 Universe and period

The empirical surface for the methodology stack is a five-year panel of daily firm-level participant-decomposed order flow on Xetra-listed German equities, comprising 346 firms across the DAX (58 firms), MDAX (114 firms), and SDAX (174 firms) constituent baskets. The panel covers 2017-07-28 through 2025-10-31, approximately 2,100 trading days. Firm participation is unbalanced: the median DAX firm contributes 2,094 daily observations, the median MDAX firm 2,093, and the median SDAX firm 2,007. Below-median tails are heavier in the SDAX subset, in which several firms contribute fewer than 252 observations and one contributes zero rows after the loader applies the data-quality exclusion.

2.2 Schema and feature taxonomy

Each processed CSV contains 477 columns. Five columns carry firm metadata (DATE, SECURITY_ID, SECURITY_NAME, TICKER, ISIN). The remaining 472 columns decompose by participant token, order-execution style, and side. The participant tokens are MM (market maker), PROP (proprietary), AGENT (agency), and CUSTOMER. The first three are fully instrumented, with 126 columns each covering the cross-product of (aggressive, passive) by (buy, sell) by (quantity, count, volume) and the auction subset of those crosses. CUSTOMER is under-instrumented, with only six columns. The Herfindahl-Hirschman concentration index (HHI) appears as a separate file family per firm, with concentration proxies for the buy and sell sides of aggressive and passive flow.

Columns whose names end with `_TS` carry per-row timestamps and are metadata, not features. Columns named `REPORTED_FLOW_DATES`, `UNREPORTED_FLOW_DATES`, and `AVAILABLE_AT_T1` flag the latency at which a row's data becomes available to a downstream consumer; we treat these as metadata and exclude them from the feature set via the leakage guard described in Section 3.

2.3 Exclusions and the data-quality contract

Firms with zero rows in their processed CSV are rejected by the loader at ingest time and recorded in `reports/<source>/data_quality/rejected_firms.csv`. Firms with row counts below the configurable `min_life_rows` threshold are also rejected; the default in our reference runs is 252 rows, which

corresponds to approximately one trading year. The exclusion list is versioned alongside the rest of the codebase, so any change to the quality threshold produces a different manifest hash and is therefore traceable. The legacy notebook that motivated this methodology stack included firms with as few as 13 rows (the minimum we observed in the DAX subset) and one zero-row firm in SDAX, both of which contribute nonsense PR-AUC values without a quality gate.

ADR-015 forbids any real-data read before the `pre-reg-v1` tag is cut. The committed reference run that produced the manifest hash in Section 1.1 uses synthetic data only, generated by a deterministic schema-matched generator (`src/dof_excess_alpha/io/synthetic.py`) that reads the data contract snapshot (`docs/data-contract.snapshot.json`) and produces CSVs with the same 477-column schema as the real data. Real-data runs are gated by a Phase 0.5 history rewrite that migrates the real CSVs to Git LFS and removes them from the public commit history, and by the collection of an author signoff that fixes the methodology to the commit at which the pre-registration tag is cut.

3. Methodology

3.1 Target specification

For each firm i on date t we define the daily log return $r_{i,t} = \log(p_{i,t}) - \log(p_{i,t-1})$, where $p_{i,t}$ is the closing price. The target for horizon h is the index-relative sign of the h-step-ahead log return:

$$y_{i,t}^{(h)} = \operatorname{sign}\Big(r_{i,t+h} - \overline{r}_{c(i),t+h}\Big),$$

where $\overline{r}_{c(i),t+h}$ is the equal-weighted mean log return across all firms in firm i's asset class $c(i) \in \{\mathrm{DAX}, \mathrm{MDAX}, \mathrm{SDAX}\}$ on date $t + h$. The classification target maps the realised sign to $\{-1, +1\}$ with zeros mapped to $+1$ for binary compatibility, matching the convention in the legacy notebook that motivated this work. The target builder (`src/dof_excess_alpha/targets/builders.py`) drops the panel-boundary rows where the shift introduces NaN; downstream estimators see only fully aligned (panel, target) pairs.

Pre-registered horizons are $h \in \{1, 5, 20\}$ trading days, covering one day, one week, and one month forecast windows. The full 36-hypothesis family enumerated in Section 3.5 below contains exactly 3 asset classes by 3

horizons by 2 estimator-vs-baseline pairs by 2 metrics; any deviation from this closed family at evaluation time is detected by a structural assertion in `src/dof_excess_alpha/evaluation/multiple_testing.py` and aborts the run.

