MYALGOPORTFOLIO · PROCESO

Claude Code Trading: análisis metodológico del proceso de generación de estrategias con LLMs

Primera versión · enero–abril de 2026

Laura GP · 19 de noviembre de 2026

Resumen

El proyecto demostró que un equipo de agentes LLM puede operar de forma autónoma una fábrica completa de estrategias, pero no demostró que las estrategias resultantes tengan ventaja. De 412 estrategias que superaron la validación fuera de muestra, la mediana del Profit Factor cae de 1,37 a 0,99 en el año siguiente, y el 51 % pierde dinero.

Este documento analiza la primera versión de Claude Code Trading (enero–abril de 2026) a partir de su código, sus skills, sus 135 transcripts de sesión y su base de 1.070 estrategias. Describe el proceso metodológico, lo que se consiguió y sus limitaciones.

La conclusión es que la aportación del proyecto es de ingeniería (arquitectura, trazabilidad, automatización) y que su debilidad es de inferencia estadística: miles de pruebas sin corrección, un periodo fuera de muestra reutilizado y una memoria de lecciones que filtra resultados de validación al diseño. Las dos cosas son útiles para la tesis: la primera como base del paper de generación hypothesis-driven, la segunda como evidencia piloto del paper IS/OOS.

1. Introducción y contexto

El objetivo de la tesis es un método de portfolio de estrategias de trading con fiabilidad de ser rentable. La primera versión del proyecto atacó la parte anterior al portfolio: producir muchas estrategias candidatas con un LLM y filtrarlas.

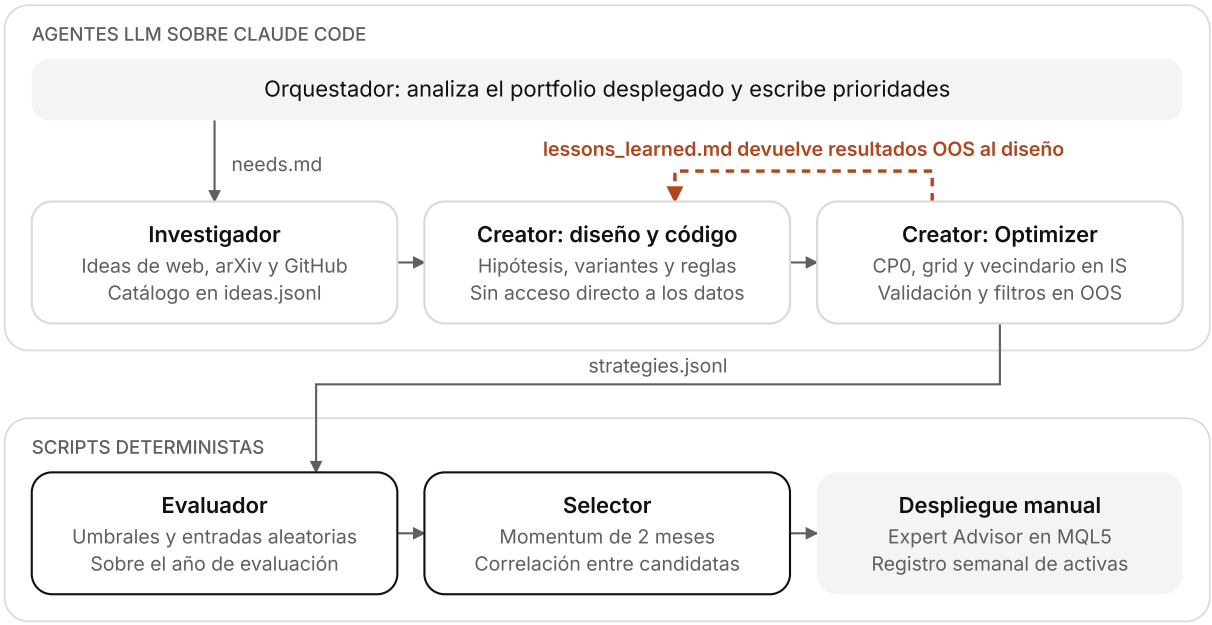
La pregunta implícita era doble. Primero, si un LLM puede hacer el trabajo de un trader cuantitativo: leer una idea, formular hipótesis, programarla, optimizarla y decidir si se guarda. Segundo, si una cadena de filtros sucesivos (IS, OOS, evaluación, momentum) basta para quedarse solo con estrategias que funcionen después.

