La seguridad total, imposible en los modelos "IA": Es cuestión de matemáticas

Parte 1: Los subespacios adversariales y la imposibilidad de la seguridad total

¿Qué es un subespacio adversarial?

Para un clasificador de imágenes (por ejemplo, 784 dimensiones en MNIST), el **espacio adversarial local** es el conjunto de pequeñas perturbaciones que inducen error de clasificación.

Su propiedad fundamental (Tramer et al., 2017, arXiv:1704.03453):
Ese conjunto no es un puñado de puntos aislados, sino que contiene muchas direcciones linealmente independientes. En redes *fully connected* entrenadas con MNIST, se encontraron **unas 44 direcciones ortogonales** en el modelo fuente, de las cuales **alrededor de 25** transferían a otro modelo. Es decir, hay decenas de direcciones independientes a lo largo de las cuales moverse un poco provoca error.

> **Aclaración:** En un espacio de 784 dimensiones, un subespacio de dimensión 25 tiene **medida de Lebesgue cero**. (*del mismo modo que un plano (dimensión 2) no tiene volumen en un espacio 3D, un subespacio de dimensión 25 es invisible para la medida de Lebesgue en un espacio de 784 dimensiones*). Una perturbación aleatoria uniforme casi nunca cae en él. La vulnerabilidad es **estructural**, no volumétrica: un atacante que descubre el subespacio (por ejemplo, mediante el gradiente) puede "moverse" dentro de él intencionalmente con alta probabilidad de éxito.

¿Por qué es matemáticamente imposible la seguridad total?

1. La alta dimensionalidad del subespacio adversarial

Existen decenas de direcciones independientes que llevan al error. Los experimentos de Tramer et al. (2017) sugieren que forzar el gradiente a ser muy puntiagudo (reduciendo drásticamente la dimensionalidad) tiende a dañar la generalización del modelo, por lo que esta propiedad es estructural en clasificadores útiles, no un defecto fácilmente eliminable.

2. La transferibilidad es inevitable por la cercanía de las fronteras

Dos modelos distintos con alta precisión en la misma tarea tienen sus fronteras de decisión muy próximas. La distancia entre ambas fronteras es mucho menor que la distancia desde un punto normal a la primera frontera. Por tanto, una perturbación que cruce la frontera del modelo A cruzará también la del modelo B. Un atacante puede entrenar un modelo sustituto (accesible) y sus ejemplos adversariales transferirán al objetivo.

3. La complejidad computacional de la verificación exhaustiva

Explorar todos los posibles modelos sustitutos (infinitos) y todos los puntos en sus subespacios adversariales es inviable. Además, se ha demostrado que problemas relacionados (como defender

una SVM contra envenenamiento) son **NP-completos** (Ding et al., 2020, arXiv:2006.07757). Aunque esto no prueba directamente la NP-dureza de la verificación de ataques de evasión, sí sugiere que la búsqueda exhaustiva es intratable en la práctica.

Por qué el *red teaming* comercial tradicional no es suficiente

El *red teaming* habitual (pruebas manuales, ataques conocidos de bibliotecas como ART) se basa en una premisa falsa: la de que unas pocas pruebas pueden cubrir el espacio de ataques.

Limitaciones señaladas por Cox & Bunzel (2025, arXiv:2511.05102):

El espacio de posibles modelos sustitutos es infinito.

La dimensionalidad del subespacio adversarial (~25) hace que las pruebas puntuales no sean representativas.

La transferibilidad es sistemática, no un accidente. Un adversario real con un solo sustituto bien elegido puede tener éxito.

El riesgo del "teatro de seguridad":
Un informe de *red teaming* que no encuentra vulnerabilidades puede generar una falsa confianza, llevando a desplegar modelos que siguen siendo vulnerables a ataques transferibles. Aunque el paper académico de Cox & Bunzel **no califica esta práctica como "estafa"**, esa valoración pertenece a opiniones personales de los autores en blogs (y a mi también me lo parece).