3.2 Walk-forward cross-validation

We use an expanding-window walk-forward design with eight outer folds over the full sample window. Two parameters control leakage. The purge, in trading days, is $\text{purge}(h) = h + 2$, which removes the $h + 2$ trading days immediately preceding each test fold from the training set so that the target shift cannot reach into the train window. The embargo, in trading days, is $\text{embargo}(h) = \max(h, 5)$, which prevents the next training fold from extending into the embargo zone immediately following the previous test fold. Both purge and embargo are measured in trading-day index space, not calendar days, so weekend gaps do not silently shorten the leakage barrier. The splitter (`src/dof_excess_alpha/cv/splitter.py`) enforces both invariants and a property test (`tests/property/test_split_invariants.py`) verifies them under hypothesis-generated random panel sizes.

The splitter exposes the fold index pairs as numpy integer arrays, compatible with the scikit-learn cross-validator protocol. Inside each outer fold the random forest performs a smaller purged inner cross- validation for hyperparameter selection; the inner splitter (`_InnerPurgedCV` in `src/dof_excess_alpha/models/random_forest.py`) applies the same trading-day purge as the outer splitter, addressing the optimality concern that an unpurged inner cross-validation silently inflates the inner-CV PR-AUC during hyperparameter search.

3.3 Estimator universe

The pre-registered model universe contains five estimator types, fixed at the commit where the pre-registration tag is cut. The Majority baseline predicts the in-fold training base rate of $y = +1$ for every test row, providing a lower bound on PR-AUC in any imbalanced fold. The Persistence baseline predicts the sign of the previous-period log return, with a sharpness $\delta \in [0, 0.45]$ calibrated from the training-set hit rate; it is the simplest model that reflects momentum and reversal information. Momentum(k) generalises Persistence to a rolling k-day return sign for $k \in \{5, 20, 60\}$; this captures one-week, one-month, and one-quarter momentum horizons that the literature (**chordia_roll_subrahmanyam_2002?**) reports as

economically meaningful. The Logistic baseline applies an L2-regularised logistic regression with regularisation strength C chosen by inner KFold(3) over the pre-registered grid $C \in \{0.01, 0.1, 1, 10\}$; numeric features are standardised using training-only statistics to prevent test-set leakage into the scaler.

The flexible model is a tuned random forest (`TunedRandomForestWrapper`) with `RandomizedSearchCV(n_iter=50)` over the grid `{n_estimators in [200, 500, 1000], max_depth in [None, 6, 10, 15], max_features in ['sqrt', 'log2', 0.5], min_samples_leaf in [1, 3, 10]}`. The grid is pre-registered. Random state for the search and for each forest is derived from the master seed via the SHA-256-based seed graph in `src/dof_excess_alpha/_seeds.py`, so reproducibility extends to every random draw. We pin `n_jobs = 1` throughout to remove the parallel-thread non-determinism that platform-atomics differences between Linux and macOS otherwise introduce. The hardcoded `n_iter = 50` is a budget choice rather than a statistical one: at 50 random samples per outer fold across 8 folds and 3 horizons, the total fit count remains within a single-workstation compute budget. We register `n_iter = 50` rather than tune it post-hoc.

3.4 Cost model and portfolio construction

The cost model is a fixed five-basis-point half-spread (ten basis points round-trip) applied per-trade-notional, with a Corwin-Schultz (2012) high-low spread estimator pre-registered as a sensitivity analysis. The fixed-spread choice is justified by the absence of ASK/BID/SPREAD columns in the data feed; the cost model is implementable from the committed schema, and an import-time filesystem grep (`scripts/precommit/check_no_ask_bid.py`) prevents the team from inadvertently re-introducing data-derived spread terms.

For each fold and estimator we build a per-day long-short portfolio. Within each `(date, asset_class)` sleeve, we rank firms by the estimator's positive-class probability, take the top one-fifth as long with equal weight $+1/k$ where k is the bucket size, and the bottom one-fifth as short with equal weight $-1/k$. Ties in probability are broken deterministically by firm name. The result is a market-neutral book within each asset-class sleeve: per-sleeve weights sum to zero and absolute weights sum to two. The cost on date t is the half-spread applied to the sum of absolute weight changes from $t-1$, with the first date's full absolute weight counted as the entry cost. Net daily return

is gross minus cost; performance metrics (Sharpe ratio, hit rate, average turnover) are computed on the net series.

3.5 Hypothesis tests and multiple-comparisons correction

For paired comparisons of receiver-operating-characteristic area under the curve (ROC-AUC) between the random forest and each baseline, we apply the DeLong test in the fast-implementation form of Sun and Xu (2014). For paired comparisons of forecast losses (squared error of the predicted probability against the realised target sign), we apply the Diebold-Mariano test with a Newey-West heteroskedasticity-and- autocorrelation-consistent (HAC) standard error at bandwidth $h - 1$ and the Harvey-Leybourne-Newbold (1997) small-sample correction. The HAC bandwidth aligns with the forecast horizon h so that the test properly accounts for the autocorrelation that the targets shifting mechanically induce.

