POPPER: Automated Hypothesis Validation with Agentic Sequential Falsifications

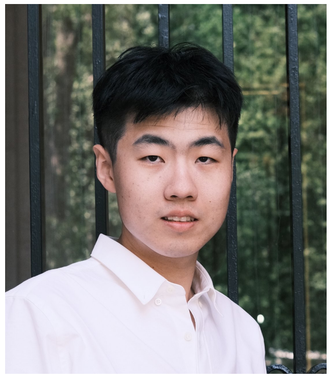
Ying Jin

Data Science Initiative & Harvard Medical School

Harvard University

International Seminar on Selective Inference, April 22, 2025

Collaborators



Kexin Huang
Stanford CS

Ryan Li
Stanford CS

Michael Li
Stanford CS

Emmanuel Candès
Stanford Stats & Math

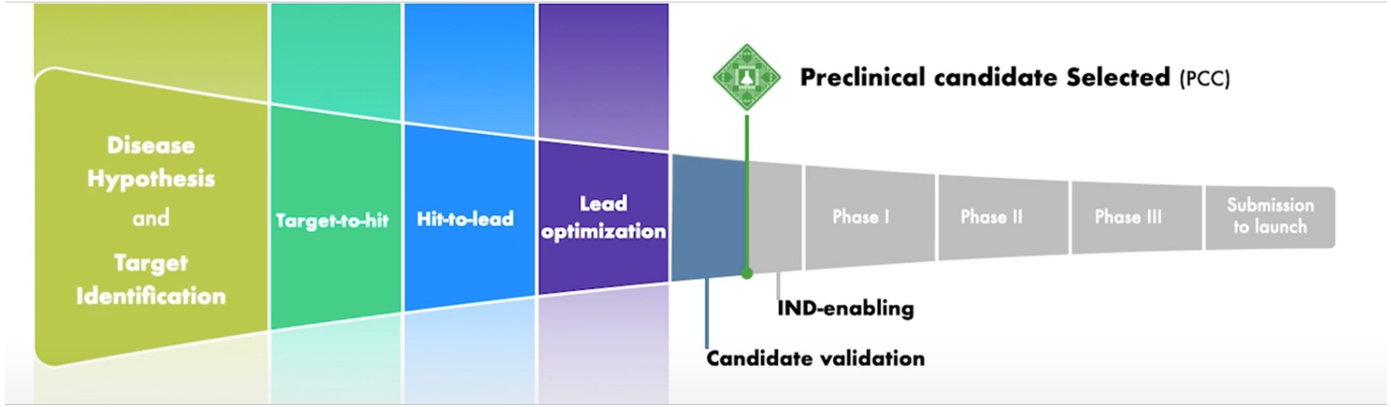
Jure Leskovec
Stanford CS

Stanford Artificial Intelligence

HDSI | Harvard Data Science Initiative

Stanford | Department of Statistics
SCHOOL OF HUMANITIES AND SCIENCES

AI for scientific discovery



Scientific discovery in the age of artificial intelligence

Hanchen Wang, Tianfan Fu, Yuanqi Du, Wenhao Gao, Kexin Huang, Ziming Liu, Payal Chandak, Shengchao Liu, Peter Van Katwyk, Andreea Deac, Anima Anandkumar, Karianne Bergen, Carla P. Gomes, Shirley Ho, Pushmeet Kohli, Joan Lasenby, Jure Leskovec, Tie-Yan Liu, Arjun Manrai, Debora Marks, Bharath Ramsundar, Le Song, Jimeng Sun, Jian Tang, ... Marinka Zitnik + Show authors

Nature **620**, 47–60 (2023) | Cite this article

151k Accesses | **588** Citations | **743** Altmetric | Metrics

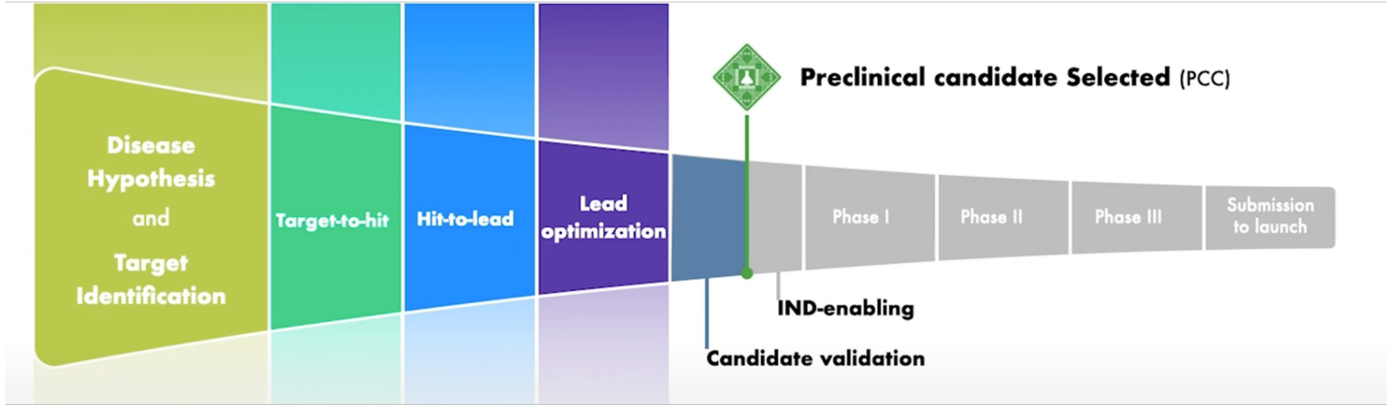
Recent emerging paradigm

AI as prediction engines → AI as automation agents

Predict drug binding affinities
Predict protein structures
Simulate particle interactions

Generate hypotheses
Design experiments (wet/dry lab)
Extract & synthesize knowledge

AI for scientific discovery



Scientific discovery in the age of artificial intelligence

Hanchen Wang, Tianfan Fu, Yuanqi Du, Wenhao Gao, Kexin Huang, Ziming Liu, Payal Chandak, Shengchao Liu, Peter Van Katwyk, Andreea Deac, Anima Anandkumar, Karianne Bergen, Carla P. Gomes, Shirley Ho, Pushmeet Kohli, Joan Lasenby, Jure Leskovec, Tie-Yan Liu, Arjun Manrai, Debora Marks, Bharath Ramsundar, Le Song, Jimeng Sun, Jian Tang, ... Marinka Zitnik + Show authors

Nature **620**, 47–60 (2023) | Cite this article

151k Accesses | **588** Citations | **743** Altmetric | Metrics

Recent emerging paradigm

AI as prediction engines → AI as automation agents

Accelerating scientific breakthroughs with an AI co-scientist [Google research]

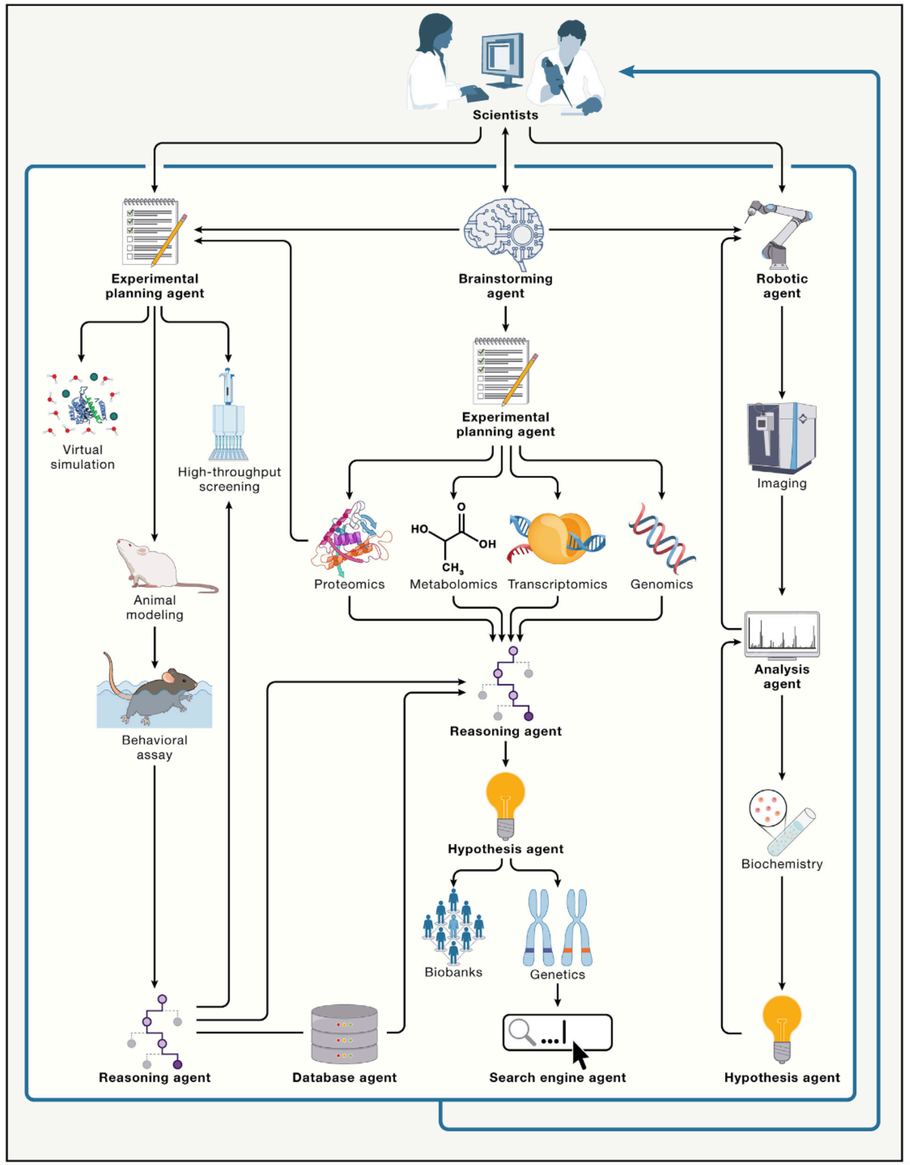
February 19, 2025 · Juraj Gottweis, Google Fellow, and Vivek Natarajan, Research Lead

The AI Scientist: Towards Fully Automated Open-Ended Scientific Discovery

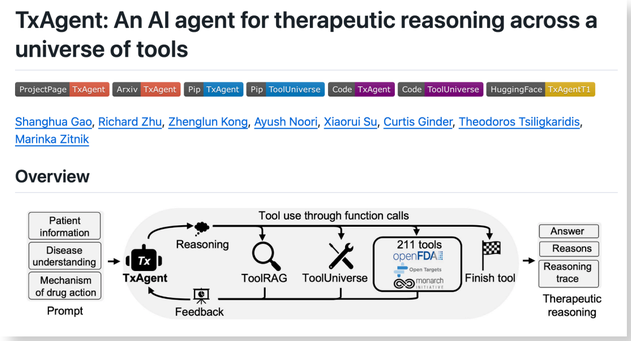
August 13, 2024 [Sakana.ai]

AI agents for scientific discovery

[Cell]



AI agents: computer programs that can use tools, internal knowledge, and reasoning capabilities to conduct various tasks.



(Representative frameworks later than Popper)

Mostly rely on LLMs for tool use, code generation, data exploration, etc.

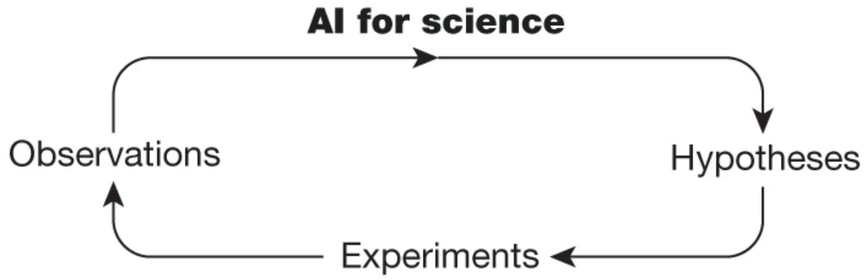
- How to make AI agents work like human scientists?
- How to measure the quality of scientific discoveries made by AI agents?

