Facsímil 08

La ciencia de los datos

Capítulo 08

Recapitulación de ciencia de datos

686f6c61 · 2026

Capítulo 08: Recapitulación de ciencia de datos

Entrando en el cierre

Este capítulo no introduce una técnica aislada. Sirve para comprobar si el facsímil entero se ha convertido en criterio de ingeniería. Después de hablar de contratos, calidad, splits, embeddings, slices, operación y causalidad, la pregunta ya no es “qué métrica sale”, sino “qué decisión puedo defender con lo que sé”.

Ese matiz cambia la forma de trabajar. Un dataset puede tener columnas correctas y aun así no estar listo. Un experimento puede estar bien planteado y aun así no tener muestra suficiente. Un sistema puede mejorar la media global y empeorar justo el segmento que más nos importa. Por eso el cierre del facsímil no pide una respuesta decorativa: pide reunir evidencias, ejecutar un gate, escribir una decisión y dejar claro qué se corrige antes de publicar.

Qué deberías poder hacer al terminar

Este facsímil empezó con una idea sencilla: en IA, los datos no son un trámite. Son parte del sistema. Si el dataset está mal definido, si el contrato no existe, si el split engaña, si los slices fallan, si falta trazabilidad o si confundimos predicción con intervención, el modelo puede parecer sofisticado y aun así tomar malas decisiones.

Al cerrar el facsímil deberías poder hacer esto:

RESULTADO DE APRENDIZAJE	EVIDENCIA DE QUE LO SABES HACER
Auditar un dataset antes de usarlo.	Revisas contrato, columnas, valores, licencias, linaje y trazabilidad.
Diseñar una evaluación honesta.	Separas train, validation y test sin leakage.
Leer features y embeddings como decisiones.	No conviertes columnas en vectores sin contrato.
Medir slices críticos.	No publicas una media global si falla un segmento importante.
Operar datos en producción.	Escribes SLIs, SLOs, runbooks, trazas y gates.
Diseñar un experimento aplicable.	Separas predicción de intervención, ATE de CATE y A/B de observacional.
Entregar una decisión defendible.	Generas scorecard, reporte y documento de salida.

Esta tabla no está pensada como una lista de objetivos de un temario. Es más útil leerla como una pequeña prueba de realidad. Si una persona te da un CSV, un modelo, una métrica y una fecha de publicación, ¿qué preguntas harías antes de aceptar la release? ¿Qué pedirías ver? ¿Qué parte podrías automatizar y qué parte exigiría revisión humana? En ciencia de datos aplicada a IA, saber programar una comprobación es solo la mitad; la otra mitad es saber cuándo esa comprobación cambia una decisión.

La diferencia con una práctica puramente académica está ahí. En una práctica puedes aprobar con una respuesta correcta. En un sistema real tienes que dejar rastro de por qué esa respuesta era defendible en ese momento, con esos datos, esos límites y esa evidencia. Por eso este cierre insiste tanto en contratos, trazas, slices, gates y experimentos. Son palabras poco vistosas, pero sostienen el trabajo cuando el proyecto deja de ser una demo.

La frase de cierre:

> La ciencia de datos en IA no consiste en encontrar una métrica bonita. Consiste en construir una decisión que aguante preguntas.

Lo que hemos construido

El recorrido del facsímil puede leerse como una cadena:

La cadena no es una ocurrencia editorial. Datasheets for Datasets consolidó la idea de documentar motivación, composición, recogida, usos y límites de un dataset antes de ponerlo a circular.[1] Model Cards hizo algo parecido para modelos, obligándonos a declarar uso previsto, métricas, grupos evaluados y limitaciones.[2] Y el ML Test Score recordaba una cosa muy de ingeniería: un sistema de ML no está maduro solo porque tenga una métrica alta; necesita tests de datos, monitorización, reproducibilidad y gestión de cambios.[3]

CAPÍTULO	PREGUNTA	ARTEFACTO MENTAL
01	¿Qué dato tenemos y de dónde viene?	Contrato, linaje y dataset card.
02	¿El dato cumple lo mínimo?	Gate de calidad.
03	¿Medimos sin engañarnos?	Split y manifiesto de evaluación.
04	¿Cómo representamos la información?	Features, embeddings y búsqueda.
05	¿A quién falla la decisión?	Slices, políticas y mitigación.
06	¿Qué pasa cuando llega producción?	DataOps, drift, trazas y postmortem.
07	¿Una acción cambia algo?	Experimento, causalidad y rollout.