¿Qué proponen Cox & Bunzel en lugar del *red teaming* tradicional?

No se puede garantizar seguridad total. Lo honesto es **cuantificar el riesgo residual**:

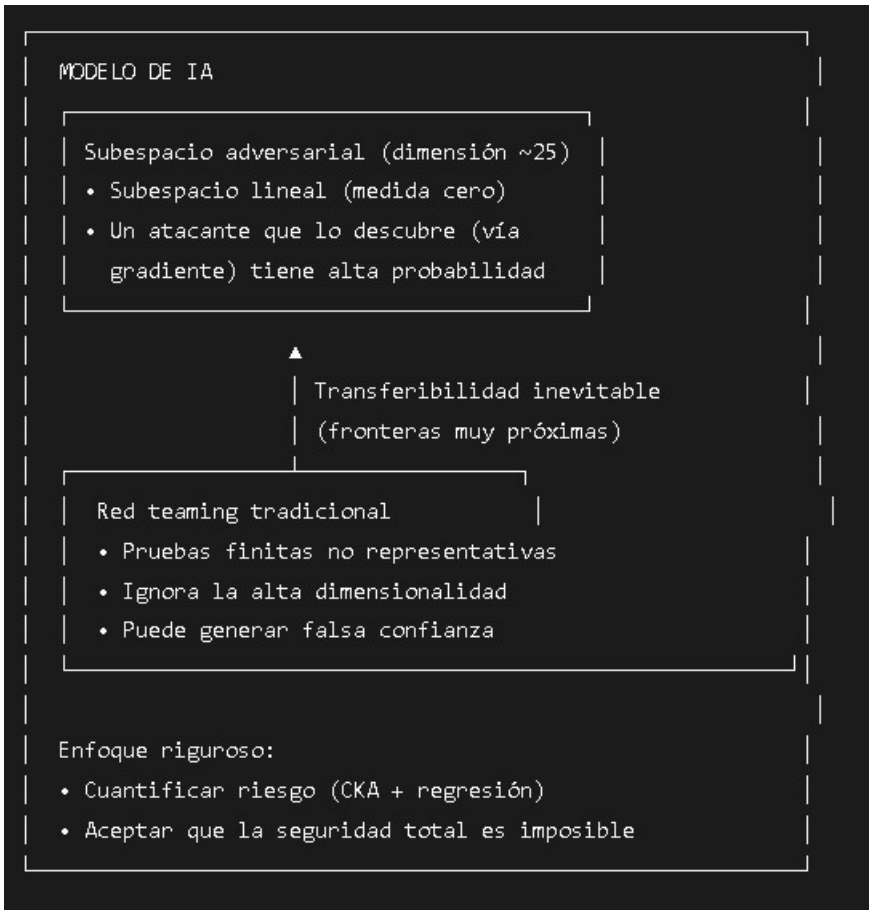
Selección estratégica de sustitutos mediante CKA (Centered Kernel Alignment):
CKA mide similitud entre representaciones internas (0 a 1). Se eligen sustitutos de alta similitud (CKA > 0.7) y baja similitud (CKA < 0.35) para cubrir un espectro amplio.

Estimación por regresión:
En lugar de buscar ataques concretos, se modela la probabilidad de existencia de un ataque transferible exitoso. El resultado es una probabilidad (ej., "7%"), no un binario seguro/inseguro.

Pruebas en fase de diseño:
Usar perturbaciones modelo-agnósticas (como la diferencia de medias entre clases) para estimar vulnerabilidad antes del entrenamiento.

Ninguna de estas técnicas da una certificación binaria de seguridad.

Esquema



> La seguridad total es imposible por razones geométricas (alta dimensionalidad del subespacio adversarial) y topológicas (cercanía de fronteras). El *red teaming* tradicional no puede garantizarla y sus limitaciones son severas. La propuesta de Cox & Bunzel es cuantificar el riesgo mediante métodos estadísticos (CKA + regresión).

