From Natural-Language Hypotheses to Verified Data-Mining Discoveries

Phat T. Tran-Truong[1], Son X. Ha[2], Tung Q. Nguyen[3], Xuan-Bach Le[1,*]
[1]Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology (HCMUT), VNU-HCM, Vietnam
[2]The Business School, RMIT University, Ho Chi Minh City, Vietnam [3]FPT University Can Tho Campus, Vietnam
phatttt@hcmut.edu.vn, lexuanbach@hcmut.edu.vn, ha.son@rmit.edu.vn, NguyenQuangTung0902@gmail.com
ORCID: 0000-0003-3199-6333, 0000-0002-5336-0566, 0009-0003-5046-5535, 0009-0003-6848-5403 *Corresponding author

Abstract—Fluent LLM hypotheses are useful data-mining candidates, but unsafe as discoveries: a generated claim may be unsupported, confounded, multiplicity-induced, or unstable. We study LLM-guided verifiable discovery: we let an LLM suggest constrained natural-language hypotheses, but we ask a separate verifier to decide which claims deserve to be called discoveries. VeraDM compiles temporal, sequence, subgroup-treatment, and conjunctive hypotheses into typed data-mining queries, then accepts them only after support checks, permutation testing, multiple-testing correction, matched-contrast validation, and held-out replication. Every accepted or rejected claim becomes an auditable evidence card. Under the split and firewall assumptions, we prove firewall-preservation and held-out FDR results, give BY and multi-round variants, and show a power-separation result explaining why an informative proposer preserves power where exhaustive BH collapses. In experiments, synthetic failure-mode studies, semi-synthetic injections, statistically sound pattern-mining baselines, multi-provider proposer diagnostics, timestamped Bank/Bike/Healthcare case studies, and audit controls show that VeraDM rejects plausible invalid claims while producing readable, replicated discoveries.

Index Terms—statistically sound pattern mining, trustworthy and interpretable data mining, large language models for data mining, mining time-evolving data, false discovery control, matched-contrast validation.

I. Introduction

Interpretable discovery is a central goal of data mining. Association rules, contrast sets, emerging patterns, subgroup descriptions, and rule lists [1], [2], [3], [4], [5] all seek structure that a human can inspect, question, and reuse. Their interpretability comes from a symbolic interface: a discovered pattern is a statement about a condition under which a target changes.

Large language models (LLMs) are a natural front end for this process. Given a schema, they can write hypotheses in plain language—for example, "customers with high debt and low savings are more likely to default"—that connect feature names, domain vocabulary, and candidate rule forms. Adjacent LLM-assisted analysis systems already show how useful this interface can be [6], [7]. The difficulty is that the interface is more fluent than it is trustworthy. A confident, grammatical claim may name nonexistent fields, describe under-supported patterns, or report associations that held only in the sample that inspired them [8]. Even a syntactically valid claim can be an artifact of data dredging or distribution shift [9], [10], [11].

We therefore argue that LLM-guided data mining needs a strict separation between proposal and acceptance: the LLM proposes, but a verifier compiles each claim into a typed query and accepts it only after explicit evidence gates. This separation is the paper's central design choice. It lets us use the model for what it does well, namely finding plausible and human-readable candidates, while leaving authority over what counts as a discovery to statistical tests, held-out evidence, and an auditable record.

We introduce VeraDM [12], a framework for LLM-guided data mining with verifiable discovery. Throughout, a "natural-language hypothesis" means a constrained grammar (Sec. III), not open-ended language understanding: the compiler fails closed on anything it cannot typecheck. The LLM's flexibility is in which typed claims to propose, and the verifier alone decides which claims are accepted. Concretely, on Bank Marketing, VeraDM accepts "long call duration and low euribor ⇒ subscription" (validation effect .59, held-out .58) yet rejects the equally fluent "contacted by cellular ⇒ subscription", which shows no held-out effect, and it reports both as evidence cards that name the gate that decided them.

Contributions. (1) Protocol and compiler: we formulate LLM-guided verifiable discovery as proposal/acceptance separation and give a typed compiler from constrained natural-language hypotheses to executable pattern queries. (2) Verifier and statistical results: we combine support, permutation testing, BH/BY correction, matched contrast, and held-out replication, prove firewall-scoped FDR results, and analyze weighted and multi-round variants. (3) Benchmarks and evidence cards: we release synthetic failure regimes, real-covariate injections, timestamped case studies, proposer diagnostics, and auditable evidence cards, including a controlled pilot in which cards improve human trust judgments over prose-only claims.

Table I is our reader's map: what each major claim is, where we support it, and what limitation remains.

II. Related Work

Pattern and rule mining. Association rules [1], contrast sets [2], emerging patterns [3], subgroup discovery [4], [5], [13], [14], [15], [16], pattern-set mining [17], [18], and rule-list learning [19], [20] define symbolic pattern families

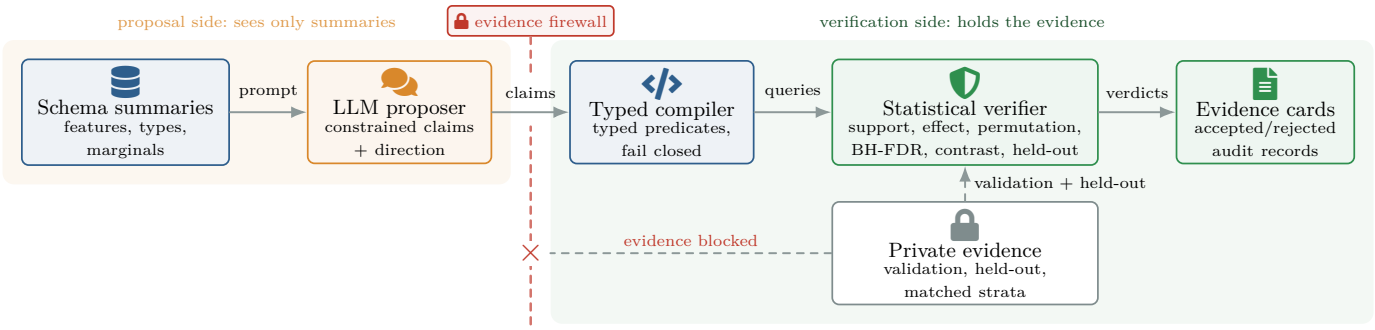

Fig. 1. VeraDM splits discovery into a proposal side and a verification side; validation and held-out evidence stay behind the firewall.

TABLE I
Claim-to-evidence map for the main argument.

Claim	Where	Key evidence	Limitation
proposal–acceptance split	Alg. 1, Fig. 1	LLM proposes, verifier accepts	constrained grammar
held-out FDR firewall	Thm. 1	89 prompts, 0 leaks; BY survival	public-data contamination risk
power separation	Thm. 3	Tables IV, V	depends on proposer ρ
matched contrast matters	Table VII	stable confounder rejected	exact-match overlap varies
evidence cards are auditable	Table XIV	87.5% vs. 69.8% human accuracy	controlled pilot
reproducibility	Table XV	per-card CSVs and result stems	cached proposer outputs

and search them. VeraDM keeps that interpretable rule interface but changes how candidates enter the tested family: constrained natural language proposes a frozen batch, and a separate verifier decides acceptance.