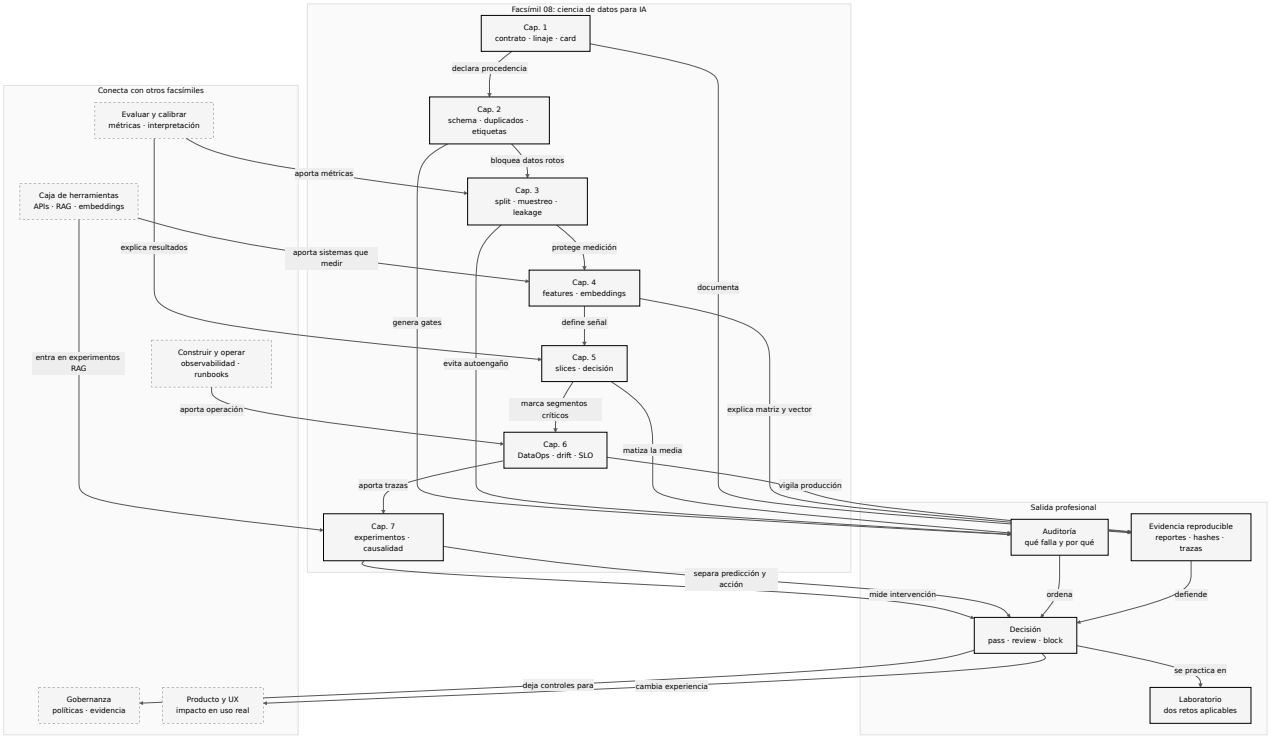
Leída así, la ciencia de datos para IA deja de ser una colección de técnicas y se convierte en una disciplina de evidencia. Cada capítulo añade una defensa contra una forma distinta de autoengaño: usar datos sin permiso, limpiar sin contrato, medir con leakage, vectorizar sin trazabilidad, publicar una media global, operar sin runbook o llamar causal a una correlación. La cadena completa importa porque el sistema falla por el eslabón más débil, no por el capítulo que mejor entendimos.

