LUMINOMÁTICA (LUM)

Metrología de la Razón.

▶ **¿Qué es metrología?** La metrología es la ciencia de la medición: define cómo medir algo con precisión, trazabilidad y reproducibilidad. Un termómetro calibrado, una regla estandarizada, una báscula certificada son todos instrumentos metrológicos. LUM extiende esta idea al terreno del conocimiento científico: en vez de medir temperatura o longitud, mide la madurez de un campo científico, es decir, si su evidencia es lo suficientemente coherente, auditable y consistente como para justificar acción. La expresión 'Metrología de la Razón' no es una metáfora poética: es el programa técnico del sistema. Igual que la metrología física requiere instrumentos calibrados con trazabilidad a patrones internacionales, LUM requiere índices computables con calibración externa y trazabilidad criptográfica.

Sistema Gnoseo-Operatorio Unificado para Demarcación,

Cierre Científico (CLARION) y Planes de No-Cierre (PSNC),

con Contrato Auditable

Edición integrada para ensayo: arquitectura por fricciones, variables computables,

reglas de decisión (GREEN/AMBER/RED/BLACK/INVALID), gobernanza de versiones y emisión de bundles verificables,

Mtro. **Julio David Rojas Aguayo,** Laboratorio de Luminomática | Ciudad de México, México, 24 de marzo de 2026

LEYENDA DE AUDITABILIDAD (Release v0.1.0)
Este release publica un **Hello World auditable** (un campo completo con bundle reproducible), pero **no publica** el dataset completo de validación multi-campo por razones de confidencialidad. Sí publica la **Contract Specification v1.0.0** (CONTRACT.md, schema JSON y ejemplos) en GitHub y Zenodo (DOI: 10.5281/zenodo.19211260, 24-mar-2026).
Por tanto:

1. Los KPIs globales (AUC/ECE, etc.) se consideran **internos** y **no auditables externamente** en este release.
2. El documento debe interpretarse como **SPEC + DEMO**, no como validación independiente a gran escala.
3. Se publican artefactos congelados (z_scales, thresholds) y un bundle completo para reproducibilidad mínima.

ÍNDICE POR FRICCIONES

Sección	Función	Qué demuestra	Puente hacia siguiente
1. El problema que la filosofía nombró pero no pudo medir	Abre la fricción: vocabulario vs. procedimiento	Los criterios clásicos fallan por falta de operacionalización, no por error conceptual	Nombrar el problema es el primer paso; computarlo es el segundo.
2. Arquitectura LUM: tres capas apiladas	Presenta la solución estructural	LUM no es una fórmula aislada sino un pipeline: demarcación → tipado → medición → decisión → auditoría	El pipeline necesita medir algo real: ahí entran los cuatro ejes.
3. Los cuatro ejes ortogonales: qué mide cada uno y cómo falla	Fricción central: medición real vs. numerología	IPU, CPV, A_norm y κ_conf capturan dimensiones distintas del cierre; cada uno puede fallar de modo distinto	Medir los ejes produce un score; convertirlo en decisión requiere el modelo.
4. La ecuación maestra: cierre como evento probabilístico en ventana Δ	Fricción: permanencia vs. temporalidad	El modelo clog-log con offset log(Δ) convierte los ejes en probabilidad de cierre calibrada; no en verdad	La probabilidad sola no basta; necesita una regla de decisión con guardarraíles.
5. Semáforo y guardarraíles: GREEN/AMBER/RED/BLACK/INVALID	Fricción: decisión honesta vs. GREEN fraudulento	Cada estado tiene condición específica, consecuencia concreta y lo que prohíbe hacer	Cuando no hay GREEN, se necesita un plan de salida estructurado: el PSNC.
6. PSNC: la ingeniería del no-cierre	Convierte el no-cierre en protocolo accionable	El PSNC no es un consuelo; es un diagnóstico causal con acciones concretas según el tipo de deficiencia	Todo sistema de decisión necesita gobernanza para no corromperse: ahí entra la capa de auditoría.
7. Gobernanza, versionado y huella criptográfica	Fricción: sistema auditable vs. sistema sin trazabilidad	El versionado, el drift detection y el footprint SHA-256 hacen que el dictamen sea	El bundle necesita un contrato formal para volverse

		reconstruible y que la corrupción deje huella	institucionalmente deployable.
8. El Contrato LUM-I/O: el dictamen como objeto verificable	Cierra la arquitectura con el bundle auditable	El contrato especifica la tupla (INPUT, CONFIG, MODEL, OUTPUT, AUDIT) y define INVALID como estado de violación del contrato	Todo sistema tiene límites: la conclusión los declara honestamente.
9. Alcances, límites y respuesta al contraargumento del problema de Goodhart	Aparato crítico: dónde falla LUM y por qué es legítimo igual	LUM-PE no “resuelve” Goodhart por decreto: lo convierte en condición explícita de auditoría. El sistema no promete imposibilidad del gaming; promete detección temprana operable, evidencia de sobrecierre, y degradación automática del dictamen (BLACK/INVALID) cuando aparecen señales de optimización adversarial.	—

LUMINOMÁTICA (LUM): Sistema Gnoseo-Operatorio Unificado para Demarcación, Cierre Científico y Planes de No-Cierre, con Contrato Auditable

«Es propio del hombre cultivado buscar la exactitud en cada materia en la medida en que la naturaleza del asunto lo permite, y sería igualmente absurdo aceptar razonamientos probables de un matemático que exigir demostraciones a un retórico.»

— Aristóteles, Ética a Nicómaco, I, 3

«Sólo cuando las operaciones de un campo se cierran sobre sus propios resultados, neutralizando al sujeto operatorio, puede hablarse de ciencia en sentido estricto.»

— Gustavo Bueno, Teoría del cierre categorial, Vol. I

RESUMEN

Presentamos Luminomática (LUM), un sistema formal computable para determinar cuándo un campo científico ha alcanzado cierre suficiente para justificar acción, y qué hacer cuando no lo ha alcanzado. LUM trata el cierre como un evento probabilístico dentro de una ventana temporal Δ, cuantificado por cuatro ejes ortogonales: defectología (IPU), centralidad de prácticas (CPV), estructura topológica (A_norm) y densidad de contradicción (κ_conf). Estos ejes se integran mediante un modelo de regresión con enlace complementario log-log, corregido para eventos raros y calibrado externamente. La regla de decisión clasifica el estado de un campo en cuatro categorías: VERDE (CLARION, cierre operativo), ÁMBAR/GRIS (señal parcial), ROJO (no-cierre) y NEGRO (sobrecierre patológico). Cuando no hay cierre, LUM produce un Plan de Salida de No-Cierre (PSNC) estructurado con acciones específicas según el diagnóstico. El sistema se cierra con un contrato auditable (LUM-I/O), gobernanza de versiones y huella criptográfica SHA-256. El resultado es el primer criterio metrológico de demarcación científica que es simultáneamente computable, temporalmente revisable, accionable y auditable por terceros.

Palabras clave: demarcación científica; cierre categorial; metaciencia; criterio computable; GLM; reproducibilidad; gobernanza; CLARION; PSNC.

> ▶ **Guía rápida para lectores nuevos:** Si es la primera vez que encuentras este sistema, aquí están los conceptos clave que necesitas para seguir el texto. Un campo científico es el conjunto de investigaciones, datos, modelos y teorías organizados alrededor de un fenómeno (por ejemplo: psicología social, terapia génica, física de partículas). Cierre científico (CLARION) significa que ese campo ha alcanzado suficiente coherencia interna y externa para justificar acción institucional sobre sus resultados. No es una declaración de verdad absoluta: es una declaración de madurez operativa. Probabilidad calibrada es una probabilidad que se ha verificado empíricamente: si el sistema dice 80%, en el 80% de los casos similares efectivamente ocurre lo que predice. El contrato (LUM-I/O) es el registro formal y verificable de todos los parámetros de una evaluación, de modo que cualquier tercero pueda auditarla o reproducirla. El semáforo

> (VERDE/ÁMBAR/ROJO/NEGRO/INVÁLIDO) es la decisión operativa que produce el sistema a partir de sus cálculos.

Este ensayo es la especificación narrativa de LUM-PE. El código fuente, el dataset sintético y la documentación técnica completa están disponibles de forma abierta en GitHub y Zenodo (ver sección "Recursos y código" al final de este documento).

---

1. El problema que la filosofía nombró pero no pudo medir

Existe una diferencia entre diagnosticar una enfermedad y saber cuándo el paciente está curado. La filosofía de la ciencia ha sido, durante casi un siglo, extraordinariamente hábil en el primer tipo de tarea y casi inútil en el segundo.

Karl Popper identificó correctamente que la falsabilidad separa la ciencia de la metafísica. Pero falsabilidad en principio no equivale a suficiencia empírica para actuar: una teoría puede ser falsable y sin embargo estar tan rodeada de parches ad hoc que ningún resultado la derribe en la práctica. Thomas Kuhn describió la dinámica real de cómo los científicos trabajan —bajo paradigmas que estructuran qué preguntas son legítimas y qué evidencia cuenta— pero su teoría es retrospectiva: explica por qué hubo revolución después de que ocurrió, no cuándo un paradigma está lo suficientemente maduro para guiar política pública. Imre Lakatos refinó a Kuhn con la distinción entre programas de investigación progresivos y degenerativos, un paso adelante enorme, pero tampoco especificó umbrales cuantitativos ni procedimientos que un evaluador externo pueda reproducir. Paul Feyerabend, en el otro extremo, disolvió la pregunta misma argumentando que no existen reglas metodológicas universales.

Gustavo Bueno formuló, desde el Materialismo Filosófico, el concepto de cierre categorial: un campo tiene cierre cuando sus operaciones son estables, se conectan con otras a través de puentes, y producen identidades sintéticas que neutralizan al sujeto operatorio —es decir, que el resultado no depende de quién lo ejecuta sino de qué aguanta auditoría. Este concepto es quizás el más cercano a lo que LUM operacionaliza. Su limitación es que Bueno no tradujo el cierre categorial a un algoritmo: describió la condición pero no especificó cómo medirla.