Statistically sound mining and adaptive inference. BH-FDR [9], [21], permutation tests [22], Tarone-style testing [23], [24], Webb-style adjustments [25], [26], Kingfisher [27], LAMP-style mining [28], SPuManTE [29], and statistically sound pattern surveys [30] study enumerated families and correct over those search spaces, and we compare against representative approximations of this line (Tarone-style, Westfall–Young maxT-style, and Subgroup-BH) rather than re-implementing every system. We make no claim of superiority over LAMP or SPuManTE, and a direct empirical comparison with them is left to future work. VeraDM instead controls a language-proposed batch after it has been frozen behind a firewall. Selective and online inference [10], [31], [32], [33], [34] study repeated or adaptive testing, and sample splitting [35] and post-selection inference [36], [37] offer stronger frameworks. We adopt sample splitting because it stays valid for an arbitrarily adaptive language proposer, whereas post-selection inference would require a tractable description of the selection event. VeraDM therefore freezes the batch first, then tests and replicates it on held-out data.

Robustness and LLM-assisted analysis. Invariant prediction and domain generalization seek stability across environments [38], [39], [40], and matching and counterfactual comparisons diagnose confounding [11], [41], [42]. VeraDM uses matched strata as a diagnostic gate, not as a causal identifier. LLM-assisted analysis and visualization systems [6], [7], [43], [44], [45], [46] make analysis easier but usually optimize for insight generation. VeraDM frames the LLM as a candidate generator for statistically validated pattern discovery. Our question is narrower and, for data mining, more consequential: after language proposes a claim, what evidence makes it safe to publish?

III. Problem Setup and the VeraDM Framework

Problem setup. Let $D = \{(x_i, y_i, s_i)\}_{i=1}^{n}$ have typed features x_i, a binary target y_i, and a split label $s_i \in \{\text{summ}, \text{val}, \text{hold}\}$. The proposer observes only a statistic $S(D_{\text{summ}})$ (schema, marginal summaries, discovery-set associations). A hypothesis h pairs a natural-language statement with a compiled query $q_h = (A_h, r_h, \eta_h)$, where A_h is a conjunction of predicates, $r_h \in \{-1, +1\}$ is the asserted direction, and η_h names the estimand kind. The association estimand is $\theta_h = \mathbb{P}(Y{=}1 \mid A_h{=}1) - \mathbb{P}(Y{=}1 \mid A_h{=}0)$, and a subgroup-treatment estimand instead extracts a treatment predicate T_h and subgroup G_h. We write $\Delta_{\text{val}}(h)$ and $\Delta_{\text{hold}}(h)$ for the plug-in estimates of θ_h on the validation and held-out splits. The verifier accepts h iff the signed effect $r_h\theta_h$ is large enough, statistically reliable, robust to matched contrast, and replicated on held-out data. The rest of the section turns that sentence into the concrete workflow of Fig. 1: what the proposer may see, what the compiler is allowed to emit, which gates decide acceptance, and what evidence card a reader receives at the end.

Proposal. We start deliberately upstream of testing. The proposal module receives a schema and (optionally) discovery-set summaries, and emits constrained hypotheses of the form IF <predicate> [AND ...] THEN y=1 increases/decreases, where the asserted direction is reported as ↑ (increase) or ↓ (decrease).

Compiler. We then remove ambiguity before any statistical decision is made. The compiler maps each hypothesis to a typed query and fails closed on any unknown

feature, illegal operator, or missing direction, which is what makes the firewall claim of Proposition 1 checkable. It supports equality predicates, numeric thresholds, temporal windows (EVENT x WITHIN k DAYS $\Rightarrow$ event_x_within_kd), ordered sequences (SEQUENCE $x \rightarrow y$ WITHIN k DAYS), and subgroup "treatment" clauses (TREATMENT x AMONG g) that we read associationally rather than causally.

Gates. Once a claim is typed, we ask it to survive a sequence of increasingly strict checks rather than a single significance test. For each compiled query, VeraDM applies six gates in sequence. The support gate requires validation coverage of at least τ, checked before any held-out record is read; the held-out replication gate rechecks coverage on the held-out split. The validation effect gate requires $|\Delta_{\text{val}}(h)|$ to exceed ϵ. A one-sided permutation test runs R rounds, using the signed effect as the test statistic and label permutation as the null, and the same test on the held-out split supplies the held-out p-values; held-out effect confidence intervals use a normal approximation. BH-FDR then corrects this held-out p-value family. The matched-contrast gate compares exact-match context strata and requires the signed matched effect to exceed ϵ. Finally, held-out replication requires the signed held-out effect and its CI to support the asserted direction. Exact matching can be low-overlap, which would make a matched effect of zero misleading, so the gate also reports the fraction of mass retained in non-empty overlap strata and abstains when this falls below 0.5. Abstention is conservative rather than a soft warning: the card is reported as not certified instead of accepted, so a low-overlap rule never counts as a discovery. We surface this fraction for every accepted card (Table VIII).

Treatment clauses. For a TREATMENT x AMONG g hypothesis, matched contrast is a diagnostic that screens confounding on the context Z_h (either LLM-suggested or a default set), not a causal identification strategy. We therefore report an accepted treatment card as a treatment-associated contrast: an association that survived matching on Z_h, not an identified causal effect.

Evidence cards. The reader-facing output is a card rather than a standalone claim, because we want the verdict and the reasons for it to travel together. Each accepted or rejected card contains the constrained natural-language hypothesis, compiled query, support, validation effect, permutation p, adjusted q, matched contrast, held-out effect and CI, and rejection reasons (Table II). A redundancy-aware post-pruning step removes accepted cards whose validation coverage has Jaccard overlap > 0.9 with a stronger accepted rule, or that are strict predicate supersets of one.

TABLE II
Evidence-card schema.

Field group	Contents
claim/query	constrained claim, typed predicate, asserted direction
validation	support, effect, permutation p, matched contrast
held-out	support, effect, CI, adjusted q, BY flag when requested
decision	accepted/rejected, failed gates, redundancy-pruning note
audit	split identifiers, result-file stem, proposer/model metadata

Algorithm 1: VeraDM: LLM-Guided Verifiable Discovery

Input: dataset D, discovery summary $S(D_{\text{summ}})$, proposal model M, thresholds τ, ϵ, α

prompt M with $S(D_{\text{summ}})$ to obtain hypotheses H
optionally ask M for context variables Z_h and an identifying assumption for each h
compile each $h \in H$ into typed query $q_h = (A_h, r_h, \eta_h)$; reject failures
forall compiled h do
    estimate $\Delta_{\text{val}}(h)$, support, permutation p_h
    compute matched-strata contrast C_h using Z_h or defaults
    estimate $\Delta_{\text{hold}}(h)$, p_h^{hold}, CI
apply BH correction to the held-out p-value family
accept h iff support, effect, validation screen, held-out FDR, contrast, and replication gates pass
prune accepted cards by coverage overlap and predicate supersets
return accepted and rejected evidence cards

IV. Theoretical Analysis

The formal results are meant to capture the discipline in Fig. 1. Under the stated split and firewall assumptions, we allow the LLM to adapt to discovery-set summaries, but not to the evidence used for final acceptance. This is the narrow condition that lets us combine flexible proposal with ordinary multiple-testing logic. Table III maps each result to the operational concern it addresses. Split the sample into mutually independent $D_{\text{summ}}, D_{\text{val}}, D_{\text{hold}}$. The proposer observes only $S(D_{\text{summ}})$ and external randomness independent of $(D_{\text{val}}, D_{\text{hold}})$, and emits a compiled batch $H_1, \ldots, H_m$. Let p_i^{hold} be a held-out p-value for $H_{0,i} : r_i\theta_i \leq 0$.