El sistema se construyó sobre Claude Code entre el 12 de febrero y el 22 de abril de 2026, con estrategias fechadas desde enero. Operó sobre 18 activos en CFD de Darwinex (índices de EE. UU., Europa y Japón, pares de divisas, oro, petróleo y gas) con barras de 5 minutos.

2. Descripción del proceso metodológico

El proceso es una cadena de cinco etapas en la que cada una usa un tramo de datos distinto y solo deja pasar a la siguiente lo que supera sus umbrales. Las tres primeras las ejecutan agentes LLM; las dos últimas, scripts deterministas.

La memoria de lecciones devuelve resultados de validación al diseño



El bucle marcado es el punto débil del diseño: el Creator no lee los datos, pero sí los resultados de validación de sesiones anteriores.

2.1 Arquitectura de agentes

El sistema usa tres roles sobre Claude Code (Agent Teams), definidos en nueve skills y un `CLAUDE.md` común.

- **Orquestador.** Analiza el portfolio desplegado, busca noticias y calendario económico, identifica huecos (por activo, dirección y tipo) y los escribe en `needs.md`. Lanza al resto y, cada diez estrategias nuevas, ejecuta evaluador y selector.
- **Investigador.** Busca ideas en web, arXiv, GitHub y YouTube y las guarda en `ideas.jsonl`. De las 928 ideas, 644 vienen de la web, 208 las generó el propio LLM, 43 de arXiv, 28 de GitHub y 5 de YouTube.
- **Creators (hasta cuatro en paralelo).** Cada uno reclama una idea y recorre cuatro skills en serie: Conceptualizer, Programmer, Optimizer y Saver.

Los agentes no se hablan entre sí salvo por avisos de progreso. Se coordinan por ficheros: `needs.md` (prioridades, solo lo escribe el Orquestador), `ideas.jsonl` y `strategies.jsonl` (con bloqueo atómico para evitar que dos Creators cojan la misma idea) y `lessons_learned.md`

(memoria compartida de lo que no funciona). Cada sesión deja un transcript en lenguaje natural con el razonamiento de cada decisión.

2.2 Datos y partición temporal

El histórico se corta en cuatro tramos consecutivos sin solape, contados hacia atrás desde el último dato disponible.

TRAMO	DURACIÓN	USO	QUIÉN LO USA
In-sample (IS)	2 años	Prueba inicial y grid search	Creator (Optimizer)
Out-of-sample (OOS)	1 año	Validación de los parámetros fijados y elección de filtro de régimen	Creator (Optimizer)
Evaluación	1 año	Umbrales de métricas y test de entradas aleatorias	Script evaluador
Corto plazo (CP)	2 meses	Filtro de momentum reciente	Script selector

Una regla explícita prohíbe al LLM leer o inspeccionar los ficheros de datos. Solo puede llamar a las funciones de backtest, con el argumento de que las estrategias deben salir de hipótesis de mercado y no de patrones vistos en el histórico.

2.3 Generación de la estrategia

El Conceptualizer convierte una idea en una o varias hipótesis y cada hipótesis en tres a cinco variantes. Cada variante lleva reglas de entrada y salida en lenguaje natural, rangos para todos los parámetros numéricos y una justificación de los activos y timeframes elegidos. Antes de diseñar lee `lessons_learned.md`.

El Programmer traduce las reglas a tres funciones Python parametrizadas (indicadores, entrada y salida) y las prueba con valores por defecto. Cada combinación de variante, activo, timeframe y dirección cuenta como una estrategia independiente.

2.4 Optimización y validación (Optimizer)

El Optimizer aplica tres puntos de control y un paso final de filtros.