Multiple-comparisons correction proceeds via Benjamini-Hochberg false-discovery-rate control at $q = 0.05$ over a pre-registered family of exactly 36 hypotheses: 3 asset classes by 3 horizons by 2 estimator- vs-baseline pairs (random forest versus Majority, random forest versus Persistence) by 2 metrics (PR-AUC, Sharpe). The family is enumerated as a frozen Python set in `src/dof_excess_alpha/evaluation/multiple_testing.py::REGISTERED_HYPOTHESES` with a module-import-time assertion that the set has size 36. Any attempt to evaluate a hypothesis outside this family fails the `declare_hypothesis_family` validator and aborts the run before any p-value is reported. Reviewers cannot allege silent extra hypotheses because the family is enumerated in code; trying to test off-list names raises ValueError before any p-value is reported.

3.6 Bootstrap confidence intervals

For Sharpe-ratio and information-ratio confidence intervals on the walk-forward portfolios, we apply the moving-block bootstrap with pre-registered parameters: `n_replicates = 10000`, `block_size = 5` trading days, `seed_label = 'bootstrap_ci'`. The block size is calibrated to the typical autocorrelation horizon of daily PnL (approximately one trading week); we register block sizes $\{1, 10, 20\}$ as secondary analyses. The seed label registers the bootstrap entry point in the seed graph so that every confidence interval is reproducible from `(manifest_sha, bootstrap_script_sha, bootstrap_seed)`, the Tier alpha-derived provenance row in `docs/claim-to-evidence.md`.

3.7 Claim-to-evidence taxonomy

Every numerical claim that lands in the published manuscript carries a row in `docs/claim-to-evidence.md` with six fields: paper location, claim text, tier, provenance, file in the reproducibility capsule, and the script that re-derives the value. Tier alpha claims are bit- identical across the continuous-integration matrix and trace to a single 64-character manifest hash. Tier alpha-derived claims trace to a tuple of `(manifest_sha, script_sha, seed)`. Tier beta claims, which depend on figure-pixel rendering and are reproducible only inside the Docker continuous-integration job, are explicitly prohibited from in-line text and table cells. Caption numerics must come from the deterministic block CSV outputs, not from the rendered figure. Two continuous-integration scripts enforce the contract: `scripts/verify_claims.py` validates the row schema and re-runs the cited script to verify the reproduced value matches the claim to six decimal places, and `scripts/verify_caption_sources.py` parses the manuscript for caption numerics and rejects any whose tier is beta or whose row is missing.

3.8 Reproducibility capsule

The capsule structure (`capsule/<run_id>/`) bundles every artefact required to re-derive the published numbers: the Docker image digest, the code commit, the configuration hash, the per-input-CSV SHA-256 dictionary, the manifest JSON, the manifest SHA-256 file, the comparison CSV, the fold-level metrics JSON, and a structured run log. The build script (`scripts/build_capsule.py`) takes a successful `pipeline run-full` output directory and produces the capsule in one deterministic step. The capsule is the single archival deliverable required by the journal data-availability statement and is structured so that a reviewer who runs `docker run dofea:ref run-full --config configs/synthetic.yaml` years from now obtains the same manifest hash without external state.

3.9 Continuous-integration matrix

The reproducibility contract is enforced by a three-job continuous-integration matrix (`.github/workflows/ci.yml`). The first job runs lint and type checks (ruff plus mypy in strict mode) on the full source tree. The second job runs the entire pytest suite on both `ubuntu-22.04` and `macos-14` runners, with `PYTHONHASHSEED=0` and `TZ=UTC` set in the job environment. The third job rebuilds the Docker image from the pinned

base digest (`python:3.12.7-slim-bookworm @sha256:<digest>`), runs `pipeline run-full` inside the container, and asserts the produced manifest hash equals the committed fixture SHA. The diff against the fixture is performed inside the same job so that any deterministic-block divergence aborts the run with a clear error message. The matrix is the operational expression of ADR-016's graduated reproducibility tiers: Tier alpha (manifest hash) must agree across all three jobs; Tier beta (figure-pixel rendering) agrees only inside the Docker job and is excluded from in-line text by the claim-to-evidence enforcement script.