A formal framework to ground the discovery process is needed

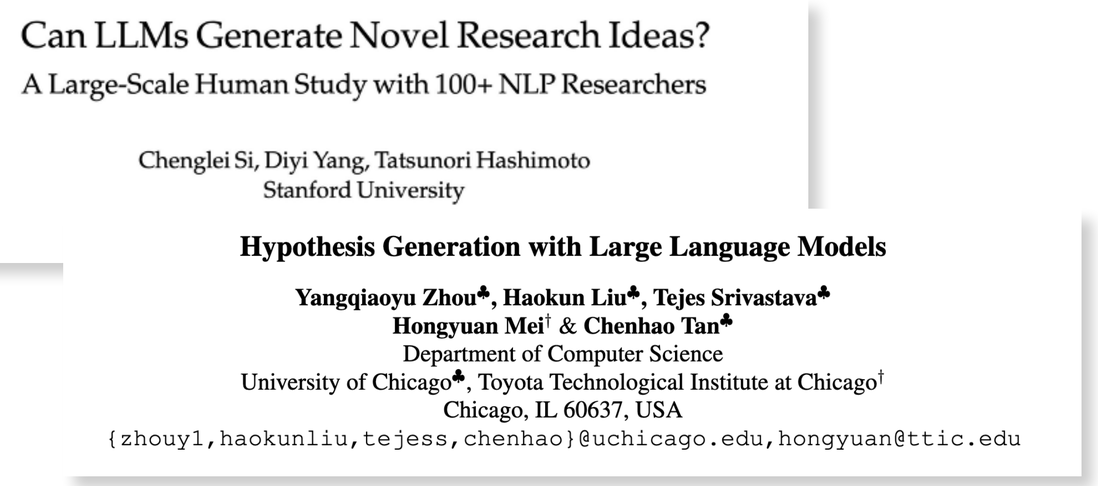
- This talk: some initial thoughts & explorations from a statistician's viewpoint

Grounding AI agents for scientific discovery?

Activities of AI agents can be framed as gathering evidence to test research hypotheses



A critical related thread: AI hypotheses generation



Can LLMs Generate Novel Research Ideas?

A Large-Scale Human Study with 100+ NLP Researchers

Chenglei Si, Diyi Yang, Tatsunori Hashimoto
Stanford University

Hypothesis Generation with Large Language Models

Yangqiaoyu Zhou[♣], **Haokun Liu**[♣], **Tejes Srivastava**[♣]
Hongyuan Mei[†] & **Chenhao Tan**[♣]
Department of Computer Science
University of Chicago[♣], Toyota Technological Institute at Chicago[†]
Chicago, IL 60637, USA
{zhouy1,haokunliu,tejess,chenhao}@uchicago.edu,hongyuan@ttic.edu

Important to make the process more rigorous

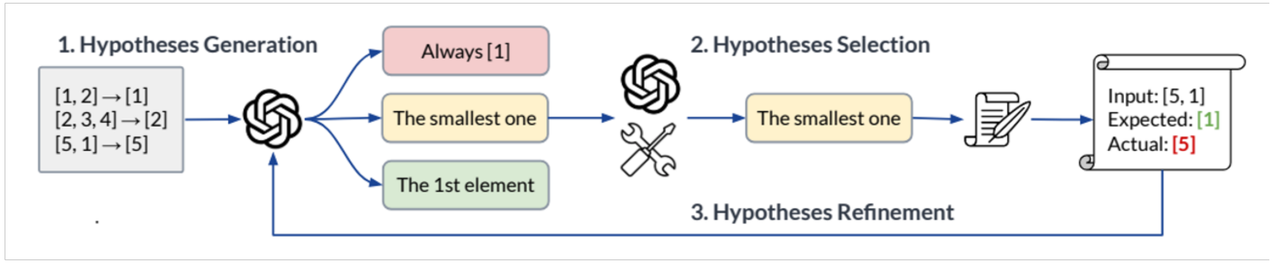
- Reasoning process has hallucinations
- Open-ended exploration hard to reach confident conclusions

Important to gauge their plausibility at scale

- Hypotheses generation also has hallucinations
- Not affordable/worthwhile pursing every generated hypothesis

This talk: automating hypothesis testing with AI agents

Hypothesis validation with AI agents



Common type of “hypothesis” in the literature

How to **rigorously** validate **free-form** hypothesis **at scale**?

Challenge: Hypothesis of interest can be free-form, abstract, hard to directly test with data

- E.g. LRRC32 is a drug target for Ulcerative colitis (UC)

Hypothesis

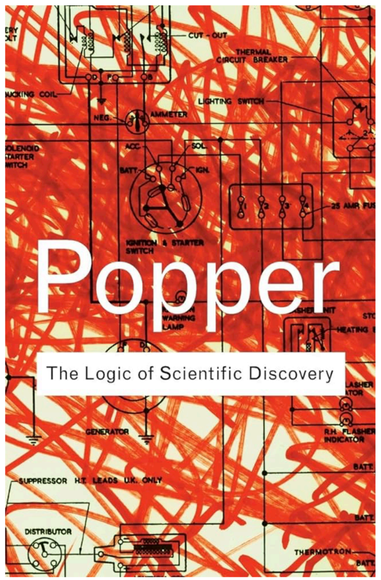
Pre-defined α

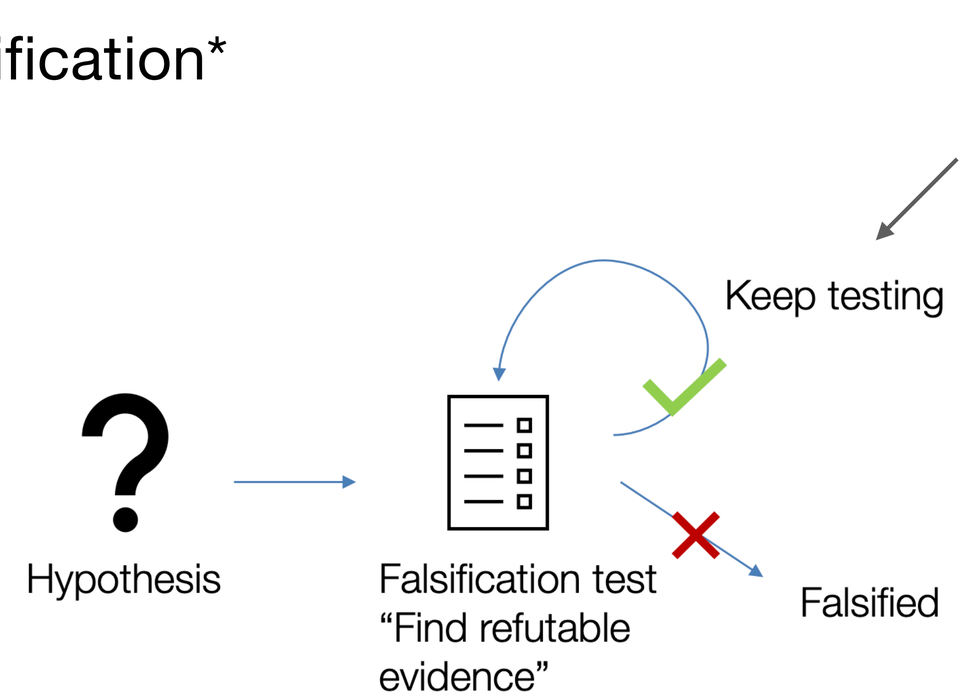
→ **AI agent** → Reject or not

Same as classical hypothesis testing, except that it’s done by AI agents.

How does a scientist validate a hypothesis?

Popper: testing by falsification*





* We take Popper’s perspective on the falsification process (nature of scientific testing), but not necessarily more broadly his view of what is scientific or not.

Why falsification tests?

Lots of scientific hypotheses cannot be directly validated with data, but their implications can

- Hypothesis (null): LRRC32 is not a drug target for Ulcerative colitis (UC)
- Implications of the null hypothesis
  - LRRC32 expression should not be significantly high at UC-relevant cell types
- Falsification experiment
  - Null: LRRC32 expression is **not high** in UC cell types
  - Alternative: LRRC32 expression is **high** in UC cell types

Testable implications provide the opportunity to connect scientific theory to observations by comparing the implications (predictions) of a general hypothesis with real-world evidence (Godfrey-Smith, 2023)

Why falsification tests?

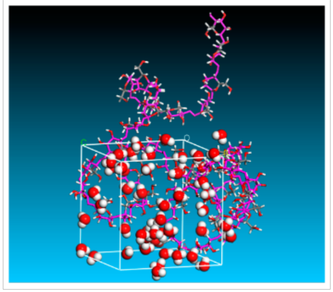
Lots of scientific hypotheses cannot be directly validated with data, but their implications can

Hypothesis → Implications → Experiments

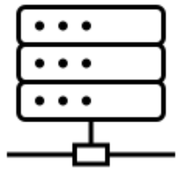
💡 Let agents conduct the falsification process!

Wet-lab

Dry-lab



Generate →



Data

→ Statistical test → P-value for testing an implication

Active data: New data produced for the experiment

Static data: existing massive hypothesis-free database for data analysis

Building an agent by imitating the falsification process

Designing a falsification experiment requires

- Reasoning
- Domain knowledge
- Creativity

We use an LLM as experiment design agent

- Input: the main hypothesis H
- Output: a designed experiment
  - Implied sub-hypothesis
  - The null and alternative
  - Experimental protocol (what data to collect/analyze)

Building an agent by imitating the falsification process

Designing a falsification experiment requires

- Reasoning
- Domain knowledge
- Creativity

We use an LLM as experiment design agent

Execution agent implements the experiment

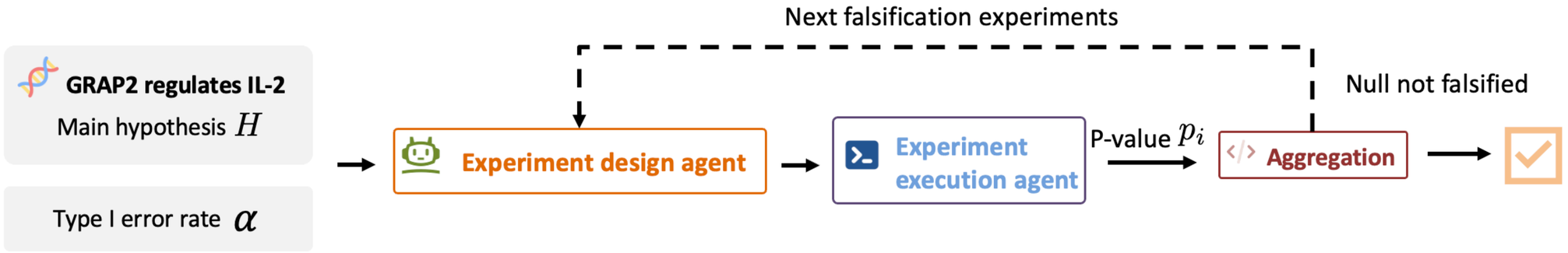
- Data analysis
- Wet-lab data collection (dream)
- Computer simulations

Basically, another LLM

- Input: the main hypothesis H
- Output: a designed experiment
  - Implied sub-hypothesis
  - The null and alternative
  - Experimental protocol (what data to collect/analyze)

- Input: the experimental design
- Output: p-values for testing the sub-hypothesis

