Same or Different?
classical and ML methods for two-sample testing in Science

Gaia Grosso
gaiag795@mit.edu

March 30, 2026

A few words about myself

Gaia Grosso, IAIFI fellow (MIT, Harvard)

- Background education: Particle Physics
- Research interest:
  - Signal-agnostic discovery in collider experiments with ML
  - Tools for statistical detection with ML
    - Role of inductive bias and ML design
    - Data-driven scientific discovery at scale
    - Validation and Monitoring of ML models

https://iaifi.org/

Outline of the lecture

Two-sample test for scientific applications

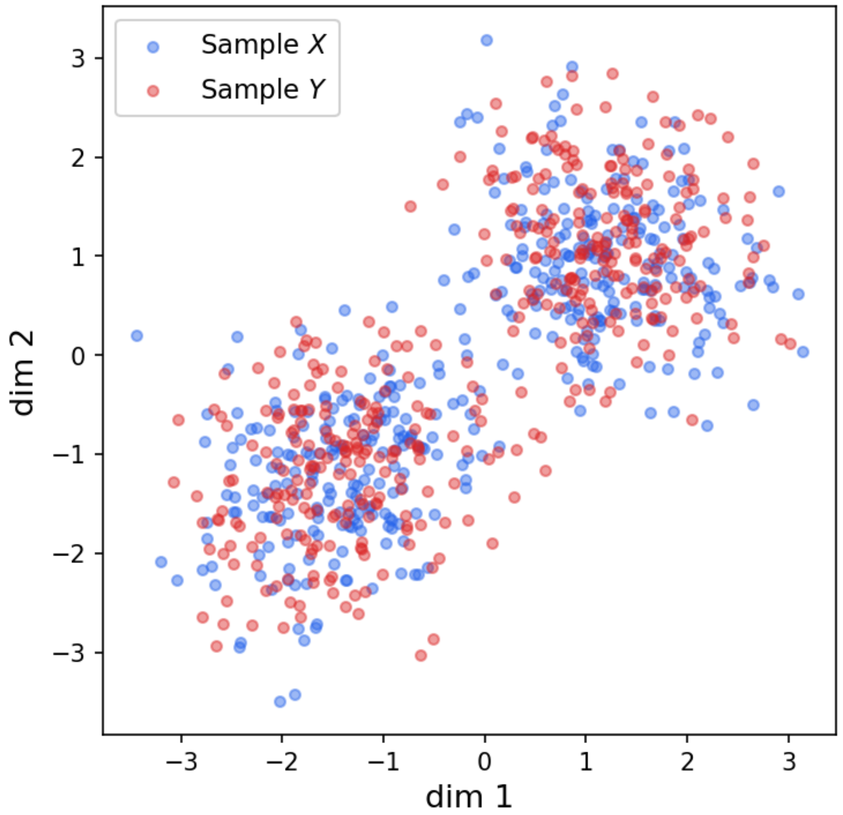
1. DEFINITION
2. RELEVANCE
3. CHALLENGES
4. METHODS: from classical to ML, caveats, good practice
5. USE CASE: Scientific Discovery

Problem statement

- A collection of N_X independent observations: $X = \{x_i \in \mathbb{R}^d, x_i \sim P_X\}_{i=1}^{N_X}$
- A collection of N_Y independent observations: $Y = \{y_i \in \mathbb{R}^d, y_i \sim P_Y\}_{i=1}^{N_Y}$

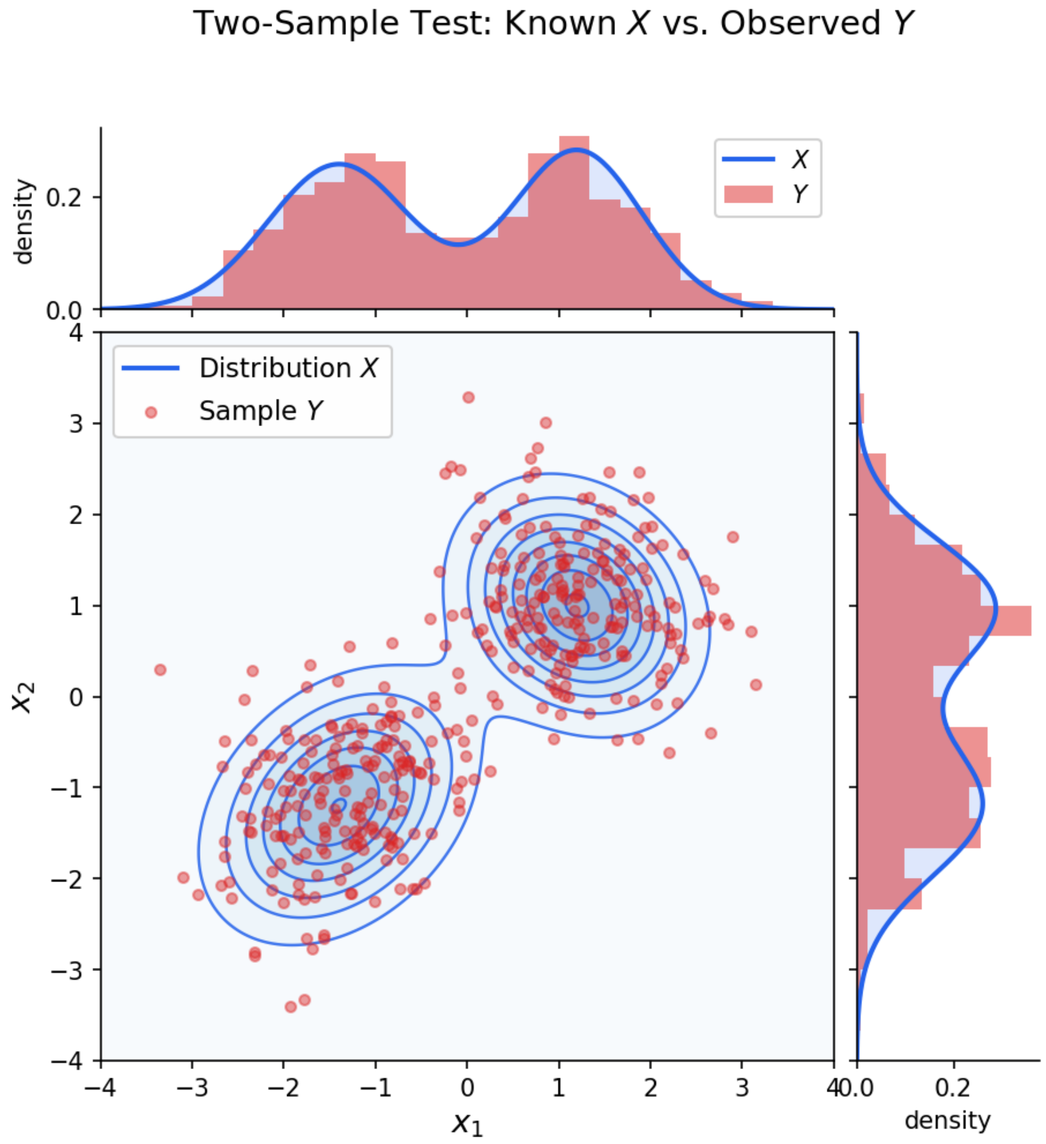
Do X and Y come form the same generative process?