Proposition 1 (Compiler soundness and firewall preservation): Every accepted compiler output $C(u) = q$ is a well-typed query referring only to schema fields or predeclared engineered features. If u is generated from $S(D_{\text{summ}})$ and external randomness, then $C(u)$ is measurable with respect to the same information and independent of D_{hold} conditional on $S(D_{\text{summ}})$.

Proof sketch. The compiler performs deterministic parsing, schema lookup, type and direction checking, and it does not query validation or held-out records. Composition of the proposer with C cannot introduce dependence on D_{hold}. □

Theorem 1 (Held-out FDR under an adaptive proposer):

TABLE III
Theory map linking results to operational concerns.

Result	Main assumption	Reader payoff	Check
compiler firewall	fail-closed parser	blocks hidden evidence access	prompt audit
held-out FDR	frozen batch; PRDS	BH valid on held-out family	null sweep
BY fallback	arbitrary dependence	conservative fallback	BY survival
weighted BH	pre-hold-out weights	optional power gain	Tab. XIII
power separation	high proposer ρ	small batch preserves power	Tabs. IV, V
adaptive lower bound	reused verdicts/data	rules out naive interaction	multi-round
wealth spending	fresh shards; spent α_t	interaction with budget	multi-round

Assume (A1) $D_{\text{summ}}, D_{\text{val}}, D_{\text{hold}}$ are mutually independent, (A2) $(H_1, \ldots, H_m)$ is measurable w.r.t. $S(D_{\text{summ}})$ and external randomness independent of $(D_{\text{val}}, D_{\text{hold}})$, (A3) each true-null held-out p-value is super-uniform conditional on the proposed hypothesis, and (A4) the held-out p-values satisfy PRDS on the null coordinates. Applying BH at level α to $\{p_i^{\text{hold}}\}$ yields FDR $\leq |\mathcal{H}_0|/m \cdot \alpha \leq \alpha$, where $\mathcal{H}_0$ is the set of true nulls in the batch.

Proof sketch. Condition on the realized batch. By (A1)–(A2) the held-out sample is independent of which H_i was proposed. Conditional on the batch, (A3)–(A4) are the BH conditions [9], [21]. Integrating gives the marginal bound. □

Which family is corrected, and screening. The key bookkeeping point is simple but important: BH corrects only the held-out p-values for the frozen compiled batch $\{H_i\}_{i=1}^m$, fixed by the firewall before any held-out record is touched. Support, effect, and matched contrast act as a pre-screen computed from $S(D_{\text{summ}})$ and D_{val} only. Since these are independent of D_{hold} by (A1), screening selects the tested coordinates without conditioning on the held-out p-values, so applying BH to the screened subfamily preserves the bound (validation-screen failures are dropped, not assigned p-values and then filtered post hoc). Gate order and the scope of the guarantee. Made explicit, the order is: (i) support, validation effect, and matched contrast are computed from $S(D_{\text{summ}})$ and D_{val} alone and select the tested subfamily; (ii) BH is applied once, to the held-out p-values of that frozen subfamily, and Theorem 1 is a statement about this step; (iii) the remaining checks—held-out support, held-out effect and CI direction, and redundancy pruning—act after BH and can only remove cards. The displayed set is therefore a subset of the BH-controlled set. Because filtering after BH can in principle change the realized false-discovery proportion, we state Theorem 1 for the BH step and read the post-BH gates as conservative, discovery-reducing checks rather than as part of the FDR argument. BH vs. BY in practice. We recommend BH as the default when proposed rules are approximately independent or PRDS, and BY (Corollary 1) when the batch contains many heavily overlapping predicates whose held-out statistics are strongly co-dependent. In all three case studies the families overlap yet every BH accept also survives BY (Table XI), so we report BH with BY as a routine robustness check rather than switching defaults.

Corollary 1 (Arbitrary dependence): If PRDS is not assumed, the Benjamini–Yekutieli correction (BH with critical values divided by $c_m = \sum_{k=1}^m 1/k$) controls FDR at level α under arbitrary dependence [21].

Theorem 2 (LLM-confidence weighted BH): For nonnegative weights $w_i = g(c_i)/\bar{g}$ built from the proposer's stated confidence c_i, with $m^{-1}\sum_i w_i = 1$ and measurable w.r.t. $(S(D_{\text{summ}}), \text{rand. ind. of } D_{\text{hold}})$, applying BH to weighted p-values p_i^{hold}/w_i [34] controls FDR at level α under the conditions of Theorem 1.

Theorem 3 (Power separation from an informative proposal prior): Consider an expanded family of size M containing a non-null hypothesis with effect $\theta > 0$. Let an exhaustive verifier test all M claims by BH at level α, and let a firewall-safe proposer output $m \ll M$ claims that contain the non-null hypothesis with probability ρ independent of D_{hold}. Under standard sub-Gaussian one-sided tests, exhaustive testing requires $n \geq c_1\theta^{-2}\log(M/\alpha)$ held-out samples for constant detection probability, while the proposed family requires $n \geq c_2\theta^{-2}\log(m/\alpha)$ conditional on proposal. The unconditional proposed-family power is at least ρ times its conditional power.

Proof sketch. Passing BH in a family of size k needs a null tail of order α/k, so detecting θ with constant probability requires $n = \Omega(\theta^{-2}\log(k/\alpha))$. Substitute $k = M$ and $k = m$. The firewall ensures the proposed family does not invalidate the p-values. □

Theorem 4 (No free FDR control under verdict-adaptive proposal): For any protocol that lets a proposer observe verdicts and then test a new nonempty batch at per-round level at least $\alpha_0 > 0$ without a finite alpha budget, there exists an all-null sequence with valid conditional p-values whose cumulative FDR after T rounds is at least $1-(1-\alpha_0)^T \to 1$.

Theorem 5 (Validation-only BH can be anti-conservative): If the proposer can observe a nonconstant statistic of the split used to compute the BH-controlled p-values, there exists a null distribution and a proposer for which BH at any fixed α has FDR $\to 1$ as the number of candidate features grows.

Theorem 6 (Wealth-spending FWER for verdict-adaptive proposers): For T rounds with $\mathcal{F}_{t-1}$ the past, suppose (B1) $D_{\text{hold}}^{(t)}$ is independent of $\mathcal{F}_{t-1}$, (B2) true-null p-values are super-uniform conditional on $(\mathcal{F}_{t-1}, B_t)$, (B3) within-round acceptance controls FWER at α_t (e.g. Bonferroni), and (B4) the predictable schedule satisfies $\sum_t \alpha_t \leq \alpha$. Then $\mathbb{P}(V \geq 1) \leq \alpha$ across all rounds, so FDR $\leq \alpha$.

A deployable corollary initializes $W_0 = \alpha$, fixes predictable fractions $\gamma_t \geq 0$ with $\sum_t \gamma_t \leq 1$, spends

$\alpha_t = \gamma_t W_0$, and updates $W_t = W_{t-1} - \alpha_t$. Public-data caveat. Theorem 1 assumes the proposer has not seen the held-out records. On public benchmarks this is threatened by pretraining contamination, so we treat public-data results as empirical stress tests rather than theorem-backed deployments. The practical safeguard we recommend is to draw the held-out split from records dated after the proposer's training cutoff, or to keep it in a private store the proposer never sees; only then does the implementation-level firewall coincide with the independence the theorem assumes.

