Decision-PGA and the Need for Decision-State Diagnostics

Zachary D. Michels, PhD
May 19, 2026

A Prototype Vocabulary for Uncertainty Shape in Agentic AI Workflows

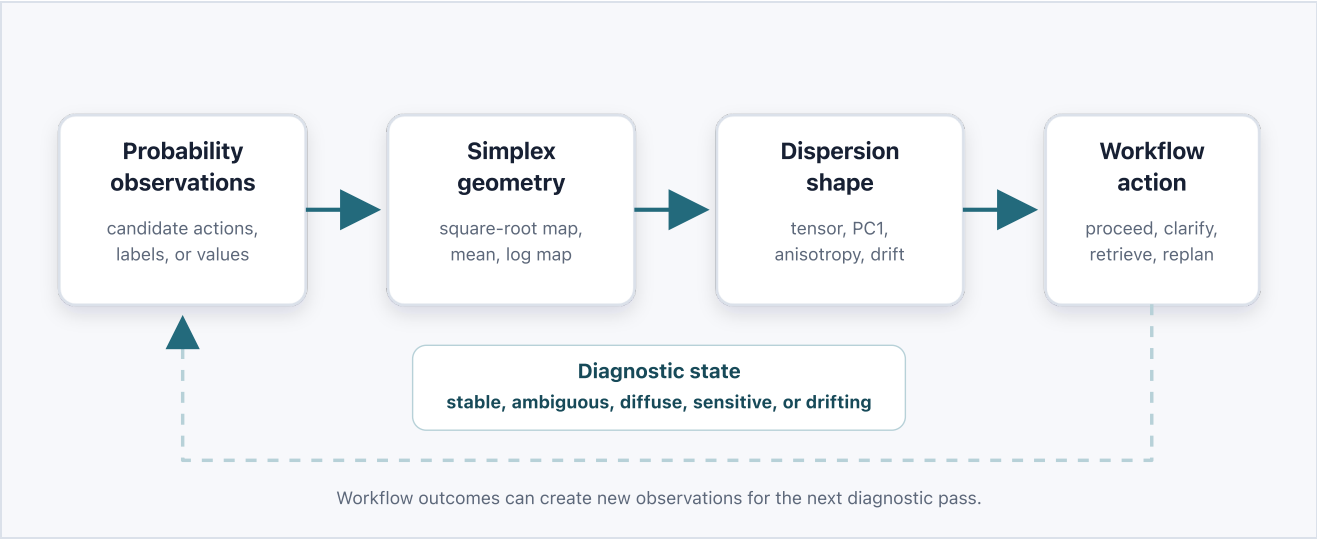
AI systems are moving from one-shot answer generation toward workflow participation. They read documents, retrieve evidence, extract fields, route requests, call tools, compare options, and recommend next actions. A person may experience that as a very ordinary queue: invoices waiting for review, contract dates to extract, messages to triage, or policy questions that need the right source. The uncertainty problem changes when an AI system is not merely writing an answer, but helping decide what should happen next in that queue.

In that setting, "low confidence" is too blunt. A workflow needs more practical questions. Is the extracted due date stable enough to accept? Is the system mostly torn between two plausible contract dates? Is it uncertain because the attachment it needs is missing? Did a threshold rule flip the recommended action? Did later pages in a packet contradict earlier pages?

Decision-PGA is a deliberately narrow prototype method for studying those questions. In its current form, it analyzes clouds of categorical probability vectors on the probability simplex using Fisher-Rao/square-root geometry, then describes the shape of the resulting uncertainty. The aim is not to replace task evaluation or human review. The aim is to make a decision state easier to inspect, compare, and route.

The "PGA" in Decision-PGA refers to Principal Geodesic Analysis: a manifold analogue of Principal Component Analysis (PCA). Where PCA summarizes variation with linear directions in a Euclidean vector space, PGA summarizes variation with geodesic directions on a curved space. The method lineage here comes from Fletcher, Lu, Pizer, and Joshi's 2004 paper, "Principal Geodesic Analysis for the Study of Nonlinear Statistics of Shape," which introduced PGA for statistical shape analysis on nonlinear Riemannian spaces. Decision-PGA borrows that mean, log-map, and tangent-dispersion idea, but applies it to a different object: probability clouds over candidate decisions.