> ▶ **Cierre categorial en términos simples:** Imagina un laboratorio donde varios técnicos independientes, usando los mismos protocolos, obtienen siempre el mismo resultado al medir la misma muestra. El resultado ya no depende de quién lo ejecuta, sino del método. Eso es cierre: cuando la operación "neutraliza al sujeto", es decir, cuando la persona que mide deja de importar porque el procedimiento es lo suficientemente estricto y su resultado es verificable. Bueno llamó a esto "identidades sintéticas": resultados que emergen de la operación misma y pueden auditarse externamente. LUM operacionaliza esta condición: en vez de describirla filosóficamente, la convierte en índices computables.

La crisis de reproducibilidad que golpeó a la psicología, la biomedicina y la economía en la primera década de este siglo reveló que el problema no es académico. Cuando los resultados de laboratorio no se replican —y el Proyecto de Reproducibilidad en Psicología encontró que cerca de la mitad de los efectos publicados no se sostienen bajo condiciones equivalentes— las consecuencias van más allá de la vida académica: intervenciones de salud pública se basan en hallazgos que colapsan, medicamentos se aprueban sobre evidencia que no resiste replicación, políticas sociales se diseñan sobre efectos que resultan ser artefactos metodológicos.

El problema estructural es que no existe, hasta LUM, un criterio operativo que responda a la pregunta correcta. La pregunta correcta no es: ¿esta teoría es científica? Sino: ¿el cuerpo de evidencia de este campo es suficientemente coherente para justificar acción? La primera es una

pregunta filosófica sobre la naturaleza de la ciencia; la segunda es una pregunta metrológica sobre el estado actual de un campo. LUM responde la segunda.

1.1 Por qué los criterios existentes fallan en el lugar preciso que importa

Los cuatro grandes criterios de demarcación fallan en dimensiones distintas, y vale la pena ser quirúrgico sobre dónde falla cada uno, porque LUM no pretende sustituirlos como filosofía: los complementa como procedimiento.

Criterio	Aporte real	Falla en...	Consecuencia práctica
Popper (falsabilidad)	Separa ciencia de metafísica. Identifica estructura lógica de la teoría.	Accionabilidad y revisabilidad. No dice cuándo hay suficiente corroboración.	Teorías falsables pueden sobrevivir indefinidamente con parches.
Kuhn (paradigmas)	Captura la dinámica social de la ciencia en acción.	Las cuatro: no da umbrales, no es prospectivo, no es auditable.	Solo describe el cambio después de que ocurrió.
Lakatos (programas)	Distingue progreso de degeneración. Evalúa trayectoria.	Accionabilidad y auditabilidad. La evaluación es retrospectiva.	No puede decir cuándo actuar ahora.
Bueno (cierre categorial)	Especifica la condición filosófica del cierre: identidades sintéticas bajo operaciones estables.	Auditabilidad y accionabilidad. No hay algoritmo ni umbral.	Concepto rico, imposible de implementar institucionalmente.
LUM	Opera a nivel de campo, no de teoría. Produce probabilidad calibrada en ventana temporal.	Satisface las cuatro dimensiones: scope, revisabilidad, accionabilidad, auditabilidad.	Primer criterio deployable institucionalmente.

La tabla anterior no es un argumento de superioridad filosófica: es un mapa de funciones. LUM no refuta a Popper; opera en una capa diferente, la capa de la decisión institucional sobre campos, no sobre teorías individuales.

1.2 El problema de la unidad de análisis

Aquí yace la confusión más productiva de la filosofía de la ciencia: las teorías y los campos son unidades de análisis distintas que producen preguntas distintas. Popper evalúa teorías. Lakatos evalúa programas de investigación. Kuhn evalúa paradigmas. Pero cuando una agencia regulatoria pregunta si la terapia génica está lo suficientemente desarrollada para aprobar un protocolo, o cuando un fondo de investigación decide si el priming social merece financiamiento, la unidad relevante es el campo en su conjunto, con su heterogeneidad interna, sus controversias activas, su historia de replicaciones fallidas y exitosas.

LUM resuelve esto con el Axioma A1 (Field Scope): el cierre es propiedad del campo, no de la teoría ni del estudio individual. El campo es la aggregación de prácticas de investigación, datos, modelos

y teorías organizados alrededor de un conjunto común de fenómenos. Esta distinción no es semántica: cambia completamente qué se mide y cómo se mide.

2. Arquitectura LUM: tres capas apiladas

LUM no es una ecuación con nombre bonito. Es un sistema de procesamiento con tres capas funcionales que deben ejecutarse en orden, porque cada capa filtra los errores que la siguiente no puede detectar sola.

Capa	Nombre	Pregunta que responde	Fallo que previene
−1	Demarcación gnoseológica (anti-mito)	¿Esto es un problema resoluble con operadores, reglas de juego y criterios de contradicción, o es una totalización sin recorte?	GREEN fraudulento por sobrecierre de marco: correr el pipeline sobre un objeto mal recortado.
0	Tipado por dominio + selección de pack de verificaciones	¿Qué tipo de campo es este y qué cuenta como solución válida en su dominio?	Aplicar criterios de cierre de ciencias naturales a problemas de ingeniería, o viceversa.
1	Medición LUM + Decisión + Gobernanza + Bundle auditable	¿Los cuatro ejes, el modelo probabilístico y el sistema de decisión producen un dictamen reproducible?	Dictámenes sin trazabilidad que no pueden ser auditados ni detectan deriva.

El flujo completo queda: DEMARCATE → TYPE → SELECT_PACK → VERIFY → MEASURE_LUM → DECIDE (CLARION/PSNC/BLACK) → EMIT (bundle+hash). Cada flecha es una operación con input y output especificados.

2.1 Capa −1: La cuchilla gnoseológica

La primera pregunta que LUM formula no es estadística sino filosófica en sentido estricto: ¿el objeto sobre el que se va a evaluar cierre tiene operadores, reglas del juego y criterio de contradicción? Si la respuesta es no, el sistema no entra al pipeline. Devuelve PSNC-D (Plan de Salida de No-Cierre por Demarcación).

Los seis tests que el sistema ejecuta internamente son: (1) ¿qué acciones concretas existen sobre el objeto —medir, probar, demostrar, simular, intervenir?; (2) ¿qué cuenta como evidencia o validez en este dominio —axiomas, replicación, identificación causal, pruebas A/B?; (3) ¿qué refutaría la respuesta?; (4) ¿qué queda explícitamente fuera del alcance (scope_out)?; (5) ¿en qué condiciones de frontera falla el modelo?; (6) ¿puede auditarse la cadena de decisiones?

Si falla alguno de los tres primeros o el recorte es totalizante, el objeto se clasifica como M-TOT (mito por totalización) o I-NOR (ideología normativa), y el sistema devuelve PSNC-D en lugar de continuar. Preguntas del tipo «¿Qué es La Conciencia?», «¿Es La IA peligrosa?» o «¿Es La Naturaleza sostenible?» activan M-TOT. No porque sean preguntas inválidas —son preguntas filosóficamente ricas— sino porque no tienen el recorte operatorio necesario para aplicarles LUM. Convertirlas a problemas tratables requiere primero definir el campo concreto sobre el que se va a operar.

2.2 Capa 0: El tipado por dominio y los packs de verificaciones

Una vez que el objeto pasa la demarcación, el sistema lo tipifica según dominio: formal (matemática, lógica, verificación), natural (física, biología, química, medición experimental), social (economía, psicología, ciencias políticas con identificación causal) o ingeniería (P-TEC: diseño, optimización,

pruebas de especificación). El tipado importa porque «solución» no significa lo mismo en todos los dominios.

En ciencias formales, una solución requiere demostración verificable desde axiomas, contraejemplo o verificación automática. La contradicción es un paso inválido o una inconsistencia con los axiomas. En ciencias naturales, requiere un modelo con predicciones, validación fuera de muestra y control de confusores. La contradicción son predicciones incompatibles con datos bajo condiciones equivalentes. En ciencias sociales, requiere identificación causal explícita (DiD, IV, RDD, SCM), pre-trends, placebos, robustez a especificaciones alternativas y validez externa. La contradicción es heterogeneidad no explicada por moderadores plausibles. En ingeniería, requiere especificación verificable, tests unitarios e integración, métricas de performance y seguridad.

Esta diferenciación no es burocrática: impide que un campo declare GREEN por haber cumplido los criterios de otro dominio. Un campo social que no demuestra identificación causal pero tiene estadísticas bonitas no puede obtener GREEN aunque su p_cal sea alta, porque el pack SOC_DiD_v3 incluye un gatillo BLACK específico para esa situación.

2.3 La trinidad M1-M2-M3 como condición del cierre

El Materialismo Filosófico de Bueno distingue tres momentos en el conocimiento: M1 (físico-corporal: aparatos, señales, datos, registros), M2 (práctico-social: operaciones, protocolos, instituciones, sanciones) y M3 (simbólico-formal: modelos, ecuaciones, índices, lenguajes formales). LUM hereda esta estructura y la operacionaliza: el cierre requiere que los tres estén trabados.

No hay cierre con solo M3 (ecuaciones sin datos ni protocolos), ni con solo M1 (datos brutos sin modelo ni protocolo de verificación), ni con solo M2 (consenso institucional o comité sin datos ni modelo). El cierre real emerge cuando las identidades sintéticas producidas por el campo se sostienen independientemente de quién ejecuta la operación: cuando el resultado depende del método, no del metodólogo.

3. Los cuatro ejes ortogonales: qué mide cada uno y cómo falla

Los cuatro índices de LUM no son arbitrarios ni decorativos. Cada uno captura una dimensión del cierre que los demás no pueden capturar, y esa ortogonalidad es la que permite combinarlos en un modelo sin que se cancelen ni dupliquen.

3.1 IPU — Índice de defectología (integridad interna)

IPU mide la frecuencia y gravedad de los defectos metodológicos en la ventana de evidencia: retractaciones, correcciones mayores, errores de análisis documentados. Un campo con IPU alto tiene poca actividad defectológica severa; un campo con IPU bajo tiene defectos frecuentes o graves que erosionan la confianza en sus resultados.