$$\mathrm{H}_0 : P_X = P_Y$$
$$\mathrm{H}_1 : P_X \neq P_Y$$



Problem statement

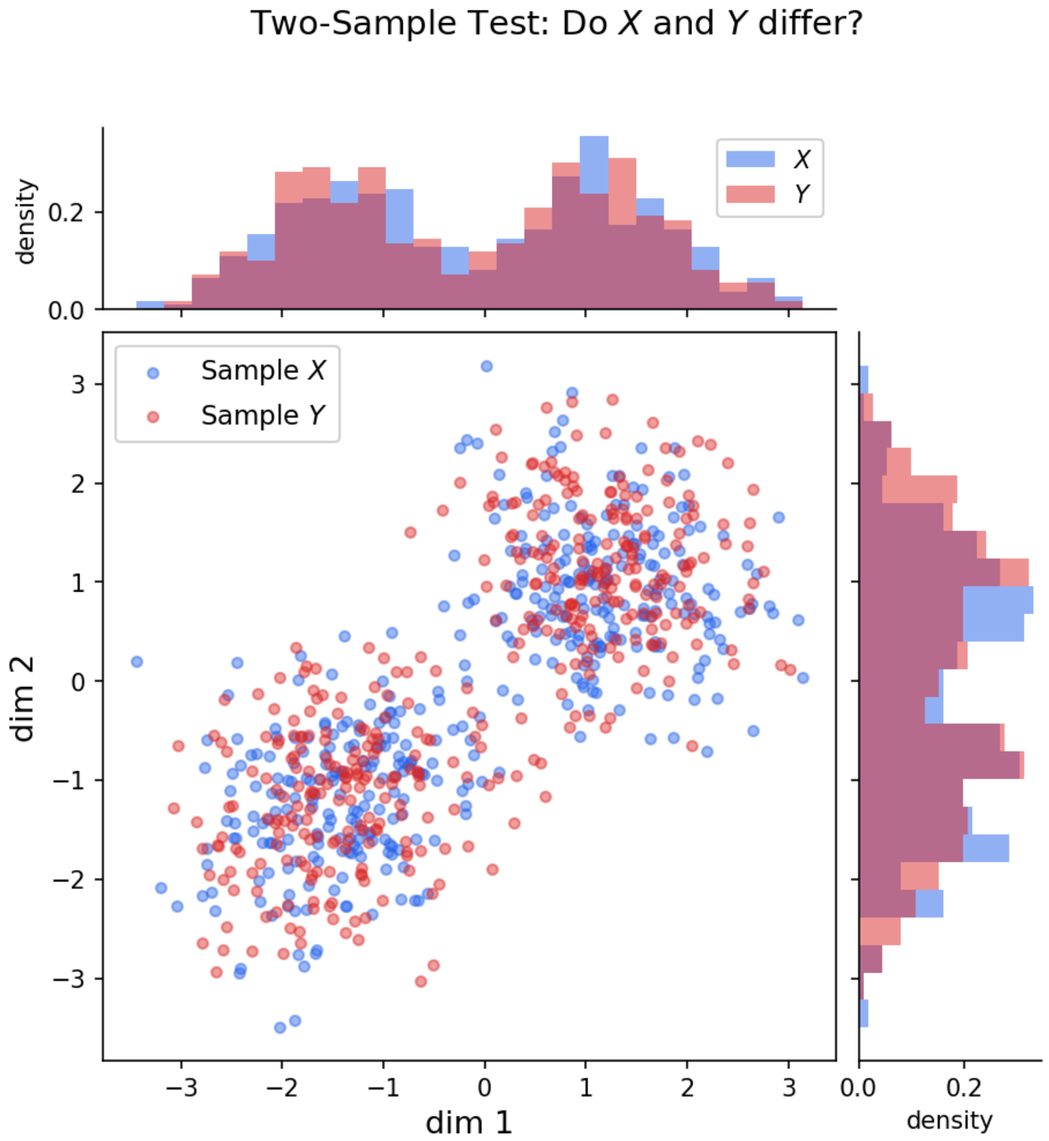
- If either P_X or P_Y is known in close form → goodness of fit test



Problem statement

- If either P_X or P_Y is known in close form $\rightarrow$ goodness of fit test
- If both P_X and P_Y are unknown $\rightarrow$ two-sample test