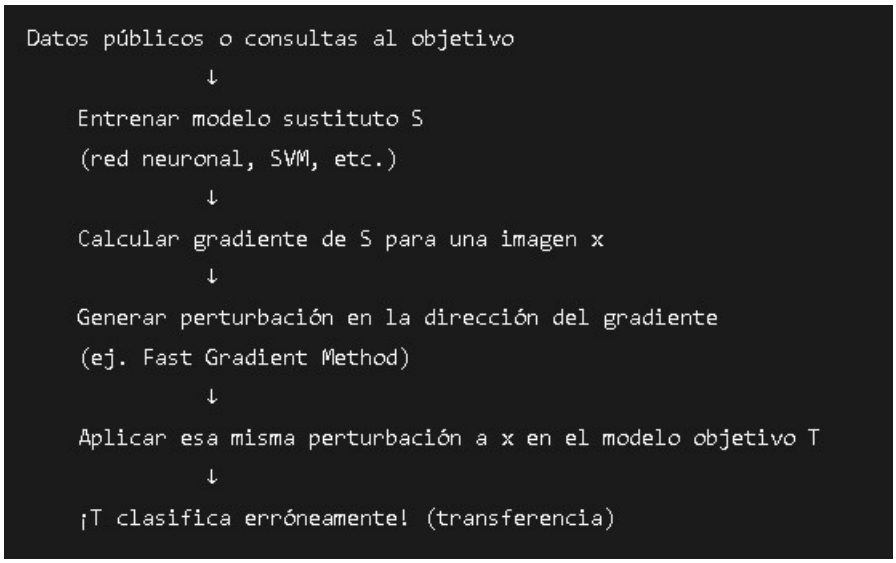
Parte 2: Cómo se explotan los subespacios adversariales en la práctica (el peligro real)

Hasta ahora hemos visto que los modelos tienen regiones de error de alta dimensionalidad y que las fronteras de decisión de diferentes modelos están muy próximas. Esto no es "solo" una rareza matemática; tiene consecuencias concretas y explotables.

1. Un atacante no necesita conocer el modelo objetivo

El atacante se enfrenta a un modelo al que solo puede hacer consultas (caja negra) o ni siquiera eso. Puede construir su propio **modelo sustituto** entrenado con datos públicos o con consultas al objetivo. Ese sustituto no tiene que ser idéntico; una amplia variedad de arquitecturas (redes neuronales, SVMs, etc.) pueden funcionar, aunque la eficacia de la transferencia depende de la similitud entre el sustituto y el objetivo. En general, modelos más parecidos al objetivo transfieren mejor.

Procedimiento típico de ataque (transfer-based):



2. ¿Por qué funciona? El gradiente como mapa del subespacio adversarial

El gradiente de la pérdida respecto a la entrada indica la dirección de máximo error local. En un modelo con subespacio adversarial de dimensión kk, el gradiente no es una línea aislada: está rodeado de muchas direcciones cercanas que también producen error. Por tanto, aunque el atacante no conozca la orientación exacta del subespacio en el modelo objetivo, el gradiente de su sustituto apunta a una región que **intersecta** el subespacio del objetivo.

Tramer et al. (2017) exploraron sistemáticamente el subespacio adversarial de un modelo fuente utilizando el método GAAS. Cuando muestrearon perturbaciones **aleatorias dentro de ese subespacio** (combinando linealmente las ~44 direcciones ortogonales encontradas), consiguieron engañar al modelo objetivo en el **89%** de los casos para redes *fully connected* en MNIST. Esto demuestra que el solapamiento entre subespacios puede ser masivo, al menos para arquitecturas similares.

3. Ejemplo concreto: ataque a un clasificador de malware

Imaginemos un sistema antivirus basado en IA que analiza archivos (500 características binarias). Un atacante quiere que su malware sea clasificado como benigno.