V. Experiments

The evaluation follows the verifier's own decision path. We ask whether proposers produce usable constrained hypotheses (RQ1), whether gates reject plausible but invalid claims (RQ2), when the proposal layer adds value beyond statistically sound pattern mining (RQ3), and whether accepted evidence cards replicate in later data (RQ4). Table XV maps every result to its backing file.

Pre-specified operating choices. Thresholds are fixed before looking at held-out labels: real temporal studies use $\alpha = 0.10$, support $\tau = 0.025$, and effect $\epsilon = 0.035$ (Bank), $\epsilon = 0.050$ (Bike), $\epsilon = 0.025$ (Healthcare, with weaker 30-day readmission effects). Permutation tests use 500 rounds ($p_{\min} = 1/501 \approx 0.002$). Discovery, validation, and held-out partitions follow calendar time where available.

What is actually run. Although the analysis proves several properties, the deployed procedure is a single pass of Algorithm 1: freeze a batch on discovery summaries, run the gate stack on an untouched held-out split with BH (Theorem 1), and report cards. The weighted, multi-round, and lower-bound results (Theorems 2, 6, 4, 5) are optional or explanatory.

Baselines. LLMOnly, DescriptiveStats, StatisticalFilter, HeldoutVerifier, and FullVerifier are ablation arms whose per-arm numbers are in the released artifact; the gate-level summary is Fig. 2A. BH-Exhaustive enumerates singleton/pair rules and applies BH. Tarone-style, Westfall–Young maxT-style [47], and Subgroup-BH are published-style approximations of statistically sound mining, and StableEnv-BH is an environment-stable validation/held-out variant.

A. Gate Validity and Multiplicity

Can plausible invalid claims pass the verifier? Fig. 2 (left) gives the gate-necessity matrix on a 50-seed failure-mode benchmark: each regime isolates one failure (shifted rule, null flood, rare high-lift, confounding, small-n artifact), and removing the corresponding gate admits the invalid family while the full verifier rejects it. The right panel sweeps null-proposal count: raw p-value testing reaches FDR = 0.96 at 1000 nulls while BH remains near nominal and BY is conservative.

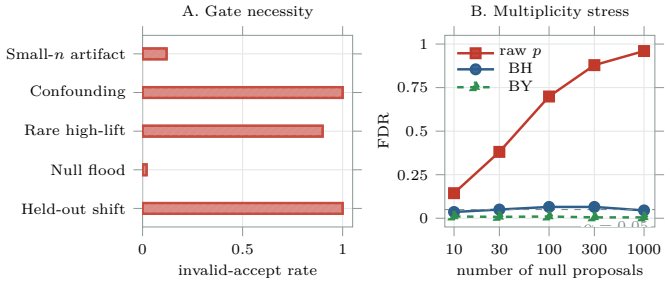

Fig. 2. Gate necessity and multiplicity stress. Left: each bar is the invalid-accept rate when the relevant gate is removed, and the full verifier accepts zero invalid claims in every regime, so its all-zero bars are omitted. Right: as null proposals grow, raw p-value testing loses FDR control while BH stays near nominal and BY is conservative.

TABLE IV
Power separation at controlled FDR.

	Exhaustive+BH		Random-25		VeraDM	
M	rec.↑	FDR↓	rec.↑	FDR↓	rec.↑	FDR↓
10^2	0.945	0.087	0.240	0.097	0.829	0.082
10^3	0.792	0.101	0.033	0.092	0.843	0.095
10^4	0.601	0.098	0.003	0.101	0.853	0.091
10^5	0.387	0.106	0.000	0.088	0.847	0.090

B. Power Separation: the LLM Prior Has Operational Value

When does the proposer add value? Table IV evaluates Theorem 3 in an expanded temporal/sequence/treatment family. Exhaustive+BH tests the full family, Random-25+BH tests a size-matched random shortlist, and VeraDM tests 25 firewall-safe proposals. As M grows from 10^2 to 10^5, exhaustive recall collapses from 0.945 to 0.387 under the multiplicity burden, the random shortlist drops to 0, and VeraDM keeps recall ≈ 0.85 at empirical FDR ≈ 0.09. The Random-25 control rules out the trivial "any small family helps" explanation: the prior needs to be informative.

C. Multi-Provider Inclusion Rate and Power Separation

The power-separation result (Theorem 3) hinges on the proposer's inclusion rate ρ, the probability that a firewall-safe shortlist contains the injected non-null hypothesis. To test whether real LLMs clear the ρ bar in practice, we measure ρ and held-out recall at the hardest expanded-family setting $M = 10^5$ across seven proposers and three expanded-language schemas (temporal, sequence, and treatment) at "guessable" difficulty, using 50 seeds for ρ and 500 for recall. Three reference arms calibrate the scale: an Oracle-25 that always includes the truth, a Human-Expert-25 hand-written shortlist, and Exhaustive+BH (Table V).

Two findings stand out. First, inclusion rate is strongly schema- and provider-dependent: Mistral-Large and Gemini-2.5-Pro clear $\rho \geq 0.70$ on multiple schemas, whereas otherwise-strong models have $\rho = 0$ on entire

schemas—GPT-4o and Claude-Sonnet-4 propose no temporal family at all, and their recall collapses accordingly. This is a coverage failure rather than a parsing one: GPT-4o, for instance, compiles at rate 1.0 on the real schemas below, so its proposals are well formed but never contain the injected temporal-window hypothesis. Second, wherever ρ is high, the corresponding proposer keeps held-out recall well above Exhaustive+BH, and above the Human-Expert-25 shortlist once $\rho \geq 0.80$, matching the theorem's prediction that recall tracks measured ρ, not model size or reputation. The practical lesson of the diagnostic is that a proposer's per-schema ρ can be measured before it is trusted.

D. Pattern-Mining Baselines and Confounding

Is the matched-contrast gate redundant? Table VI compares VeraDM against statistically sound pattern-mining baselines on the singleton/pair family. The table uses two synthetic diagnostics that should not be conflated with the theorem-backed held-out FDR claim: Emp. FDR counts any accepted rule that is not an injected truth as false, including redundant variants of a true signal feature, whereas the redundancy-adjusted FDR does not penalize such variants. The redundancy-adjusted column is the quantity our held-out guarantee (Theorem 1) targets, namely genuine false discoveries, and it is 0.00 for VeraDM and StableEnv-BH. VeraDM's Emp. FDR = 0.77 here therefore reflects redundant true-signal variants rather than uncontrolled error. Validation-only methods recover true rules but accept shifted ones, whereas StableEnv-BH rejects the shifted rule by requiring environment stability. To distinguish VeraDM from StableEnv-BH, Table VII reports a confounded-but-stable injection: the marginal rule replicates across environments but vanishes within matched-context strata. StableEnv-BH accepts the stable confounder, whereas VeraDM rejects it via the matched-contrast gate while preserving true-rule recall, so the gate stack is non-redundant on ground truth. Matched contrast also drives accepted-card max-SMD to 0, and coverage pruning removes redundant accepts (Table VIII). Bike is the most sensitive to this cutoff: its mean retained overlap (0.597) is the lowest and the nearest to the 0.5 threshold. No accepted Bike card abstains at 0.5, but tightening the threshold toward 0.6 would begin to flag them, whereas Bank and Healthcare (0.925 and 0.983) stay far from it. Retained mass is therefore the diagnostic that tells a reader when to distrust a matched contrast: a value near the 0.5 floor means the contrast rests on a small matched subpopulation, whichever variables Z_h contains.