La forma estándar del índice es:

> ▶ **Cómo leer la fórmula de IPU:** La fórmula $IPU = 1 - (\sum_i w_i \cdot E_i) / (\sum_i w_i)$ se lee así: E_i es la severidad del defecto i (0 = defecto menor, 1 = retractación total o fraude). w_i es el peso del defecto i, que puede ser 1 para todos por defecto, o proporcional al número de citas del estudio afectado (un error en un paper muy citado pesa más). El cociente $(\sum w_i \cdot E_i) / (\sum w_i)$ es el promedio ponderado de defectos, entre 0 y 1. Restar ese cociente de 1 invierte la escala: un campo con muchos defectos graves tiene IPU cercano a 0; un campo con pocas retractaciones y errores menores tiene IPU cercano a 1. Ejemplo concreto: si un campo tiene 3 estudios de peso igual, con severidades 0.1, 0.2 y 0.0, el promedio ponderado es (0.1+0.2+0.0)/3 ≈ 0.10 y IPU ≈ 0.90 (campo sano). Si los defectos son 0.8, 0.9 y 0.7 (tres retractaciones graves), IPU ≈ 0.20 (campo con problemas severos de integridad).

$$IPU = 1 - (\Sigma_i \, w_i \cdot E_i) \, / \, (\Sigma_i \, w_i)$$

donde E_i es la severidad del defecto i y w_i es su peso relativo. La versión robusta usa estandarización MAD (Median Absolute Deviation) para atenuar el efecto de outliers: calcula z-scores robustos y los pasa por una función logística, produciendo $IPU \in [0,1]$.

Lo que IPU no hace: no mide si el campo tiene razón. Mide la higiene metodológica de su evidencia. Un campo puede tener IPU alto y producir conocimiento erróneo; puede tener IPU bajo y estar en proceso de corrección productiva. IPU es una condición necesaria, no suficiente, del cierre.

Cómo falla IPU: (a) fraude por omisión —no reportar defectos para mantener IPU alto—; (b) sesgos de publicación que ocultan errores del corpus; (c) ventanas temporales muy cortas que no capturan el historial completo. El sistema de auditoría obliga a declarar el evidence_set con hash, lo que hace la omisión trazable.

3.2 CPV — Índice de centralidad (coherencia práctica)

CPV mide qué tan agrupadas están las prácticas, modelos y datos del campo en torno a un núcleo central. Un campo con CPV alto tiene prácticas convergentes; uno con CPV bajo tiene prácticas dispersas o contradictorias entre sí.

CPV tiene dos variantes según el dominio. CPV_s (espacial/geométrico) usa distancia de Mahalanobis robusta entre cada estudio y el núcleo central del campo; CPV predictiva usa la probabilidad de que una observación pertenezca al núcleo según un modelo de clasificación. La variante se declara en cpv_semantics_version y no puede cambiarse sin un bump de versión.

> ▶ **¿Qué es la distancia de Mahalanobis?** La distancia euclidiana común (la que conocemos del teorema de Pitágoras) trata todas las dimensiones como iguales e independientes. La distancia de Mahalanobis es más inteligente: ajusta la distancia según la forma y correlación de los datos. Ejemplo: en un campo donde los estudios normalmente tienen alta muestra y bajo p-valor, un estudio con muestra media y p-

> valor medio podría estar 'lejos del núcleo' según Mahalanobis aunque geométricamente no parezca tan alejado. Robusto significa que usa mediana y MAD (Desviación Absoluta de la Mediana) en lugar de media y desviación estándar, lo que hace que unos pocos estudios extremos no distorsionen la medición del núcleo central. En términos de CPV: si la mayoría de los estudios del campo forman un núcleo compacto y coherente, CPV será alto. Si los estudios están dispersos sin núcleo claro, CPV será bajo, independientemente de cuántos estudios haya.

La regla de normalización de CPV es de las más cuidadas del sistema: CPV pasa por clipping ($\varepsilon_{low}=10^{-6}$, $\varepsilon_{high}=1-10^{-6}$), luego por logit, luego por z-score con escalas congeladas por versión. Este pipeline impide que valores extremos distorsionen el modelo y garantiza comparabilidad entre evaluaciones.

Cómo falla CPV: (a) definición circular del núcleo —si el núcleo se define retrospectivamente sobre los estudios que se evalúan, CPV mide autoconsistencia, no centralidad real—; (b) monoparadigmatismo artificial —un campo forzado por presión institucional a converger superficialmente puede tener CPV alto con diversidad real baja—; (c) cambio de la definición semántica sin versionar, que haría incomparables dictámenes históricos. Por eso la versión de la semántica CPV es un campo obligatorio del contrato.

3.3 A_norm — Actividad topológica normalizada (estructura de la evidencia)

A_norm cuantifica la estructura topológica del cuerpo de evidencia mediante homología persistente: mide qué tan conectada, coherente y estable es la red de evidencias en la ventana de análisis. Donde IPU mide errores y CPV mide convergencia de prácticas, A_norm mide la arquitectura global de la evidencia.

> ▶ **¿Qué es la homología persistente?** La homología persistente es una técnica matemática del Análisis Topológico de Datos (TDA) que estudia la "forma" de una nube de puntos, en particular sus agujeros y componentes conectados, y cómo persisten esas características al variar la escala de observación. Intuición: imagina que tienes un mapa con ciudades (estudios). ¿Están todas conectadas por carreteras (referencias cruzadas, replaciones)? ¿Forman grupos aislados (islas de evidencia)? ¿Hay un 'agujero' en el centro, es decir, una zona de evidencia ausente que nadie ha estudiado? La homología persistente detecta precisamente eso: componentes conectadas (¿cuántas islas?), ciclos (¿hay circularidad en las citas?) y vacíos (¿qué zonas del espacio teórico no tienen evidencia?). Un campo con A_norm alto tiene su evidencia bien conectada, sin islas aisladas y sin vacíos inexplicables. Un campo con A_norm bajo tiene evidencia fragmentada: muchos estudios que no se hablan entre sí, que no forman una red coherente capaz de producir predicciones robustas.

La intuición geométrica es la siguiente: si la evidencia de un campo forma una nube de puntos bien conectada, con estructura estable bajo perturbaciones, A_norm es alto. Si forma grupos aislados, archipiélagos de evidencia que no se conectan, o si es muy sensible a remover algunos estudios, A_norm es bajo. El campo tiene evidencia pero no tiene estructura.

En la práctica, para campos con restricciones computacionales, A_norm se aproxima mediante A_norm* (versión robusta): mide la estabilidad paramétrica del modelo del campo, la modularidad de sus lemas o resultados principales, y la sensibilidad a ablaciones del conjunto de evidencia.

El par A_norm/A_norm* tiene una función irreemplazable en el sistema: es posible tener IPU alto (pocas retractaciones), CPV alto (prácticas convergentes) y aun así tener evidencia fragmentada, no conectada, incapaz de producir predicciones fuera de muestra. A_norm captura esa dimensión que los otros ejes pasan por alto.

Cómo falla A_norm: (a) sensibilidad a la definición de qué constituye un «estudio» o «nodo» en la red de evidencia; (b) dificultad computacional real para campos con cientos de miles de publicaciones; (c) interpretación errónea: A_norm no mide si el campo tiene muchos estudios, sino si esos estudios forman una estructura coherente.

3.4 κ_conf — Densidad de contradicción (consistencia interna)

κ_conf mide la inconsistencia entre resultados y modelos dentro del campo. No es simplemente el número de estudios que se contradicen: es la densidad de incompatibilidades bajo la definición operacional de incompatibilidad que el evaluador debe preregistrar.

El índice derivado Cons = 1 − κ_conf representa la consistencia interna y entra en el modelo con coeficiente $\beta_\kappa \geq 0$ (más consistencia nunca puede reducir la probabilidad de cierre; esto es una restricción normativa del modelo).

La diferencia entre κ_conf y los demás ejes es que κ_conf requiere que el evaluador defina ex ante qué cuenta como contradicción en su dominio. Esta definición debe ser preregistrada y forma parte del contrato LUM-I/O. Sin esa definición, κ_conf no puede medirse: se llama a la inconsistencia «diversidad» o «perspectivas complementarias» para evitar que registre incompatibilidad real. El preregistro bloquea esa trampa.

Cómo falla κ_conf: (a) definición laxa de incompatibilidad que permite que todo sea «complementario»; (b) sesgos de confirmación en la selección del corpus que oculten estudios contradictorios; (c) campos con heterogeneidad real y legítima —donde los resultados varían según el contexto de forma explicable— podrían verse penalizados si la definición de incompatibilidad no incorpora moderadores.

3.5 Variables auxiliares: Conf, coverage/Sh y diagnósticos espectrales

Además de los cuatro ejes, LUM obliga a medir tres variables auxiliares que operan como diagnósticos del sistema:

Conf (incertidumbre operacional): mide la incertidumbre del evaluador sobre los propios índices, derivada de la calidad y completitud de los datos disponibles. Si Conf < θ_Conf, el sistema bloquea automáticamente el estado GREEN —porque la incertidumbre sobre los índices es tan alta que la probabilidad de cierre calculada no es fiable.

coverage/Sh (cobertura/shadow): mide qué fracción del campo está representada en el corpus de evaluación. Sh = 1 − coverage representa la «sombra» —lo que no se ve. Un campo con Sh alta puede tener evidencia positiva pero el evaluador no tiene acceso a ella, lo que invalida cualquier dictamen fuerte.

Diagnósticos espectrales (H_s, SNR, I): entropia espectral de la evidencia, señal/ruido y nivel de interferencia. Estos parámetros orientan el PSNC cuando hay no-cierre: H_s alta con SNR baja indica ruido de medición (la solución es mejorar los instrumentos); I alto indica interferencia entre mecanismos (la solución es rediseñar el modelo causal).

4. La ecuación maestra: cierre como evento probabilístico en ventana Δ

El salto conceptual más importante de LUM es tratar el cierre científico como un evento —no como un estado permanente. Esta diferencia parece sutil pero cambia todo.

Un estado permanente dice: «este campo es ciencia» o «no lo es». Un evento dice: «en la ventana temporal (t, t+Δ], la probabilidad de que ocurra al menos un evento de cierre es p». La primera formulación es binaria y estática. La segunda es probabilística, temporal y revisable.