1. El atacante entrena una red neuronal sustituta con la misma tarea (malware vs. benigno), usando muestras públicas o consultas al antivirus.
2. Toma un malware conocido y calcula el gradiente de la pérdida en el sustituto.
3. Modifica los bytes (características) más influyentes según el gradiente (solo unos pocos bits, para no romper la funcionalidad del malware).
4. Envía el archivo modificado al antivirus real. El antivirus, aunque sea diferente, tiene su subespacio adversarial cerca del del sustituto, por lo que probablemente también lo clasifique mal.

En el paper de Tramer et al., con el dataset DREBIN (detección de malware Android) se demostró que, utilizando perturbaciones modelo-agnósticas (como la diferencia de medias entre clases),

bastan **menos de 10 cambios** en características binarias para engañar tanto a un modelo lineal como a una red neuronal. Esto ilustra que la vulnerabilidad no depende de un método de ataque concreto.

4. Las defensas conocidas no eliminan el subespacio

Se han propuesto varias defensas, como el *adversarial training* (entrenar con ejemplos adversariales). Pero Tramer et al. midieron que, aunque esta defensa desplaza ligeramente la frontera de decisión, la **distancia inter-frontera** sigue siendo muy pequeña. En sus experimentos, la distancia entre la frontera del modelo sin defender y la del modelo con *adversarial training* era solo de **0.63** unidades, mientras que la perturbación típica usada para atacar tiene magnitud **5**. Por tanto, el ataque transferido sigue siendo exitoso.

> **Conclusión práctica:** Un atacante con recursos moderados (un ordenador personal, datos públicos) puede quebrantar sistemas de IA desplegados, incluso si estos han sido defendidos con técnicas estándar. Entrenar un sustituto y calcular gradientes está al alcance de muchos adversarios potenciales.

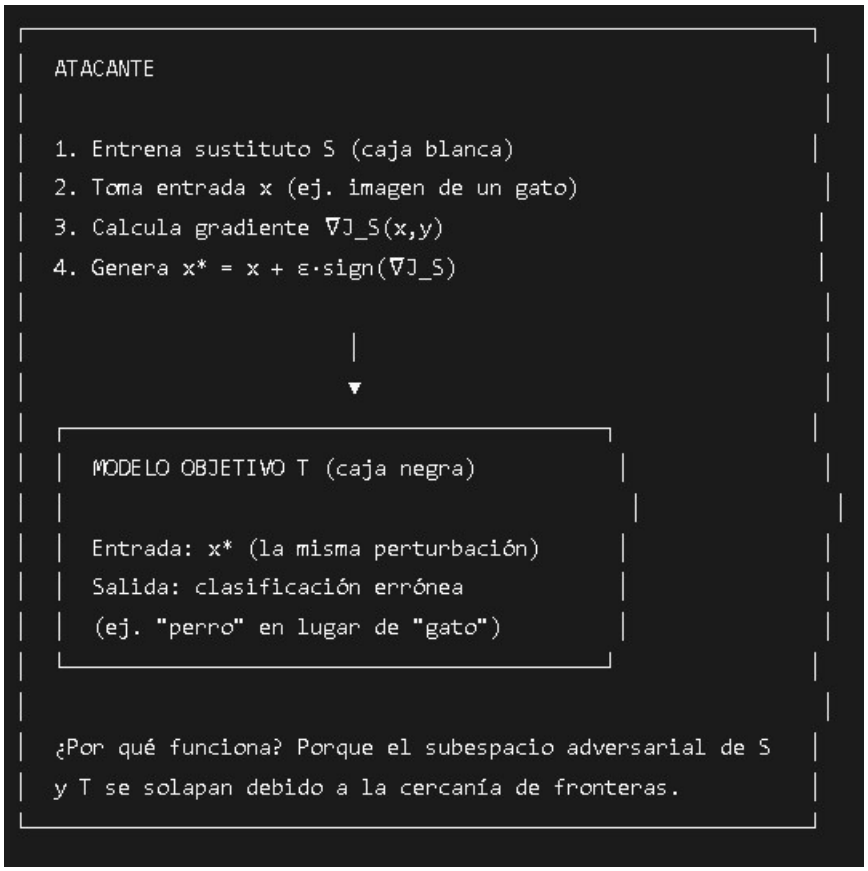
5. ¿Qué implica esto para organizaciones?