PASO	DATOS	QUÉ HACE	CRITERIO
CP0	IS	Backtest con el punto medio de cada rango	PF < 0,70 descarta; 0,70–0,85 permite hasta 2 cambios de lógica; 0,85–1,15 va a grid; > 1,15 con 30 operaciones salta el grid
Grid + CP1	IS	Búsqueda exhaustiva (máximo 100 combinaciones) y selección por vecindario	Descarta si menos del 10 % de combinaciones es rentable o si ningún punto tiene todos sus vecinos con PF > 1,0
OOS + CP2	OOS	Backtest con los parámetros fijados	PF ≥ 1,10; al menos 10 operaciones; drawdown ≤ 1,5 veces el de IS; PF ≥ 0,6 veces el de IS
Filtros de régimen	IS y OOS	Prueba SMA 200, ADX > 25, ATR relativo y anchura de Bollinger, solos y combinados	Se añade el filtro que más mejora el ratio beneficio/drawdown en OOS

Los cambios de lógica salen de un menú cerrado: quitar una condición, cambiar el tipo de señal o de salida, invertir la dirección o relajar un umbral. Si tres variantes seguidas de una hipótesis fallan, se abandona la hipótesis. Lo aprobado se guarda con su código y sus métricas; lo descartado genera una entrada en `lessons_learned.md`.

2.5 Evaluación (script)

El evaluador ejecuta cada estrategia guardada sobre el año de evaluación, en dos capas.

- **Capa 1, métricas mínimas:** PF > 1,2, más de 30 operaciones, PnL medio positivo y drawdown inferior a 3.500 USD.
- **Capa 2, test de entradas aleatorias (Monte Carlo):** 1.000 estrategias aleatorias con el mismo número de operaciones, la misma dirección y una duración media similar. La estrategia pasa si su PF supera al de las aleatorias con p < 0,15.

El diseño es unidireccional: el resultado del evaluador no vuelve al Creator para que retoque la estrategia.

2.6 Selección por momentum

El selector es el último filtro automático y el que decide qué estrategias se recomiendan para operar. Ejecuta las aprobadas por el evaluador sobre los dos últimos meses (el tramo CP) y

marca como «en momentum» las que tienen PF > 1,5, más de 10 operaciones y más del 40 % de aciertos.

Las que pasan reciben `portfolio: true`; las demás quedan guardadas pero fuera de la recomendación. El criterio es más exigente que el del evaluador (PF 1,5 frente a 1,2) y se mide sobre muchas menos operaciones (más de 10 frente a más de 30).

La idea de fondo es que una estrategia que ha funcionado en las últimas semanas tiene más probabilidad de seguir funcionando en las siguientes. Es decir, que el rendimiento de una estrategia persiste a corto plazo. El proceso no contrasta esa idea en ningún momento (ver limitación 4.2.3).

2.7 Selección por correlación

La selección por correlación es un informe, no un filtro: mide cuánto se parecen las candidatas, pero no descarta ninguna. La decisión de cuáles combinar queda en manos de quien despliega.

El script vuelve a ejecutar cada estrategia marcada para portfolio sobre todo lo posterior al IS (OOS, evaluación y CP) y calcula tres cosas:

- Correlación del PnL diario y semanal entre cada par, señalando los pares con correlación de 0,5 o más.
- Solape en mercado: qué porcentaje del tiempo coinciden en posición, señalando los que superan el 30 %.
- Drawdown del conjunto frente a la suma de drawdowns individuales, como medida del beneficio de diversificar.

Con las 8 candidatas del registro, ningún par llegó a 0,5 de correlación semanal y solo un par superó el 30 % de solape.

2.8 Despliegue

El despliegue es manual. Una skill genera el Expert Advisor en MQL5 a partir del código Python, la estrategia se marca como desplegada y un registro semanal anota qué estrategias estuvieron activas cada semana.