Real life



Two-sample test

$$t : (\mathbb{R}^{N_X \times d}, \mathbb{R}^{N_Y \times d}) \to \mathbb{R}$$
$$(X, Y) \to t(X, Y)$$

t is a similarity metric
between X and Y

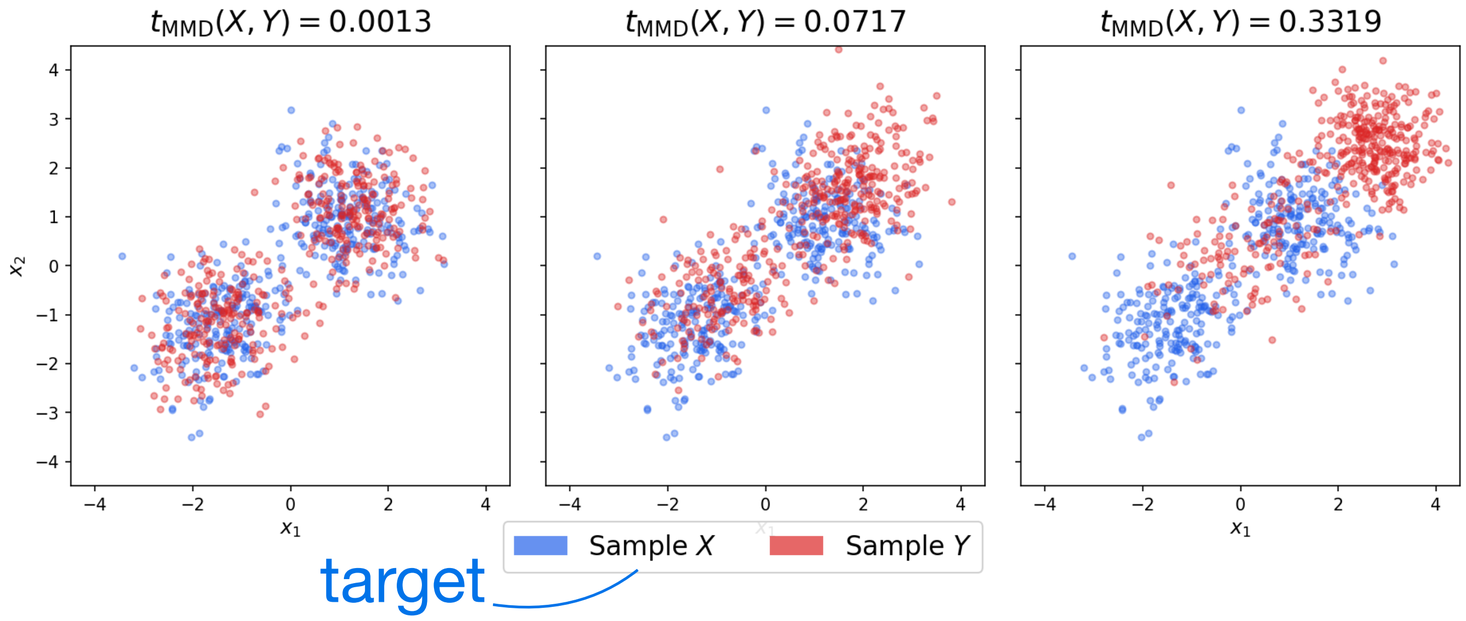
Two-sample test

$$t : (\mathbb{R}^{N_X \times d}, \mathbb{R}^{N_Y \times d}) \to \mathbb{R}$$
$$(X, Y) \to t(X, Y)$$

t is a similarity metric between X and Y

Can be used to **compare**

Which sample Y is more compatible with X?

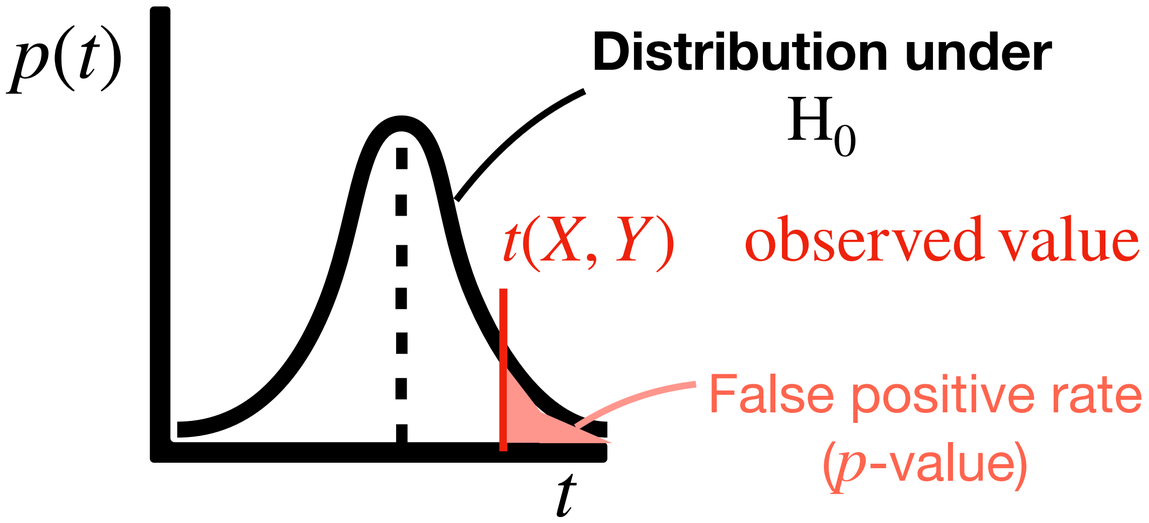


Two-sample test

$$t : (\mathbb{R}^{N_X \times d}, \mathbb{R}^{N_Y \times d}) \to \mathbb{R}$$
$$(X, Y) \to t(X, Y)$$