Building an agent by imitating the falsification process



Designing a falsification experiment requires

- Reasoning
- Domain knowledge
- Creativity

We use an LLM as experiment design agent

Execution agent implements the experiment

- Data analysis
- Wet-lab data collection (dream)
- Computer simulations

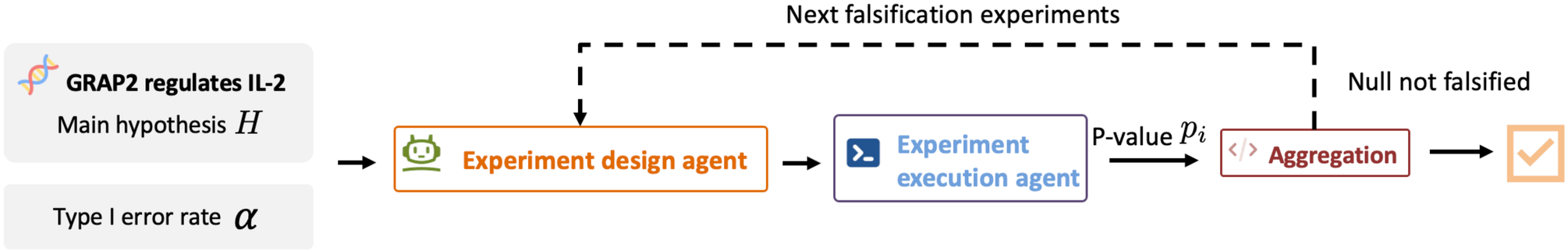
Basically, another LLM

- Input: the main hypothesis H
- Output: a designed experiment
  - Implied sub-hypothesis
  - The null and alternative
  - Experimental protocol (what data to collect/analyze)

- Input: the experimental design
- Output: p-values for testing the sub-hypothesis

Finally, want the process to continue if one round of falsification isn't enough: Aggregation!

Sequential safe testing with error control



Want the process to continue if one round of falsification isn't enough. Ideally,

- Adaptively determine whether to continue or not
- Combine diverse evidence from multiple rounds that test different sub-hypotheses
- Maintain statistical rigor for final output

JOURNAL ARTICLE

Safe testing FREE

Peter Grünwald ✉ , Rianne de Heide , Wouter Koolen Author Notes

Journal of the Royal Statistical Society Series B: Statistical Methodology, Volume 86, Issue 5, November 2024, Pages 1091–1128, https://doi.org/10.1093/jrsssb/qkae011

Published: 07 March 2024 **Article history** ▾

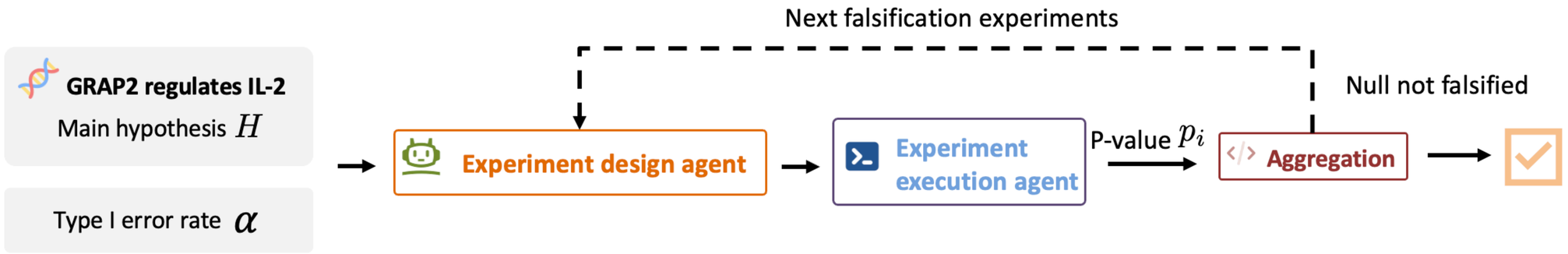
Sequential testing with e-values!

E-value: non-negative random variable whose expectation under the null is at most 1. [Vovk and Wang, 2021]

Benefits of e-values:

easy to combine, flexible to construct, optional stopping, valid testing, interpretable as "wealth"…

POPPER: theoretical framework



Goal: valid type-I error control $\mathbb{P}(\text{reject}) \leq \alpha$ if the main null hypothesis H_0 is true

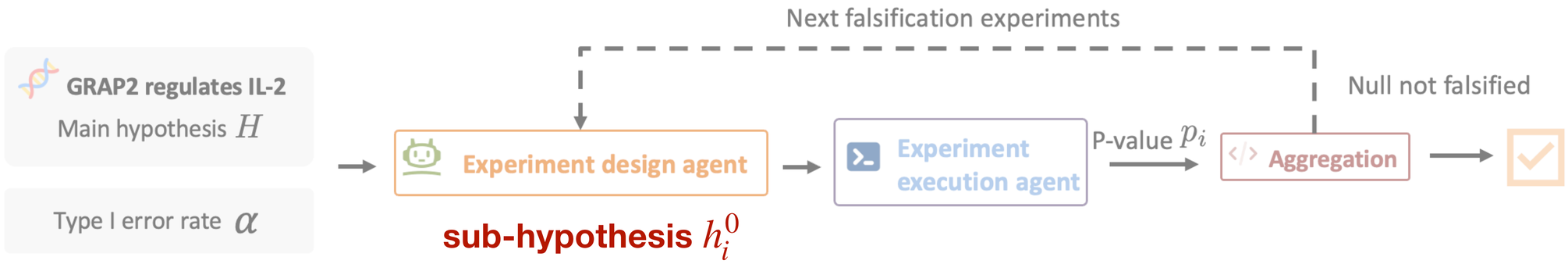
- Rigorous error control for scientific discovery
- Quantify strength of evidence even for early-stage exploration

What is needed in the whole process to guarantee type-I error control?

1. **Implication**
2. **E-value validity**
3. **Optional stopping**

Then, aggregating the e-values in existing rounds leads to a valid e-value for testing H_0

POPPER: theoretical framework

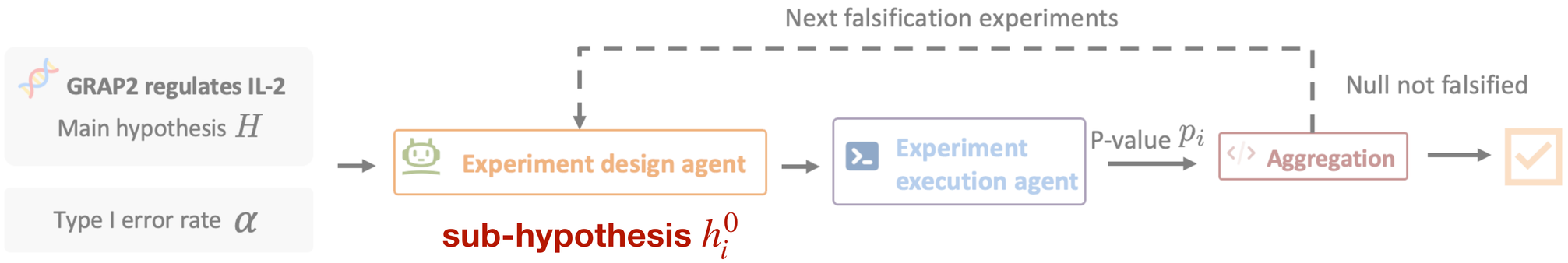


1. **Implication**: In each round i, the design agent proposes a sub-hypothesis h_i^0 such that H_0 implies h_i^0

The falsification sub-hypothesis needs to be very relevant to the main hypothesis such that if the null of the main hypothesis is true, then the null of sub-hypothesis must also be true.

- Null of the main hypothesis:
  - LRRC32 is **not** a drug target for Ulcerative colitis (UC)
- Null of the sub-hypothesis:
  - LRRC32 expression is **not** high in UC cell types

POPPER: satisfying condition 1



1. **Implication**: In each round i, the design agent proposes a sub-hypothesis h_i^0 such that H_0 implies h_i^0

- Self-refine based prompting

```
First produce an initial falsification test proposal.

Then, in each round i, you will do the following:
(1) critic: ask if the main hypothesis is null, is this test also null? be rigorous. this
    is super important, otherwise, the test is invalid. Is it redundant on capabilities
    with existing tests? Is it overlapping with failed tests? Can this be answered and
    implemented based on the given data?
(2) reflect: how to improve this test definition.
```

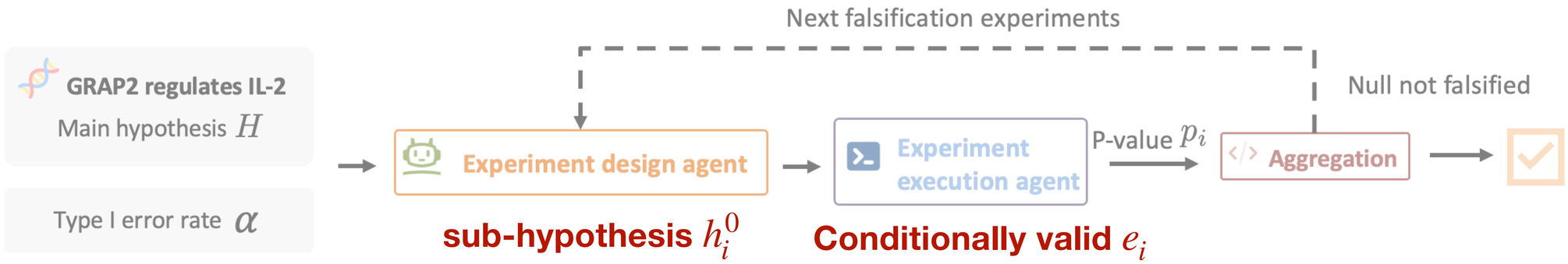
- Relevance checker

$$R(h) \in [0, 1]$$

Requires strong reasoning capabilities!

```
Rubric:

1.0 - Highly Relevant: The sub-hypothesis provides direct evidence or a clear mechanistic
    insight that strongly supports or refutes the main hypothesis. The test is specific to
     variables or mechanisms involved in the main hypothesis, with significant predictive
    value.
0.8 - Strongly Relevant: The test addresses a major component of the main hypothesis,
    providing substantial supporting or refuting evidence, and shows strong mechanistic
    alignment. The results would significantly impact the confidence in the main
    hypothesis.
0.6 - Moderately Relevant: The test examines elements supporting the main hypothesis
    without direct mechanistic insight. Some aspects align with the main hypothesis,
    offering moderate predictive value.
0.4 - Slightly Relevant: The test is related to the main hypothesis but provides limited
    direct evidence. It explores loosely associated variables and has minimal predictive
    value.
0.2 - Barely Relevant: The test is tangentially related, providing minimal information
    that could impact the main hypothesis, with no clear mechanistic link and negligible
    predictive value.
0.1 - Irrelevant: The sub-hypothesis does not provide relevant evidence or mechanistic
    connection to the main hypothesis, with no predictive value.
```

