Nothing to fit

Kenneth Pernyér, Reflective Labs
Stockholm, Sweden
kenneth.pernyer@reflective.se · reflective.se · ORCID 0009-0006-4543-1156

A great deal of what organizations call machine learning is a mean, a variance, a linear fit and a threshold. This paper describes Prism, the closed-form analytics extension of my convergence substrate: eleven packs — descriptive statistics, z-score anomalies, linear regression, logistic classification, Gaussian naive Bayes, k-means segmentation, exponential smoothing, moving-average trend detection, weighted multi-criteria ranking, pairwise similarity and fuzzy inference — exposed as Suggestors that propose into a governed context. None of them has a training loop, and I argue that this is a substrate property rather than a modesty. A closed-form pack is a pure function of the facts it reads, so it satisfies the determinism, idempotency and commutativity axioms of the substrate for free; a fitted model satisfies them only relative to a pinned artifact, which is a strictly weaker statement and one that has to be maintained. That difference is exactly where I draw the boundary with the extension that does fit models. I also report a small piece of engineering I have come to rely on: the authority boundary between analytics and promotion is enforced by tests that are required not to compile, so that an attempt to give an analytic pack promotion power fails at build time rather than in review.

1. Introduction

There's a well-known observation that most industrial machine learning is regression with better marketing. It's unkind and roughly true, and the useful version of it isn't a joke about practitioners but a question about architecture: if a mean, a z-score and a linear fit answer the question, what does an organization gain by answering it with something that must be trained, versioned, monitored and eventually retrained?

The two-cultures argument (Breiman, 2001) framed this as a choice between models that describe data-generating mechanisms and models that predict well without one. My concern is narrower and more operational. In a governed decision system, the difference between a closed-form computation and a fitted model is not primarily accuracy. It's that one of them is a function and the other is a function *of an artifact*, and the substrate's guarantees are stated over functions (Pernyér, 2025a).

Prism is where the closed-form half lives. It gives a Formation somewhere to ask a data-driven question — is this value anomalous, what is the trend, which of these candidates ranks highest — and get a proposal back, with the engine retaining authority over what becomes fact.

2. Eleven packs

The list is deliberately unfashionable. It's also, in my experience, most of what gets asked. An organization wanting to know whether last week's refund rate is unusual is asking for a z-score; one wanting to know which of forty suppliers to shortlist is asking for a weighted rank; one wanting next quarter's demand is usually well served by exponential smoothing and poorly served by anything that requires a data engineer.

Pack	Method
Descriptive statistics	Mean, median, variance, percentiles
Anomaly detection	Z-score
Trend detection	Moving average
Forecasting	Exponential smoothing
Regression	Linear least squares
Classification	Logistic
Naive Bayes	Gaussian, priors supplied
Segmentation	K-means, centroids supplied
Ranking	Weighted multi-criteria
Similarity	Pairwise vector similarity
Fuzzy inference	Membership functions and rules

Table 1 | The packs. Every one is closed-form: given its inputs and its declared parameters it computes an answer without consulting a stored artifact. Where a method normally requires fitting — naive Bayes priors, k-means centroids, regression coefficients — the parameters are supplied to the pack rather than learned by it.

© 2025 Reflective Labs™. All rights reserved

The interpretability literature makes the stronger version of this argument for high-stakes settings: where a decision affects someone, an intrinsically interpretable model is generally preferable to an opaque one plus a post-hoc explanation (Rudin, 2019). I agree, and note that the governance framing sharpens it. In my substrate every promoted fact carries provenance, and the provenance of a z-score is *the mean, the standard deviation and the threshold*. There's nothing to explain afterwards because the computation is the explanation.

3. Why closed form is a substrate property

Here is the argument that makes this a paper rather than a feature list.

> **Definition (Closed-form pack).** A Suggestor s is closed-form if there's a pure function g and a declared parameter set Θ, carried in the proposal, such that $s(K) = g(R_s(K), \Theta)$ where $R_s(K)$ is the subset of context it reads. No component of s's output depends on state outside $R_s(K)$ and Θ.