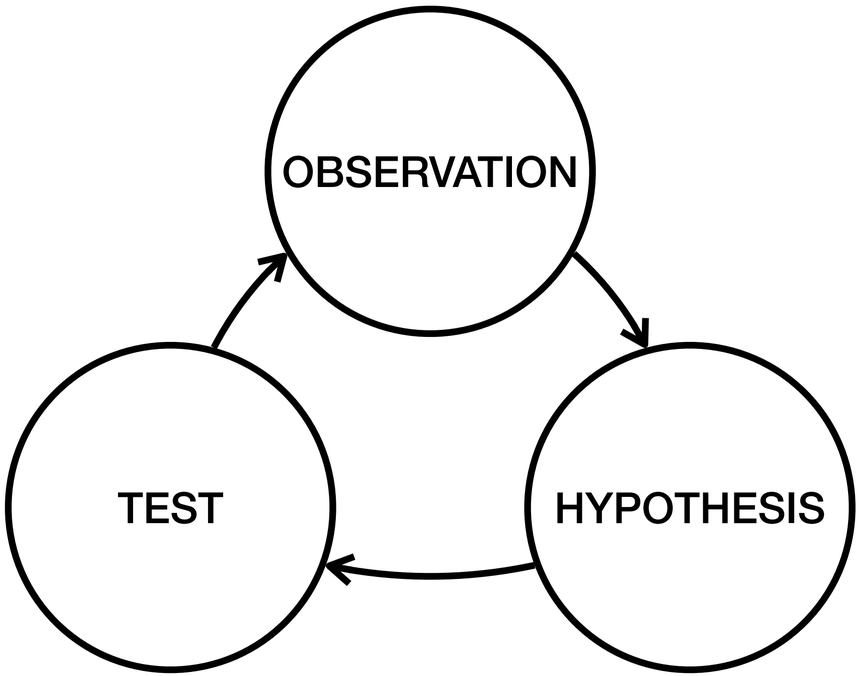
t is a similarity metric between X and Y

To **asses significance** we compute the p-value (aka calibrate)



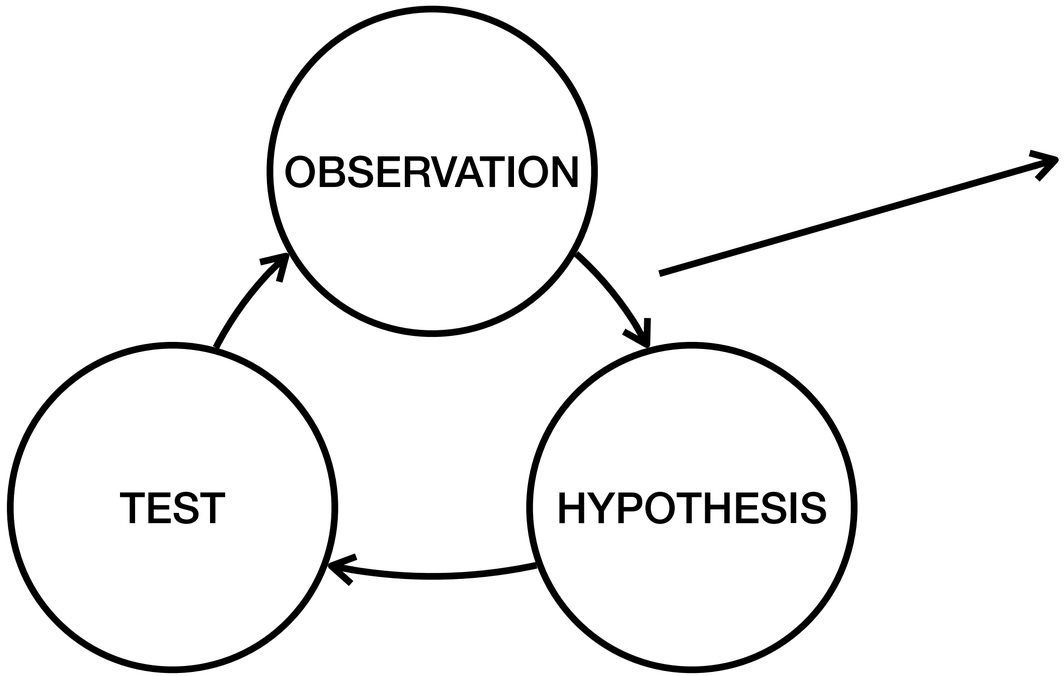
Learning begins when expectations fail

Scientific method



Learning begins when expectations fail

Scientific method

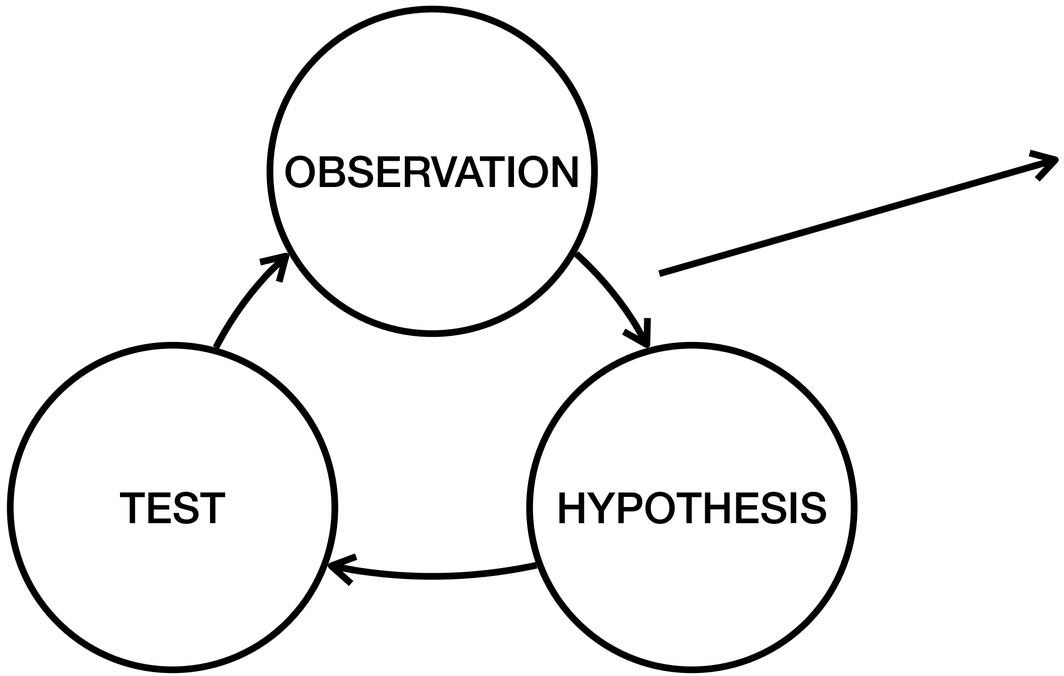


There is something in the observed data that my current **model of the world (P_X)** cannot explain