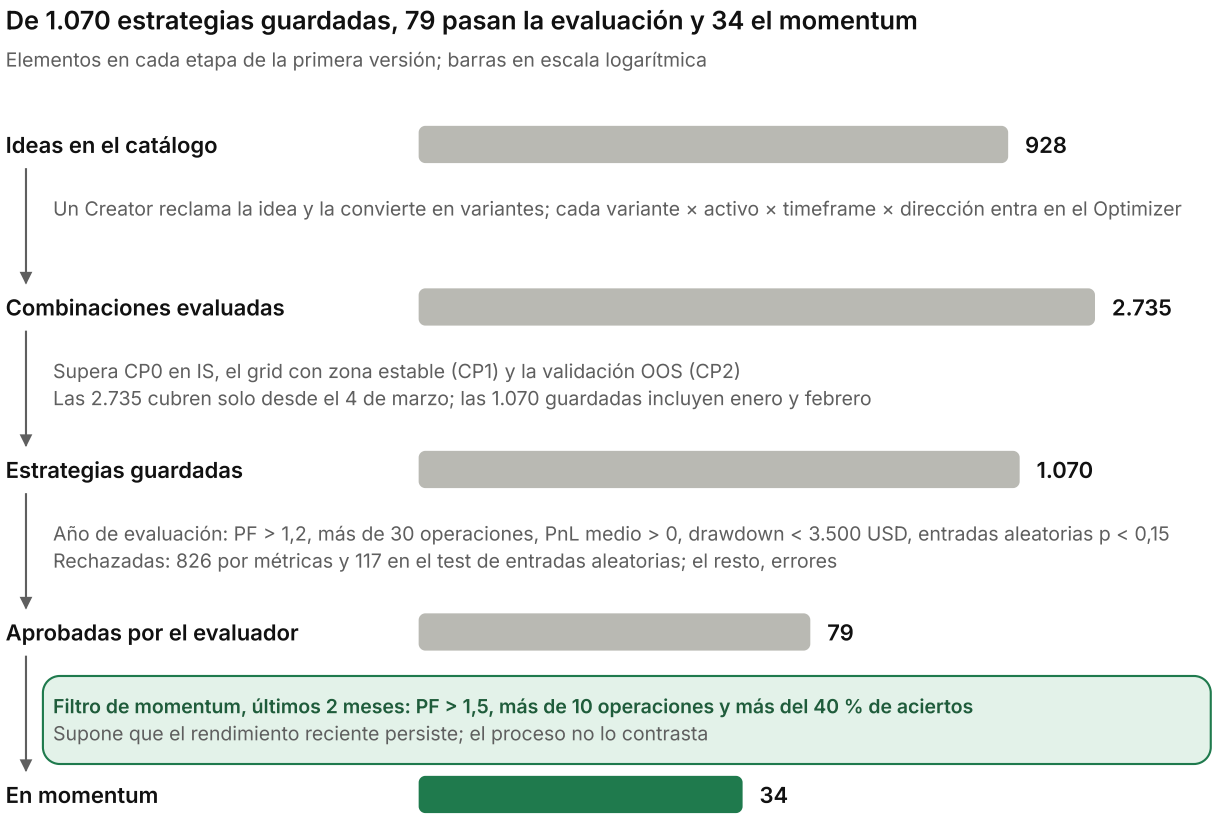
2.9 Motor de backtest

El motor es propio y los agentes no pueden modificarlo. Las señales se evalúan al cierre de la barra y se ejecutan en la apertura de la siguiente. Admite una sola posición a la vez por estrategia, con lote fijo y un coste de spread por operación tomado del catálogo de activos. Calcula 21 métricas, entre ellas Profit Factor, Sharpe y Sortino por operación, drawdown en dólares y alfa frente a comprar y mantener.

3. Qué se consiguió

Se consiguió una arquitectura que funciona de principio a fin: de una idea en texto a un Expert Advisor ejecutable, sin intervención humana hasta el despliegue. Los logros son de ingeniería y de generación de datos, no de rentabilidad demostrada.

3.1 Volumen producido



Las cifras salen de `strategies.jsonl`, `ideas.jsonl` y los 135 transcripts tal como están hoy en la carpeta. Las etapas no forman un embudo limpio porque el esquema de datos cambió durante el proyecto.

3.2 Una arquitectura en la que un LLM crea estrategias

- **El LLM propone y el código determinista juzga.** El modelo formula hipótesis y escribe reglas, pero no puede tocar el motor de backtest ni las métricas. Es la separación que hace auditable el resultado.
- **Un formato de código común para todas las estrategias.** Tres funciones parametrizadas bastan para que cualquier idea sea ejecutable, optimizable y traducible a MQL5. Cada estrategia guarda su código, así que se puede volver a evaluar con otros datos, costes o criterios.

- **Paralelismo sin coordinación central.** Cuatro Creators trabajan a la vez sobre el mismo catálogo con un bloqueo atómico, y el Orquestador solo fija prioridades.
- **Generación dirigida por las necesidades del portfolio.** `needs.md` cierra el ciclo entre lo desplegado y lo que se busca (por ejemplo, el desequilibrio del 90 % en largos detectado en abril).