También cambia la manera de estudiar el facsímil. No hace falta memorizar cada archivo ni cada paso. Lo importante es reconocer el patrón: antes de usar datos, declaro qué son; antes de evaluar, protejo la separación; antes de vectorizar, entiendo qué señal estoy fabricando; antes de publicar, miro a quién falla; antes de escalar, observo producción; y antes de afirmar impacto, diseño una intervención medible. Si ese patrón queda dentro, el lector puede llevarlo a otro dominio aunque cambien los nombres de columnas, proveedores o herramientas.

Si una persona termina este facsímil y solo recuerda una cosa, que sea esta: **cada métrica necesita una historia de procedencia, una unidad de análisis, una ventana, un contrato y una decisión permitida**.

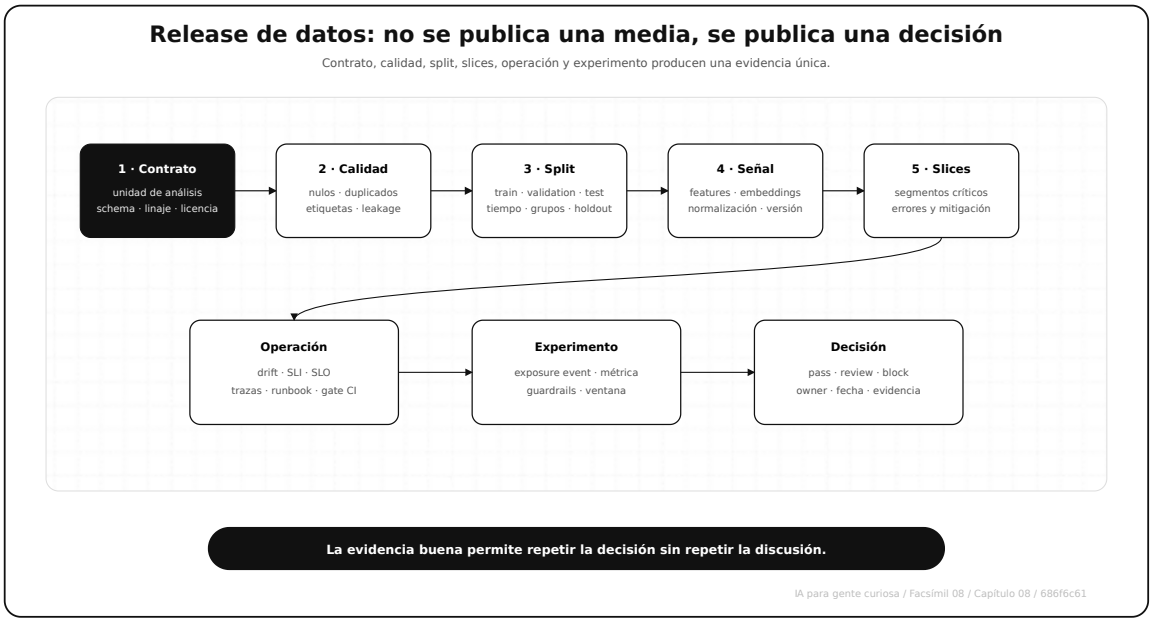
Cómo encaja todo

Este mapa une el facsímil completo. No intenta repetir cada tabla, sino mostrar el flujo profesional: del dato inicial a la decisión de publicación. La lectura buena no es lineal del todo: contrato, calidad, split, representación, slices, operación y causalidad se revisan entre sí.



Anatomía visual: de dataset a decisión

El cierre del facsímil necesitaba una figura propia porque la ciencia de datos no termina en el dataframe. Termina cuando alguien puede explicar qué dato entró, qué se midió, qué se corrigió, qué quedó fuera y qué decisión se tomó.