The substrate requires nine invariants, three of which constrain the Suggestor directly: determinism, idempotency and commutativity. For a closed-form pack they are theorems rather than obligations.

> **Proposition (Free invariants).** Every closed-form pack is deterministic, idempotent and commutes with every other Suggestor.

Determinism is immediate: g is a function, so equal inputs give equal outputs. *Idempotency* follows because the proposal is content-addressed and the context is monotone, so a second application proposes the same fact, which the merge absorbs. *Commutativity* holds because s reads only $R_s(K)$ and the context only grows: if another Suggestor ran first, R_s is either unchanged or extended, and on the extended read set g is evaluated on strictly more information without any dependence on when it arrived. Appendix B gives this carefully.

Contrast a fitted model. Let s_θ be a Suggestor backed by an artifact θ. Then $s_\theta(K) = g(R_s(K), \theta)$, which is a function — for a fixed θ. Determinism therefore holds *relative to a pinned artifact*, and the pin is a piece of operational discipline rather than a mathematical fact. Retrain the model and the same context yields a different proposal; ship the retrained artifact without pinning it in the fleet version and the substrate's uniqueness result silently loses its hypothesis, in exactly the way the planning paper describes for learned authority (Pernyér, 2025b).

This isn't an argument against fitted models. It's an argument for knowing which kind you have, which is why the extension that fits models is a different extension with a different contract, discussed in Section 4.

Prism	Crucible
Fuzzy inference from expert rules	Fuzzy systems whose parameters are learned
Gaussian naive Bayes, priors given	Random forest, structure learned
Logistic and linear fits, weights given	Kernel methods and boosted trees
K-means inference, centroids given	Everything requiring a training loop
No artifacts	Artifacts, registry, monitoring, deployment

Table 2 | Prism never fits; Crucible never owns expert rules. The line isn't drawn by algorithm family — the same family appears on both sides — but by whether the parameters arrive with the question or come from a stored artifact.

4. The boundary with fitting

The division of labour is stated as a rule about what each side may own.

Drawing the line at *fitting* rather than at *algorithm* is the choice worth defending. A logistic regression whose coefficients an analyst wrote down is closed-form; the same logistic regression with coefficients fitted last Tuesday isn't, despite being the same arithmetic. What changes isn't the mathematics but whether the answer depends on history. The substrate cares about exactly that, and nothing else about the model.

5. A boundary the compiler enforces

Analytics extensions accrete authority. The pull is constant and reasonable — the pack has just computed that this transaction is anomalous with a z-score of 4.1; why should it not simply mark the transaction? — and every time it is given in to, the promotion path acquires another entrance.

© 2025 Reflective Labs™. All rights reserved

Prism resists this with tests that must fail to compile. Alongside ordinary unit, integration, property and negative tests, the crate carries compile-fail tests that attempt to use Converge's promotion surface from an analytic pack. The suite passes when those files *do not build* and the compiler's message matches. An engineer who widens the boundary discovers it at build time, in the diff, with the error naming the rule.

I mention this because it generalizes past my stack and past Rust. An architectural invariant that lives in a document is a suggestion; the same invariant expressed as a type that can't be constructed, plus a regression test that the construction still fails, is enforcement with a maintenance cost close to zero. It's the same move as the type boundary between coordination and commitment described in the substrate paper, applied one layer down.

6. Limitations

The packs are shallow by design and there are questions they answer badly. A z-score assumes something close to symmetry and constant variance; organizational data is frequently neither, and the pack will report an anomaly on a Monday because Mondays are different. Exponential smoothing has no seasonality term. K-means requires the number of clusters and centroids to arrive with the question, which pushes a modelling decision onto the caller who may not know they are making one.

I don't currently ship diagnostics that would catch these. A residual plot, a normality check, a silhouette score — the tools an analyst would reach for (Hastie et al., 2009, Tukey, 1977) — exist in the ecosystem and not in the packs, and a proposal that carries its parameters but not its diagnostics is telling the truth incompletely.

The Burn-based inference path in the crate is an example rather than a capability. It demonstrates that a neural inference step can register as a Suggestor with supplied weights; it isn't a serving stack, and treating it as one would be a mistake.