3.3 Trazabilidad

Cada decisión queda escrita con su motivo: por qué se eligió un activo, en qué punto de control cayó una variante y qué se aprendió. Los descartes registrados se reparten entre CP1 (935, sin zona estable de parámetros), CP2 (517, fallo fuera de muestra) y CP0 (374, la lógica no funciona). Pocos sistemas de generación guardan los fracasos; aquí son la mayor parte del registro.

3.4 Un resultado negativo bien medido

El propio sistema detectó que sus filtros previos no bastaban: el evaluador rechazó el 88 % de lo que los Creators habían guardado. Ese dato, con 1.070 estrategias que conservan sus métricas en tres tramos, es un conjunto de datos reutilizable para estudiar qué predice el rendimiento posterior.

En las ocho estrategias analizadas para portfolio, la correlación semanal media fue de −0,01 y el drawdown conjunto quedó un 35 % por debajo de la suma de los individuales.

4. Limitaciones

La limitación central es que el proceso busca entre miles de variantes y después trata a las supervivientes como si se hubieran probado una sola vez. El dato que lo muestra está en la propia base de estrategias.

TRAMO	MEDIANA DEL PROFIT FACTOR (412 ESTRATEGIAS APROBADAS EN OOS)
In-sample	1,26
Out-of-sample	1,37
Evaluación (año siguiente)	0,99

En el tramo de evaluación, el 51 % de esas estrategias tiene PF inferior a 1 y solo el 25 % llega a 1,2. La correlación de rangos entre el PF de IS y el de evaluación es 0,27, y entre OOS y evaluación 0,19. Que el OOS salga mejor que el IS no indica robustez: es la huella de haber seleccionado por OOS. La muestra solo incluye supervivientes, así que estas correlaciones son un indicio y no una estimación.

4.1 Validez estadística

1. **Pruebas múltiples sin corrección.** Hay al menos 2.735 intentos registrados y cada uno incluye un grid y hasta dos cambios de lógica. Los umbrales son los de una prueba única (PF ≥ 1,10 con 10 operaciones, p < 0,15). Con ese volumen, decenas de aprobados por azar son lo esperable.
2. **El tramo OOS no es fuera de muestra.** Se usa para elegir el filtro de régimen, lo que lo convierte en un segundo tramo de ajuste.
3. **La memoria de lecciones filtra la validación al diseño.** `lessons_learned.md` (4.424 líneas) guarda resultados OOS con activo, timeframe y parámetros, y se lee antes de diseñar. `needs.md` dirige la búsqueda hacia lo «confirmado» en OOS. El agente no mira los datos, pero sí aprende de ellos por esta vía.
4. **El tramo de evaluación también se gasta.** Las 1.070 estrategias se miden contra el mismo año y el Orquestador reorienta la generación tras cada evaluación.
5. **Muestras pequeñas.** Diez operaciones bastan para validar en OOS y para el momentum de dos meses. Un Profit Factor sobre 10 o 30 operaciones tiene un error muy superior a la distancia entre 1,0 y 1,1.
6. **Demasiados parámetros para tan pocos datos.** La mediana es de 5 parámetros por estrategia (máximo 10) sobre dos años de IS.
7. **Umbrales sin justificación.** Ningún valor (0,70, 1,15, 1,10, 0,6, 1,5 veces, p < 0,15) procede de un experimento documentado en el proyecto.
8. **Test de entradas aleatorias condicionado.** Solo se aplica a lo que ya pasó PF > 1,2, de modo que su p-valor no es el de una prueba independiente. De 197 estrategias, 80 pasan con p < 0,15 y 35 con p < 0,05.

4.2 Mal de concepto

1. **Las hipótesis no son hipótesis.** Son combinaciones de indicadores sin un mecanismo de mercado que explique por qué deberían funcionar. El menú de cambios permite invertir la dirección cuando la idea falla, algo incompatible con partir de una explicación. El proceso es data-driven con vocabulario hypothesis-driven.
2. **Prohibir mirar los datos no evita el sesgo.** El LLM conoce por entrenamiento lo que hicieron el Nasdaq o el oro en los años usados como IS, OOS y evaluación. Las ideas que provienen de webs (644 de 928) también se publicaron después de ver esos mercados. Hay sesgo de anticipación aunque el agente no abra ningún fichero.
3. **El momentum de estrategias es un supuesto.** Seleccionar por los últimos dos meses presupone que el rendimiento reciente persiste. El proyecto no lo comprueba.
4. **Confusión entre ventaja y beta.** El 78 % de las estrategias son largas y 298 son sobre el Nasdaq, en años alcistas. El test de entradas aleatorias lo corrige en parte; los filtros anteriores, no.