La revisión de datos une contrato, calidad, medición, segmentos, operación y experimento. El resultado útil no es “la métrica sube”, sino una decisión trazable.

En el día a día: quién mira qué

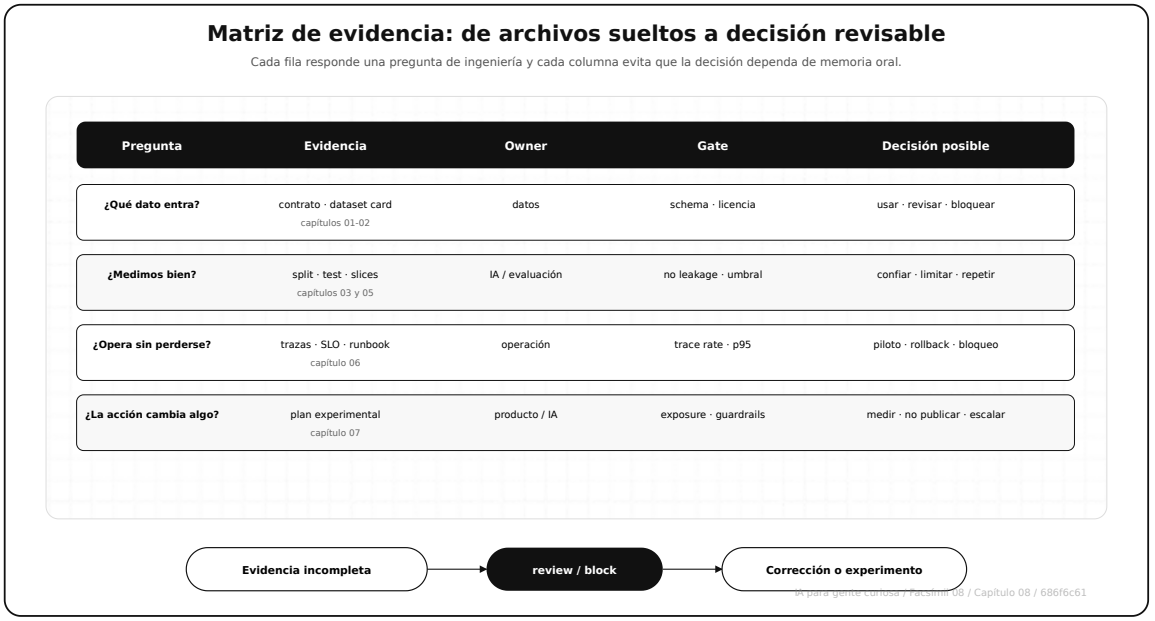
En una práctica universitaria puede parecer que todo termina cuando el script imprime `block` o `review`. En un equipo real, ese es solo el principio de la conversación. Una persona de datos mirará si el contrato y el linaje son defendibles. Una persona de IA mirará si el split, la métrica y los slices permiten confiar en la evaluación. Una persona de producto preguntará si el fallo afecta a usuarios reales y qué alcance se puede permitir. Operación preguntará si hay trazas, SLOs y rollback. Y una persona con responsabilidad de cumplimiento o gobernanza preguntará si la evidencia se conserva, si hay owner y si la decisión se puede reconstruir dentro de tres meses.

Esta separación de responsabilidades no es burocracia. Amershi y otros documentaron que los sistemas de aprendizaje automático introducen fricciones específicas en requisitos, datos, evaluación, monitorización y coordinación entre perfiles.[4] El NIST AI RMF insiste en una idea compatible: gestionar riesgo de IA exige mapear, medir, gestionar y gobernar, no solo optimizar una métrica.[5] Traducido a este facsímil: cada evidencia debe tener una pregunta, un archivo, un responsable y una decisión posible.

