

Manifiesto

ML Engineer para Oncología

Luis Palomero, PhD

v1.0 · junio 2026

Anti-pilares (lo que no soy)

Si buscas a alguien que haga alguna de estas cosas, probablemente no soy la persona adecuada.

- No soy el que pone un transformer donde basta una regresión logística.
- No soy el que optimiza el 0.1 % de AUC a costa de explicabilidad.
- No soy el que entrega un notebook con celdas fuera de orden.
- No soy el que dice «funciona en producción» sin haberlo visto.
- No soy el que usa IA para generar código que no entiende.

Declaración de intenciones

Construyo pipelines y modelos oncológicos que sean:

Reproducibles: porque un resultado que no se puede replicar no es un resultado.

Comprensibles: porque si un clínico o un revisor no entiende lo que hago, he fracasado en mi trabajo.

Robustos: porque en oncología no hay «funciona casi siempre».

Honestos: porque un método *fancy* que nadie entiende no es innovación, es ruido.

Mi objetivo no es impresionar con complejidad técnica. Es generar confianza a través de claridad y solidez metodológica.

Principio rector: simplicidad radical

La simplicidad no es pereza. Es la decisión activa de eliminar todo lo que no aporta.

Un pipeline simple es:

- Explicable en 5 minutos a un clínico.
- Reparable por otro ingeniero sin llamarme.
- Modificable sin cascadas de efectos secundarios.

Regla: si una abstracción oculta lógica relevante para el resultado, es magia. Y la magia no tiene cabida aquí.

Pilares

1. Reproducibilidad y robustez en producción

En investigación oncológica, el peor escenario no es que un modelo falle. Es que funcione una vez y no se sepa por qué.

- Pincho versiones de todo: código, datos, dependencias, entorno.
- Los pipelines son deterministas. Semilla fija. Orden explícito.
- Cada pipeline se ejecuta en un contenedor. No hay «en mi máquina funciona».
- Los experimentos dejan traza: qué se ejecutó, con qué parámetros, sobre qué datos.

2. Altos estándares de programación

El código no es solo para la máquina. Es para la persona que lo leerá dentro de seis meses (que probablemente sea yo).

- Type hints. Tests. Nombres que explican intención, no implementación.
- Una función, una responsabilidad.
- El testing no es una métrica. Es una red de seguridad que permite moverse rápido sin miedo a romper.
- Prefiero código aburrido y correcto a código brillante y frágil.

3. Simplicidad en métodos (sin magia)

El método más sofisticado no es el mejor. El mejor es el que el investigador puede explicar en la sección de métodos del paper.

- Una regresión logística bien construida vale más que una red neuronal que nadie entiende.
- Cada transformación de datos tiene un porqué. Si no puedes justificarla, no la hagas.
- Si el método necesita un tutorial de 20 minutos para entenderlo, busca una alternativa más simple.
- La complejidad se justifica, no se asume.

4. Narrativa coherente

Un análisis oncológico cuenta una historia. Desde la pregunta clínica hasta la interpretación.

- Cada pipeline se lee como un artículo: introducción (qué preguntamos), métodos (cómo lo hicimos), resultados (qué encontramos).
- El código es legible como narrativa. Los comentarios explican el porqué, no el qué.
- Los outputs tienen contexto. Un número sin su significado no es un resultado.
- La historia la cuenta el investigador. Mi trabajo es darle las herramientas para contarla bien.

5. IA como herramienta de soporte

La inteligencia artificial es un asistente, no un sustituto. Acelera, no decide.

- La IA me ayuda a escribir tests, refactorizar código, explorar alternativas.
- La IA no toma decisiones metodológicas. No interpreta resultados. No firma papers.
- Todo output de IA se revisa. Todo. Sin excepción.
- **Transparencia:** cuando entrego un trabajo, especifico qué partes han sido asistidas por IA y cómo se han verificado.
- El juicio clínico y la experiencia del investigador son irremplazables.
- Uso la IA para eliminar trabajo tedioso, no para eliminar pensamiento crítico.

Filosofía de testing

Si no escribes código, el testing te importa menos de lo que debería. En términos simples: Cuando modifico un pipeline, ejecuto verificaciones automáticas que comprueban que los resultados no han cambiado. Si algo se rompe, los tests me lo dicen antes de entregarte nada.

- Los resultados que recibes son consistentes entre versiones del pipeline.
- Puedo modificar el código sin riesgo de errores silenciosos.
- Cada pipeline se prueba con datos sintéticos donde el resultado correcto es conocido.

En una frase: si ejecutas el mismo análisis dos veces con los mismos datos y código, obtienes exactamente el mismo resultado. Eso es testing.

Validación experimental

El modelado computacional genera hipótesis. La validación experimental las confirma. Un resultado bioinformático — un gen candidato, una firma pronóstica, una vía enriquecida — es un punto de partida, no una conclusión. Para que sea publicable necesita, idealmente:

- Validación en una cohorte independiente.
- Contraste con datos de proteína (IHC, western blot) cuando sea posible.
- Revisión por un clínico o biólogo que evalúe la plausibilidad biológica.

Mi responsabilidad es dejar claros los límites de lo que los datos permiten afirmar. Si es correlacional, se dice. Si necesita confirmación experimental, se dice.

Cómo trabajamos juntos

Lo que siempre recibirás

- **Pipeline reproducible:** scripts organizados, dependencias declaradas, instrucciones de ejecución.
- **Reporte en PDF:** métodos, resultados, figuras publicables y código fuente.
- **Datos versionados:** cada resultado vinculado a una versión específica de código y datos.
- **Resultados consistentes:** testing automático garantiza que no hay errores silenciosos.
- **Dos rondas de revisión incluidas:** cambios de alcance se presupuestan aparte.
- **Código fuente completo:** pipelines, scripts y configuración. Si quieres revisarlo, ejecutarlo o adaptarlo, te lo entrego sin reservas.
- **Comunicación clara:** si algo necesita ajuste, lo sabrás antes de que sea un problema.

Qué espero de ti

- **Contexto biológico:** qué pregunta responder, qué sabes ya, qué te preocupa.
- **Datos en bruto:** sin procesar, con metadatos claros.
- **Revisión crítica:** si algo no te cuadra, dímelo.
- **Honestidad sobre el uso esperado:** publicación Q1, grant o exploración — el esfuerzo es distinto.

Límite de competencia

Yo sé de datos, pipelines y modelos. Tú sabes de biología tumoral, microambiente inmune e interpretación clínica. El resultado es mejor cuando no confundimos los roles.

No interpreto resultados biológicos. No firmo papers como coautor a menos que contribuya intelectualmente.

Pero diseño cada análisis para que sus resultados puedan ser interpretados desde una perspectiva biológica. Un hallazgo sin contexto biológico no es un hallazgo. Por eso los outputs incluyen información de contexto, anotaciones funcionales y referencias que permitan al investigador evaluar la relevancia sin tener que bucear en código.

Mi trabajo termina donde empieza el juicio del investigador. Pero intento dejarle el mejor punto de partida posible.