(Two-sample test)

Learning begins when expectations fail

Scientific method

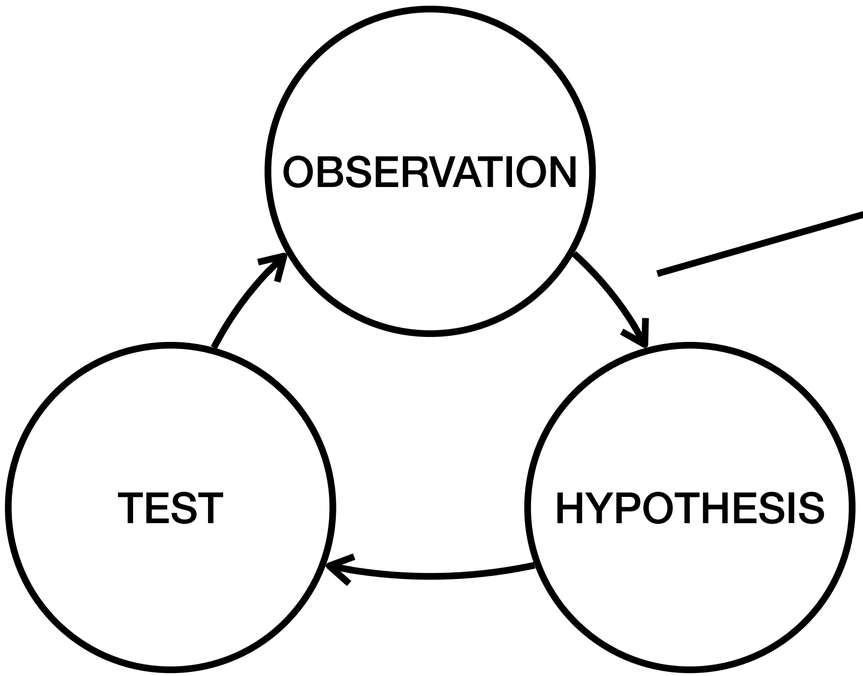


Monitoring

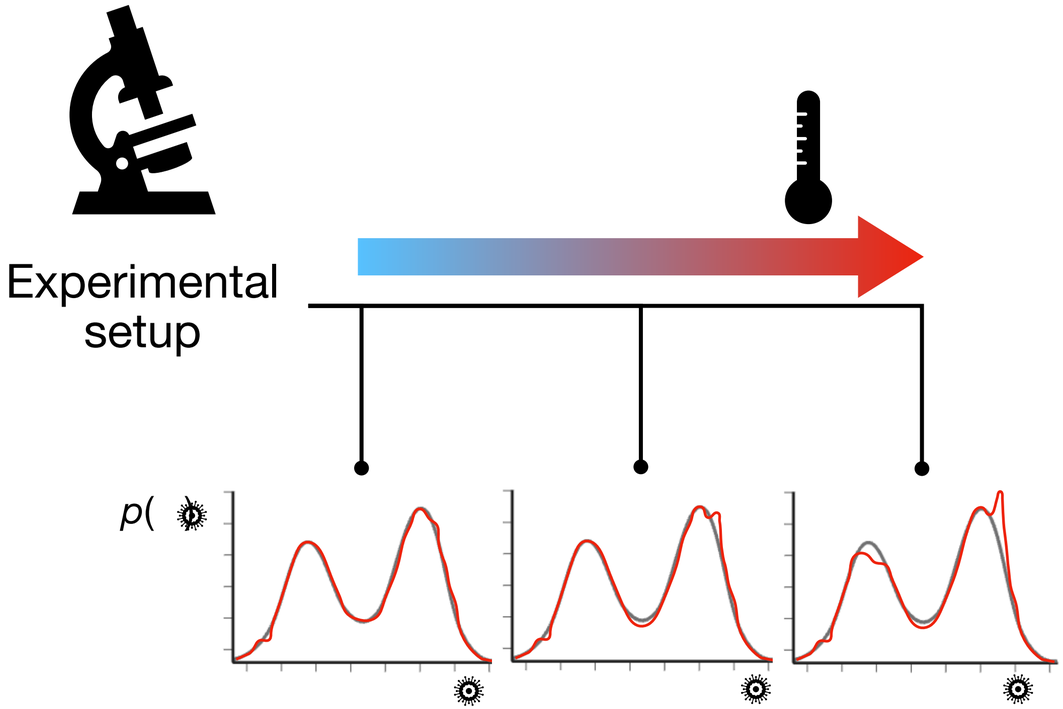
There is something in the observed data that my current **understanding of the experimental apparatus** (P_X) cannot explain