Por eso no basta con entregar outputs. El alumno debería poder decir: “esta evidencia sostiene este check; este check permite esta decisión; esta decisión deja este riesgo residual; y este riesgo define el siguiente experimento”. Cuando esa cadena existe, el trabajo se puede revisar. Cuando no existe, solo tenemos una carpeta con archivos.

Anatomía visual: matriz de evidencia de release

Esta figura añade la parte que suele faltar cuando se habla de ciencia de datos de forma demasiado abstracta: el expediente. Un expediente de release no es una carpeta enorme; es una matriz donde cada fila responde a una pregunta técnica y cada columna obliga a conectar evidencia, owner y decisión.



> La matriz obliga a que cada pregunta tenga evidencia, owner, gate y decisión. Así el trabajo deja de ser una lista de archivos y se convierte en una revisión profesional.

Cuadernos para practicar

Has cuidado los datos a lo largo del facsímil; estos cuadernos te dejan ver, en directo, las dos formas más comunes de engañarte con ellos. Son *notebooks* que se abren en Google Colab —gratis, en el navegador— o te puedes descargar. Cada uno, explicado paso a paso, con salidas reales, y anclado al capítulo del que sale.

Leakage: cómo te engañas sin querer

Qué practicas: provocar y detectar la fuga de datos, el error que infla resultados sin dar error. **Dónde encaja:** capítulos 2 y 3 (calidad de datos y *leakage*; *splits* y medir sin engañarse). **Qué necesitas:** un navegador. Corre en CPU; sin claves.

Montas el escenario más honesto posible: 120 muestras, 1000 columnas de ruido y una etiqueta totalmente aleatoria. No hay nada que aprender. Y aun así consigues que un modelo presuma de un **77,5%** de acierto... haciendo trampa sin querer: seleccionas las «mejores» columnas mirando el dataset entero *antes* de partir en train y test. Cuando lo haces bien —el test escondido hasta el final, la selección dentro de la validación— el número cae al **45%**, el del puro azar. La fuga inflaba **32 puntos** sobre datos sin señal alguna.

Casi todos los «modelos increíbles» que fracasan al desplegarse tenían fuga. Es silenciosa: no da error, da buenas noticias falsas.

► **Abrir en Google Colab** ↓ **Descargar el cuaderno (.ipynb)**

Slices y sesgo: el promedio miente

Qué practicas: descubrir que un buen acierto global puede esconder un fallo grave en un subgrupo. **Dónde encaja:** capítulo 5 (slices, sesgos y decisión algorítmica). **Qué necesitas:** un navegador. Corre en CPU; sin claves.

Entrenas un modelo sobre dos grupos: uno mayoritario (85%) y otro minoritario (15%) con un patrón distinto. El acierto global sale en un tranquilizador **83%**. Pero cuando partes por grupos, el promedio confiesa: el mayoritario va al **92%** y el minoritario se desploma al **36%**, peor que lanzar una moneda. El global se parecía al del grupo grande porque son la mayoría de los casos y mandan en la media; el pequeño casi no la mueve... pero recibe un servicio mucho peor.

Un modelo que va «bien de media» puede ser inútil o injusto para un colectivo concreto. Si decides sobre personas, el promedio no te exime.

► **Abrir en Google Colab** ↓ **Descargar el cuaderno (.ipynb)**

Vocabulario aprendido

Estas palabras no son adorno. Son el vocabulario mínimo para discutir el trabajo con alguien de datos, IA, producto u operación sin que todo se convierta en “me parece que”. Si puedes usar estos términos para explicar tu entrega, probablemente has entendido el cierre del facsímil.