This article is a personal technical perspective, not an institutional statement. It uses no patient data, is not clinical validation, and does not describe a medical device or clinical decision support product.



Decision-PGA treats repeated probability-like observations as a cloud, maps that cloud through probability-simplex geometry, and returns a diagnostic state that can inform the next workflow action.

What this article means by decision-state

The phrase "decision-state" can easily become too large. In this article, I use it in a deliberately narrow and observable sense: a decision-state is a probability-like distribution over an explicit set of candidate decisions in a specified task context.

For example, suppose a document workflow is looking at an invoice due date. The candidate decisions might be: accept the extracted value, ask the user which date they mean, retrieve another attachment, flag the case for review, or defer until the packet is re-read. A decision-state is not the whole mind of the model. It is the observed pattern of support across those specific options.

That definition is intentionally modest. It does not require access to hidden model activations, private reasoning traces, or a full latent representation of agent cognition. It also does not assume that every agent trace lives on a smooth behavioral manifold. For now, Decision-PGA asks a smaller question: when we can observe repeated probability-like estimates over candidate decisions, does the shape of that probability cloud add useful diagnostic information beyond scalar uncertainty scores?

The larger possibility is exciting: future systems may combine probability clouds with richer telemetry such as tool traces, retrieval paths, memory updates, and planner states. But that is a research trajectory, not a claim made by the current prototype.

The gap between confidence and action

Many AI tools already expose useful signals: confidence scores, token probabilities, calibrated probabilities, retrieval scores, agreement rates, human review flags, and task-specific benchmarks. Those signals matter. The gap appears when a system must decide what to do next.

Consider two document cases that both look "uncertain" by a scalar score. In the first, a contract has two dates on the same page: an effective date and a signature date. The system is not generally confused; it is mostly split between "accept this date" and "ask which date definition the user intended." In the second case, a purchase request refers to an attachment that is not present. The system's probability mass is spread across retrieve context, clarify, review, and defer because the evidence itself is incomplete.

Those should not feel like the same workflow state. The first case suggests a targeted clarification. The second suggests finding the missing source. Entropy can warn that both cases are uncertain, but it does not always tell a person what kind of uncertainty they are looking at. A decision-state diagnostic tries to preserve that shape:

- if repeated observations stay tightly around one action, the extraction may be stable enough to accept;
- if the cloud stretches mainly between two choices, the next useful action may be a targeted clarification;
- if the cloud spreads across many choices, the workflow may need more context rather than a yes/no decision;
- if small changes flip the action near a threshold, the case may deserve review;
- if the preferred action changes over a sequence of pages or tool calls, the workflow may need to pause, segment, or replan.

The practical question is modest but important: can we build diagnostic tools that help AI workflows choose safer and more useful next actions under uncertainty?

A minimal operational example

Imagine a document extraction tool that has read the same field several ways: through repeated model samples, OCR perturbations, neighboring page context, reviewer votes, or a small set of rule checks. Each pass returns support for the same five workflow actions: accept the extraction, ask for clarification, retrieve more context, flag for review, or defer.

A simple score can tell us that the tool is uncertain. Decision-PGA asks a more workable follow-up: what does the uncertainty look like? Is the cloud tightly clustered around accept? Is it stretched between accept and clarify? Is it diffuse because the source document is incomplete? Is it moving from accept to defer as later pages introduce contradictory evidence?

This is the practical niche Decision-PGA is meant to explore. It is not trying to make the document extraction decision by itself. It is trying to help a workflow choose a better next step when a human would otherwise only see "confidence: low."

A small companion page makes this operational shape concrete with synthetic document-extraction triage cases: Document Extraction Triage Demo.

The public prototype code is available at github.com/zmichels/Decision-PGA.

Why healthcare is a useful application lens

Healthcare is not the only place this matters, but it is a useful lens because the workflows are high-accountability, document-heavy, and full of decisions where "low confidence" is too vague to be operationally helpful.

Examples worth studying include:

- message or request triage, where candidate actions might include answer, clarify, route, schedule, retrieve context, or escalate;
- policy and guideline retrieval, where sources may be relevant but incomplete, outdated, or in tension;
- medical document extraction, where uncertainty may reflect two plausible values, source-span ambiguity, table-row confusion, or OCR sensitivity;
- trial matching and eligibility review, where some criteria are clearly met, some are missing, and some conflict across sources;
- operational agents that coordinate multi-step work and may drift as they gather context.