4.1 El evento de cierre (CLARION)

El evento de cierre se define como la existencia de una identidad sintética auditada en la ventana (t, t+Δ]. En términos computacionales:

$$E\_\Delta(t) = \{\exists \text{ IS auditada en } (t, t+\Delta]\}$$

y la probabilidad de cierre es:

$$p\_\Delta(t) = P(E\_\Delta(t) \mid F\_t)$$

donde F_t es la información disponible hasta el tiempo t. Este condicionamiento es crucial: p_Δ(t) no es la probabilidad de que el campo «tenga razón»; es la probabilidad de que, dado lo que sabemos hasta ahora, se produzca cierre en la siguiente ventana. Anti-leakage: mirar al futuro invalida el dictamen.

4.2 La ecuación maestra

Los cuatro ejes se integran mediante un modelo de regresión con enlace complementario log-log (clog-log), diseñado para modelar la probabilidad de eventos raros en ventanas de tiempo:

> ▶ **La ecuación maestra, término por término:** $\log(-\log(1 - p\_\Delta(t))) = \log(\Delta) + \beta_0 + \alpha \cdot IPU^d + \beta \cdot CPV^d + \gamma \cdot A\_norm\_z + \beta\kappa \cdot Cons\_z$. El lado izquierdo ($\log(-\log(1-p))$) es la transformación clog-log de la probabilidad: convierte $p \in (0,1)$ en un número real que puede ser cualquier valor, lo que permite hacer regresión lineal. $\log(\Delta)$ como offset: Δ es el tamaño de la ventana temporal (por ejemplo, 5 años). Este término no tiene coeficiente estimado: se suma directamente. Garantiza que si doblas la ventana, la probabilidad de ver al menos un evento de cierre aumenta de forma matemáticamente correcta. β_0 es la constante del modelo (intercepto), que captura la dificultad base de lograr cierre en cualquier campo. α, β, γ, βκ son los coeficientes que indican cuánto contribuye cada eje a la probabilidad de cierre, estimados con datos históricos de campos con cierre conocido. IPU^d, CPV^d, A_norm_z, Cons_z son las versiones estandarizadas (z-score) de cada índice, con media 0 y desviación estándar 1 para hacerlos comparables. La restricción $\beta\kappa \geq 0$ es una restricción normativa: más consistencia interna (Cons = 1 − κ_conf) nunca puede reducir la probabilidad de cierre. Si el modelo estadístico sugiere lo contrario, hay un error en el corpus.

$$\log(-\log(1 - p\_\Delta(t))) = \log(\Delta) + \beta_0 + \alpha \cdot IPU^z + \beta \cdot CPV^z + \gamma \cdot A\_norm\_z + \beta\kappa \cdot Cons\_z$$

Cada componente tiene una justificación operacional, no solo matemática:

log(Δ) como offset: garantiza invarianza por ventana. Si duplicas Δ, la probabilidad de ver al menos un evento de cierre aumenta de forma correcta, sin cambiar los coeficientes del modelo. Sin este offset, cambiar Δ requeriría reentrenar el modelo completo.

El enlace clog-log es la elección correcta para eventos raros en tiempo continuo. El enlace logístico estándar asume simetría alrededor de p=0.5, lo que no es válido cuando la mayoría de los campos están en no-cierre. El clog-log modela correctamente la asimetría de la distribución de cierres.

> ▶ **¿Por qué clog-log y no regresión logística?** La regresión logística es el modelo estándar para predecir probabilidades binarias (cierre/no-cierre). Sin embargo, asume que el efecto de los predictores es simétrico alrededor de $p = 0.5$, lo que resulta adecuado cuando los dos resultados son aproximadamente igual de frecuentes. En el mundo real de la demarcación científica, los cierres son eventos raros: la mayoría de los campos están en no-cierre en la mayoría de las ventanas de evaluación. Esta asimetría hace que la logística subestime la dificultad del cierre. El modelo clog-log (log-log complementario) corresponde a la distribución de valores extremos de Gumbel y es matemáticamente equivalente al modelo de riesgos proporcionales de Cox para datos de supervivencia. Modela correctamente eventos raros: la curva de probabilidad sube lentamente hasta valores altos de los predictores, y luego se acelera, reflejando que el cierre es difícil de alcanzar pero, una vez que los índices son altos, la probabilidad sube rápidamente. En términos prácticos: con logística, un campo con índices moderados podría parecer cerca del cierre; con clog-log, ese campo correctamente se clasifica en no-cierre hasta que los índices alcanzan valores sustancialmente altos.

La restricción $\beta\kappa \geq 0$ es normativa: más consistencia (mayor Cons = 1 − κ_conf) nunca puede reducir la probabilidad de cierre. Si el modelo estimado produce $\beta\kappa < 0$, el resultado es inválido y debe investigarse el corpus.

La corrección de Firth para eventos raros reduce el sesgo de máxima verosimilitud estándar cuando el número de eventos de cierre observados en el conjunto de entrenamiento es pequeño. Sin esta corrección, el modelo sobreestima sistemáticamente la probabilidad de no-cierre.

4.3 Calibración externa: la diferencia entre score y probabilidad

Este punto es donde LUM establece una distinción que la mayoría de los sistemas similares evitan: p_raw (la probabilidad cruda del modelo) es un score, no una probabilidad en el sentido de frecuencia relativa. Para convertirlo en probabilidad calibrada, se requiere calibración externa con un conjunto de datos independiente.

La calibración puede ser de Platt (ajuste logístico del score) o isotónica (monotónica no paramétrica). El método, el hash del dataset de calibración, el procedimiento de split y la fecha deben quedar registrados en el contrato. Las métricas obligatorias de calibración son ECE (Error de Calibración Esperado) y Brier Score.

> ▶ **¿Qué son ECE y Brier Score?** ECE (Error de Calibración Esperado) mide si las probabilidades del modelo corresponden a frecuencias reales. Si el modelo dice “80% de probabilidad” para un conjunto de casos, el ECE mide si aproximadamente el 80% de esos casos realmente tuvieron el resultado. ECE = 0 es calibración perfecta; ECE = 0.10 significa que el modelo se equivoca en promedio 10 puntos porcentuales en sus probabilidades. Ejemplo: si LUM asigna p_close = 0.85 a 20 campos, y solo 14 de esos 20 (70%) realmente alcanzan cierre, ECE ≈ 0.15 (el modelo está sobreestimando). Brier Score es la media de los errores cuadráticos entre la probabilidad predicha y el resultado real (0 = cierre ocurrió, 1 = no ocurrió). Brier Score = $(1/N) \sum(p_{predicha} - resultado_{real})^2$. Varía entre 0 (perfecto) y 1 (terrible). Un Brier Score de 0.25 equivale al de un modelo que siempre predice 0.5 (sin información). La diferencia clave: ECE mide calibración (¿las probabilidades son honestas?); Brier Score mide calibración Y discriminación juntas (¿el modelo es preciso?). LUM obliga a reportar ambas porque un modelo puede tener buen Brier Score pero mala calibración, o viceversa.

La regla es explícita: si no hay calibración, p_close es score y no puede reportarse como probabilidad; el estado GREEN requiere AND_min estricto y CPV y coverage altos para compensar la incertidumbre de calibración. Si ECE supera el umbral en flujo durante múltiples ventanas, el sistema debe recalibrarse y versionar.

Esta distinción tiene consecuencias institucionales concretas: un dictamen que reporta p_close=0.82 sin calibración externa no está afirmando que la probabilidad de cierre es 82%; está afirmando que el score del modelo es 0.82, lo que requiere interpretación diferente.

5. Semáforo y guardarraíles: GREEN/AMBER/RED/BLACK/INVALID

La probabilidad calibrada por sí sola no produce una decisión. Para que el sistema sea accionable, LUM implementa una regla de decisión con cuatro estados, un guardarraíl de mínimos obligatorios y una condición de invalidez del contrato.

5.1 Los cuatro estados y su semántica operacional

Estado	Condición de activación	Qué permite	Qué prohíbe
VERDE (CLARION)	p_cal ≥ 0.80 Y AND_min OK. Con calibración externa válida.	Acción basada en el campo. Intervenciones bajo monitoreo de drift.	Tratar el dictamen como permanente. Requiere revisión en [t, t+2Δ].
ÁMBAR/GRIS (señal parcial)	0.40 ≤ p_cal < 0.80. O p_cal ≥ 0.80 pero AND_min falla con ECE_decil ≤ 5%.	Investigación bajo PSNC. Financiamiento condicionado a milestones.	Acciones de alto impacto irreversibles. Claims de cierre sin más evidencia.
ROJO (no-cierre)	p_cal < 0.40.	PSNC obligatorio. Diagnóstico causal. Investigación básica.	Intervenciones de política basadas en el campo. Financiamiento sin condiciones.
NEGRO (sobrecierre)	Cierre degenerado: circularidad, supresión de comparaciones, leakage, monocriterio colapsado. **Nota (Release 0.1.0):** la robustez adversarial del estado NEGRO aún no está validada; se reporta como defensa **teórica** hasta completar red-team	Auditoría externa obligatoria. PSNC-D (demarcación).	Cualquier dictamen de cierre. Publicación del semáforo. Acción institucional.

5.2 AND_min: el guardarraíl que impide el GREEN débil

AND_min es una condición necesaria para que el GREEN sea audit-grade —es decir, confiable para uso institucional. La condición es:

$$\text{AND\_min: } (\text{IPU} \geq \theta\_\text{IPU}) \wedge (\text{A\_norm o A\_norm}^* \geq \theta\_\text{A}) \wedge (\text{Conf} \geq \theta\_\text{Conf})$$

> ▶ **AND_min en lenguaje llano:** AND_min es un requisito de mínimos que deben cumplirse simultáneamente (por eso AND, no OR) para que un VERDE sea confiable a nivel institucional. Piénsalo como un avión: puede tener velocidad suficiente (IPU alto) pero si el tren de aterrizaje falla (CPV bajo) o hay niebla densa (coverage insuficiente), no puedes declarar aterrizaje seguro. AND_min es la salvaguarda: dice "incluso si la probabilidad calculada es alta, no puedo declarar VERDE institucional si los cimientos del cálculo tienen grietas detectables". El VERDE-AUDIT es el término medio: la probabilidad es alta, AND_min falla en algún punto, pero ECE por decil es aceptable. El sistema no dice ni GREEN ni RED: dice "prometedor pero requiere verificación en [t, t+2Δ]".