Among the arms compared here, only VeraDM simultaneously recovers true rules, refuses the shifted and the stable-confounded rule, replicates on held-out data, and admits an LLM proposer.

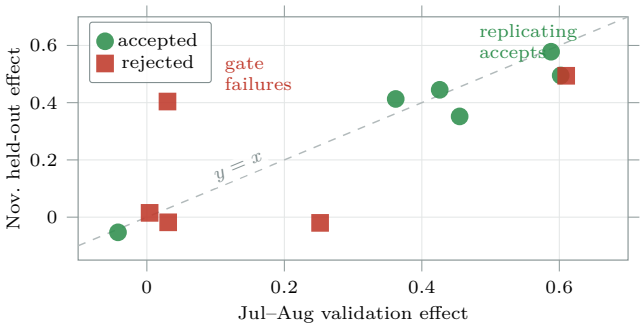

Fig. 3. Selected Bank evidence cards; accepted cards cluster near the $y{=}x$ replication line.

E. Semi-Synthetic Injection and Real Schemas

Do the gates behave on real covariates with known injected truth? We inject known true and known-null rules into real covariates over 30 seeds (Table IX). On Adult, VeraDM recovers all injected truths with no null accepts; on BankTemporal it recovers one of three; and on German Credit none, as expected for a small dataset under held-out verification. UCI Adult Income and German Credit [48], [49] additionally serve as schema feasibility checks for the compiler.

F. Timestamped Bank Marketing Case Study

Do accepted cards replicate in later data? We treat Bank Marketing [50] as a campaign discovery problem: early months suggest targeting hypotheses, but useful discoveries should survive later campaigns. Discovery uses May–June calls (n = 19,087, subscription rate 7.6%), validation uses July–August (n = 13,352, 9.8%), and held-out replication uses November (n = 4,101, 10.1%). The candidate set contains 39 hypotheses, of which 14 pass all gates and 25 are rejected (Table X, Fig. 3). Repeated small p/q values are the 500-round permutation floor, not a BH cap. Bike Sharing [51] (36 hypotheses, 22 accepted, with a calendar-time split into Jan–Jun 2011 discovery, Jul–Dec 2011 validation, all of 2012 held-out) and Healthcare readmission [52] (38 hypotheses, 24 accepted, encounter-order split) follow the same pattern, with per-dataset summaries in Table XI, threshold sensitivity discussed below, and out-of-distribution utility in Fig. 4.

Sweeping the two operating thresholds one at a time gives the expected recall/strictness tradeoff rather than a new tuning procedure. Raising support τ over $\{.01, .025, .05\}$ at fixed ϵ takes Bank from 19 to 14 to 9 accepted rules and Bike from 24 to 22 to 19; raising the effect threshold ϵ at fixed τ takes Bank from 18 ($\epsilon = .025$) to 14 (.035) to 12 (.05), and Bike from 23 to 22.

Fig. 4 treats accepted patterns as auditable features in downstream out-of-distribution (OOD) prediction, on month- and education-based splits of Bank Marketing, Adult, and Online Shoppers [53]. We use this as a shift-stability check, distinct from discovery FDR. In an addi-

TABLE V
Full L1 multi-provider benchmark.

	Inclusion rate $\rho\uparrow$ [95% CI]			Recall @ $M{=}10^5\uparrow$ [95% CI]		
Model	Temporal	Sequence	Treatment	Temporal	Sequence	Treatment
Reference arms						
Oracle-25	—	—	—	0.980 [0.971,0.986]	0.980 [0.972,0.988]	0.985 [0.979,0.992]
Human-Expert-25	—	—	—	0.716 [0.693,0.739]	0.701 [0.677,0.724]	0.702 [0.678,0.727]
Exhaustive+BH	—	—	—	0.397 [0.370,0.423]	0.388 [0.361,0.415]	0.399 [0.371,0.426]
LLM proposers (VeraDM arm)						
Mistral-Large	0.94 [0.87,1.00]	1.00 [1.00,1.00]	0.26 [0.14,0.38]	0.912 [0.898,0.927]	0.987 [0.981,0.993]	0.271 [0.248,0.294]
Gemini-2.5-Pro	0.70 [0.57,0.83]	0.86 [0.76,0.96]	0.90 [0.82,0.98]	0.671 [0.647,0.695]	0.841 [0.823,0.860]	0.876 [0.859,0.893]
Llama-3.1-70B	0.22 [0.11,0.33]	0.96 [0.91,1.00]	0.46 [0.32,0.60]	0.216 [0.196,0.236]	0.948 [0.937,0.959]	0.429 [0.404,0.453]
GPT-4o	0.00 [0.00,0.00]	0.80 [0.69,0.91]	0.18 [0.07,0.29]	0.000 [0.000,0.000]	0.780 [0.758,0.802]	0.183 [0.163,0.203]
Qwen-2.5-72B	0.04 [0.00,0.10]	0.82 [0.71,0.93]	0.04 [0.00,0.09]	0.047 [0.037,0.058]	0.811 [0.790,0.831]	0.044 [0.033,0.055]
Claude-Sonnet-4	0.00 [0.00,0.00]	0.32 [0.19,0.45]	0.00 [0.00,0.00]	0.000 [0.000,0.000]	0.319 [0.294,0.343]	0.000 [0.000,0.000]
GPT-4o-mini	0.06 [0.00,0.13]	0.12 [0.03,0.21]	0.34 [0.21,0.47]	0.061 [0.048,0.073]	0.111 [0.096,0.127]	0.337 [0.313,0.361]

TABLE VI
Statistically sound mining baselines (published-style approximations) on singleton/pair rules.

Method	Acc. ↑	Recall ↑	Emp. FDR↓	Redund. FDR↓	Shift ↓
BH-Exhaustive [9]	21.4	1.00	0.86	0.30	1.0
Tarone-style [23]	0.0	0.00	0.00	0.00	0.0
WY maxT-style [47]	20.7	1.00	0.85	0.29	1.0
Subgroup-BH [14]	22.2	1.00	0.86	0.32	1.0
StableEnv-BH [38]	13.0	1.00	0.77	0.00	0.0
VeraDM	13.0	1.00	0.77	0.00	0.0

TABLE VII
Confounded-but-stable benchmark.

Method	Accepted	Recall↑	Emp. FDR↓	Confounder accept↓
StableEnv-BH	2.0	1.00	0.50	1.00
VeraDM	1.0	1.00	0.00	0.00

TABLE VIII
Balance and redundancy diagnostics.

Diagnostic	Bank	Bike	Health.
mean max-SMD before match	0.451	1.609	0.322
max-SMD after exact context	0.000	0.000	0.000
mass retained in overlap	0.925	0.597	0.983
Redundancy pruning (accepted before → after)			
Adult 17 → 14 (−3)	German 7 → 6 (−1)		

TABLE IX
Semi-synthetic injected-rule benchmark on real covariates, 30 seeds.