These examples should be treated as research and evaluation targets, not as claims of deployment readiness. A diagnostic can describe the shape of a decision state; it does not determine clinical truth, prove safety, or replace domain review.

The broader public context supports caution. The FDA maintains information on AI-enabled medical devices and clinical decision support software: https://www.fda.gov/medical-devices/software-medical-device-samd/artificial-intelligence-enabled-medical-devices and https://www.fda.gov/medical-devices/software-medical-device-samd/clinical-decision-support-software-frequently-asked-questions-faqs. The ONC HTI-1 rule addresses transparency for predictive decision support interventions in certified health IT: https://healthit.gov/regulations/hti-rules/hti-1-final-rule/. The WHO has published guidance on ethics and governance of AI for health: https://www.who.int/publications/i/item/9789240029200.

The point is not that Decision-PGA solves these governance questions. It does not. The point is that governance and evaluation need inspectable technical signals, and decision-state diagnostics may become one useful category of such signals.

What Decision-PGA does

Decision-PGA starts with a representation pipeline:

1. define a candidate decision set;
2. gather repeated probability-like observations over that set;
3. normalize those observations into probability vectors;
4. analyze the resulting probability cloud.

The labels might be actions, extracted field values, evidence clusters, routing choices, or review outcomes. The current prototype then:

1. maps probability vectors to the positive unit sphere with the square-root transform;
2. computes an intrinsic mean;
3. log-maps samples into a tangent space;
4. computes a dispersion tensor and eigensystem;
5. reports shape metrics such as total dispersion, PC1 fraction, anisotropy ratio, margin, label switching, and half-window geodesic drift.

That produces a compact diagnostic contract:

State	Possible workflow interpretation
Stable	The same due date, route, or action keeps winning; proceed if the task is in scope and other checks pass.
Binary ambiguity	The workflow is mostly torn between two options, such as two plausible dates; ask a targeted question or compare top candidates.
Diffuse uncertainty	No action clearly explains the case; gather evidence, retrieve a missing attachment, or broaden context.
Boundary sensitive	A small change flips the action near a threshold; inspect assumptions, thresholds, or constraints.
Regime shift	Later evidence changes the preferred action; segment the task, replan, or escalate.

This contract is intentionally small. It is meant to be usable by software systems, not just by notebooks. A local command-line tool, report generator, or agent tool can call the diagnostic and receive a structured result.

Just as importantly, the contract is bounded. It does not decide whether the underlying answer is true. It does not explain a model's internal reasoning. It does not replace task-specific evaluation. It is a diagnostic layer over one observable object: a probability cloud over candidate decisions.

Why geometry, and why be cautious about it?

Probability vectors live on a simplex, not in ordinary unconstrained Euclidean space. Decision-PGA uses the square-root transform to place probability vectors on the positive unit sphere, where Fisher-Rao geometry becomes easier to work with. The result is a way to analyze dispersion directions, not only dispersion amount.

That matters because uncertainty can have shape. A cloud stretched along one direction is different from a cloud spread broadly across many choices. Two states may have similar entropy while suggesting different next actions. PGA metrics such as the first principal geodesic fraction and anisotropy ratio are attempts to capture that distinction.

The geometric framing should be treated as a hypothesis to test, not as mathematical ornament. If simpler tools such as entropy, margin, agreement, calibration, switch rate, or ordinary trajectory clustering explain the same behavior just as well, Decision-PGA should say so. The value of PGA-style analysis depends on whether intrinsic dispersion shape reveals operationally useful distinctions that simpler baselines miss.

That grounding matters because Decision-PGA is not inventing "PGA" as a brand name. It is adapting an existing geometric-statistics idea to a decision diagnostics setting. The adaptation itself still needs evidence: the shape of a probability cloud may be useful in some workflow states and unnecessary in others.

Application patterns worth testing

Tool and action selection

Agentic systems frequently choose among tools, routes, or next actions. A support agent might decide whether to answer directly, search a knowledge base, open a ticket, or route to a specialist. A diagnostic could distinguish a stable tool choice from a two-tool ambiguity or diffuse uncertainty across the action set.