POPPER: theoretical framework

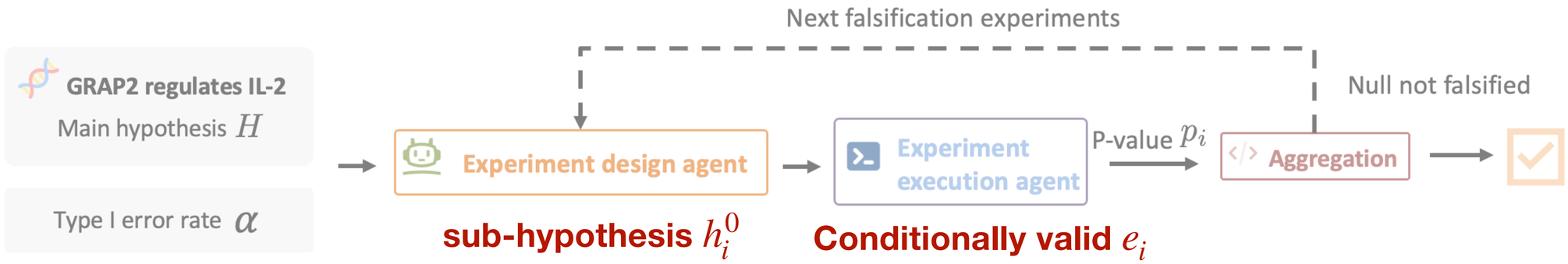


2. **E-value validity**: Let $\mathscr{D}_i$ be the data used in all rounds before i, then the execution agent in round i produces a nonnegative random variable e_i obeying $\mathbb{E}[e_i \mid \mathscr{D}_i] \leq 1$ under the sub-null h_i^0

Satisfied if

- Data is not reused
  - With static data (fixed database), decision to use a dataset doesn't involve the data
  - With active data collection, naturally true since data is collected anew
- Statistical analysis for getting the e-value is well executed

POPPER: satisfying condition 2

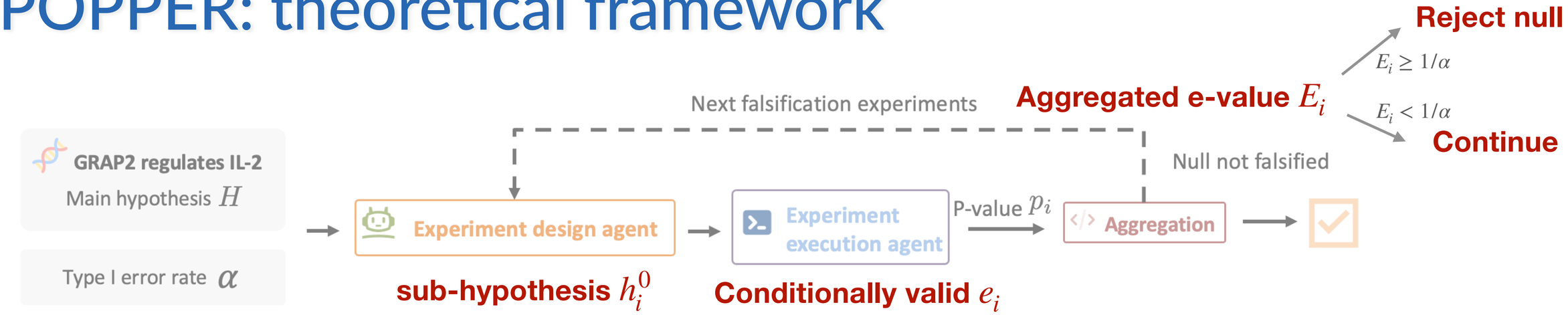


2. **E-value validity**: Let $\mathscr{D}_i$ be the data used in all rounds before i, then the execution agent in round i produces a nonnegative random variable e_i obeying $\mathbb{E}[e_i \mid \mathscr{D}_i] \leq 1$ under the sub-null h_i^0

- Controlling information flow
  - Meta-data only access (design agent sees only the schema, not the raw data)
  - No raw data from previous round is used in the next round
- Calibrate e-values from p-value (instead of directly asking for an e-value)
  - We find current LLMs are good at getting valid p-values, but not at producing e-values directly
  - So we first ask the agent to compute a (conditionally valid) p-value and convert it to an e-value

$$e_i = \kappa \times p_i^{\kappa-1}, \quad \kappa \in (0,1).$$

POPPER: theoretical framework



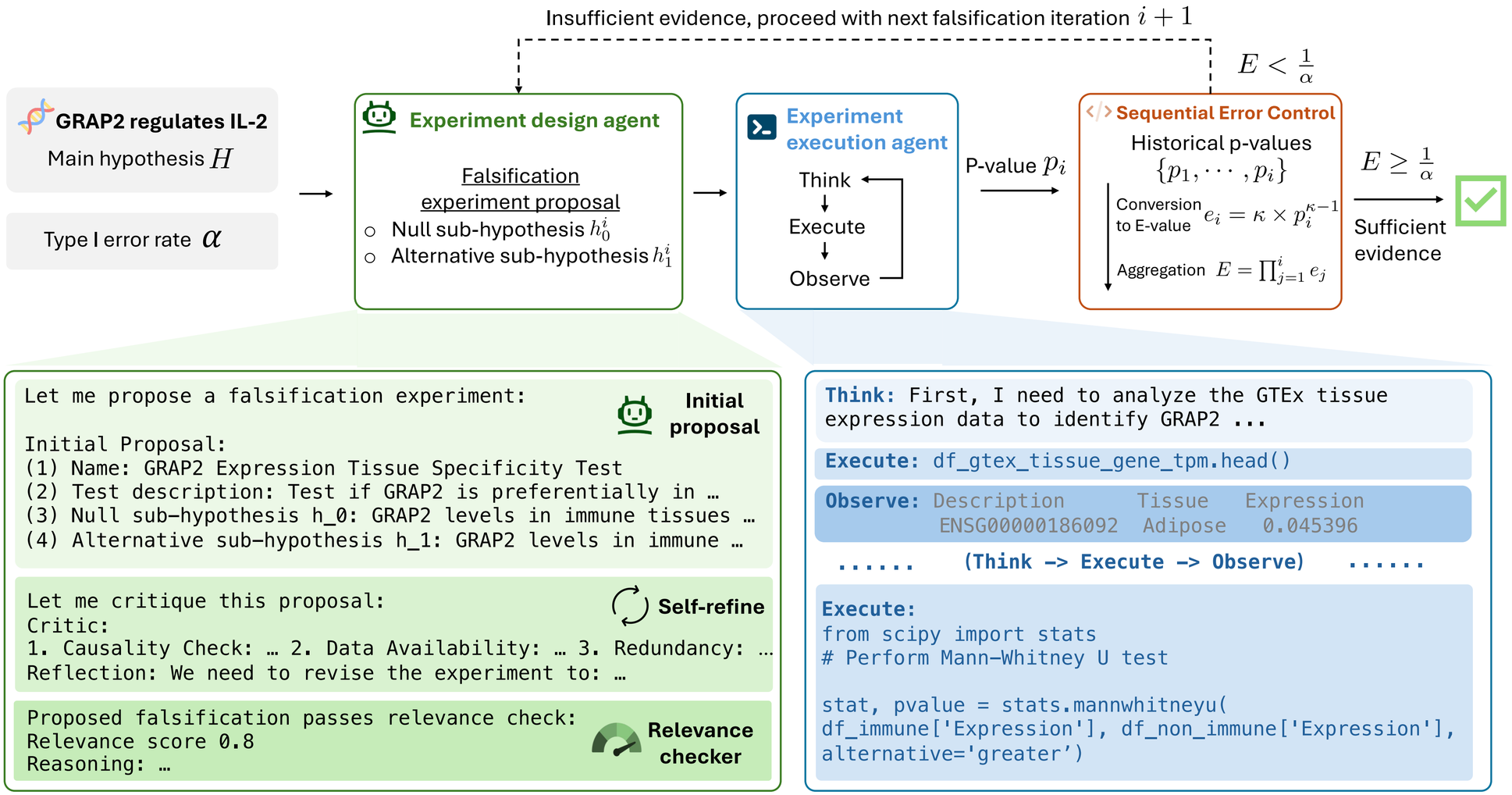
1. **Implication**: In each round i, the design agent proposes a sub-hypothesis h_i^0 such that H_0 implies h_i^0

2. **E-value validity**: Let $\mathscr{D}_i$ be the data used in all rounds before i, then the execution agent in round i produces a nonnegative random variable e_i obeying $\mathbb{E}[e_i \mid \mathscr{D}_i] \leq 1$ under the sub-null h_i^0

3. **Stopping**: Define aggregated e-value as $E_i = \prod_{t \leq i} e_t$. We stop if and only if $i \geq T_0$ or $E_i \geq 1/\alpha$

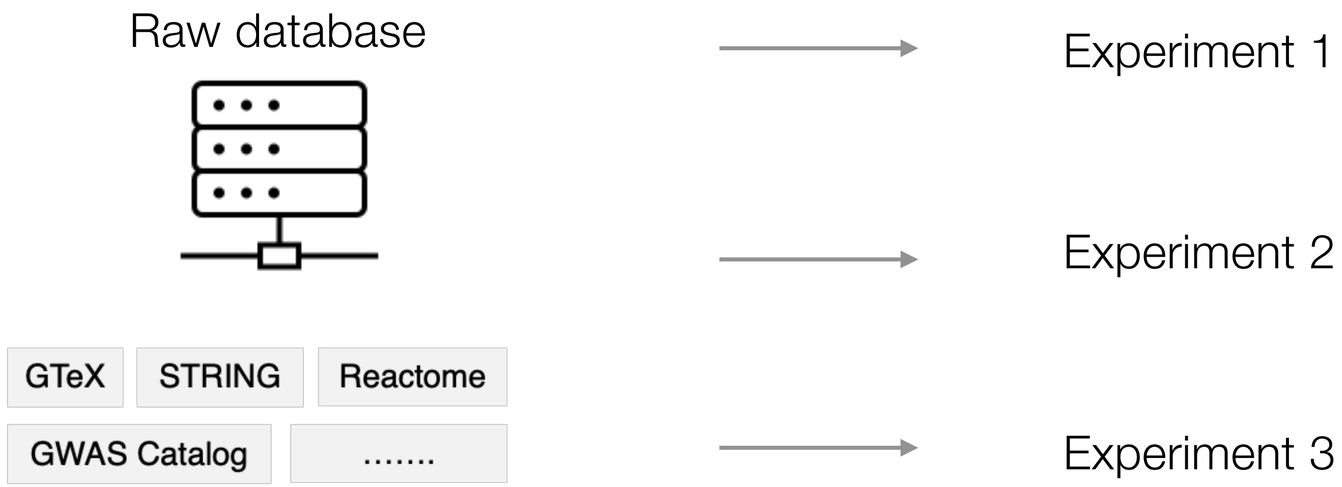
The terminal e-value E_τ obeys $\mathbb{E}[E_\tau] \leq 1$ under the global null H_0, hence the test $\mathbf{1}\{E_\tau \geq 1/\alpha\}$ controls the type-I error

POPPER


Insufficient evidence, proceed with next falsification iteration $i+1$
$E < \frac{1}{\alpha}$
GRAP2 regulates IL-2
Main hypothesis H
Type I error rate α
Experiment design agent
Falsification experiment proposal
Null sub-hypothesis h_0^i
Alternative sub-hypothesis h_1^i
Experiment execution agent
Think
Execute
Observe
P-value p_i
Sequential Error Control
Historical p-values $\{p_1, \cdots, p_i\}$
Conversion to E-value $e_i = \kappa \times p_i^{\kappa-1}$
Aggregation $E = \prod_{j=1}^{i} e_j$
$E \geq \frac{1}{\alpha}$
Sufficient evidence
Let me propose a falsification experiment:
Initial proposal
Initial Proposal:
(1) Name: GRAP2 Expression Tissue Specificity Test
(2) Test description: Test if GRAP2 is preferentially in …
(3) Null sub-hypothesis h_0: GRAP2 levels in immune tissues …
(4) Alternative sub-hypothesis h_1: GRAP2 levels in immune …
Let me critique this proposal:
Self-refine
Critic:
1. Causality Check: … 2. Data Availability: … 3. Redundancy: …
Reflection: We need to revise the experiment to: …
Proposed falsification passes relevance check:
Relevance score 0.8
Reasoning: …
Relevance checker
Think: First, I need to analyze the GTEx tissue expression data to identify GRAP2 ...
Execute: df_gtex_tissue_gene_tpm.head()
Observe: Description Tissue Expression
ENSG00000186092 Adipose 0.045396
...... (Think -> Execute -> Observe) ......
Execute:
from scipy import stats
# Perform Mann-Whitney U test
stat, pvalue = stats.mannwhitneyu(
df_immune['Expression'], df_non_immune['Expression'],
alternative='greater')