Dataset	Recall↑	Emp. FDR↓	Accepted
Adult	1.00	0.00	3.0
German	0.00	0.00	0.0
BankTemporal	0.33	0.00	1.0

tional Mushroom [54] habitat split, SCPM remains robust where pooled patterns collapse (0.92 vs. 0.29 accuracy).

G. Multi-Round Adaptive Discovery and LLM Proposer Diagnostics

We split each Bank and Bike held-out period into three fresh confirmatory shards and run a verdict-adaptive proposer with predictable schedule $(\alpha_1, \alpha_2, \alpha_3) = (0.05, 0.03, 0.02)$. Within each round, acceptance uses Bonferroni on the fresh shard plus the standard gates. Bank accepts 0/0/4 duration-based claims across rounds, and Bike accepts 6/3/5. Prior verdicts may shape later prompts, but no round reuses a confirmatory shard after its evidence card is revealed.

The constrained grammar makes proposal failures visible rather than mysterious. API-model proposals compile reliably: gpt-4o-mini and the stronger gpt-4o each generate 24–25 unique hypotheses per dataset at compile success 1.0. Local open-weight small language models (SLMs) are weaker stress tests: smollm2:135m produces no parseable hypotheses, llama3.2:3b compiles at 0.33, qwen2.5:0.5b at 0.75 (2.0 unique), and llama3.2:1b reaches 0.80 (4.8 unique). Because the verifier is unchanged across proposers, poor proposals are absorbed at the compiler or evidence-card stage. On the real schemas the accepted rules are mostly within the enumerable singleton/pair space with a cross-family minority, verified at 9.5–22.7 hypotheses per second, and the grammar compiles every valid expanded-language kind while rejecting every invalid one (Table XII). Confidence-weighted BH (Theorem 2) recovers power on Bank temporal when weights are calibrated but inflates FDR if miscalibrated weights reuse the calibrated level (Table XIII).

H. Auditability: Human Study and Firewall Leakage Audit

Can the firewall and reporting unit be inspected? Do evidence cards help humans? The evidence card is VeraDM's reporting unit, so we test it directly with a within-subject controlled pilot (16 non-author volunteers, anonymized IDs, authors excluded, using 12 items from Bank/Bike/Healthcare balanced over accepted/rejected verdicts, randomized across the two conditions for 96 judgments per condition): each judge sees a reported claim, shown either as an evidence card or as the natural-language claim (prose) only, and labels it trustworthy or not; the label is then scored against the held-out ground-truth verdict. Card judgments are correct 87.5% of the time versus 69.8% for prose, a within-subject gain of +17.7

TABLE X
Selected Bank Marketing evidence cards.

Decision	Rule	Sup.	Val.	Hold.	Reason
acc.	duration300↑ AND euribor↑	.04	.59	.58	passes
acc.	recent prior contact↑	.03	.60	.50	passes
acc.	euribor low↑	.11	.36	.41	passes
acc.	duration↑ ≥ 600	.09	.46	.35	passes
acc.	blue collar↓ AND campaign↓	.07	−.04	−.05	passes
rej.	poutcome success↑	.02	.61	.49	support
rej.	confidence↑ ≥ −40	.48	.03	.40	weak contrast
rej.	cellular↑	.90	.03	−.02	no replication
rej.	university↑	.35	.00	.02	FDR

TABLE XI
Bike and Healthcare case-study summaries.

Dataset	Hyp.	BH acc.	BY surv.	max-SMD after↓	overlap kept↑
Bank	39	14	14	0.000	0.925
Bike	36	22	22	0.000	0.597
Healthcare	38	24	24	0.000	0.983

points (95% CI [2.1, 33.3], sign-test $p = 0.035$, with 12 improving, 3 worsening, and 1 tie), at a modest time cost (median 47.5 vs. 29.0 s). The auditable card, not merely the underlying statistics, improves human discrimination of trustworthy claims.

Does the firewall hold in practice? The held-out FDR result (Theorem 1) rests on the proposer not seeing held-out data. We audited all 89 proposer prompts across the case studies: every prompt contained the schema and discovery summaries (100%), and none included any validation or held-out record (leak rate 0.000). The firewall is thus satisfied operationally, not only by construction. This audit is implementation-level: it certifies that no validation or held-out record entered a prompt at run time, but it cannot certify statistical independence from a public dataset the proposer may have seen during pretraining (Sec. IV). Under label randomization (Bank, 50 seeds with permuted targets) the verifier accepts 0 rules on every seed (Table XIV), a direct negative control for an FDR-control claim. The compiler itself passes an 80-item gold set with predicate F1 1.00 (Table XVI). It fails closed in the strict sense of rejecting any claim that does not type-check against the schema, and the 0.80 invalid-rejection rate reflects that the remaining gold-set items are well typed but semantically invalid, so they compile and are instead caught downstream by the statistical gates, which leaves firewall safety unaffected.

I. Cost and Reproducibility

Verification is inexpensive: throughput is inversely proportional to the permutation budget R (Table XII), a full Bank or Bike run completes in seconds on one CPU core with no GPU, and proposal is a single batched LLM call.

Reproducibility and artifact. The released artifact [12] bundles proposer prompts, cached LLM outputs, per-card accepted/rejected records, and one result file per table/figure. Table XV maps each manuscript element to its backing result file so a reader can reproduce any number without navigating the source tree. All thresholds are fixed before held-out testing (Sec. V). Synthetic studies

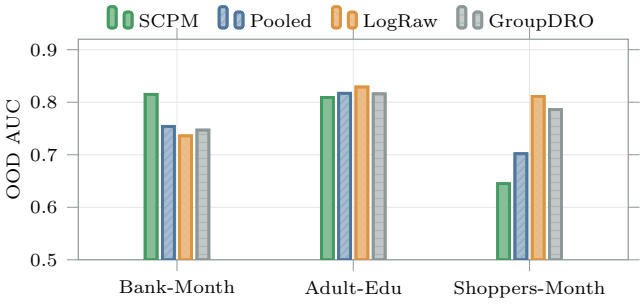

Fig. 4. Accepted-pattern features remain competitive under OOD shift.

TABLE XII
Novelty, throughput, and language coverage. Throughput is profiled on the two clean timestamped datasets (Bank and Bike).

	Bank	Bike	Health.
accepted / enumerable	14/14	22/21	24/24
cross-family / expanded-lang.	5/0	7/1	5/0
hyp./s @ 200/500/1000 rounds	19/9/5	40/23/14	—
Language coverage: 4 kinds, valid-compile 1.00, invalid-rejection 1.00, 5/5 expanded items verified.			

use 50 seeds and permutation tests 500 rounds; the public datasets are UCI Adult, German Credit, Bank Marketing, Bike Sharing, Online Shoppers, Mushroom, and Diabetes 130-US hospitals [48], [49], [50], [51], [53], [54], [52]; and verification runs CPU-only.

VI. Discussion

Our experiments support the design principle introduced at the start: in LLM-guided data mining, proposal and acceptance need different mechanisms. An LLM is a useful source of interpretable candidate rules, but fluency alone does not tell us whether a claim holds. A plausible claim can be confounded, under-supported, a multiplicity artifact, or unstable in later data, so VeraDM treats the model as a candidate generator and the verifier as the authority on what counts as a discovery.