Learning begins when expectations fail

Scientific method

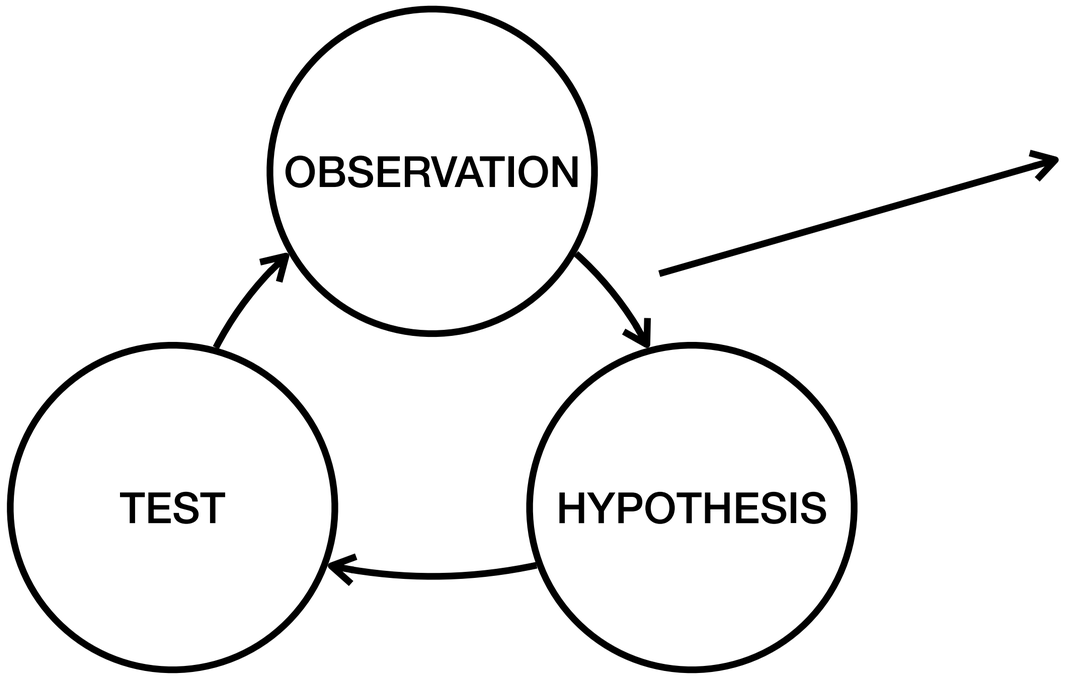


Monitoring



Learning begins when expectations fail

Scientific method

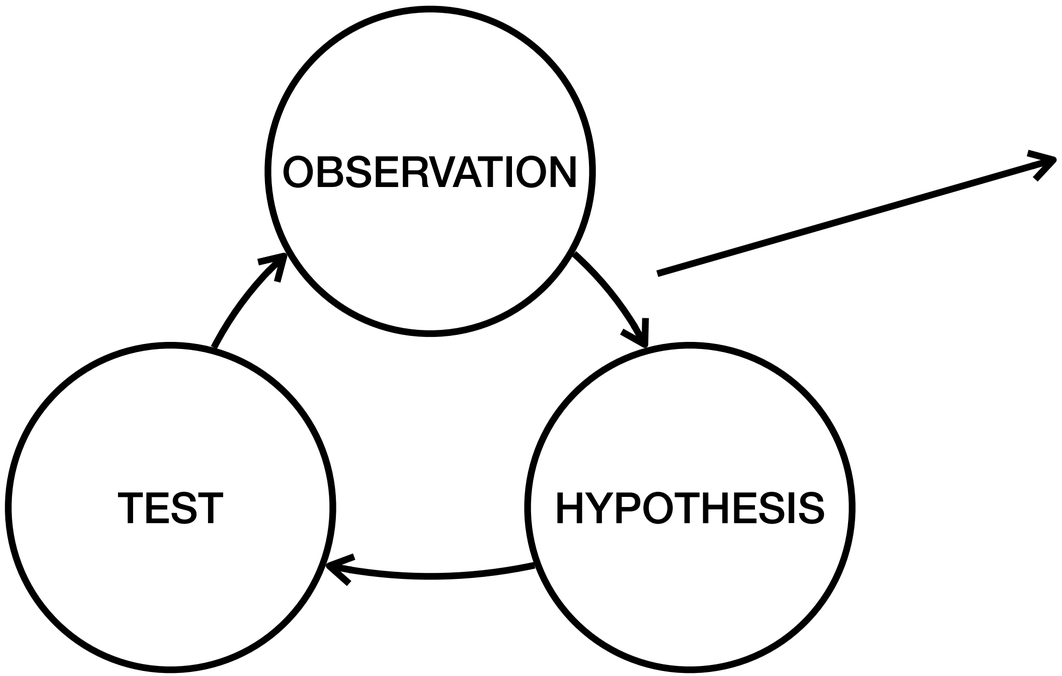


Validation

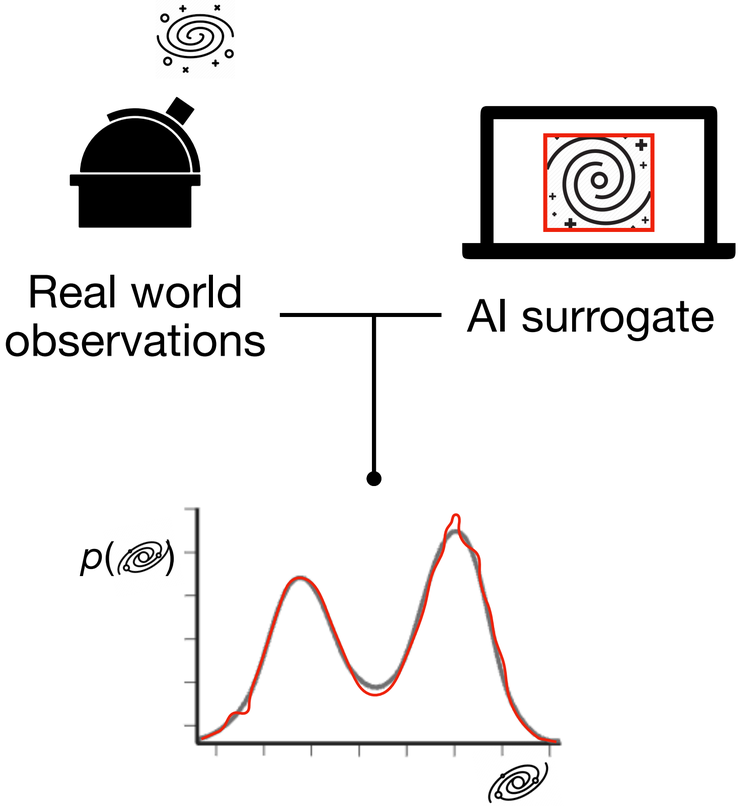
There is something in the observed data that my **data simulation** (P_X) cannot explain