Retrieval and evidence conflict

Retrieval systems can surface sources that are relevant but not mutually consistent. In a policy question, one source may look current while another contains an exception. A diagnostic over candidate evidence clusters or answer decisions could help decide whether to answer, retrieve more, cite uncertainty, or route to review.

Document extraction

Extraction systems often face ambiguous spans, conflicting values, incomplete tables, or uncertain entity associations. A decision-state diagnostic could separate a stable extracted value from a two-value dispute, a missing-document problem, or a table-row association problem.

Multi-step workflow monitoring

An agent may begin with one plan, gather new evidence, and gradually move into a different decision regime. A case may begin as "accept this extraction" and end as "defer, because a later page contradicts the earlier value." Sliding-window diagnostics could help identify when a trajectory should be segmented, replanned, or escalated.

What must be proven next

The next step is not a bigger claim. It is better evidence. Decision-PGA needs tests that ask where it adds value over simpler metrics, where it is redundant, and where the geometric assumptions fail.

Useful near-term tests include:

- explicit representation pipelines that define the candidate decision set, sampling procedure, and metric object being analyzed;
- entropy-matched examples where binary ambiguity and diffuse uncertainty should lead to different actions;
- fixture suites for tool selection, retrieval conflict, and document extraction;
- held-out synthetic scenarios with known decision states;
- comparisons against entropy, margin, agreement, calibration, drift, switch-rate, PCA, and clustering baselines;
- stability checks across sample count, candidate-set design, and perturbation strategy;
- readable reports that state when PGA does not add value.

For healthcare-adjacent examples, the first benchmark should use synthetic or public, non-patient fixtures. No clinical or operational claim should depend on private data, anecdote, or unreviewed workflow assumptions.

How to try the prototype

The current code is an initial public research release. It is intended for synthetic examples, collaborator testing, agent-tooling experiments, and critique:

- public repository: github.com/zmichels/Decision-PGA;
- synthetic demo fixture: Document Extraction Triage Demo;
- local examples for probability clouds, model-output-shaped observations, sampled responses, provider-score-shaped payloads, and evaluation reports.

The prototype should still be treated carefully. Examples should remain synthetic or explicitly public. Negative or redundant findings should be reported alongside promising ones. Healthcare-adjacent examples should be treated as evaluation targets that require separate governance before real-world use.

Decision-PGA may turn out to be most useful as a small observability layer: a local diagnostic that helps an AI workflow decide whether to proceed, clarify, gather evidence, inspect sensitivity, segment, replan, or escalate.

If that narrow layer proves useful, the larger research direction becomes more interesting: decision-state diagnostics could eventually connect probability clouds, trace telemetry, retrieval behavior, and tool-use trajectories into a more general science of agent workflow monitoring. That is the horizon, not the starting claim.

Selected references

- Fletcher, P. T., Lu, C., Pizer, S. M., & Joshi, S. (2004). "Principal Geodesic Analysis for the Study of Nonlinear Statistics of Shape." IEEE Transactions on Medical Imaging, 23(8), 995-1005. https://doi.org/10.1109/TMI.2004.831793
- Zhang, M., & Fletcher, P. T. (2013). "Probabilistic Principal Geodesic Analysis." Advances in Neural Information Processing Systems 26. https://papers.nips.cc/paper/5133-probabilistic-principal-geodesic-analysis
- Kuhn, L., Gal, Y., & Farquhar, S. (2023). "Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation." https://arxiv.org/abs/2302.09664

Conclusion

As AI systems become more active participants in workflows, they will need ways to expose not only what they chose, but what kind of uncertainty surrounded the choice. Confidence scores are part of that story, but they are not the whole story.

Decision-PGA proposes a compact way to describe uncertainty shape over candidate decisions. It is early, model-neutral, and intentionally modest. The important claim is not that the method is ready for high-stakes deployment, nor that PGA is the inevitable geometry of agent cognition. PGA is not the inevitable geometry of agent cognition; it is one testable lens on a narrower observable object. The important claim is that decision-state diagnostics deserve to be made visible, tested, and improved before agentic AI systems become routine infrastructure.