The same separation makes the FDR result usable in practice, with two operating modes. In the default batch mode, control is conditional on a frozen proposed set tested against an untouched held-out split (Theorem 1). The moment the proposer is allowed to see verdicts and react, a single reused held-out set is no longer enough (Theorem 4), and the interactive mode restores control by spending a predictable wealth schedule against fresh confirmatory shards (Theorem 6). Intuitively, letting the model re-query a reused held-out set after seeing verdicts is like re-rolling a test until it passes, and the wealth schedule simply caps the total significance budget those re-rolls may spend.

Two results explain where the LLM layer earns its place. Power separation (Theorem 3) shows that the prior's job is to shrink the tested family from a vast semantic space to a small firewall-safe batch without invalidating

TABLE XIII
Confidence-weighted BH on Bank temporal.

Method	Disc.	Emp. FDR↓	Power↑
Uniform BH	0	0.00	0.00
Weighted BH (calibrated)	21	0.00	1.00
Weighted BH (miscalibrated)	0	0.00	0.00
miscalibrated @ calib. α	10	1.00	0.00

TABLE XIV
Audit dashboard for assumptions and reporting.

Audit	Main result
firewall prompt scan	89 prompts audited, 0 held-out leaks
label randomization	0 accepts over 50 Bank seeds
BY dependence check	survival 1.00 on all three (min q 0.011–0.014)
human card pilot	87.5% card vs. 69.8% prose accuracy
verification throughput	Bank 9.49 hyp./s, Bike 22.68 hyp./s

TABLE XV
Artifact manifest: manuscript element → result-file stem.

Element	Result-file stem
Fig. 2	gate_necessity, null_fdr_sweep
Table IV	power_separation
Table V	l1_rho, l1_power_sep
Table VI	significant_pattern_baselines
Table VII	confounded_stable
Table IX	real_injection
Table X, Fig. 3	bank_temporal_case
Table XI	bike_/healthcare_case
threshold sweep	threshold_sensitivity
Fig. 4	real_environment
Table XII	novelty_over_enumeration, permutation_runtime
Table XIII	weighted_bh_confidence
Table VIII	contrast_balance_diagnostics, redundancy_paper
Table XVI	compiler_gold_set, label_randomization_audit
Table XIV	human_audit, firewall_audit, dependence_robust

BH, and the confounded-but-stable benchmark shows the matched-contrast gate rejecting stable confounders that environment-stability filtering would accept. The proposer plays two roles depending on the search space. On the timestamped real schemas, where accepted rules mostly lie in the enumerable singleton/pair space, its value is budget efficiency. It recovers the same discoveries from a batch of a few dozen proposals instead of exhaustive search, together with 5 to 7 cross-family conjunctions per dataset that singleton/pair enumeration would not surface as a unit (Table XII). On the expanded-language L1 families (M up to 10^5), the effect is larger: exhaustive BH collapses under multiplicity, while a high-inclusion-rate proposer preserves held-out power (Table V).

Finally, auditability here is more than a formatting choice. Evidence cards measurably improve human judgment over prose (+17.7 points, $p = 0.035$), the prompt audit finds zero held-out leakage, label randomization accepts nothing over 50 seeds, and every real-data accept survives Benjamini–Yekutieli under arbitrary dependence.

A. Practical Deployment and Cost

VeraDM is inexpensive to run and easy to drop in. Proposal is a batched LLM call (about 25 hypotheses per call) and verification is one pass with no fine-tuning: throughput is 9.5–22.7 hypotheses/s at 500 permutations and falls inversely with the permutation budget, which simply trades runtime for a lower p-value floor (Table XII). Because the verifier is identical across proposers, VeraDM integrates as a thin wrapper around an existing model or mining pipeline rather than a replacement for it. The operator's main lever is the support/effect threshold, and the number of discoveries responds to it smoothly and predictably, so the recall/strictness tradeoff can be dialed without re-engineering the gates. We interpret the out-of-distribution results in the same spirit: accepted-pattern features stay reliable under shift where pooled-pattern features collapse (Fig. 4), which we read as auditing value, since verification yields durable patterns, rather than as a claim to better prediction.

B. Limitations

The compiler targets binary outcomes and a limited rule language. Matched contrast uses exact-match strata, and propensity weighting and doubly robust estimation are natural extensions. The Healthcare split is an encounter-order proxy, so Bank and Bike remain the clean timestamped evaluations. Our splitting and permutation arguments also treat rows as exchangeable observational units: a Bank row is one contact, a Bike row one calendar record, and a Healthcare row one encounter, so calendar-ordered serial dependence and repeated patients across encounters are not modeled, and block- or cluster-aware permutation is the natural extension. Baselines include exact exhaustive BH and published-style corrections, not bundled reference implementations of every prior system. The LLM proposer study uses cached API and local SLM batches rather than a full temperature/prompt/scale sweep. Finally, the held-out FDR result invokes PRDS for its BH step. This is plausible but not verified for heavily overlapping rule families, so in practice we treat BY survival (Table XIV) as the operative robustness check and recommend BY whenever the proposed batch is strongly co-dependent. Every dataset here is public, so the held-out guarantee carries the pretraining-contamination caveat of Sec. IV, and the firewall targets the non-public, proprietary setting where that risk is absent, which makes an at-scale proprietary deployment the priority next step. The human study is a 16-participant pilot whose interval ($[2.1, 33.3]$) reflects that size, with a pre-registered $n \geq 30$ replication planned.

VII. Conclusion

We studied LLM-guided data mining under a simple discipline: an LLM may propose constrained natural-language hypotheses, but discoveries should be accepted only after evidence checks. VeraDM turns this discipline into a working system: a firewall-preserving compiler, a multi-gate verifier with held-out replication and matched

TABLE XVI
Compiler and label-randomization audits.

Audit	Items	Compile↑	Reject inv.↑	F1 (pred.)↑	Direction↑
Label permute (Bank)	39	0 accepts over 50 seeds			
Compiler gold set	80	1.00	0.80	1.00	1.00

contrast, evidence cards as an auditable reporting unit, and a power-separation result that characterizes when the proposal layer helps. The broader lesson is the one we want readers to carry forward: an LLM is valuable as a hypothesis generator, not as a discovery authority. Credibility should rest on typed queries, controlled testing, held-out replication, and transparent reporting, not on how convincing a claim sounds. We hope the verifiable-discovery framing, and the benchmark we release with it, make that separation easier to demand in LLM-assisted analysis.

Acknowledgement

We acknowledge Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for supporting this study.