Learning begins when expectations fail

Scientific method

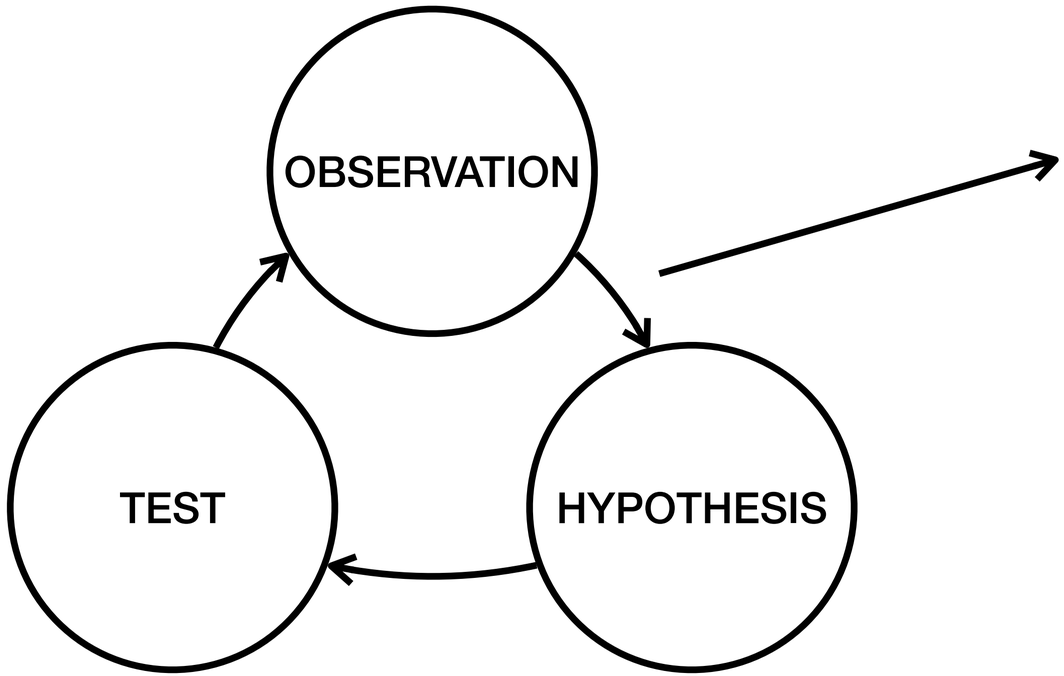


Validation



Learning begins when expectations fail

Scientific method

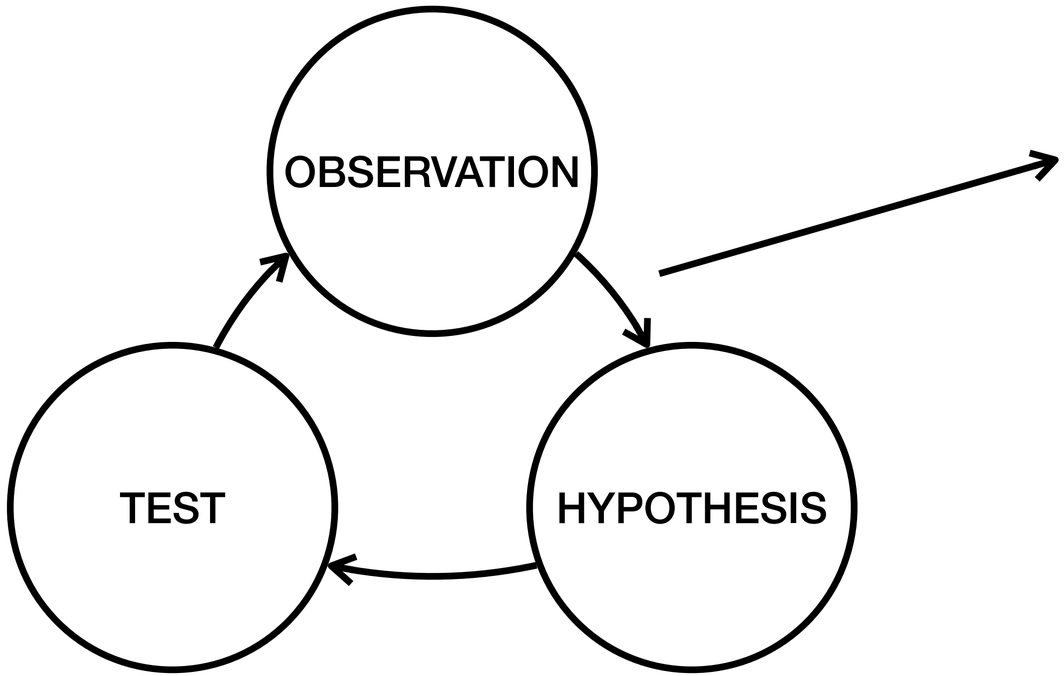


Discovery

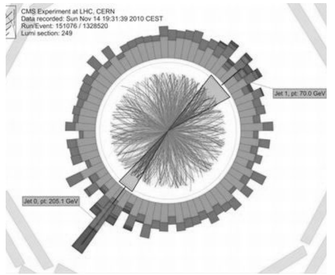
There is something in the observed data that my **current scientific understanding of Nature** (P_X) cannot explain