Si p_cal ≥ 0.80 pero AND_min falla, el sistema puede reportar VERDE-AUDIT (una subcategoría que obliga a remedición en [t, t+2Δ]) en lugar de VERDE pleno, siempre que ECE por decil sea ≤ 5%. La lógica es la siguiente: una probabilidad alta con baja integridad de los datos subyacentes puede ser un artefacto de la calibración. VERDE-AUDIT señaliza «la probabilidad es alta pero hay algo en la arquitectura de los datos que requiere verificación».

Los umbrales θ_IPU, θ_A y θ_Conf no tienen valores universales: se calculan mediante una función de utilidad explícita U = π·TPR − c·FPR (donde π es el beneficio de acertadamente clasificar GREEN y c es el costo de un falso positivo). Estos parámetros se fijan por dominio, se publican y se someten a análisis de sensibilidad: cambios de ±10% en los costos no deben mover U más del 2%.

Valores de referencia (release defaults)

En ausencia de calibración específica por dominio, se adoptan los siguientes valores iniciales:

$\tau_V = 0.80, \tau_R = 0.40$

$\theta_{IPU} = 0.65, \theta_A = 0.55, \theta_{Conf} = 0.60$

Estos valores son provisionales y deben ser recalculados por dominio usando la función de utilidad U=π·TPR−c·FPR (con π y c publicados).

5.3 BLACK: el estado que la mayoría de los sistemas de métricas ignora

El estado NEGRO es la contribución más contraintuitiva de LUM al campo de la gobernanza científica. La mayoría de los sistemas de métricas distinguen «bien» de «mal». LUM distingue cuatro categorías, y la más peligrosa no es el no-cierre sino el sobrecierre patológico.

> ▶ **¿Qué es el problema de Goodhart?** La ley de Goodhart, formulada por el economista Charles Goodhart, dice: "Cuando una medida se convierte en un objetivo, deja de ser una buena medida". Ejemplo clásico: si un hospital es evaluado por tiempo de espera, puede optimizar ese número dando de alta a pacientes antes de que estén listos, o clasificando consultas para evitar registrar esperas largas. El tiempo de espera baja, pero la calidad de atención no mejora. En ciencia: si un campo es evaluado por número de publicaciones en revistas de alto impacto, los investigadores optimizan para publicar (p-hacking, HARKing, fragmentación de resultados), no para producir conocimiento sólido. LUM no resuelve Goodhart por decreto: ningún sistema puede. Lo que LUM hace es convertir el gaming en detectable: el estado NEGRO, el leakage_score, el missing_comparisons_ratio y el entropy_protocols son métricas diseñadas específicamente para detectar cuando alguien optimiza el sistema en lugar de mejorar el campo. Si el gaming es sofisticado y no deja señales en esas métricas, LUM no lo detecta. Eso es un límite honesto que el sistema declara en la Sección 9.

Un campo en NEGRO no tiene poco cierre: tiene demasiado, pero del tipo equivocado. Las señales de sobrecierre son: (a) monocriterio colapsado —todos los estudios usan el mismo método y los resultados son perfectamente consistentes porque no pueden ser de otra manera—; (b) supresión de comparaciones —se desalientan o imposibilitan los estudios que podrían falsificar el consenso—; (c) leakage —la información del futuro contamina el entrenamiento, produciendo sobreestimación sistemática de cierre—; (d) circularidad de validación —el mismo corpus se usa para entrenar y validar el modelo.

El sistema computa métricas específicas de sobrecierre: entropy_protocols (qué tan variable es la metodología), missing_comparisons_ratio (qué fracción de comparaciones posibles no se realizan), leakage_score y validation_dependency. Cuando estas métricas superan los umbrales θ_H, θ_S, θ_L y θ_C, el sistema activa NEGRO independientemente de p_cal.

La consecuencia institucional del NEGRO es bloqueo total: no se publica semáforo, no se toman acciones basadas en el campo, y se activa auditoría externa. El NEGRO no dice que el campo está equivocado; dice que sus mecanismos de validación están comprometidos y no pueden producir dictámenes confiables.

Protocolo Goodhart (Red Team + Alarmas + Consecuencias)

Objetivo: LUM-PE no postula que el gaming sea imposible. Postula que, cuando aparece, debe disparar señales computables que **bloqueen acción** y fuercen auditoría adversarial.

5.3.1 Señales mínimas de Goodhart institucional (gatillos de sobrecierre)

Se definen cuatro métricas mínimas, todas reportadas en OUTPUT.overclosure_metrics:

- leakage_score (fuga temporal/labels/p-hacking): si ≥ 0.80 ⇒ **BLACK automático**
- missing_comparisons_ratio (supresión comparativa): si ≥ 0.80 ⇒ **BLACK automático**
- validation_dependency (circularidad/autoridad como muleta): si ≥ 0.85 ⇒ **BLACK automático**
- entropy_protocols (monocultura operacional): si 0 < ≤ 0.05 ⇒ **BLACK automático**

5.3.2 Consecuencia operativa (no solo diagnóstico)

Si state = BLACK:

1. PSNC.required = true
2. **Prohibido** publicar “GREEN” en cualquier documento institucional derivado
3. El dictamen se etiqueta como **NO-ACTION** hasta completar auditoría adversarial

5.3.3 Red Team obligatorio (para declarar robustez de BLACK)

Un estado BLACK se considera “defensa teórica” hasta completar un ciclo de red teaming:

- Ataques: p-hacking de ventanas, leakage inducido, supresión de comparaciones, “core laundering” (forzar CPV alto), ingeniería de κ_conf artificialmente baja
- Éxito del red team = producir GREEN manteniendo fraude real
- Éxito de LUM = degradar a AMBER/BLACK/INVALID con evidencia

Salida del red team: release_artifacts/red_team/rt_report_v0.1.0.md con escenarios, métricas, bundles y huellas.

5.3.4 Separación institucional (no como deseo, como requisito verificable)

LUM-PE exige separación de roles como condición auditable:

- Definición de thresholds (Role T)
- Evaluación (Role E)
- Auditoría (Role A)
  En releases públicos, se documenta governance_manifest.json con hashes de configuración y firmas (o responsables). Si no existe, el release se etiqueta **AUDIT: LIMITED**.

5.4 INVALID: cuando el contrato no se cumple

INVALID es técnicamente un metaestado, no un resultado del modelo: es la respuesta del sistema cuando el contrato LUM-I/O está incompleto, inconsistente o viola sus reglas lógicas internas. Un dictamen INVALID no puede publicar semáforo porque el sistema no completó una ejecución válida.

Las condiciones que producen INVALID incluyen: falta de cualquier campo obligatorio del contrato; $\tau_V \leq \tau_R$ (el umbral GREEN no es mayor que el umbral RED, lo que hace la decisión imposible); absence de calibración cuando se reporta p_close_calibrated; Cons ≠ 1 − κ_conf (inconsistencia interna del reporte); o leakage confirmado en el evidence_set.

5.5. Paquete de Reproducibilidad (Release Gate)

Este documento y el motor LUM solo pueden considerarse **auditables** si se publican los siguientes artefactos mínimos:
(1) evidence_set_table: lista de campos evaluados + ventana Δ + estado LUM por campo + (IPU, CPV, A_norm/A*, κ_conf, Conf, coverage/Sh).

A* vs A_norm: pérdida por aproximación (Loss Budget)

En implementación real, A_norm puede operar como aproximación de A*. Para que A_norm no degrade el sistema a "caricatura práctica", se exige publicar un **loss budget** por dominio:

- corr(A_norm, A*) en un set de referencia (mínimo n=30 instancias por dominio)
- error medio absoluto MAE(A_norm, A*)
- condición de validez: si corr < 0.70 o MAE > 0.15, A_norm queda marcado como **UNTRUSTED** y el sistema bloquea GREEN por AND_min (θ_A no puede cumplirse vía A_norm).

Si A* no puede computarse por limitaciones técnicas, el release se etiqueta **AUDIT: LIMITED** y se publica explícitamente: "A_norm usado sin validación contra A*

(2) z_scales.json: medias y desviaciones estándar **congeladas por versión** (z_scales_id).
(3) thresholds.json: valores de referencia de θ_{IPU}, θ_A y θ_{Conf} (o reglas de derivación + ejemplos por dominio).
(4) calibration_manifest.json: método (Platt/isotónica), hash del dataset externo, split, fecha y métricas (ECE, Brier).
Si cualquiera falta, el dictamen debe etiquetarse como: "RELEASE: LIMITED / NO-AUDIT".

z_scales_id (congelación obligatoria)
La normalización z-score usa escalas **congeladas por versión** publicadas en z_scales.json. Si el evaluador calcula sus propias escalas, el dictamen queda marcado como **NO-COMPARABLE**.

5.6 Paquete de Evidencia Pública (Hello World) — Obligatorio para v0.1.0

Para evitar que LUM-PE sea interpretado como "caja negra sin datos", este release incluye un **Hello World auditable**: un único campo de referencia **completo**, publicado como un bundle reproducible.
Contenido mínimo del Hello World (archivos públicos):

1. release_artifacts/hello_world/evidence_set_table.csv (1 campo, 1 fila)
2. release_artifacts/hello_world/z_scales.json (escalas congeladas)
3. release_artifacts/hello_world/thresholds.json (umbrales)
4. release_artifacts/hello_world/calibration_manifest.json (calibración: declarada o "uncalibrated")
5. release_artifacts/hello_world/bundle.json (INPUT/CONFIG/MODEL/OUTPUT/AUDIT)

Regla: si el dataset completo (p.ej., “45 campos”) no puede publicarse por confidencialidad, el proyecto se declara **AUDIT: LIMITED** y el Hello World funciona como **prueba mínima de integridad** (la tubería corre, produce huella y se valida de extremo a extremo).
Etiqueta de Release:

- **AUDIT: FULL** = dataset completo + artefactos congelados por versión + calibración reproducible.
- **AUDIT: LIMITED** = Hello World auditable + tabla parcial agregada (sin datos sensibles), sin claims de performance generalizable.

6. PSNC: la ingeniería del no-cierre

El Plan de Salida de No-Cierre (PSNC) es la parte del sistema que más frecuentemente se subestima. En la mayoría de los marcos de evaluación científica, cuando el resultado es negativo la respuesta es «se necesita más investigación», lo cual es tan informativo como decir que si tienes hambre debes comer.