Instantiation of POPPER with static data

Static data: massive atlas of hypothesis-free diverse multi-scale and multi-modal datasets

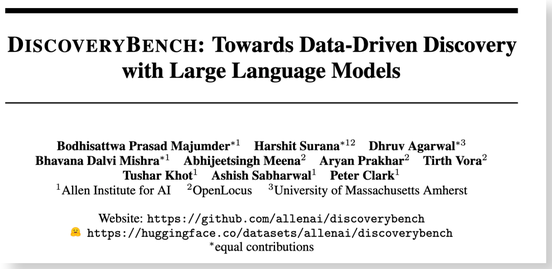


Experiment execution: python coding environment

Evaluation domain

- Target validation in biology (TargetVal):
  - A key task in drug discovery
  - Validating hypothesis of the form “Gene A regulates Phenotype B”
  - Ground truth: expensive CRISPR experiment

- Discovery Bench: 86 non-null hypotheses mined from literature with paired data
  - Sociology, biology, humanities, economics, engineering, and meta-science
  - “From 1700 BCE onwards, the use of hatchets and swords increased while the use of daggers decreased”



DISCOVERYBENCH: Towards Data-Driven Discovery with Large Language Models

Bodhisattwa Prasad Majumder*[1] Harshit Surana*[12] Dhruv Agarwal*[3]
Bhavana Dalvi Mishra*[1] Abhijeetsingh Meena[2] Aryan Prakhar[2] Tirth Vora[2]
Tushar Khot[1] Ashish Sabharwal[1] Peter Clark[1]
[1]Allen Institute for AI [2]OpenLocus [3]University of Massachusetts Amherst

Website: https://github.com/allenai/discoverybench
https://huggingface.co/datasets/allenai/discoverybench
*equal contributions

Curated static database for biology

df_gtex_tissue_gene_tpm:
{'Description': 'ENSG00000186092', 'Tissue': 'Adipose - Subcutaneous', 'Expression': 0.0453961, 'Gene': 'OR4F5'}

df_gwas_catalog:
{'DATE ADDED TO CATALOG': '2018-07-30', 'PUBMEDID': 27618447, 'FIRST AUTHOR': 'Surendran P', 'DATE': '2016-10-01', 'JOURNAL': 'Nat Genet', 'LINK': 'www.ncbi.nlm.nih.g
hypertension.', 'DISEASE/TRAIT': 'systolic blood pressure', 'INITIAL SAMPLE SIZE': 'up to 165,276 European ancestry individuals, up to 27,487 South Asian ancestry in
Hispanic individuals, 22,077 African American individuals', 'REGION': '3q26.2', 'CHR_ID': '3', 'CHR_POS': '169383111', 'REPORTED GENE(S)': 'MECOM', 'MAPPED_GENE': 'M
'DOWNSTREAM_GENE_DISTANCE': nan, 'STRONGEST SNP-RISK ALLELE': 'rs448378-A', 'SNPS': 'rs448378', 'MERGED': 0, 'SNP_ID_CURRENT': 448378.0, 'CONTEXT': 'intron_variant',
BETA': 0.0196, '95% CI (TEXT)': 'mmHg decrease', 'PLATFORM [SNPS PASSING QC]': 'Illumina [242296]', 'CNV': 'N'}

df_gene_info:
{'gene_id': 'ENSG00000228037', 'transcript_id': 'ENST00000424215', 'chr': '1', 'gene_start': 2581560, 'gene_end': 2584533, 'strand': 1, 'transcript_start': 2581560,
'gene_type': 'lncRNA'}

df_genetic_interaction:
{'interaction_id': 206363, 'gene_a_id': 'YCR011C', 'gene_b_id': 'YCL025C', 'experimental_system_type': 'genetic', 'pubmed_id': 'PUBMED:16269340', 'organism_id_a': 55

df_genebass_synonymous_filtered:
{'annotation': 'synonymous', 'Pvalue': 0.72199, 'Pvalue_Burden': 0.85198, 'Pvalue_SKAT': 0.52215, 'BETA_Burden': 0.0034703, 'SE_Burden': 0.018598, 'gene': 'TSPAN6',

df_genebass_missense_LC_filtered:
{'annotation': 'missense_LC', 'Pvalue': 0.85434, 'Pvalue_Burden': 0.69637, 'Pvalue_SKAT': 0.87135, 'BETA_Burden': -0.0046974, 'SE_Burden': 0.012037, 'gene': 'TSPAN6'

df_genebass_pLoF_filtered:
{'annotation': 'pLoF', 'Pvalue': 0.59145, 'Pvalue_Burden': 0.96398, 'Pvalue_SKAT': 0.42053, 'BETA_Burden': -0.00053797, 'SE_Burden': 0.011913, 'gene': 'DPM1', 'pheno_

df_affinity_capture_ms:
{'interaction_id': 241421, 'gene_a_id': 'ENSG00000169217', 'gene_b_id': 'ENSG00000144028', 'experimental_system_type': 'physical', 'pubmed_id': 'PUBMED:17353931', 'o

df_two_hybrid:
{'interaction_id': 269293, 'gene_a_id': 'ENSG00000114395', 'gene_b_id': 'ENSG00000109103', 'experimental_system_type': 'physical', 'pubmed_id': 'PUBMED:16169070', 'o

df_synthetic_growth_defect:
{'interaction_id': 299191, 'gene_a_id': 'SPBC30D10.04', 'gene_b_id': 'SPAC14C4.13', 'experimental_system_type': 'genetic', 'pubmed_id': 'PUBMED:18931302', 'organism_

df_affinity_capture_rna:
{'interaction_id': 637589, 'gene_a_id': 'YDR515W', 'gene_b_id': 'YAL030W', 'experimental_system_type': 'physical', 'pubmed_id': 'PUBMED:22271760', 'organism_id_a': 5

df_co_fractionation:
{'interaction_id': 740709, 'gene_a_id': 'ENSG00000160201', 'gene_b_id': 'ENSG00000063244', 'experimental_system_type': 'physical', 'pubmed_id': 'PUBMED:22939629', 'o

df_synthetic_lethality:
{'interaction_id': 818564, 'gene_a_id': 'YLR418C', 'gene_b_id': 'YLR103C', 'experimental_system_type': 'genetic', 'pubmed_id': 'PUBMED:23390603', 'organism_id_a': 55

df_dosage_growth_defect:
{'interaction_id': 908565, 'gene_a_id': 'YBL009W', 'gene_b_id': 'YPR058W', 'experimental_system_type': 'genetic', 'pubmed_id': 'PUBMED:22282571', 'organism_id_a': 55

df_proximity_label_ms:
{'interaction_id': 913336, 'gene_a_id': 'ENSG00000206560', 'gene_b_id': 'ENSG00000103194', 'experimental_system_type': 'physical', 'pubmed_id': 'PUBMED:24255178', 'o

df_synthetic_rescue:
{'interaction_id': 986367, 'gene_a_id': 'YJR017C', 'gene_b_id': 'YHR114W', 'experimental_system_type': 'genetic', 'pubmed_id': 'PUBMED:24470217', 'organism_id_a': 55

df_reconstituted_complex:
{'interaction_id': 2841862, 'gene_a_id': 'ENSG00000169083', 'gene_b_id': 'ENSG00000097007', 'experimental_system_type': 'physical', 'pubmed_id': 'PUBMED:24728074', '

df_eqtl_ukbb:
{'rs_id': 'rs111972115', 'gene_id': 'ENSG00000134250', 'cell_type_name': 'CEDAR-MONOCYTE_CD14', 'qtl_score': 4.851273983601853, 'gene_name': 'NOTCH2'}

df_pqtl_ukbb:
{'rs_id': 'rs41265213', 'gene_id': 'ENSG00000160712', 'cell_type_name': 'SCALLOP_2020-UBERON_0001969', 'qtl_score': 5.13715334001706, 'gene_name': 'IL6R'}

df_sqtl_ukbb:
{'rs_id': 'rs12756687', 'gene_id': 'ENSG00000265491', 'cell_type_name': 'GTEx-sQTL-Brain_Caudate_basal_ganglia', 'qtl_score': 3.19846622732464, 'gene_name': 'RNF115'

df_variant_table:
{'RS': 'rs116587930', 'ID': '1:727841_G_A', 'CHR': 1, 'POS': 727841, 'A1': 'G', 'A2': 'A', 'MAF': 0.0507035}

df_trait:
{'pheno_name': '120_1', 'description': 'Birth weight known; Yes - pounds and ounces'}

We curated a database of 85 million observations (rows) across 22 sources in biology for our experiment.

Typical biostatisticians work with a few datasets only; the agent can perform joint analysis across massive datasets from different modalities and sub-fields of biology.

This is a difficult task that requires expertise and familiarity with many cross-disciplinary skills.

A real example

Problem Statement

```
Main Hypothesis:
Gene GRAP2 regulates the production of Interleukin-2 (IL-2).

Data sources:
df_gtex_tissue_gene_tpm:
{'Description': 'ENSG00000186092', 'Tissue': 'Adipose - Subcutaneous',
 'Expression': 0.0453961, 'Gene': 'OR4F5'}

df_gene_info:
{'gene_id': 'ENSG00000228037', 'transcript_id': 'ENST00000424215',
 'chr': '1', 'gene_start': 2581560, 'gene_end': 2584533, 'strand': 1,
 'transcript_start': 2581560, 'transcript_end': 2584533,
 'tss': 2581560, 'transcript_is_canonical': 1.0, 'gene_name': nan,
 'percentage_gene_gc_content': 51.11, 'gene_type': 'lncRNA'}

df_genetic_interaction:
{'interaction_id': 206363, 'gene_a_id': 'YCR011C', 'gene_b_id': 'YCL025C',
 'experimental_system_type': 'genetic', 'pubmed_id': 'PUBMED:16269340',
 'organism_id_a': 559292, 'organism_id_b': 559292,
 'throughput_type': 'High Throughput', 'experimental_score': '-5.6431'}

# 19 more datasets omitted for presentation
```

A real example, continued

Experiment Design Agent

Round 4

Let me propose an initial falsification test:

Name: LCP2 eQTL-IL2 Regulatory Region Test

Test description: Test if genetic variants that affect LCP2 expression (eQTLs) are enriched in regulatory regions near the IL2 gene, which would suggest direct regulatory relationship.

Null sub-hypothesis h_0: Genetic variants that are eQTLs for LCP2 are not significantly enriched in regulatory regions near IL2 gene compared to background.

Alternative sub-hypothesis h_1: Genetic variants that are eQTLs for LCP2 are significantly enriched in regulatory regions near IL2 gene compared to background.

Let me critique this proposal:

Critic:
1. Causality Direction: If the main hypothesis (LCP2 regulates IL2) is null, this test might still show significance if there's a third factor affecting both genes, or if IL2 regulates LCP2. Therefore, the test doesn't fully satisfy the crucial property that null main hypothesis implies null sub-hypothesis.

2. Data Availability: Looking at the provided data sources, while we have eQTL data from UKBB (df_eqtl_ukbb), we don't have comprehensive regulatory region annotations or IL2 genomic location data to properly implement this test.

3. Non-redundancy: While this test examines a different aspect than previous tests (genetic regulation vs. expression correlation or protein interactions), it may partially overlap with the variant-phenotype associations tested in Round 3.