5. **No hay método de portfolio.** Era el objetivo del proyecto y es lo que falta. No hay asignación de capital ni dimensionado: todo se mide a lote fijo.

4.3 Mal planteado: medición e implementación

1. **Stops solo al cierre de barra.** En 815 estrategias el stop se comprueba solo con el precio de cierre de cada barra, y la salida se ejecuta en la apertura de la siguiente; los Expert Advisors hacen lo mismo. Así, el stop no limita la pérdida de una operación: en una barra de 1 o 4 horas el precio puede ir mucho más allá antes de cerrar, o abrir con hueco. Además, el drawdown se calcula con operaciones cerradas y no recoge la pérdida flotante dentro de cada operación.
2. **Costes incompletos.** Solo se descuenta el spread. No hay deslizamiento ni financiación nocturna de los CFD.
3. **El protocolo lo ejecuta quien es evaluado.** Los puntos de control están escritos en lenguaje natural y los aplica el mismo agente que quiere aprobar estrategias. Nada verifica que los siga.
4. **Una sola métrica de decisión.** Casi todo se decide por Profit Factor.
5. **No se mide el resultado real.** No hay comparación detallada entre backtest y operativa.
6. **Consumo de uso muy alto.** Cuatro Creators, un Investigador y un Orquestador trabajando a la vez agotan rápido el límite de uso de Claude. El ritmo de generación depende más de ese límite que del diseño del proceso. Además, no se mide cuánto uso cuesta cada estrategia, así que no se puede saber si el paralelismo compensa ni cuánto costaría repetir el experimento.

5. Próximos pasos

PASO	QUÉ CORRIGE	ESTADO
Definir las reglas sin optimizar: sin grid search y con un máximo de 3 parámetros por estrategia	Pruebas múltiples; demasiados parámetros	Decidido para la siguiente versión
Formular cada hipótesis con un formato fijo y su mecanismo de mercado antes de cualquier backtest	Hipótesis que son combinaciones de indicadores	Decidido para la siguiente versión
Empezar con la memoria de lecciones en blanco, sin resultados de validación que orienten el diseño	Fuga de la validación al diseño	Decidido para la siguiente versión
Reservar un tramo final que no se consulte hasta el final, posterior al corte de conocimiento del modelo	OOS y evaluación reutilizados; sesgo de anticipación	Pendiente
Registrar todos los intentos, guardar también las descartadas y corregir los umbrales por el número de pruebas	Pruebas múltiples; test de entradas aleatorias condicionado	Pendiente
Contrastar si el rendimiento de los últimos 2 meses predice el de los siguientes	Momentum como supuesto	Pendiente
Revisar el código de las estrategias que entran en portfolio y sus Expert Advisors con un agente independiente	El protocolo lo ejecuta quien es evaluado	Decidido para la siguiente versión
Poner los stops como orden real en el bróker y añadir deslizamiento y financiación nocturna	Stops al cierre de barra; costes incompletos	Pendiente
Diseñar el módulo de portfolio: tamaño ajustado a la cuenta en cada rebalanceo mensual y un máximo de 20 posiciones	No hay método de portfolio	En diseño
Comparar cada semana el resultado real con el del backtest	No se mide el resultado real	Pendiente
Medir el consumo de uso por estrategia y usar subagentes en lugar de equipos de agentes	Consumo de uso muy alto	En marcha en la siguiente versión

Primer experimento: ¿qué métrica predice el rendimiento posterior?

Las 412 estrategias con métricas en tres tramos indican que seleccionar por Profit Factor en IS u OOS predice poco. El primer experimento será comprobarlo bien: evaluar también las estrategias descartadas, fijar los tramos por fecha y calcular los resultados con y sin costes. En lugar de un único valor por tramo, se puede seguir la evolución de cada métrica a lo largo del histórico.

Lo que esta versión enseña para el portfolio

Si las métricas individuales predicen poco, el método de portfolio no puede depender de acertar con cada estrategia. Tiene que funcionar con muchas estrategias de riesgo controlado y ventaja incierta, y demostrar su valor en el conjunto.