Finally, the argument of Section 3 is about the substrate's invariants and says nothing about whether a closed-form method answers a given question *well*. A deterministic wrong answer is still wrong, and the guarantees in this paper are guarantees about reproducibility and provenance, not about accuracy.

7. Conclusion

Closed-form analytics deserve a better position in agentic architectures than they get, and the reason isn't nostalgia. In a system whose correctness rests on Suggestors being functions of context, a pack with no artifact behind it satisfies the requirements by construction, while a fitted model satisfies them by discipline. Both are legitimate; only one is free.

So the boundary is drawn at fitting, not at sophistication, and it's enforced by a compiler rather than by a review convention. What remains is the honest gap: the packs currently tell you what they computed and with which parameters, and not whether their assumptions held. That's the next thing to add, and it's a diagnostics problem rather than a modelling one.

> **Code and correspondence.** The work described here is implemented, not sketched. The source behind every claim in this paper — the engine, the extension, the fixtures and the tests that hold the invariants — can be shared on request, including the parts that did not work. Write to kenneth.pernyer@reflective.se. We are equally interested in being told where the argument is wrong.

References

Breiman, L., 2001. Statistical modeling: The two cultures. Statistical Science 16, 199–231.

Hastie, T., Tibshirani, R., Friedman, J., 2009. The Elements of Statistical Learning, 2nd ed. Springer.

Pernyér, K., 2025a. Proposal, gate, commitment: deterministic convergence as a substrate for organizational decisions. Reflective Labs technical report.

Pernyér, K., 2025b. Institutionalized disagreement: how a plan earns the right to be run. Reflective Labs technical report.

Rudin, C., 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence 1, 206–215.

Tukey, J.W., 1977. Exploratory Data Analysis. Addison-Wesley.

© 2025 Reflective Labs™. All rights reserved

A. The pack surface, operationally

P1 Ingestion Polars-backed readers for CSV, TSV and Parquet, with Excel behind a feature flag. Ingestion produces typed feature vectors, not opinions.

P2 Feature agents A `FeatureAgent` registers as a Suggestor and proposes extracted features. Extraction is itself a proposal and may be refused at the gate.

P3 Packs Each pack declares its typed input, its parameters and its typed output. Parameters travel with the request; a pack that requires a parameter it wasn't given fails rather than defaulting.

P4 Provenance Proposals carry the Prism provenance marker and a trace span at the Suggestor boundary.

P5 Boundary tests Compile-fail fixtures attempt to reach the promotion surface from pack code. The suite requires them to fail to build.

B. Free invariants, carefully

Let s be closed-form with $s(K) = g(R_s(K), \Theta)$ as in Section 3, and let Φ_s be its promotion map.

Determinism. For contexts K, K' with $R_s(K) = R_s(K')$ I have $s(K) = s(K')$ because g is a function and Θ travels with the request. No other input exists.

Idempotency. Facts are content-addressed, so the proposal produced by the second application of s carries the same identity as the first. The merge is set union, hence $\Phi_s(\Phi_s(K)) = \Phi_s(K)$.

Commutativity. Let s_1 be closed-form and s_2 any Suggestor eligible at K, and suppose s_2 runs first, giving $K' = \Phi_2(K) \supseteq K$. Two cases. If $R_1(K') = R_1(K)$ then s_1 proposes identically and the union is unaffected by order. If $R_1(K') \supset R_1(K)$ — s_2 contributed facts that s_1 reads — then s_1 proposes $g(R_1(K'), \Theta)$ in both orders provided s_1 is run after s_2 in both, which the Formation guarantees by running to a fixed point rather than once: the engine re-evaluates eligible Suggestors until no proposal is new, so the later, better-informed proposal is produced in either ordering and absorbed by the union.

The second case is where the argument is doing work, and it's worth being explicit that the property being used is the substrate's, not the pack's: it's the run-to-fixed-point discipline that turns "reads more later" into "same result regardless of order". A closed-form pack inside a single-pass pipeline wouldn't have this property.

© 2025 Reflective Labs™. All rights reserved