El PSNC de LUM produce un diagnóstico causal: identifica qué tipo de deficiencia impide el cierre y prescribe acciones concretas para resolverla. La lógica es que el no-cierre tiene múltiples causas posibles con remedios diferentes, y confundirlas produce planes de investigación equivocados.

6.1 La matriz diagnóstico-acción

Variable diagnóstica	Valor que señaliza problema	Diagnóstico	Acción PSNC
Sh (sombra/shadow)	Alta (Sh > θ_Sh)	Falta instrumentación o cobertura del corpus	Aumentar coverage: ampliar instrumentación, acceso a datos, colaboraciones.
H_s (entropía espectral)	Alta con SNR baja	La señal está oculta en ruido de medición	Denoise: mejorar instrumentos de medición, filtrado, control de ruido.
I (interferencia)	Alto	Mecanismos causales distintos están mezclados en el modelo	Separar mecanismos, rediseñar el modelo causal, segmentar el corpus.
κ_conf (contradicción)	Alta	Incompatibilidades reales no resueltas en el corpus	Resolver incompatibilidades: preregistrar definición de contradicción, replicar estudios clave, abrir comparaciones.
Conf (incertidumbre operacional)	Baja	El evaluador no tiene suficiente información para medir los ejes con confianza	Bloquear claims. Aumentar n efectivo, mejorar diseño, calcular intervalos de confianza formales.
IPU	Bajo	Alta frecuencia o gravedad de defectos metodológicos	Reducir errores: protocolos de preregistro, replicación, auditoría de métodos.

La ganancia del PSNC sobre «se necesita más investigación» es que cada acción está condicionada al diagnóstico. Si el problema es Sh alta (falta de instrumentación), agregar más estudios con los mismos instrumentos no mejora el PSNC: la acción correcta es diferente. Si el problema es I alto (mecanismos mezclados), la acción correcta es rediseño del modelo causal, no más datos con el mismo diseño.

6.2 PSNC-D: el plan de demarcación

Un tipo especial de PSNC es el PSNC-D, que se activa cuando la Capa −1 detecta M-TOT o I-NOR. Aquí el plan no es «más evidencia» sino «definir el objeto»: (a) identificar qué conceptos totalizantes causan el pseudo-problema; (b) imponer el recorte operatorio necesario; (c) traducir el mandato normativo a objetivos medibles si es posible; (d) identificar qué subproblemas sí son resolubles ya. El PSNC-D no resuelve mitos; los convierte, si se puede, en problemas con cierre.

7. Gobernanza, versionado y huella criptográfica

Un sistema de evaluación sin gobernanza es un sistema que alguien puede corromper sin dejar rastro. La gobernanza de LUM está diseñada para que la corrupción sea difícil y que cuando ocurra, deje huella detectable.

7.1 Por qué la gobernanza es un requerimiento epistémico, no burocrático

Esta distinción importa porque cambia cómo se interpreta toda la arquitectura de versionado, footprint y freeze. Un sistema de auditoría podría verse como una capa de burocracia sobre el modelo estadístico. El argumento correcto es el contrario: sin trazabilidad, el modelo estadístico no cumple el requisito de auditabilidad establecido en la Sección 1, lo que lo hace epistémicamente incompleto. Un criterio sin trazabilidad no puede ser auditado por terceros independientes; por tanto, no puede ser falsificado; por tanto, no satisface los requisitos mínimos de un criterio operacional.

7.2 El sistema de versiones

LUM distingue cuatro niveles de versionado, cada uno con su propia cadena de consecuencias:

engine_version: versión del motor de cálculo. Cambia si se modifica la lógica de cualquier índice, la ecuación maestra o la regla de decisión.

packs_version: versión de los checklists de verificación por dominio. Cambia si se agregan, modifican o eliminan checks en cualquier pack.

cpv_semantics_version: versión de la definición semántica de CPV. Cambia si se modifica la variante (espacial vs. predictiva) o la definición del núcleo central.

Delta_policy: versión de la política de ventana temporal. Cambia si se modifica Δ o si se pasa de política fija a adaptativa.

Cualquier cambio en thresholds, AND_min, normalización, overclosure_thresholds, CI_policy o hashing_policy implica un bump de versión MAJOR. Esto significa que dictámenes históricos con diferentes versiones no son directamente comparables, y el sistema lo explicita en lugar de ocultarlo.

7.3 Drift, freeze y obsolescencia

Un dictamen VERDE emitido en el tiempo t envejece. LUM implementa dos mecanismos para manejar el envejecimiento:

Drift detection: el sistema monitorea el log-loss de calibración en flujo entre ventanas consecutivas. Si el log-loss supera el umbral δ durante N ventanas, el sistema activa freeze: el dictamen actual se congela y no puede ser usado para nuevas acciones hasta que se recalibra. El descongelamiento requiere una mejora de calibración que supere un umbral de histéresis (típicamente ≤ 0.02) para evitar oscilaciones rápidas.

Obsolescencia: todo dictamen tiene una función de envejecimiento λ(t) = f(drift, Δt, ...) que cuantifica cuándo el dictamen debe ser reevaluado. Un dictamen VERDE sobre psicología social de 2015 no es aplicable en 2026 sin revisión.

7.4 La huella criptográfica (footprint SHA-256)

El footprint es la firma digital del dictamen: un hash SHA-256 calculado sobre la canonicalización del input completo (problem_spec, demarcation, solution, evidence_set, config, model_params). Su

función es doble: detectar si cualquier componente del dictamen fue modificado después de la emisión, y permitir la reconstrucción exacta del dictamen dado el mismo input.

> ▶ **¿Qué es SHA-256 y por qué importa?** Un hash criptográfico es una función matemática que convierte cualquier texto o archivo en una cadena de longitud fija (en el caso de SHA-256, 64 caracteres hexadecimales). Sus propiedades clave: (1) Determinismo: el mismo input siempre produce el mismo hash. (2) Sensibilidad extrema: cambiar un solo carácter del input produce un hash completamente diferente. (3) Irreversibilidad: no se puede reconstruir el input a partir del hash. Ejemplo: el texto “cierre” produce un hash completamente diferente al de “Cierre” (solo cambia la mayúscula). ¿Para qué sirve en LUM? Cuando se emite un dictamen, se calcula el SHA-256 de todos los inputs (datos, parámetros, configuración). Si alguien modifica el dictamen después (cambia un umbral, altera los datos), el hash ya no coincide. Es como un sello de lacre digital: si está roto, sabes que algo fue modificado. La canonicalización es el paso previo: antes de hashear, el input se convierte a una representación estándar (ordenar claves del JSON, eliminar espacios irrelevantes) para garantizar que dos representaciones del mismo dictamen produzcan el mismo hash. Sin canonicalización, un simple reordenamiento de campos daría un hash diferente aunque el contenido sea idéntico.

El footprint no garantiza que el dictamen sea correcto; garantiza que el dictamen que se publica es exactamente el que produjo el sistema con los inputs declarados. Si el footprint no coincide, el dictamen fue modificado —ya sea por error o por fraude— y debe considerarse INVALID.

La canonicalización es el proceso por el cual el input se convierte a una representación única antes de hashear: se eliminan espacios irrelevantes, se ordenan las claves del JSON, se declara el serializer_v utilizado. Sin canonicalización estándar, dos representaciones del mismo dictamen producirían hashes diferentes.

8. El Contrato LUM-I/O: el dictamen como objeto verificable

El contrato LUM-I/O es el esqueleto formal que convierte un cálculo estadístico en un dictamen institucionalmente deployable. Define la tupla de cinco componentes que debe satisfacer cualquier ejecución válida de LUM:

> ▶ **El contrato LUM-I/O en términos simples:** Piensa en el contrato como el expediente clínico de una evaluación: debe contener todo lo necesario para que cualquier médico (evaluador) independiente llegue al mismo diagnóstico dado los mismos datos. INPUT: ¿qué campo se evalúa, en qué ventana temporal, con qué evidencia? Es como el historial del paciente. Debe ser completo, trazable y no puede modificarse después sin nueva versión. CONFIG: ¿qué umbrales y parámetros se usaron? Es como el protocolo de análisis. Está congelado antes del análisis: no se puede ajustar para obtener el resultado deseado. MODEL: ¿qué modelo estadístico se ejecutó? Incluye los coeficientes estimados, el tipo de calibración y las métricas de ajuste. Es como el método diagnóstico. OUTPUT: ¿qué resultados produjo el sistema? Incluye los índices, las probabilidades con intervalos de confianza, el estado del semáforo y su justificación. Es el diagnóstico. AUDIT: ¿cómo se puede verificar todo lo anterior? Incluye el footprint SHA-256, las versiones de todos los componentes y los identificadores de los datasets usados. Es la firma del médico con sello verificable. Si cualquiera de los cinco componentes falta, el dictamen es INVÁLIDO por construcción: no porque sea incorrecto, sino porque no puede ser auditado.

$$L = \langle \text{INPUT}, \text{CONFIG}, \text{MODEL}, \text{OUTPUT}, \text{AUDIT} \rangle$$

Si cualquier componente falta o es internamente inconsistente, el sistema devuelve validity.status = INVALID y el semáforo no se publica.

8.1 INPUT: lo que debe declararse antes de medir

El INPUT tiene cinco secciones obligatorias: (a) domain —descripción del campo y sus límites exactos, incluyendo scope_in y scope_out—; (b) time_reference —el tiempo t del dictamen en formato ISO-8601 y la unidad oficial del análisis—; (c) window —el valor de Δ y sus unidades—; (d) evidence_set —el conjunto completo de datasets, documentos y protocolos, cada uno con hash o URI; (e) definitions preregistradas —la definición operacional del evento de cierre y la definición de incompatibilidad.

Las reglas duras del INPUT son: Δ no puede cambiarse sin nueva versión; event_of_closure debe ser operacional (observable, medible, re-ejecutable); incompatibility debe estar preregistrada por dominio; evidence_set debe ser completo, enumerable y hasheable. Un evidence_set no hasheable produce INVALID por construcción, porque no puede auditarse.