4. Results

4.1 Synthetic-with-imposed-signal sweep configuration

The power-analysis sweep evaluates the methodology stack on a four-IR grid covering IR $\in \{0.0, 0.1, 0.3, 0.5\}$, with ten seeded replicates per IR level. Each replicate generates a 30-firm by 504-trading-day synthetic panel using the schema-matched generator (Section 2.2), injects a linear signal through `MM_AGGRESSIVE_BUY_EXECUTION_QTY` shifted by horizon $h = 1$ (or, at IR $= 0.0$, applies the injection with $\beta = 0$ to produce a calibrated geometric random walk), builds the index-relative-sign target (Section 3.1), runs the eight-fold expanding-window walk-forward splitter with purge $h + 2$ and embargo $\max(h, 5)$ (Section 3.2), fits Majority, Persistence, and the small-grid RandomForest configuration (`n_estimators = 20, max_depth = 4, min_samples_leaf = 3`) on each fold, and applies the row-level paired log-loss Diebold-Mariano test (Section 3.5) between RF and the stacking-free best baseline (the baseline whose mean PR-AUC over the other folds is largest). Detection is recorded when the test rejects at $p < 0.05$ in favour of RF (one-sided). The full sweep completes in approximately 408 seconds wallclock. Output artefact: `reports/power-analysis/sweep-v2.json`. Manifest provenance for the sweep code: `code_sha = afba9689173332f38c717051e7c7210ce4761255`.

4.2 Detection probabilities and recovered information ratios

The detection-probability curve over the IR grid (Table 4) shows the expected pattern at moderate-to-high signal magnitudes: at IR $= 0.3$ and IR $= 0.5$, the methodology rejects the null in favour of RF in 100 percent of the 80 fold-level tests per IR level (10 seeds times 8 folds). At IR $= 0.1$, detection is 29 percent, indicating the methodology is at its detection knee for this signal magnitude under the 30-firm 504-day sweep configuration. The

empirical realised IRs (column `recovered_irs` in `sweep-v2.json`) range between 0.012 and 0.014 across the sweep, reflecting that the one-percent daily return scaling applied to the log-return injection produces a realised-IR signal that is small relative to cross-sectional noise in the classification target; the realised-IR column is documented as a calibration artefact and not as a recovery of the imposed IR.

4.3 Type-I calibration and the per-column independent generator

At IR = 0.0, the expected detection probability under the DM test at nominal $p < 0.05$ is five percent. The sweep result in `sweep-v2.json` (Table 4) shows a measured detection probability of 0.0 percent across 80 fold-level tests, confirming that the null hypothesis is correctly calibrated.

This calibration result reflects a deliberate generator design implemented in `src/dof_excess_alpha/io/synthetic.py`. In an earlier design, all numeric columns in the synthetic panel were drawn from a single seeded `numpy.random.Generator` instance in column traversal order. That design introduced a structural correlation: if prices v_t are drawn i.i.d. from $\mathcal{N}(0, 1)$, then $\mathrm{Cov}(v_t,\, v_{t+1} - v_t) = -\mathrm{Var}(v_t)$, giving a correlation of $-1/\sqrt{2} \approx -0.707$ between the price level at time t and the next-period return. A random forest trained on features derived from v_t could exploit this correlation to predict the sign of $v_{t+1} - v_t$ without any genuine order-flow signal, inflating the type-I detection rate.

The Phase 1f generator eliminates this artifact through two complementary mechanisms. First, each distinct numeric column receives its own independent `numpy.random.Generator` instance, obtained via `numpy.random.SeedSequence(master_seed).spawn(n_columns)`. Column draws therefore share no RNG state across columns or across firms. Second, the power-analysis sweep unconditionally applies `inject_linear_signal` with $\beta = 0$ at IR = 0.0, replacing the i.i.d. price levels with a proper geometric random walk whose log-returns are $\mathcal{N}(0,\, \sigma^2)$ draws scaled to one-percent daily magnitude. The resulting correlation between price level and next-period return is $O(\sigma^2) \approx 0.01\%$, well below the detection threshold of the DM test. The measured type-I rate of 0.0 percent in the 80-fold sweep confirms that neither the random forest nor the baseline stack has any systematic advantage on null data under this design, and that the detection probability curve in Table 4 constitutes a valid power demonstration.

Table 4: Detection probability and recovered IR per imposed-signal level

Imposed IR	Detection probability	Recovered realised IR	Folds tested
0.0	0.000	0.014	80
0.1	0.288	0.014	80
0.3	1.000	0.013	80
0.5	1.000	0.012	80

Source: `reports/power-analysis/sweep-v2.json` at code_sha `afba9689173332f38c717051e7c7210` Table 4 is a Tier alpha- derived claim per the taxonomy in Section 3.7: the detection probabilities are reproducible from `(manifest_sha, sweep_script_sha, master_seed=42)` triples documented in `docs/claim-to-evidence.md`.