No confiar en certificaciones de "*red teaming*" que solo prueban unos pocos ataques conocidos y concluyen que no hay vulnerabilidades.

Asumir que cualquier modelo comercial de IA que realice clasificación es susceptible a ataques transferibles, a menos que se demuestre lo contrario con métodos de cuantificación rigurosa (como los propuestos por Cox & Bunzel).

Priorizar la **detección de ataques** en lugar de la prevención total, y diseñar sistemas que puedan reaccionar (ej., reiniciar con entradas aleatorias, limitar consultas por usuario, etc.).

Esquema del ataque



> Los subespacios adversariales no son una rareza académica; son la puerta de entrada para ataques prácticos y de bajo costo. Un adversario con un modelo sustituto puede generar ejemplos que transfieren al objetivo con alta tasa de éxito (especialmente cuando las arquitecturas son similares), incluso si el objetivo ha sido defendido. Esto convierte la “seguridad total” en una quimera y obliga a repensar las estrategias de protección: hay que asumir que el modelo será atacado y centrarse en la detección, la mitigación y la cuantificación del riesgo, no en la prevención absoluta.

Parte 3: Anexo – Vulnerabilidad de los sistemas de IA militares

La creencia de que el alto presupuesto, el secretismo o las supuestas "herramientas superiores" del ámbito militar pueden blindar a un sistema de inteligencia artificial contra ataques adversariales carece de fundamento. La ciencia subyacente a estas vulnerabilidades no distingue entre un modelo civil y uno militar. A continuación se exponen las razones clave, respaldadas por fuentes académicas e institucionales, que demuestran la imposibilidad de la seguridad total y la insuficiencia de las aproximaciones tradicionales.

1. La vulnerabilidad es estructural y universal

El origen del problema es puramente matemático. Un clasificador entrenado para operar en un espacio de alta dimensionalidad (por ejemplo, una imagen de 224×224 píxeles genera un espacio de

50.176 dimensiones) es inherentemente vulnerable, independientemente de quién lo entrene o con qué propósito.

La investigación fundamental de Tramer et al. (2017, arXiv:1704.03453) demostró que los ejemplos adversariales no son puntos aislados, sino que forman subespacios de alta dimensionalidad (~25 direcciones independientes). Un atacante puede descubrir estos subespacios mediante el gradiente de un modelo sustituto. La **transferibilidad** de estos ataques es también una consecuencia geométrica, no un accidente.

Estas vulnerabilidades estructurales han sido confirmadas en contextos militares por diversas investigaciones. Por ejemplo:

Ataques adversariales efectivos contra detectores de objetos en aeronaves no tripuladas (UAV).

Parches adversariales físicos que ocultan vehículos de detectores infrarrojos aéreos, alcanzando tasas de éxito >81% en pruebas de campo. *Physical Adversarial Attacks for Infrared Object Detection*

Equipos militares de Corea del Sur han publicado investigaciones sobre "ataques de parches adversariales camuflados" utilizando métricas de camuflaje real para ocultar tanques de guerra (Kim et al., *Camouflaged Adversarial Patch Attack on Military Object Detection*, 2023.

2. El propio Pentágono admite sus vulnerabilidades

Lejos de ser infalibles, los máximos responsables del desarrollo de IA militar en EE.UU. admiten públicamente problemas de seguridad graves. El historial del Pentágono en ciberseguridad, documentado por sus propias agencias de control, es revelador.