Reflection:
We need a test that:
1. Has stronger directional causality aligned with the main hypothesis
2. Can be clearly implemented with available data
3. Provides unique evidence

Let me propose a refined test:

Name: LCP2 Immune Cell-Specific Expression Test

Test description: Test if LCP2 expression is specifically elevated in immune cell types where IL2 production is known to occur (e.g., T cells) compared to other tissues using GTEx tissue expression data.

Null sub-hypothesis h_0: LCP2 expression in immune tissues is not significantly higher than in non-immune tissues.

Alternative sub-hypothesis h_1: LCP2 expression in immune tissues is significantly higher than in non-immune tissues.

Critic of refined version:
1. Causality: If the main hypothesis is null, this test would also be null because if LCP2 doesn't regulate IL2, there's no requirement for it to be specifically expressed in IL2-producing cells.
2. Data Feasibility: We can implement this using df_gtex_tissue_gene_tpm data.
3. Non-redundancy: This test provides unique tissue-specific evidence different from previous tests.
4. Biological Relevance: This test aligns with the biological context of IL2 regulation.

This refined test appears to meet all our criteria.

A real example, continued

Relevance Checker

```
Proposed falsification test passes relevance check:
 Proposal: Main hypothesis: Gene LCP2 regulates the production of Interleukin-2 (IL
     -2).
 Falsification Test name: LCP2 Immune Cell-Specific Expression Test
 Falsification Test description: Test if LCP2 expression is specifically elevated in
      immune cell types where IL2 production is known to occur (e.g., T cells)
     compared to other tissues using GTEx tissue expression data.
 Falsification Test Null sub-hypothesis: LCP2 expression in immune tissues is not
     significantly higher than in non-immune tissues.
 Falsification Test Alternate sub-hypothesis: LCP2 expression in immune tissues is
     significantly higher than in non-immune tissues.
Relevance score 0.8
Reasoning: This sub-hypothesis test is strongly relevant to the main hypothesis for
    several reasons:

1. Mechanistic Connection: Since IL-2 is primarily produced by immune cells (
    especially T cells), testing LCP2's expression pattern in immune vs non-immune
    tissues directly addresses a key mechanistic aspect of the hypothesized
    regulatory relationship.


2. Evidence Strength: The test provides strong circumstantial evidence - if LCP2 is
    specifically expressed in IL-2-producing immune cells, it supports its potential
     role as a regulator. However, it doesn't directly prove regulation, which is
    why it's not scored as 1.0.

3. Predictive Value: The test results would significantly impact confidence in the
    main hypothesis. If LCP2 is not expressed in IL-2-producing cells, it would be
    difficult to support its role as a direct regulator. Conversely, specific
    expression in these cells would strengthen the hypothesis.

The test doesn't get a 1.0 score because while it provides strong supporting
    evidence, it doesn't directly test the regulatory relationship - LCP2 could be
    expressed in these cells but not actually regulate IL-2 production.
```

A real example, continued

Experiment Execution Agent

```
I apologize, but I notice a potential issue. Looking at the datasets provided, while
     we have GTEx tissue expression data (df_gtex_tissue_gene_tpm), we need to
    ensure that:

1. The LCP2 gene is actually present in the dataset
2. We can properly identify immune vs non-immune tissues
3. We have enough samples for a meaningful statistical comparison

Let me first check if LCP2 is in the dataset.

--------------------------------
Action: python_repl_ast
Action Input: # Check unique genes in the dataset
unique_genes = df_gtex_tissue_gene_tpm['Gene'].unique()
print(f"Number of unique genes: {len(unique_genes)}")
print("\nFirst few genes:")
print(list(unique_genes)[:5])

# Check if LCP2 is in the dataset
lcp2_data = df_gtex_tissue_gene_tpm[df_gtex_tissue_gene_tpm['Gene'] == 'LCP2']
print("\nNumber of LCP2 entries:", len(lcp2_data))
```

Observation

```
Number of unique genes: 18639

First few genes:
['OR4F5', 'OR4F29', 'OR4F16', 'SAMD11', 'NOC2L']

Number of LCP2 entries: 54
```

Experiment Execution Agent

```
I apologize, but I notice that while we have LCP2 entries in the dataset, we should
    first verify if we have a clear categorization of immune vs non-immune tissues
    and ensure we have sufficient data points for both groups before proceeding with
     the test.

--------------------------------
Action: python_repl_ast
Action Input: # Let's see what tissues we have for LCP2 and their expression values
lcp2_data = df_gtex_tissue_gene_tpm[df_gtex_tissue_gene_tpm['Gene'] == 'LCP2'][['
    Tissue', 'Expression']]
print("Number of tissues with LCP2 expression:", len(lcp2_data))
print("\nTissues and their expression values:")
print(lcp2_data.sort_values('Expression', ascending=False))
```

A real example, continued

Observation

Number of tissues with LCP2 expression: 54

Tissues and their expression values:

	Tissue	Expression
994744	Whole Blood	91.342300
882754	Spleen	63.694300
677439	Lung	35.695100
416129	Cells - EBV-transformed lymphocytes	18.756800
24164	Adipose - Visceral (Omentum)	18.473700
864089	Small Intestine - Terminal Ileum	18.356600
5499	Adipose - Subcutaneous	15.769000
733434	Nerve - Tibial	9.721680
453459	Cervix - Endocervix	9.024360
378799	Breast - Mammary Tissue	8.800150
80159	Artery - Coronary	8.208850
434794	Cervix - Ectocervix	7.951800
808094	Prostate	7.538440
117489	Bladder	7.335620
61494	Artery - Aorta	6.843810
565449	Fallopian Tube	6.762460
490789	Colon - Transverse	6.693410
938749	Thyroid	6.290940
341469	Brain - Spinal cord (cervical c-1)	6.247870
976079	Vagina	5.778990
957414	Uterus	5.630640
42829	Adrenal Gland	5.457470
901419	Stomach	5.345570
98824	Artery - Tibial	5.117030
640109	Kidney - Medulla	4.768020
696104	Minor Salivary Gland	4.435650
546784	Esophagus - Muscularis	4.373440
248144	Brain - Frontal Cortex (BA9)	4.343040
509454	Esophagus - Gastroesophageal Junction	4.264300
472124	Colon - Sigmoid	4.195200
845424	Skin - Sun Exposed (Lower leg)	4.153360
360134	Brain - Substantia nigra	3.530560
229479	Brain - Cortex	3.333310
826759	Skin - Not Sun Exposed (Suprapubic)	3.123410
528119	Esophagus - Mucosa	3.122660
789429	Pituitary	2.997920
154819	Brain - Anterior cingulate cortex (BA24)	2.725530
658774	Liver	2.721940
752099	Ovary	2.544880
621444	Kidney - Cortex	2.409530
584114	Heart - Atrial Appendage	2.365340
285474	Brain - Hypothalamus	2.184320

A real example, continued

Experiment Execution Agent

```
I can now perform the falsification test. I'll categorize the tissues into immune
    and non-immune groups and perform a statistical test to compare LCP2 expression
    between these groups.

The following tissues will be considered immune-related:

1. Whole Blood
2. Spleen
3. Cells - EBV-transformed lymphocytes

Action: python_repl_ast
Action Input: # Define immune tissues
immune_tissues = ['Whole Blood', 'Spleen', 'Cells - EBV-transformed lymphocytes']

# Create two groups for comparison
immune_expression = lcp2_data[lcp2_data['Tissue'].isin(immune_tissues)]['Expression
    ']
non_immune_expression = lcp2_data[~lcp2_data['Tissue'].isin(immune_tissues)]['
    Expression']

# Perform Mann-Whitney U test (non-parametric test since we can't assume normal
    distribution)
from scipy.stats import mannwhitneyu

stat, pvalue = mannwhitneyu(immune_expression, non_immune_expression, alternative='
    greater')
print(f"Mann-Whitney U test statistic: {stat}")
print(f"P-value: {pvalue:.2e}")

print("\nMedian expression in immune tissues:", immune_expression.median())
print("Median expression in non-immune tissues:", non_immune_expression.median())
```

Observation

```
Mann-Whitney U test statistic: 152.0
P-value: 8.06e-05

Median expression in immune tissues: 63.6943
Median expression in non-immune tissues: 4.2643
```

Experiment Execution Agent

```
Based on the statistical analysis, I can now provide a conclusive answer about the
    falsification test:

Final Answer: Falsification test passes. The null hypothesis is rejected with a p-
    value of 8.06e-05. The analysis shows that LCP2 expression is significantly
    higher in immune tissues compared to non-immune tissues. The median expression
    in immune tissues (63.69 TPM) is substantially higher than in non-immune tissues
     (4.26 TPM), supporting the alternate hypothesis that LCP2 expression is
    specifically elevated in immune tissues.
```

A real example, continued

Sequential Error Control

```
============================== Sequential Testing ==============================
List of p-values: [1.0, 1.0, 0.0942, 8.06e-05]
 Summarized sequential statistics: 22.682297641798968
 Sequential test result: sufficient evidence - PASS
================ Testing decision is sufficient evidence - PASS ================
```

Summarizer

```
Let me summarize the sequential falsification testing results:

(1) Main Scientific Hypothesis:
Gene LCP2 regulates the production of Interleukin-2 (IL-2)

(2) Result of Sequential Falsification Test:
The sequential testing passed with a combined e-value of 22.68, after completing 4
    tests.

(3) Reasoning and Analysis:
- Expression Correlation Test (Round 1) showed strong correlation between LCP2 and
    IL-2 expression
- Regulatory Network Analysis (Round 2) demonstrated significant physical
    interactions with IL-2 regulators
- Variant-Immune Phenotype Association Test (Round 3) failed to show significant
    association (p=0.0942 > 0.05)
- Immune Cell-Specific Expression Test (Round 4) showed highly significant immune
    tissue-specific expression (p=8.06e-05)

(4) Conclusion: True

(5) Rationale:
The hypothesis is supported based on multiple lines of evidence:
- Strong expression correlation between LCP2 and IL-2
- Confirmed physical interactions with IL-2 regulatory proteins
- Highly significant immune cell-specific expression pattern
- The combined e-value (22.68) indicates strong cumulative evidence
Although the variant association test didn't reach significance, the other three
    tests provided strong supporting evidence, and the overall sequential testing
    procedure passed the predetermined threshold. The multiple complementary
    approaches testing different aspects of the relationship between LCP2 and IL-2
    production collectively support the regulatory relationship.
```

POPPER achieves type-I error control & improved power

- Evaluating type-I error
  - Randomly permute relevant columns in the databases so all hypotheses are null
- Evaluating power
  - Use the original database, but test on hypotheses known to be non-null

Methods using LLM capabilities without the "statistical layer"

Methods using other aggregation of p-values / construction of e-values

Method	Type I Error ($\alpha = 0.1$)			Power		
	DiscoveryBench	TargetVal-IL2	TargetVal-IFNG	DiscoveryBench	TargetVal-IL2	TargetVal-IFNG
CodeGen	0.145±0.031✗	0.020±0.014✓	0.004±0.009✓	0.378±0.066	0.140±0.022	0.040±0.042
CodeGen (o1)	0.248±0.015✗	0.013±0.012✓	0.000±0.000✓	0.419±0.028	0.250±0.100	0.183±0.076
ReAct	0.078±0.061✓	0.000±0.000✓	0.000±0.000✓	0.383±0.017	0.010±0.022	0.020±0.045
Self-Refine	0.117±0.028✗	0.100±0.069✓	0.067±0.064✓	0.476±0.066	0.183±0.029	0.067±0.064
Fisher Combined Test	0.311±0.040✗	0.264±0.083✗	0.173±0.023✗	0.741±0.058	0.800±0.071	0.650±0.050
LLM-Likelihood ratio	0.152±0.031✗	0.016±0.014✓	0.180±0.028✗	0.428±0.034	0.185±0.074	0.357±0.132
POPPER (Ours)	0.103±0.020✓	0.082±0.046✓	0.085±0.028✓	**0.638*±0.066**	**0.580*±0.125**	**0.591*±0.069**