8.2 CONFIG: los parámetros que quedan congelados

La CONFIG congela los parámetros que definen el comportamiento del modelo para esa ejecución: los umbrales del semáforo (τ_V y τ_R, con regla τ_V > τ_R), los mínimos AND_min (θ_IPU, θ_A, θ_Conf), los parámetros de normalización (clipping ε_low y ε_high, el ID de escalas z congeladas), los umbrales de sobrecierre (θ_H, θ_S, θ_L, θ_C), la política de intervalos de confianza (método, B iteraciones mínimo 2000, seed) y la política de hashing.

La regla de oro de la CONFIG: cualquier cambio implica nueva versión MAJOR. Esto impide el ajuste post-hoc de parámetros para obtener el estado deseado.

8.3 MODEL: el backend estadístico y su calibración

El MODEL especifica el tipo de modelo (hazard_cloglog), el offset (log(Δ)), los predictores en z (IPU_z, CPV_z, A_norm_z, Cons_z) y los coeficientes estimados (α, β, γ, βκ, β_0). La restricción normativa βκ ≥ 0 debe cumplirse; si el modelo estimado la viola, el resultado es inválido.

Si se reporta p_close_calibrated, el MODEL debe incluir la sección calibration con: método (Platt o isotónica), hash del dataset de calibración, regla de split, timestamp y métricas mínimas (ECE, Brier). Sin calibración, p_close_calibrated no puede reportarse.

Modos de implementación (Tier 0 / Tier 1 / Tier 2)

Para reducir barrera de entrada, LUM-PE define tres niveles operativos:

Tier 0 — SPEC/DEMO (sin calibración, score):

- p_close_kind = "score"
- No se reporta probabilidad calibrada
- Se permite implementación con dataset interno mínimo + Hello World auditable

Tier 1 — Operacional (calibración parcial):

- Calibración externa por dominio con dataset reducido
- Se publican z_scales.json + thresholds.json + calibration_manifest.json

Tier 2 — Auditoría completa (FULL):

- Dataset completo (o tabla reproducible equivalente)
- Calibración reproducible (artefacto + hashes)
- Red team completado para BLACK

Regla: un release puede publicarse en Tier 0, pero debe rotularse **AUDIT: LIMITED**.

8.4 OUTPUT: la salida mínima obligatoria

El OUTPUT debe contener: los valores de los cuatro índices con intervalos de confianza; los valores derivados (Cons = 1 − κ_conf); las probabilidades p_close y p_close_calibrated con sus intervalos; el estado LUM con su justificación (que debe ser corta y verificable, no retórica); los diagnósticos auxiliares (Sh, H_s, SNR, I); el PSNC si aplica; y el estado de validez.

La condición de validez del OUTPUT tiene reglas lógicas internas (post-checks) que el schema no puede capturar solo: τ_V > τ_R, ε_low < ε_high, Cons = 1 − κ_conf con tolerancia, si GREEN → AND_min debe pasar, si p_close_calibrated existe → calibration debe existir, IC.low ≤ IC.high. Estas reglas se verifican como post-procesamiento del schema.

8.5 AUDIT: la trazabilidad del dictamen

La sección AUDIT contiene el footprint SHA-256 y los metadatos de auditoría necesarios para reconstruir el dictamen: versiones de todos los componentes, timestamps, identificadores de los datasets y documentos del evidence_set, y el serializer_v usado para la canonicalización.

La publication_minimum define qué campos son obligatorios para publicar el dictamen: has_footprint, has_full_config, has_ci_policy, has_uncertainty, has_evidence. Un dictamen que no cumple publication_minimum puede usarse internamente pero no puede ser citado como evidencia pública.

9. Alcances, límites y respuesta al contraargumento del problema de Goodhart

Un sistema de evaluación que no declare explícitamente sus límites es un sistema deshonesto. LUM tiene límites reales y esta sección los articula sin eufemismos.

9.1 Qué LUM no puede hacer

LUM no decide verdad metafísica. La afirmación «el campo X tiene cierre operativo» no equivale a «el campo X tiene razón» ni a «las intervenciones basadas en X son moralmente correctas». LUM mide cierre operativo suficiente para actuar bajo riesgo controlado. La evaluación del contenido científico —si la teoría es verdadera, si el modelo es correcto— queda fuera del alcance del sistema.

LUM no puede aplicarse a fenómenos intrínsecamente no estacionarios, eventos irrepetibles singulares y procesos estructuralmente dependientes del contexto. Si la producción de identidades sintéticas estables es imposible en principio para el objeto —porque el objeto cambia más rápido que cualquier ventana de evaluación posible— el sistema no puede producir un dictamen válido.

LUM no garantiza que los índices capturan todo lo relevante. IPU, CPV, A_norm y κ_conf son las dimensiones que el sistema puede operacionalizar; puede haber dimensiones del cierre que ningún índice captura. El evaluador es responsable de declarar esa limitación en el OUTPUT.

9.2 El problema de Goodhart: el contraargumento más fuerte

Charles Goodhart formuló lo que se conoce como la Ley de Goodhart: «cuando una medida se convierte en objetivo, deja de ser una buena medida». Si IPU, CPV, A_norm y κ_conf se vuelven criterios formales para obtener financiamiento, aprobación regulatoria o publicación, los actores institucionales aprenderán a optimizar las métricas sin mejorar el cierre real del campo.

Esta es la objeción más fuerte posible contra LUM, y merece una respuesta que no sea evasiva. El contraargumento tiene tres niveles.

Primer nivel: LUM hace el problema de Goodhart visible, no invisible. Cualquier sistema de métricas está sujeto al problema de Goodhart. La diferencia entre LUM y los sistemas sin especificación formal es que en LUM el gaming es detectable: el estado NEGRO, el leakage_score, el missing_comparisons_ratio y el entropy_protocols son métricas diseñadas específicamente para detectar cuando alguien está optimizando el sistema en lugar de mejorando el campo. Sin estas métricas, el gaming ocurre igualmente pero sin dejar rastro.

Segundo nivel: el versionado y la política anti-cambio post-hoc reducen el gaming oportunista. El sistema más común de gaming —ajustar los umbrales después de ver los resultados— está bloqueado por la regla de que cualquier cambio en parámetros produce nueva versión y hace los dictámenes históricos no comparables. Esto no elimina el gaming; lo hace más costoso.

Tercer nivel: el límite honesto. LUM no puede detectar gaming sistémico —cuando todo un ecosistema institucional coopera para optimizar las métricas. Si el órgano que define los thresholds, el que audita los footprints y el que evalúa los campos son el mismo actor o están capturados por el mismo interés, el sistema colapsa. LUM requiere separación institucional entre quienes definen los parámetros, quienes ejecutan las evaluaciones y quienes auditan los footprints. Esta separación no es un detalle de implementación; es un requerimiento arquitectónico del sistema. Su ausencia produce NEGRO por definición.

9.3 Condiciones de frontera: cuándo LUM falla por diseño

LUM falla en al menos cinco condiciones de frontera que el evaluador debe verificar antes de ejecutar el sistema:

(a) Cuando el evidence_set tiene Sh extremadamente alta (cobertura < 30% del corpus relevante), el dictamen no tiene suficiente información para ser fiable en ninguno de los cuatro estados.

(b) Cuando Δ es tan corta que ningún evento de cierre es posible a priori en esa ventana para el tipo de campo. El offset log(Δ) ajusta la ecuación, pero no puede compensar una ventana estructuralmente inadecuada.

(c) Cuando el field se recorta de forma que excluye sistemáticamente la evidencia negativa. Aquí AND_min y el leakage_score pueden detectar el problema, pero no siempre lo hacen si la exclusión es sofisticada.

(d) Cuando los coeficientes del modelo se estiman con menos de 30 eventos de cierre en el conjunto de entrenamiento. La corrección de Firth reduce el sesgo pero no lo elimina con muestras muy pequeñas.

(e) Cuando el evaluador tiene conflicto de interés no declarado. El contrato obliga a declarar afiliaciones y financiamiento en el INPUT, pero no puede verificar la veracidad de esas declaraciones. La auditoría externa es el único mecanismo real para este caso.

Conclusión: Lo que LUM cambia y lo que no

La ganancia concreta de LUM sobre los sistemas existentes no es filosófica sino operacional: transforma el problema de la demarcación de una pregunta que solo puede responderse retóricamente en una pregunta que puede responderse mediante un procedimiento auditable, reproducible y versionado.

Antes de LUM, la respuesta a «¿el campo X está suficientemente desarrollado para guiar acción?» dependía de la autoridad del evaluador, del consenso del gremio o de la intuición del tomador de decisiones. Después de LUM, la respuesta depende de cuatro índices medibles, una ecuación calibrada, un sistema de estados con reglas explícitas y un contrato que puede ser verificado por cualquier evaluador independiente dado el mismo evidence_set.

Lo que LUM no cambia: la ciencia seguirá siendo disputada, los campos seguirán teniendo controversias legítimas, y los actores institucionales seguirán teniendo incentivos para manipular cualquier sistema de métricas que afecte su financiamiento o reputación. LUM no resuelve el problema de la ciencia; proporciona un criterio para evaluar cuándo un campo ha alcanzado suficiente coherencia para que la acción sea racionalmente justificable. Esa es una tarea más modesta que la que promete la retórica científica habitual. Es también la única tarea que un sistema formal honesto puede cumplir.

El cierre científico, entendido como LUM lo define —como la existencia de identidades sintéticas auditadas dentro de una ventana temporal— no es el destino final del conocimiento. Es la condición mínima para actuar con responsabilidad sobre lo que el conocimiento dice.

AUDITORÍA FINAL

Tesis final

Hasta donde conocemos, no existe un estándar unificado que combine demarcación, verificación tipada (packs), modelo de evento por ventana Δ y auditoría criptográfica en un solo criterio operativo. LUM es el primer sistema formal que operacionaliza el cierre científico como evento probabilístico en ventana temporal, cuantificado por cuatro ejes ortogonales, integrado en un modelo calibrado y gobernado por un contrato auditable —produciendo dictámenes reproducibles, temporalmente revisables y falsificables por terceros independientes.

Fricciones resueltas

Fricción 1 (filosofía sin procedimiento vs. ciencia sin brújula): resuelta mediante la distinción entre criterios descriptivos (Popper, Kuhn, Lakatos, Bueno) y criterios operacionales (LUM), y la especificación de qué falla en cada criterio clásico y en qué dimensión.

Fricción 2 (permanencia vs. temporalidad): resuelta mediante la formulación del cierre como evento temporal (axioma A2 de revisabilidad) y el modelo clog-log con offset log(Δ).