TÉRMINO	QUÉ SIGNIFICA AQUÍ	CÓMO LO USARÍAS EN UNA ENTREGA
Auditoría de datos	Revisión sistemática de contrato, calidad, trazabilidad, splits, slices y decisión.	La usas para ordenar qué se comprueba antes de publicar o automatizar.
Decisión de publicación	Salida que indica si un sistema puede pasar a uso real, revisión o bloqueo.	La escribes como una decisión defendible, no como una opinión rápida.
Decisión de release de datos	Dictamen técnico sobre si los datos y sus derivados permiten automatizar, revisar o bloquear.	Lo escribes como `pass`, `review` o `block`, con checks y evidencia.
Evidencia reproducible	Reportes, hashes, manifests, trazas, contratos y salidas que otra persona puede regenerar.	La adjuntas al memo para que la revisión no dependa de confianza verbal.
Gate de datos	Regla ejecutable que impide publicar si fallan calidad, split, trazabilidad, slices o SLOs.	Lo conectas a CI o a la revisión de release.
Expediente de release	Conjunto de contrato, datos, evidencias, outputs, memo y decisión que permite revisar una publicación de datos o IA.	Lo preparas para que otra persona pueda reconstruir el porqué de la decisión.
Alcance limitado	Publicación o piloto restringido a un segmento, uso o ventana porque la evidencia todavía no permite un despliegue general.	Lo propones cuando no hay bloqueo, pero la señal aún no permite rollout general.
Owner de evidencia	Persona o equipo responsable de mantener, explicar y actualizar una evidencia concreta del expediente.	Lo asignas para que un check no quede huérfano cuando cambie el dato o el sistema.
Corrección sin relajar umbrales	Arreglar datos, trazas o contratos sin bajar el listón después de ver el fallo.	Mantienes el contrato y corriges la causa, no la métrica incómoda.
Siguiente experimento	Intervención medible para aprender si una acción mejora el sistema en un slice concreto.	Diseñas unidad, exposición, métrica primaria, guardrails y criterio de parada.
Laboratorio	Espacio guiado donde conviertes el facsímil en trabajo práctico.	Lo usas para producir artefactos que puedas enseñar, ejecutar y defender.

Dónde solía tropezar yo

TROPIEZO	POR QUÉ OCURRE	ANTÍDOTO
Querer publicar por una métrica global	La media resume demasiado.	Mirar contrato, test, slices y operación.
Arreglar el modelo antes de arreglar datos	Parece la parte más interesante.	Revisar trazabilidad y calidad primero.
Diseñar experimentos sin exposure event	La asignación parece suficiente.	Registrar exposición real y versión exacta.
Ignorar ventanas de maduración	Queremos decidir rápido.	Declarar `day_1`, `day_7` o la ventana que toque.

Antes de pasar página

Antes de cerrar el facsímil, deberías poder responder:

1. ¿Qué diferencia hay entre contrato de datos y contrato de experimento?
2. ¿Por qué un dataset con schema correcto puede quedar bloqueado?
3. ¿Qué hace que un split sea honesto?
4. ¿Por qué los slices críticos pueden cambiar una decisión?
5. ¿Qué SLI de DataOps bloquearía una publicación?
6. ¿Qué diferencia hay entre asignación y exposición?
7. ¿Por qué un experimento en `review` no es necesariamente un fracaso?
8. ¿Qué entregarías como evidencia reproducible de una decisión?
9. ¿Por qué no deberías cambiar umbrales después de mirar el resultado?
10. ¿Qué diferencia hay entre corregir trazabilidad y mejorar el modelo?
11. ¿Qué comprobaría un gate de CI de datos?

En resumen

Este cierre no pretende que salgas pensando que la ciencia de datos es una lista infinita de controles. Pretende justo lo contrario: que tengas una forma ordenada de no perderte. Cuando un proyecto crece, siempre aparecen urgencias, atajos y métricas seductoras. El

contrato, la evidencia, el gate y el experimento son la manera de seguir pensando con calma cuando el sistema ya tiene presión real.

IDEA	QUÉ TE LLEVAS
Los datos son sistema.	No son entrada pasiva del modelo.
La evaluación es una promesa.	Si el split o los slices fallan, la métrica no basta.
La operación decide.	Sin trazas, SLOs y runbooks, no hay publicación responsable.
La causalidad cambia la pregunta.	Predecir no prueba que una acción funcione.
El cierre junta todo.	La salida profesional es una decisión reproducible, no una impresión.