- Safe testing and versatile p-to-e calibration is key to error control

POPPER achieves type-I error control & improved power

Method	Type I Error ($\alpha = 0.1$)			Power		
	DiscoveryBench	TargetVal-IL2	TargetVal-IFNG	DiscoveryBench	TargetVal-IL2	TargetVal-IFNG
POPPER-NoReleCheck	$0.134_{\pm 0.021}$ ✗	$0.340_{\pm 0.139}$ ✗	$0.300_{\pm 0.113}$ ✗	$0.610_{\pm 0.042}$	$0.897_{\pm 0.004}$	$0.717_{\pm 0.126}$
POPPER-CodeGen	$0.140_{\pm 0.022}$ ✗	$0.105_{\pm 0.017}$ ✓	$0.090_{\pm 0.045}$ ✓	$0.544_{\pm 0.032}$	$0.526_{\pm 0.133}$	$0.450_{\pm 0.079}$
POPPER (Ours)	$0.103_{\pm 0.020}$ ✓	$0.082_{\pm 0.046}$ ✓	$0.085_{\pm 0.028}$ ✓	**$0.638^{*}_{\pm 0.066}$**	**$0.580^{*}_{\pm 0.125}$**	**$0.591^{*}_{\pm 0.069}$**

Compared to NoReleCheck

⇝ relevant checker is a key factor in controlling type-I error (assumption 1)

Compared to CodeGen

⇝ reasoning in execution agent improves power

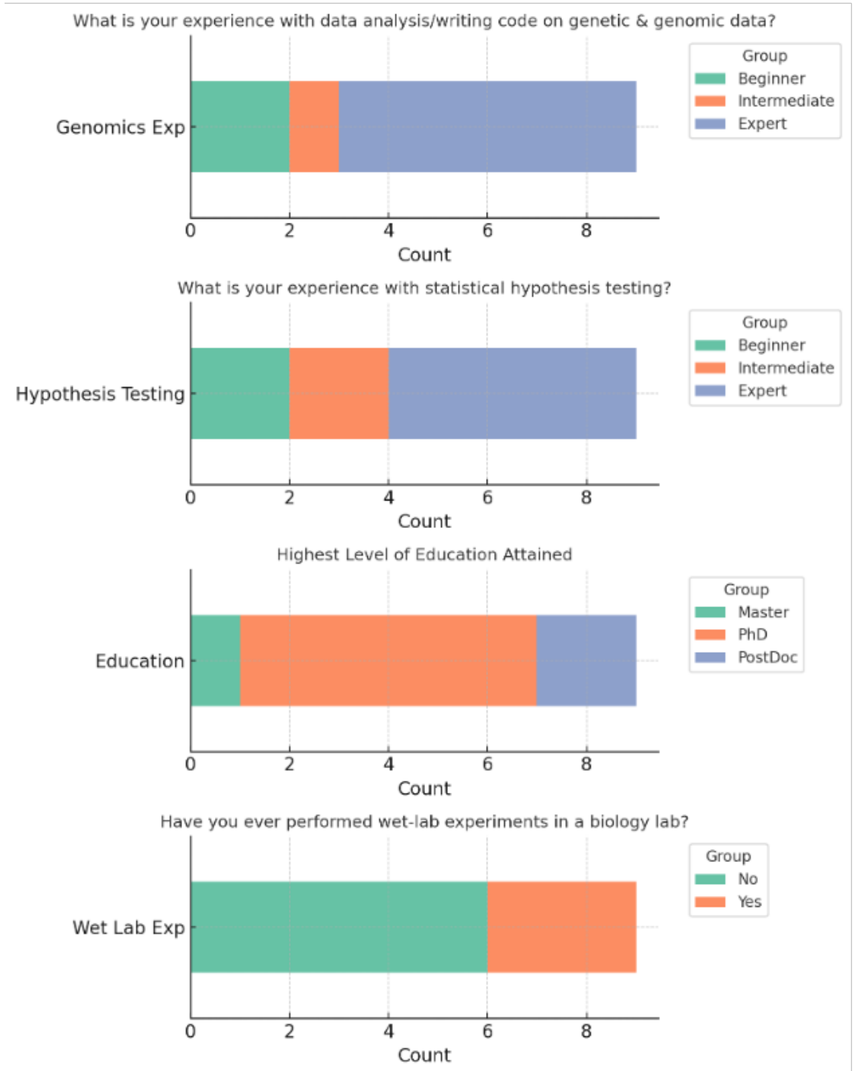
POPPER with various LLM backbones

Method	Type I Error ($\alpha = 0.1$)		Power	
	DiscoveryBench	TargetVal-IL2	DiscoveryBench	TargetVal-IL2
Claude-Haiku-3.5	0.230±0.079	0.780±0.120	0.844±0.017	0.835±0.113
Llama 3.3 70B	0.147±0.036	0.116±0.020	0.690±0.027	0.515±0.078
GPT-4o	0.143±0.039	0.096±0.043	0.730±0.054	0.385±0.102
Claude-Sonnet-3.5	0.103±0.020	0.082±0.046	0.638±0.066	**0.580***±0.125
o1	**0.091***±0.015	**0.031***±0.015	**0.654***±0.019	0.336±0.121

- The success of POPPER relies heavily on the reasoning capabilities of LLMs
  - Need to create correct implied sub-hypotheses
  - Need to generate reasonable experimental designs
  - Need to coherently generate code (tool use)
  - Need to faithfully execute the experimental plan without hallucinations

Comparing POPPER with human experts

- We recruited 9 >= graduate-level participants to conduct 2 hypothesis testing tasks from TargetVal-IL2, with access to the same databases (18 tasks are randomly assigned)



Once you open the notebook, run the following cell to start the time clock

In [1]:
```
import time
start_time = time.time()
```

Instructions

Given a biology hypothesis **"Gene MAK16 regulates the production of Interleukin-2 (IL-2)."**, your task is to validate it using the given raw databases by performing relevant data analysis, formulating statistical tests, and implementing them. The validation should be purely data-driven, not literature-driven. For statistical test, use significance level of alpha=0.1.

Output (1) If the hypothesis is valid or not given the data (2) relevant statistics (e.g. p-value, etc)

IMPORTANT

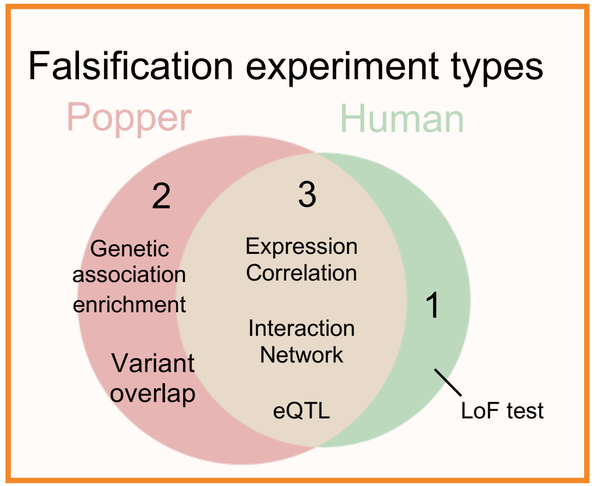
- You must only use the `database` folder in the current task folder to perform the analysis. DO NOT use the data from the other task or any external data.
- You must NOT use LLMs or internet about the direct answer to the biological hypothesis.
- You can use internet/LLMs if you are not sure about the code syntax or library usage or statistical tests or have biological questions in general.
- You can use any python library to perform the analysis.
- *The tasks are randomly sampled and may be one true & one false / all true / all false*

Here are the list of available data sources with columns and example rows:

```
df_gtex_tissue_gene_tpm:
{'Description': 'ENSG00000186092', 'Tissue': 'Adipose - Subcutaneous',
'Expression': 0.0453961, 'Gene': 'OR4F5'}

df_gene_info:
{'gene_id': 'ENSG00000228037', 'transcript_id': 'ENST00000424215', 'chr': '1',
'gene_start': 2581560, 'gene_end': 2584533, 'strand': 1, 'transcript_start':
2581560, 'transcript_end': 2584533, 'tss': 2581560, 'transcript_is_canonical':
1.0, 'gene_name': nan, 'percentage_gene_gc_content': 51.11, 'gene_type':
'lncRNA'}

df_genetic_interaction:
{'interaction_id': 206363, 'gene_a_id': 'YCR011C', 'gene_b_id': 'YCL025C',
'experimental_system_type': 'genetic', 'pubmed_id': 'PUBMED:16269340',
'organism_id_a': 559292, 'organism_id_b': 559292, 'throughput_type': 'High
Throughput', 'experimental_score': '-5.6431'}

# some dataframes are omitted for presentation purposes
```

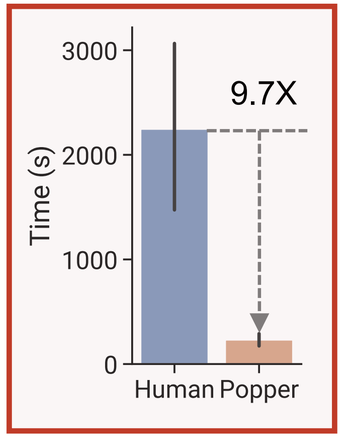
In [3]:
```
## loading the datasets
import pandas as pd
import glob

database = {}
```

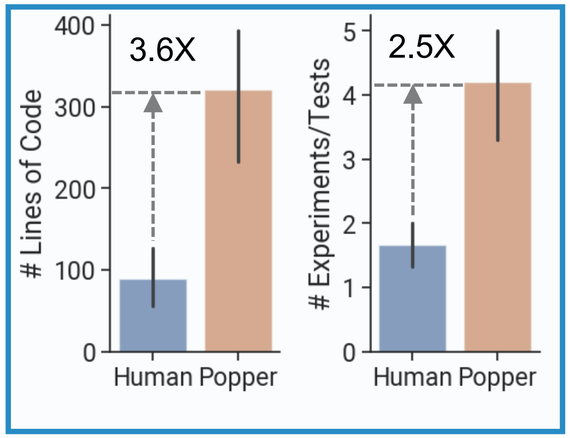
Comparing POPPER with human experts



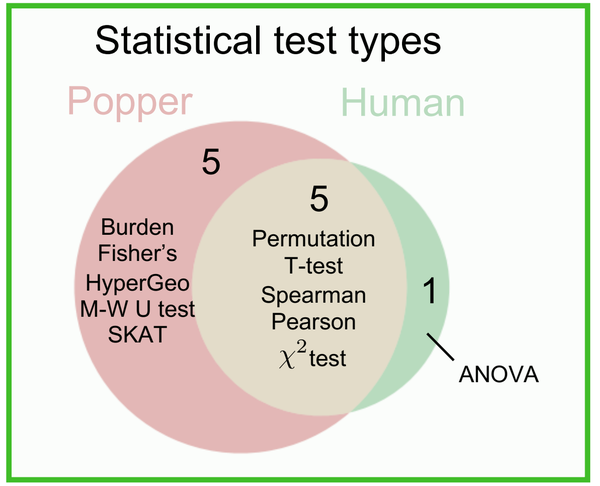
Overlap in the type of tests conducted, POPPER slightly more diverse