Fricción 3 (medición real vs. numerología): resuelta mediante la especificación operacional de cada índice, sus versiones robustas, sus modos de falla y sus mecanismos de auditoría.

Fricción 4 (decisión honesta vs. GREEN fraudulento): resuelta mediante los guardarraíles AND_min, el estado NEGRO para sobrecierre, VERDE-AUDIT para la zona de ambigüedad con alta probabilidad y ECE controlado, y INVALID para violaciones del contrato.

Fricción 5 (gobernanza auditable vs. sistema sin trazabilidad): resuelta mediante el sistema de versionado, la detección de drift, la política de freeze y el footprint SHA-256.

Afirmaciones fuertes y su soporte

Afirmación 1: «LUM produce AUC=0.88 vs. 0.74 de enfoques clásicos» — [HIPÓTESIS]: afirmada en los documentos fuente (Luminomatics.docx y LUM_Paper_DEFINITIVO_v2.docx) pero el dataset de validación no está accesible en los documentos proporcionados para esta edición. El valor se presenta como reportado por el autor, no como verificado en esta edición.

Afirmación 2: «La restricción $\beta\kappa \geq 0$ es normativa» — Soportada conceptualmente: más consistencia no puede reducir la probabilidad de cierre por definición del modelo. El soporte empírico dependería de la validación sobre los 45 campos (ver HUECO-1).

Afirmación 3: «La corrección de Firth reduce el sesgo para eventos raros» — Soportada por bibliografía estadística estándar (Firth, 1993). No requiere validación adicional en el marco LUM.

Afirmación 4: «El estado NEGRO previene el GREEN fraudulento» — Soportada funcionalmente por el diseño del sistema; su efectividad empírica ante gaming sofisticado es un [HUECO-4]: no se ha publicado una evaluación adversarial sistemática del sistema NEGRO.

Bibliografía:

1. *Barabási, A.-L., & Albert, R. (1999). Emergence of scaling in random networks. Science, 286(5439), 509–512. (doi.org)*

2. *Box, G. E. P. (1976). Science and statistics. Journal of the American Statistical Association, 71(356), 791–799. (doi.org)*

3. *Bueno, G. (1970). El papel de la filosofía en el conjunto del saber. Ciencia Nueva. (FGB - Fundación Gustavo Bueno)*

4. *Bueno, G. (1972). Ensayos materialistas. Taurus. (FGB - Fundación Gustavo Bueno)*

5. *Bueno, G. (1982). El cierre categorial aplicado a las ciencias físico-químicas. En Actas del I Congreso de Teoría y Metodología de las Ciencias (pp. 101–175). Pentalfa. (FGB - Fundación Gustavo Bueno)*

6. *Bueno, G. (1982). Gnoseología de las ciencias humanas. En Actas del I Congreso de Teoría y Metodología de las Ciencias (pp. 315–349). Pentalfa. (FGB - Fundación Gustavo Bueno)*

7. *Bueno, G. (1992–1993). Teoría del cierre categorial (Vols. 1–5). Pentalfa. (FGB - Fundación Gustavo Bueno)*

8. *Bueno, G. (1995). ¿Qué es la ciencia? La respuesta de la teoría del cierre categorial. Pentalfa. (FGB - Fundación Gustavo Bueno)*

9. *Camerer, C. F., Dreber, A., Holzmeister, F., Ho, T.-H., Huber, J., Johannesson, M., Kirchler, M., Nave, G., Nosek, B. A., Pfeiffer, T., Rand, D. G., & Wu, H. (2018). Evaluating the replicability of social science experiments in Nature and Science between 2010 and 2015. Nature Human Behaviour, 2(9), 637–644. https://doi.org/10.1038/s41562-018-0399-z (PubMed)*

10. *Campbell, D. T. (1979). Assessing the impact of planned social change. Evaluation and Program Planning, 2(1), 67–90. https://doi.org/10.1016/0149-7189(79)90048-X (DOI Resolver)*

11. *Chambers, C. D. (2013). Registered reports: A new publishing initiative at Cortex. Cortex, 49(3), 609–610. https://doi.org/10.1016/j.cortex.2012.12.016 (Orca)*

12. *Feyerabend, P. (1975). Against method: Outline of an anarchistic theory of knowledge. New Left Books. (doi.org)*

13. *Firth, D. (1993). Bias reduction of maximum likelihood estimates. Biometrika, 80(1), 27–38. https://doi.org/10.1093/biomet/80.1.27 (doi.org)*

14. *Harrell, F. E. (2015). Regression modeling strategies: With applications to linear models, logistic and ordinal regression, and survival analysis (2nd ed.). Springer. (doi.org)*

15. *Hernán, M. A., & Robins, J. M. (2020). Causal inference: What if. Chapman & Hall/CRC. (doi.org)*

16. *Ioannidis, J. P. A. (2005). Why most published research findings are false. PLoS Medicine, 2(8), e124. https://doi.org/10.1371/journal.pmed.0020124 (doi.org)*

17. *John, L. K., Loewenstein, G., & Prelec, D. (2012). Measuring the prevalence of questionable research practices with incentives for truth telling. Psychological Science, 23(5), 524–532. https://doi.org/10.1177/0956797611430953 (doi.org)*

18. *Kuhn, T. S. (2012). The structure of scientific revolutions (4th ed., 50th anniversary ed.). The University of Chicago Press. (doi.org)*

19. *Lakatos, I., & Musgrave, A. (Eds.). (1970). Criticism and the growth of knowledge. Cambridge University Press. (doi.org)*

20. *Manheim, D. (2023). Building less-flawed metrics: Understanding and creating better measurement and incentive systems. Patterns, 4(10), 100842. https://doi.org/10.1016/j.patter.2023.100842 (ScienceDirect)*

21. *Meadows, D. H. (2008). Thinking in systems: A primer. Chelsea Green Publishing. (doi.org)*

22. *Nosek, B. A., & Lakens, D. (2014). Registered reports: A method to increase the credibility of published results. Social Psychology, 45(3), 137–141. https://doi.org/10.1027/1864-9335/a000192 (Pure)*

23. *Open Science Collaboration. (2015). Estimating the reproducibility of psychological science. Science, 349(6251), Article aac4716. https://doi.org/10.1126/science.aac4716 (Science)*

24. *Pearl, J. (2009). Causality: Models, reasoning, and inference (2nd ed.). Cambridge University Press. https://doi.org/10.1017/CBO9780511803161 (doi.org)*

25. *Peng, R. D. (2011). Reproducible research in computational science. Science, 334(6060), 1226–1227. https://doi.org/10.1126/science.1213847 (doi.org)*

26. *Popper, K. R. (1959). The logic of scientific discovery. Basic Books. (doi.org)*

27. *Sandve, G. K., Nekrutenko, A., Taylor, J., & Hovig, E. (2013). Ten simple rules for reproducible computational research. PLoS Computational Biology, 9(10), e1003285. https://doi.org/10.1371/journal.pcbi.1003285 (DOI Resolver)*

28. *Simmons, J. P., Nelson, L. D., & Simonsohn, U. (2011). False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. Psychological Science, 22(11), 1359–1366. https://doi.org/10.1177/0956797611417632 (doi.org)*

29. *Simon, H. A. (1996). The sciences of the artificial (3rd ed.). MIT Press. (doi.org)*

30. *Steyerberg, E. W. (2019). Clinical prediction models (2nd ed.). Springer. (doi.org)*

31. *Stodden, V., McNutt, M., Bailey, D. H., Deelman, E., Gil, Y., Hanson, B., Heroux, M. A., Ioannidis, J. P. A., & Taufer, M. (2016). Enhancing reproducibility for computational methods. Science, 354(6317), 1240–1241. https://doi.org/10.1126/science.aah6168 (PubMed)*

32. *Strathern, M. (1997). "Improving ratings": Audit in the British university system. European Review, 5(3), 305–321. https://doi.org/10.1017/S1062798700002660 (doi.org)*

33. *Van Calster, B., Nieboer, D., Vergouwe, Y., De Cock, B., Pencina, M. J., & Steyerberg, E. W. (2016). A calibration hierarchy for risk models was defined: From utopia to empirical data. Journal of Clinical Epidemiology, 74, 167–176. https://doi.org/10.1016/j.jclinepi.2015.12.005 (PubMed)*

34. *von Bertalanffy, L. (1968). General system theory: Foundations, development, applications. George Braziller. (doi.org)*

35. *Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of "small-world" networks. Nature, 393, 440–442. (doi.org)*

36. *Dürer, A. (1514). Melencolia I [Grabado]. National Gallery of Art*

Recursos y código

Todos los artefactos de LUM-PE están disponibles públicamente bajo licencia MIT (código y especificación) y CC BY 4.0 (dataset). La Contract Specification v1.0.0 (CONTRACT.md, schema JSON y ejemplos) fue publicada el 24 de marzo de 2026 en GitHub y Zenodo.

GitHub (código fuente, especificación de contrato y documentación)

https://github.com/julespintor-tech/lum-pe

Incluye: código fuente (lum_pe_product/), dataset sintético (dataset/), validación (validation/) y Contract Specification v1.0.0 (spec/CONTRACT.md, spec/lum_contract_schema.json, spec/examples/).

Zenodo — DOI permanente, dataset sintético y Contract Spec v1.0.0

https://doi.org/10.5281/zenodo.19211260

Contiene: lum_spec_v1.0.0.zip (Contract Specification completa). Publicado el 24 de marzo de 2026. Versión anterior: 10.5281/zenodo.19142481 (Hello World auditable, v0.1.0).

OSF — Proyecto de ciencia abierta

https://osf.io/av6zy

Cómo citar este trabajo

Rojas Aguayo, Julio David (2026). *LUM-PE: Luminomatics Public Edition v0.1.0 + Contract Spec v1.0.0*. Zenodo. https://doi.org/10.5281/zenodo.19211260

— FIN DEL DOCUMENTO —

Julio David Rojas Aguayo | Laboratorio de Luminomática | CDMX, 24 de marzo de 2026

Release status: v0.1.0 (Public) — **AUDIT: LIMITED** (sin dataset completo) / **AUDIT: FULL** (si se incluye evidence_set_table + z_scales + thresholds + calibration manifest).