References

[1] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," in Proc. ACM SIGMOD, 1993, pp. 207–216.
[2] S. D. Bay and M. J. Pazzani, "Detecting group differences: Mining contrast sets," Data Min. Knowl. Discov., vol. 5, no. 3, pp. 213–246, 2001.
[3] G. Dong and J. Li, "Efficient mining of emerging patterns: Discovering trends and differences," in Proc. ACM SIGKDD, 1999, pp. 43–52.
[4] W. Klösgen, "Explora: A multipattern and multistrategy discovery assistant," in Advances in Knowledge Discovery and Data Mining, 1996, pp. 249–271.
[5] S. Wrobel, "An algorithm for multi-relational discovery of subgroups," in Proc. PKDD, 1997, pp. 78–87.
[6] V. Dibia, "LIDA: Automatic generation of grammar-agnostic visualizations and infographics using large language models," in Proc. ACL (System Demonstrations), 2023, pp. 113–126.
[7] C. Wang, J. Thompson, and B. Lee, "Data formulator: Ai-powered concept-driven visualization authoring," IEEE Trans. Vis. Comput. Graph., pp. 1–11, 2023.
[8] Z. Ji et al., "Survey of hallucination in natural language generation," ACM Comput. Surv., vol. 55, no. 12, pp. 1–38, 2023.
[9] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: A practical and powerful approach to multiple testing," J. R. Stat. Soc. B, vol. 57, no. 1, pp. 289–300, 1995.
[10] J. Taylor and R. J. Tibshirani, "Statistical learning and selective inference," Proc. Natl. Acad. Sci., vol. 112, no. 25, pp. 7629–7634, 2015.
[11] D. B. Rubin, "Estimating causal effects of treatments in randomized and nonrandomized studies," J. Educ. Psychol., vol. 66, no. 5, pp. 688–701, 1974.
[12] P. T. Tran-Truong et al., "VeraDM: Source code and artifact," https://github.com/LeoTTTPhat/VeraDM, 2026.
[13] F. Lemmerich, M. Becker, and M. Atzmueller, "Generic pattern trees for exhaustive exceptional model mining," in Proc. ECML PKDD, 2012, pp. 277–292.
[14] M. Atzmueller, "Subgroup discovery," WIREs Data Min. Knowl. Discov., vol. 5, no. 1, pp. 35–49, 2015.
[15] J. H. Friedman and N. I. Fisher, "Bump hunting in high-dimensional data," Stat. Comput., vol. 9, no. 2, pp. 123–143, 1999.
[16] N. Lavrač et al., "Decision support through subgroup discovery: Three case studies and the lessons learned," Mach. Learn., vol. 57, no. 1–2, pp. 115–143, 2004.
[17] A. J. Knobbe and E. K. Y. Ho, "Maximally informative k-itemsets and their efficient discovery," in Proc. ACM SIGKDD, 2006, pp. 237–244.
[18] B. Bringmann and S. Nijssen, "What is frequent in a single graph?" in Proc. PAKDD, 2008, pp. 858–863.
[19] R. L. Rivest, "Learning decision lists," Mach. Learn., vol. 2, no. 3, pp. 229–246, 1987.
[20] B. Letham et al., "Interpretable classifiers using rules and bayesian analysis," Ann. Appl. Stat., vol. 9, no. 3, pp. 1350–1371, 2015.
[21] Y. Benjamini and D. Yekutieli, "The control of the false discovery rate in multiple testing under dependency," Ann. Stat., vol. 29, no. 4, pp. 1165–1188, 2001.
[22] P. Good, Permutation, Parametric, and Bootstrap Tests of Hypotheses, 3rd ed. Springer, 2005.
[23] R. E. Tarone, "A modified bonferroni method for discrete data," Biometrics, vol. 46, no. 2, pp. 515–522, 1990.
[24] A. Terada et al., "Statistical significance of combinatorial regulations," Proc. Natl. Acad. Sci., vol. 110, no. 32, pp. 12 996–13 001, 2013.
[25] G. I. Webb, "Discovering significant patterns," Mach. Learn., vol. 68, no. 1, pp. 1–33, 2007.
[26] G. I. Webb, "Layered critical values: A powerful direct-adjustment approach to discovering significant patterns," Mach. Learn., vol. 71, no. 2–3, pp. 307–323, 2008.
[27] W. Hämäläinen, "Kingfisher: An efficient algorithm for searching for both positive and negative dependency rules," Knowl. Inf. Syst., vol. 32, no. 2, pp. 383–414, 2012.
[28] F. Llinares-López et al., "Fast and memory-efficient significant pattern mining via permutation testing," in Proc. ACM SIGKDD, 2015, pp. 725–734.
[29] L. Pellegrina, M. Riondato, and F. Vandin, "SPuManTE: Significant pattern mining with unconditional testing," in Proc. ACM SIGKDD, 2019, pp. 1528–1538.
[30] W. Hämäläinen and G. I. Webb, "A tutorial on statistically sound pattern discovery," Data Min. Knowl. Discov., vol. 33, no. 2, pp. 325–377, 2019.
[31] D. P. Foster and R. A. Stine, "α-investing: A procedure for sequential control of expected false discoveries," J. R. Stat. Soc. B, vol. 70, no. 2, pp. 429–444, 2008.
[32] A. Javanmard and A. Montanari, "Online rules for control of false discovery rate and false discovery exceedance," Ann. Stat., vol. 46, no. 2, pp. 526–554, 2018.
[33] A. Ramdas et al., "SAFFRON: An adaptive algorithm for online control of the false discovery rate," in Proc. ICML, 2018, pp. 4286–4294.
[34] C. R. Genovese, K. Roeder, and L. Wasserman, "False discovery control with p-value weighting," Biometrika, vol. 93, no. 3, pp. 509–524, 2006.
[35] L. Wasserman and K. Roeder, "High-dimensional variable selection," Ann. Stat., vol. 37, no. 5A, pp. 2178–2201, 2009.
[36] J. D. Lee et al., "Exact post-selection inference, with application to the lasso," Ann. Stat., vol. 44, no. 3, pp. 907–927, 2016.
[37] R. J. Tibshirani et al., "Exact post-selection inference for sequential regression procedures," J. Am. Stat. Assoc., vol. 111, no. 514, pp. 600–620, 2016.
[38] M. Arjovsky et al., "Invariant risk minimization," 2019.
[39] I. Gulrajani and D. Lopez-Paz, "In search of lost domain generalization," in Proc. ICLR, 2021.
[40] S. Sagawa et al., "Distributionally robust neural networks for group shifts," in Proc. ICLR, 2020.
[41] P. R. Rosenbaum and D. B. Rubin, "The bias due to incomplete matching," Biometrics, vol. 41, no. 1, pp. 103–116, 1985.
[42] H. Bang and J. M. Robins, "Doubly robust estimation in missing data and causal inference models," Biometrics, vol. 61, no. 4, pp. 962–973, 2005.
[43] P. Ma et al., "Insightpilot: An llm-empowered automated data exploration system," in Proc. EMNLP (System Demonstrations), 2023, pp. 346–352.
[44] S. Guo et al., "DS-Agent: Automated data science by empowering large language models with case-based reasoning," 2024.
[45] L. Weng et al., "DataLab: A unified platform for llm-powered business intelligence," 2024.
[46] Y. Han et al., "ChartLlama: A multimodal llm for chart understanding and generation," 2023.
[47] P. H. Westfall and S. S. Young, Resampling-Based Multiple Testing. Wiley, 1993.
[48] B. Becker and R. Kohavi, "Adult," 1996, [Dataset]. UCI Mach. Learn. Repository.
[49] H. Hofmann, "Statlog (German Credit Data)," 1994, [Dataset]. UCI Mach. Learn. Repository.
[50] S. Moro and P. Rita, "Bank marketing," 2014, [Dataset]. UCI Mach. Learn. Repository.
[51] H. Fanaee-T, "Bike sharing," 2013, [Dataset]. UCI Mach. Learn. Repository.
[52] B. Strack et al., "Impact of HbA1c measurement on hospital readmission rates," BioMed Res. Int., vol. 2014, pp. 1–11, 2014.
[53] C. O. Sakar and Y. Kastro, "Online shoppers purchasing intention dataset," 2018, [Dataset]. UCI Mach. Learn. Repository.
[54] "Mushroom," 1981, [Dataset]. UCI Mach. Learn. Repository.