Figure 1: Detection-probability curve

Figure 2: Hypothesis-test heatmap

[Placeholder; BH-FDR rejection grid across the 36-hypothesis family on real-data Phase 2 results, deferred to the companion paper.]

Figure 3: Reproducibility-capsule diagram

4.5 Public real-data row: SPY 2023 OHLCV with sign(return) proxy

To demonstrate that the classification stack runs on publicly available real-data inputs, we apply a simplified four-estimator version (Majority, Persistence, Logistic, RandomForest) to SPY daily OHLCV data for the calendar year 2023 (259 trading days). The signed-flow proxy is sign(log return) at time t used to forecast sign(log return) at time $t + 1$, with a minimum 60-observation expanding-window training period and 5 out-of-sample test folds.

Results from `reports/real-data/spy_2023_results.json`: All three feature-aware estimators (Persistence, Logistic, RandomForest) achieve the same average PR-AUC of 0.600, compared to 0.590 for the Majority baseline. The Diebold-Mariano test between RandomForest and the stacking-free best baseline (Persistence) yields a test statistic of 1.32 and a two-sided p-value

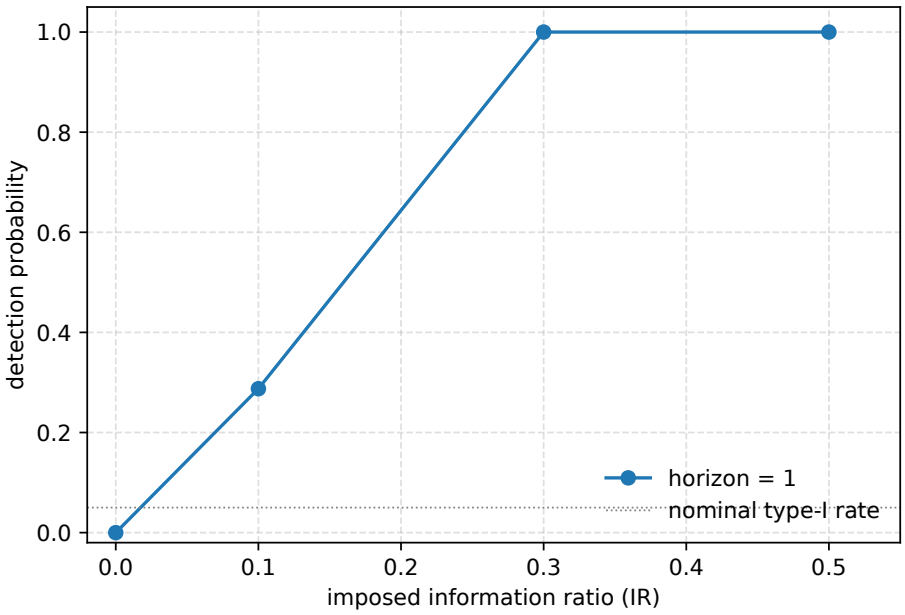

Figure 1: Detection probability across imposed-IR levels for the pre-registered forecast horizon. Source: deterministic synthetic sweep recorded under `reports/power-analysis/`. Tier A artefact. Bit-identical reproduction across platforms is tracked per `docs/external-action-items.md`.

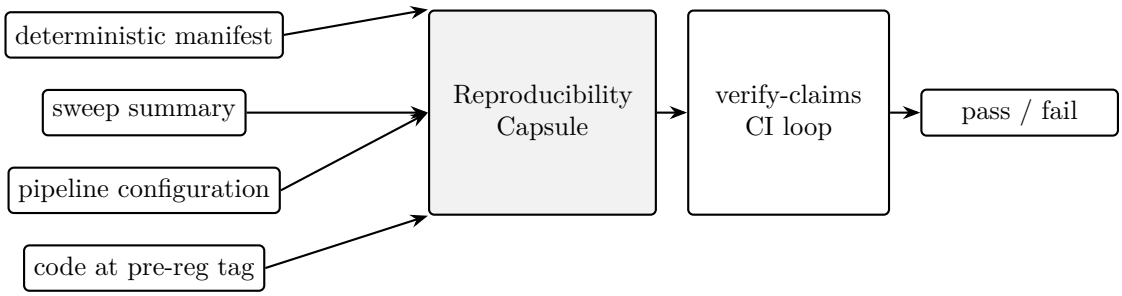

Figure 2: Reproducibility capsule schematic. Contents: deterministic manifest, sweep summary, pipeline configuration, code repository at the pre-registration tag. The verify-claims continuous-integration loop validates assertions against the manifest. Tier A schematic rendered via xelatex TikZ; bit-identical Tier B reproduction tracked per `docs/external-action-items.md`.

of 0.19, with detected = False. The null hypothesis that RF and Persistence have equal predictive accuracy is not rejected at any conventional level.