POPPER attains similar accuracy while greatly reducing the time



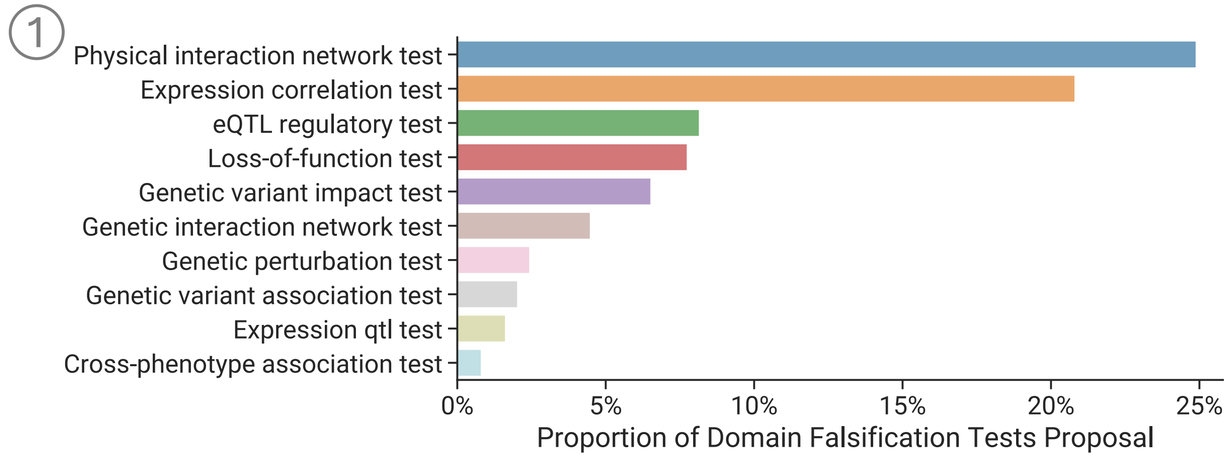


POPPER conducts more tests (more labors)

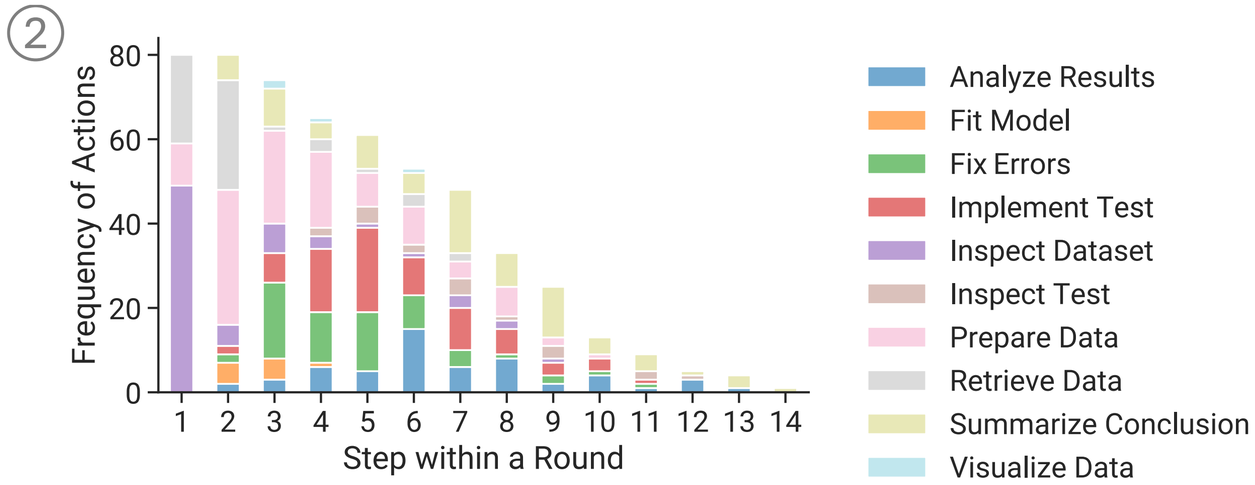


POPPER conducts more diverse tests than human scientists

Trajectory analysis



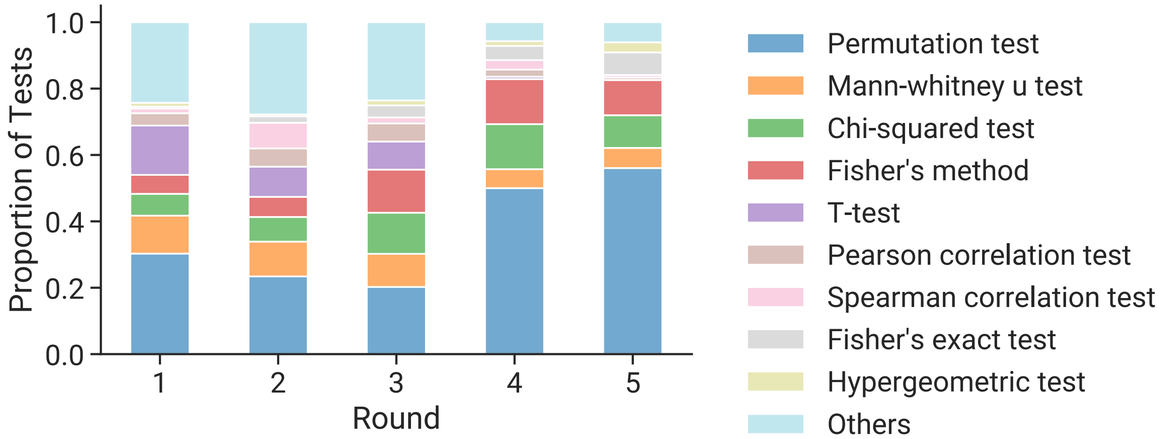
Types of tests conducted



Actions taken in various steps in each round of falsification

Trajectory analysis

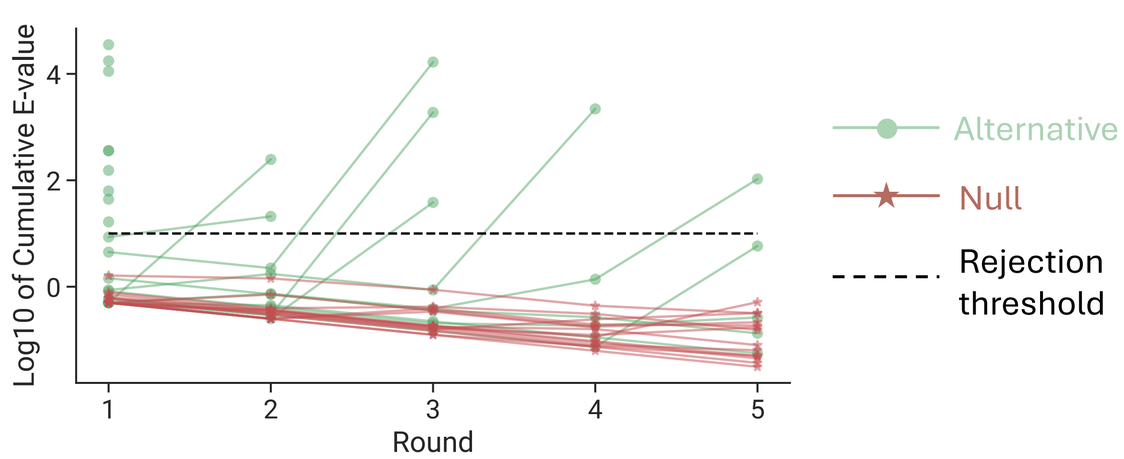
③



Specific statistical tests used in each round

- POPPER prefers nonparametric tests

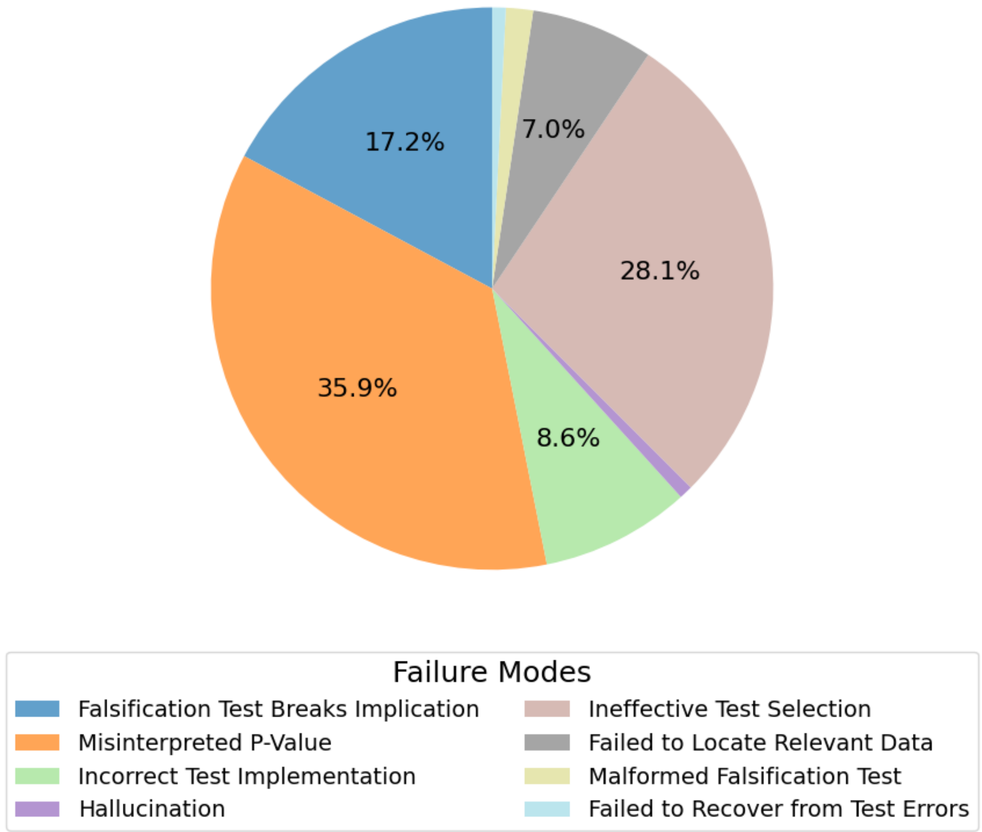
④



POPPER accumulates evidence for true hypotheses while maintaining validity for null hypotheses

- Can be conservative due to the use of e-values

When does POPPER fail to execute an experiment?



Summary

- POPPER is an agentic framework to validate a free-form hypothesis
  - Convert the main hypothesis to various implied, testable sub-hypotheses
  - Accumulate evidence from rounds of experiments to falsify the main hypothesis
  - E-value and sequential safe testing allows adaptive continuation of the process

- Unleashing the power of LLMs while maintaining statistical rigor
  - A “minimal” statistical framework to regulate scientific discovery by LLM agents
  - Safely exploit the encoded knowledge and reasoning abilities of LLM agents

- Limitations and future works
  - Relies on strong reasoning capabilities: errors are hard to detect & avoid due to randomness in generative AI
  - Need several tricks to make sure assumptions are satisfied (relevance checker, self reflection, etc.)
  - Other guarantees beyond type-I error (ongoing POPPER-v2)

Thank you!

Huang, K., Jin, Y., Li, R., Li, M. Y., Candès, E., & Leskovec, J. (2025). *Automated Hypothesis Validation with Agentic Sequential Falsifications*. arXiv preprint arXiv:2502.09858.

Paper

GitHub

POPPER Public

Automated Hypothesis Testing with Agentic Sequential Falsifications

Python 168 15 0 0 Updated on Feb 24

```python
from popper import Popper

agent = Popper(llm="claude-3-5-sonnet-20240620")
agent.register_data(data_path='path/to/data', loader_type='bio')
agent.configure(alpha=0.1)
results = agent.validate(hypothesis="Your hypothesis here")
```