Recursos para seguir: leer, construir y experimentar

La ciencia de datos es la base callada de todo lo demás: sin datos con linaje, sin splits honestos y sin mirar por slices, ningún modelo es de fiar. Aquí tienes por dónde seguir.

Para experimentar sin código. En las cajas «Pruébalo en 5 minutos» lo viste: Teachable Machine (`teachablemachine.withgoogle.com`) para provocar una fuga de datos con tus propias fotos; el Embedding Projector (`projector.tensorflow.org`) para ver cómo las features se vuelven una estructura visible con PCA y t-SNE; y el What-If Tool de Google (`pair-code.github.io/what-if-tool`) para descubrir que un buen promedio puede esconder un grupo al que el modelo trata peor.

Para construir. Las herramientas de trabajo diario: pandas para manipular datos, scikit-learn para modelos y validación, Fairlearn para medir y mitigar sesgos por subgrupo, y herramientas de calidad de datos como Great Expectations para validar esquemas y detectar problemas antes de entrenar.

Para leer. Cada capítulo enlaza sus fuentes en «Para saber más»; las ideas que más te ahorrarán disgustos son el leakage (el error que infla métricas sin que te enteres) y la evaluación por slices. Recorrer el ciclo entero, de los datos crudos a una decisión, es lo que separa «el número salió bonito» de «mi evaluación no me está engañando».

Para saber más

Amershi, S. y otros (2019). Software engineering for machine learning: A case study. *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice*, 291-300. https://doi.org/10.1109/ICSE-SEIP.2019.00042

Breck, E., Cai, S., Nielsen, E., Salib, M. y Sculley, D. (2017). The ML test score: A rubric for ML production readiness and technical debt reduction. *2017 IEEE International Conference on Big Data*, 1123-1132. https://doi.org/10.1109/BigData.2017.8258038

Gebru, T. y otros (2021). Datasheets for datasets. *Communications of the ACM*, 64(12), 86-92. https://doi.org/10.1145/3458723

Kohavi, R., Longbotham, R., Sommerfield, D. y Henne, R. M. (2009). Controlled experiments on the web: Survey and practical guide. *Data Mining and Knowledge Discovery*, 18(1), 140-181. https://doi.org/10.1007/s10618-008-0114-1

Mitchell, M. y otros (2019). Model cards for model reporting. *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 220-229. https://doi.org/10.1145/3287560.3287596

National Institute of Standards and Technology. (2023). *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*. https://doi.org/10.6028/NIST.AI.100-1

Sambasivan, N. y otros (2021). “Everyone wants to do the model work, not the data work”: Data cascades in high-stakes AI. *Proceedings of CHI 2021*, 1-15. https://doi.org/10.1145/3411764.3445518

Sculley, D. y otros (2015). Hidden technical debt in machine learning systems. *Advances in Neural Information Processing Systems 28*. https://papers.nips.cc/paper/5656-hidden-technical-debt-in-machine-learning-systems

---

Notas

1. Gebru, T. y otros (2021). Datasheets for datasets. *Communications of the ACM*, 64(12), 86-92. https://doi.org/10.1145/3458723 ↩

2. Mitchell, M. y otros (2019). Model cards for model reporting. *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 220-229. https://doi.org/10.1145/3287560.3287596 ↩

3. Breck, E., Cai, S., Nielsen, E., Salib, M. y Sculley, D. (2017). The ML test score: A rubric for ML production readiness and technical debt reduction. *2017 IEEE International Conference on Big Data*, 1123-1132. https://doi.org/10.1109/BigData.2017.8258038 ↩

4. Amershi, S. y otros (2019). Software engineering for machine learning: A case study. *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice*, 291-300. https://doi.org/10.1109/ICSE-SEIP.2019.00042 ↩

5. National Institute of Standards and Technology. (2023). *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*. https://doi.org/10.6028/NIST.AI.100-1 ↩