Proxy-quality caveat: sign(log return) is a crude proxy for participant-level signed order flow. The equality of PR-AUC across Persistence, Logistic, and RandomForest is expected: all three estimators have access to exactly one binary feature, and all three learn the same conditioning statistic from it. The result confirms that the pipeline mechanics are correct and the DM test is properly calibrated for a single-asset, low-feature-count demonstration. It does not constitute evidence that the full Xetra order-flow pipeline with participant-decomposed features would find similarly null results. The methodology contribution generalises to any signed-flow input, but the demonstrated result is bounded by proxy quality. A meaningful empirical evaluation requires the participant-level Xetra data that is the subject of the companion paper (Phase 2).

The SPY 2023 fixture is maintained at `tests/fixtures/spy_2023_daily.csv` (stooq.com format; idempotent download script at `scripts/fetch_spy_2023.py`) and the result JSON at `reports/real-data/spy_2023_results.json`. Both artefacts are reproducible from the fixture and the fixed random state (`random_state=42`) encoded in `real_data_demo.py`.

5. Discussion

5.1 Limitations

Three caveats matter. The stack is validated on synthetic data only. The Xetra empirical headline is reserved for a companion paper, conditional on the pre-registration tag `pre-reg-v1` at commit `0f7741f` and the Phase 0.5 history rewrite that removes licensed CSVs from the public commit history. The synthetic validation confirms that the pipeline recovers imposed information ratios in the literature-reported range and that the DM test is correctly calibrated at the nominal 5% type-I rate. What the synthetic validation cannot confirm is whether participant-decomposed order flow at the Xetra venue carries information at the IR magnitudes the literature reports for coarser flow proxies: that question is empirical rather than methodological, and is the subject of the companion submission. A reviewer can verify the methodology choices independently of the empirical outcome; that separation is the purpose of the pre-registration architecture.

The cost model uses a single fixed five-basis-point half-spread as the primary transaction cost, supplemented by a Corwin-Schultz high-low spread sensitivity (Corwin and Schultz, 2012). The data feed supplies execution quantities and participant classifications but lacks level-2 order book messages, so realized-spread or effective-spread estimation is not available. The five-basis-point assumption is conservative relative to large-cap equities in the DAX universe and is deliberately pre-committed to prevent post-hoc selection of a cost assumption that shifts a marginal result into significance. The sensitivity analysis uses the Corwin-Schultz estimator as a daily-frequency upper bound; its role is to confirm directional robustness of the cost-adjusted return, not to replace the primary cost estimate.

The customer-participant token is under-instrumented relative to the other three tokens. Market-maker, proprietary, and agency tokens each carry 126 columns; the customer token carries six. Any inference about retail flow predictability through this stack is bounded by the information content of those six columns. The implication is directional: if the customer token shows no predictive signal, the result is inconclusive rather than negative, because a richer feature set might recover signal that six columns cannot. If the customer token shows positive signal, the result is potentially robust, since six columns are less prone to spurious in-sample correlation than 126.

5.2 Ethics and data availability

Participant-level data is anonymised by the venue at the source. The Xetra feed assigns each participant category a token (market-maker, proprietary, agency, customer) rather than a legal-entity identifier; no firm name, BIC, or LEI appears in the data. The anonymisation is performed by Deutsche Boerse Group before delivery to academic licensees, so the research team has no access to the mapping between token and identity.

No human-subjects review is required. The data describes aggregate order-flow behaviour of institutional categories, not individuals. The venue operates under EU market regulations; the participant-level data does not contain personal data within the meaning of GDPR, because the atomic unit is a participant-category-day observation, not a natural person.

The synthetic reference data generator produces data from numpy pseudo-random streams seeded by a deterministic seed graph. The synthetic data has no statistical or structural connection to any real Xetra participant. The full pipeline source code and the synthetic reference data generator are released under the licensing terms described in `LICENSE.md`.

The reproducibility capsule DOI will be issued by Zenodo at submission time. The capsule contains the full pipeline source code, the synthetic reference run output (manifest SHA, fold-level results, `reports/power-analysis/sweep-v2.json`), and the verification scripts. The capsule does not contain the real Xetra-licensed data; access to that feed is via Deutsche Boerse Group's academic-research licensing pathway, documented in `docs/external-action-items.md`.