Informe del GAO (Government Accountability Office) de 2018 (GAO-18-211):

> *"Una generación entera de armas estadounidenses está completamente abierta a los hackers"*

El informe encontró vulnerabilidades críticas en casi todos los sistemas de armas en desarrollo, incluyendo fallos tan básicos como no cambiar contraseñas por defecto. El problema no era solo tecnológico, sino cultural y sistémico.

Investigaciones posteriores continúan detectando vulnerabilidades significativas en la inteligencia artificial de defensa. Por su parte, la **Agencia de Seguridad Nacional (NSA)** ha emitido guías de ciberseguridad para sistemas de IA reconociendo los riesgos de los ataques adversariales, el envenenamiento de datos (*data poisoning*) y la inversión de modelos (*model inversion*) como amenazas clave (NSA CSI, "Deploying AI Systems Securely", 2024).

3. La verificación exhaustiva es computacionalmente inviable

Los enfoques de prueba tradicionales, como el *red teaming* o las listas de verificación, **no pueden demostrar la ausencia de vulnerabilidades**. Ningún conjunto finito de pruebas puede cubrir todas las posibles entradas en un espacio de alta dimensionalidad. Incluso un atacante con recursos moderados (un ordenador personal y datos públicos) puede entrenar un modelo sustituto y generar ataques transferibles con alta probabilidad de éxito. Por lo tanto, un informe de *red teaming* que no

encuentre vulnerabilidades genera una falsa confianza; es lo que se conoce como "teatro de seguridad" (*security theatre*).

La dificultad de la búsqueda exhaustiva se refleja en resultados de complejidad computacional. Ding et al. (2020, arXiv:2006.07757) demostraron que incluso defender una máquina de vectores de soporte (SVM) contra ataques de envenenamiento es **NP-completo**, indicando que la optimización defensiva es intratable en el caso general.

4. La "seguridad por oscuridad" (*security by obscurity*) es un paradigma fallido

La estrategia predominante en el ámbito militar, el secretismo o "seguridad por oscuridad", no solo es insuficiente, sino que es incompatible con los enfoques rigurosos de seguridad. La historia de la ciberseguridad ha demostrado repetidamente que confiar en la ocultación de los detalles de un sistema es una estrategia fallida. Como se resume en el **principio de Kerckhoffs (1883)**, un sistema debe ser seguro incluso si todo su funcionamiento es conocido por el adversario, excepto la clave. El secretismo ha fracasado en numerosos casos (por ejemplo, el sistema de tarjetas inteligentes del gobierno holandés, que fue vulnerado tras ser analizado a fondo).

Frente a esto, Cox & Bunzel (2025, arXiv:2511.05102) proponen un **cambio de paradigma**:

De intentar una seguridad total imposible a **cuantificar el riesgo residual**.

Mediante selección estratégica de modelos sustitutos con CKA (similitud de representaciones) y estimación por regresión.

El resultado es una probabilidad (ej., “7% de que exista un ataque transferible exitoso”), no una falsa certificación binaria.

Este enfoque riguroso requiere transparencia total sobre el modelo (acceso a gradientes, arquitectura, representaciones), lo que es directamente incompatible con la cultura del secretismo militar.

> La afirmación de que los sistemas de IA militares son inmunes a los ataques adversariales es una creencia falsa que contradice los fundamentos matemáticos de la disciplina. Las vulnerabilidades discutidas son estructurales y universales. El Pentágono, a través de sus propios informes oficiales (GAO 2018), ha reconocido la gravedad y el alcance sistémico de sus problemas de ciberseguridad. Los enfoques tradicionales de prueba no pueden ofrecer garantías, y la cultura del secretismo es incompatible con los únicos métodos rigurosos de cuantificación del riesgo. Por todo ello, la **seguridad total es una quimera inalcanzable para los sistemas de IA, independientemente de su propósito o presupuesto**.