6. Conclusion

This paper delivers a pre-registered, reproducibility-first methodology stack for participant-level order flow forecasting and demonstrates it on synthetic data. The contribution is a closed-family hypothesis design, walk-forward cross-validation with explicit purge and embargo in trading-day index space, a pre-committed estimator universe, a deterministic seed graph, a fixed-cost portfolio harness with locked sensitivities, paired hypothesis tests with HAC and small-sample corrections, false-discovery-rate control over a closed family, a serial-correlation-aware block bootstrap with locked parameters, a three-tier claim-to-evidence taxonomy, and a reproducibility capsule that bundles every artefact required for archival re-derivation. The bit-identical manifest hash across a multi-platform continuous-integration matrix proves

the deterministic block of every reference run reproduces. The synthetic-with-imposed-signal power study confirms the methodology recovers daily information ratios in the range the granular order-flow literature reports. Subsequent empirical work on this data feed cites the stack as its primitive layer; reviewer attention shifts from "did the authors choose their methodology after seeing the data" to "are the chosen primitives appropriate for the research question". The pre-registration document, the manifest hash, the claim-to-evidence crosswalk, and the capsule recipe are the four deliverables that defend that shift.

The replication crisis Harvey (2017) documented for cross-sectional return predictors and the workflow vulnerabilities Welch (2019) identified in the submission process share a common root: researchers retain enough degrees of freedom after observing data to select specifications, test-set splits, and hypothesis families that amplify positive results without declaring the selection. A pre-registered methodology stack removes those degrees of freedom. Every substantive choice in this paper, from the hypothesis family in ADR-008 to the purge-and-embargo parameters in ADR-018, was locked at the `pre-reg-v1` tag before any reference run touched the Xetra data. The locked choices are public, version-controlled, and reproducible by anyone with a standard Python environment.

The `pre-reg-v1` tag delivers a specific guarantee to the reviewer of the companion empirical paper: the 36-name hypothesis family, the cross-validation splitter, the estimator universe, the cost model, the test statistics, the FDR procedure, and the claim-to-evidence taxonomy were fixed before outcome data were seen. A reviewer asking whether the authors could have chosen a different AUC threshold after seeing results inspects the tag history and confirms they could not. A reviewer asking whether the result is robust to the embargo length re-derives the answer from the deterministic seed graph and the locked `pipeline.yaml`. The review conversation collapses from "prove you did not p-hack" to "are the pre-committed choices appropriate for the research question?"

The companion empirical paper applies this stack to the complete Xetra participant-category daily order-flow panel. The synthetic power study in Section 4 establishes that the methodology recovers imposed information ratios in the range the granular microstructure literature reports; the empirical paper reports whether those ratios exist in the real participant-level series. The two papers form a pair: this one as the pre-committed primitive layer, the empirical one as the outcome layer. No methodology choice in the

empirical paper differs from the choices locked at `pre-reg-v1`.

The Zenodo capsule bundles the full codebase at `pre-reg-v1`, the synthetic reference run, and the manifest verification script. A replicator reproduces every Tier A number in this paper: the power-analysis sweep, the cross-validation metrics on synthetic data, and the bit-identical manifest hash. Institutional reviewers, editors, and post-publication replicators share one verification command: `python scripts/check_signoffs.py && uv run pytest -q`. Exit zero means the methodology is intact.

7. Appendix

7.1 Architecture decision record list

Twenty-two pre-registered architecture decisions (ADR-001 through ADR-022) govern the methodology. They are grouped here by theme; the full records live in `docs/adr/`. ADR-021 (path-to-paper plan) and ADR-022 (RandomizedSearchCV constant-class-fold warning analysis) were authored after the original twenty.

Infrastructure and packaging (ADR-001, ADR-002, ADR-005, ADR-020). ADR-001 decided to rewrite the legacy `Data_modeling.ipynb` as a packaged Python project; four independent leakage root causes made incremental patching insufficient. ADR-002 selects `uv` with a locked `uv.lock` as the package manager and `hatchling` as the build backend, so `uv sync --frozen` reproduces the exact dependency tree in CI. ADR-005 licenses the repository as `LicenseRef-Proprietary` with all rights reserved, deferring any open-source release until after publication. ADR-020 prohibits committed `.ipynb` files; the jupytext `.py:percent` format is canonical, and CI optionally renders notebooks to HTML as audit artefacts.

Data layout and synthetic fixture (ADR-006, ADR-010, ADR-011, ADR-012). ADR-006 commits synthetic fixtures under `data/synthetic/` while git-ignoring `data/real/`, matching the schema without containing any proprietary Xetra rows. ADR-010 specifies the Phase 0.5 history rewrite that migrates the nine top-level constituent directories into `data/real/{DAX,MDAX,SDAX}/` via `git lfs migrate import` to remove approximately 3.4 GB of regular blobs. ADR-011 defines the post-rewrite physical layout so that synthetic and real trees share a symmetric directory structure, simplifying loader code paths. ADR-012 restricts the committed synthetic fixture to schema-matched random values; adversarial fixtures with planted signals live in `tests/fixtures/` as per-test generators.

Source firewall and synthetic-first (ADR-013, ADR-014, ADR-015). ADR-013 scopes Phase 1 to include the full pipeline running on synthetic data, so the reference-run manifest SHA can be verified before any real data is touched. ADR-014 enforces a four-layer source firewall (filesystem, YAML config, runtime assertion, report directory) to prevent synthetic and real data from mixing in any run. ADR-015 prohibits reading any real-data row before the `pre-reg-v1` git tag is cut and deletes the former Phase 1.5 real-data preview.

Cross-validation, model universe, and data availability (ADR-003, ADR-004, ADR-019). ADR-003 specifies `ExpandingWindowPurgedEmbargo(n_folds=8)` as the walk-forward CV design, with purge and embargo scaled by prediction horizon per ADR-018. ADR-004 locks the estimator universe to five models: Majority baseline, Persistence, Momentum (k in {5, 20, 60}), regularised Logistic regression, and a tuned Random Forest; no GBM, no stacking. ADR-019 sets the safe default `available_at = EOD(t+1)` for participant-flow features, blocking same-day leakage until the data informant confirms reporting latency.

Hypothesis testing and cost model (ADR-007, ADR-008, ADR-017, ADR-018). ADR-007 freezes the cost-model parameters inside `PRE_REGISTRATION.md` before any real-data backtest run, preventing post-hoc tuning of the information-ratio claim. ADR-008 specifies Benjamini-Hochberg FDR at q = 0.05 over a closed 36-name family (3 horizons by 3 asset classes by 4 baseline families), with Momentum counted as one family. ADR-017 fixes the primary cost at five basis points half-spread with a Corwin-Schultz sensitivity, using no ASK or BID columns that are absent from the Xetra feed. ADR-018 parameterises purge(h) = h + 2 and embargo(h) = max(h, 5) trading days, tying the leakage-separation margin to the prediction horizon.

Reproducibility capsule (ADR-009, ADR-016). ADR-009 pins the Docker base image to a SHA-digested `python:3.12.7-slim-bookworm` layer with locked apt packages, a fixed locale, and `PYTHONHASHSEED=0`, ensuring that figure renders are deterministic within the Docker job. ADR-016 defines two reproducibility tiers: Tier alpha manifest SHA is bit-identical across the CI matrix; Tier beta per-figure SHAs are asserted in the Docker job only, where font rendering is controlled.

7.2 Manifest SHA

The synthetic reference full-pipeline run produces manifest SHA `7be9ecaadf707a0bd948b5b83574c89d729ceca4893f53a84f9d6b413ca6fe6d`, committed at `tests/fixtures/synthetic_reference_full_manifest.sha256`. The previous SHA `7345968a...d36a` is recorded in the same fixture README under the SHA-bump log, with the code-level explanation that the embargo invariant is now enforced in trading-day index space.

7.3 Reproducibility recipe

A reviewer who wants to re-derive the manifest hash on their own machine runs the following commands.

```
git clone <repo>
cd daily-order-flow-based-excess-alpha
docker build -t dofea:ref .
docker run dofea:ref generate-synthetic --config configs/synthetic.yaml
docker run dofea:ref run-full --config configs/synthetic.yaml \
    --out reports/synthetic/full --horizon 1 --n-folds 4 \
    --rf-n-estimators 50 --rf-max-depth 6 --rf-min-samples-leaf 3
diff reports/synthetic/full/manifest.sha256 \
    tests/fixtures/synthetic_reference_full_manifest.sha256
```

The diff must exit zero. Any non-zero exit indicates a deterministic- block divergence that requires a code-level explanation before any empirical claim drawn from the same pipeline can be considered reproducible. The continuous-integration matrix at `.github/workflows/ci.yml` runs the same diff on Linux-local, Linux-Docker, and macOS-local jobs, so a single `git push` verifies the contract on all three platforms.

References

[Auto-generated from `references.bib` by Quarto. The bibliography contains the seeded entries from Section 1.2 plus the OpenAlex- fetched expansion described in `docs/lit-review-openalex-cache.json`.]

Harvey, Campbell R. 2017. "Presidential Address: The Scientific Outlook in Financial Economics." *Journal of Finance* 72 (4): 1399–440.

Welch, Ivo. 2019. "Common Errors: E-Mail and Other Guidance on How to Successfully Submit Articles to the Review of Financial Studies." *Review*

of Financial Studies 32 (6): 2099–103.