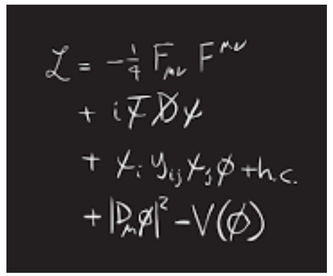
Learning begins when expectations fail

Scientific method



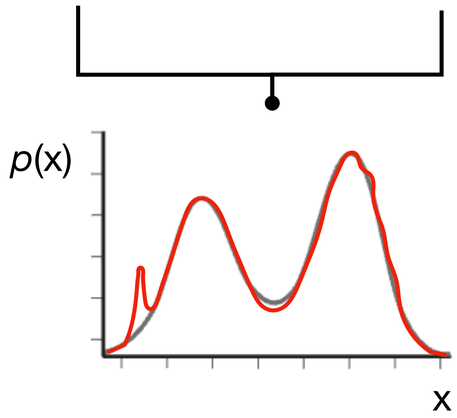
Discovery





Experiment

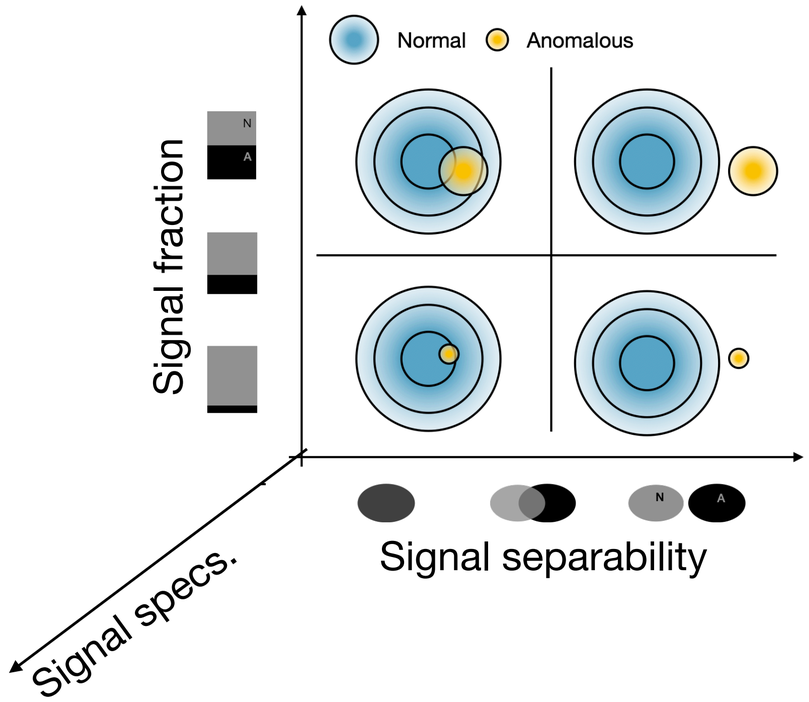
Theoretical model



What makes two-sample test hard

Intrinsic challenges:

- Signal specification: lack of a priori knowledge about the “signal” and its properties
- Signal frequency: how rare is the signal
- Signal separability: How discriminant is the given data representation



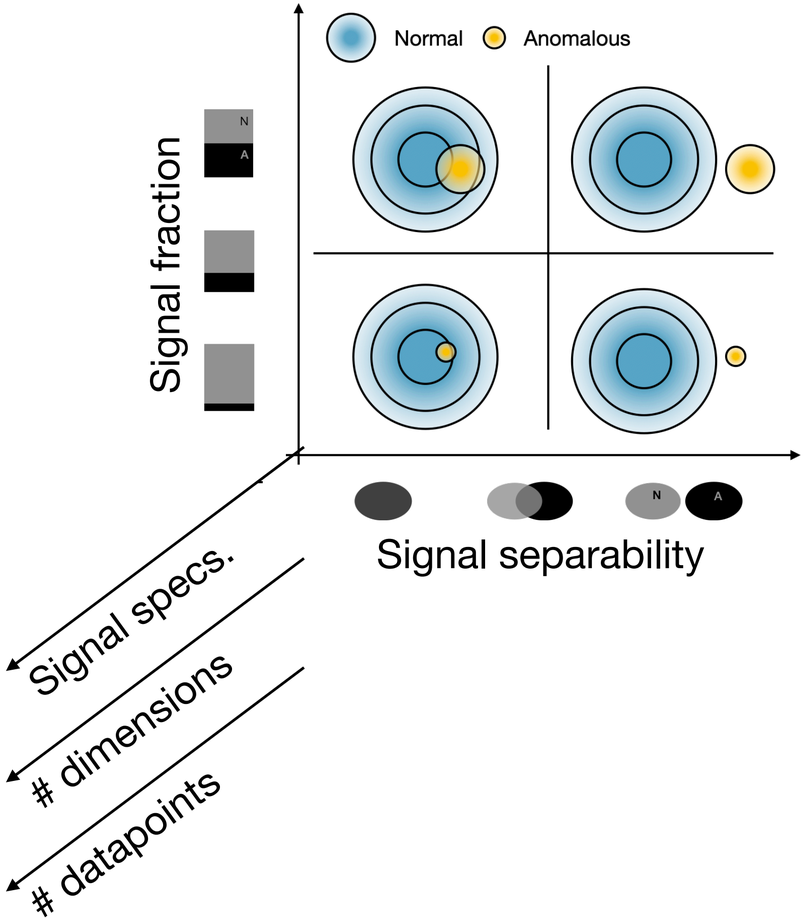
What makes two-sample test hard

Intrinsic challenges:

- Signal specification: lack of a priori knowledge about the “signal” and its properties
- Signal frequency: how rare is the signal
- Signal separability: How discriminant is the given data representation

Technical/engineering challenges:

- Size of the dataset: handle large samples is computationally hard
- Data representation: what is the data structure, what is its dimensionality



An evolving field

- Harder scientific problems
- More complex and larger datasets
- More advanced and efficient computational tools
- From 1D to nD tests
- From classical to ML based

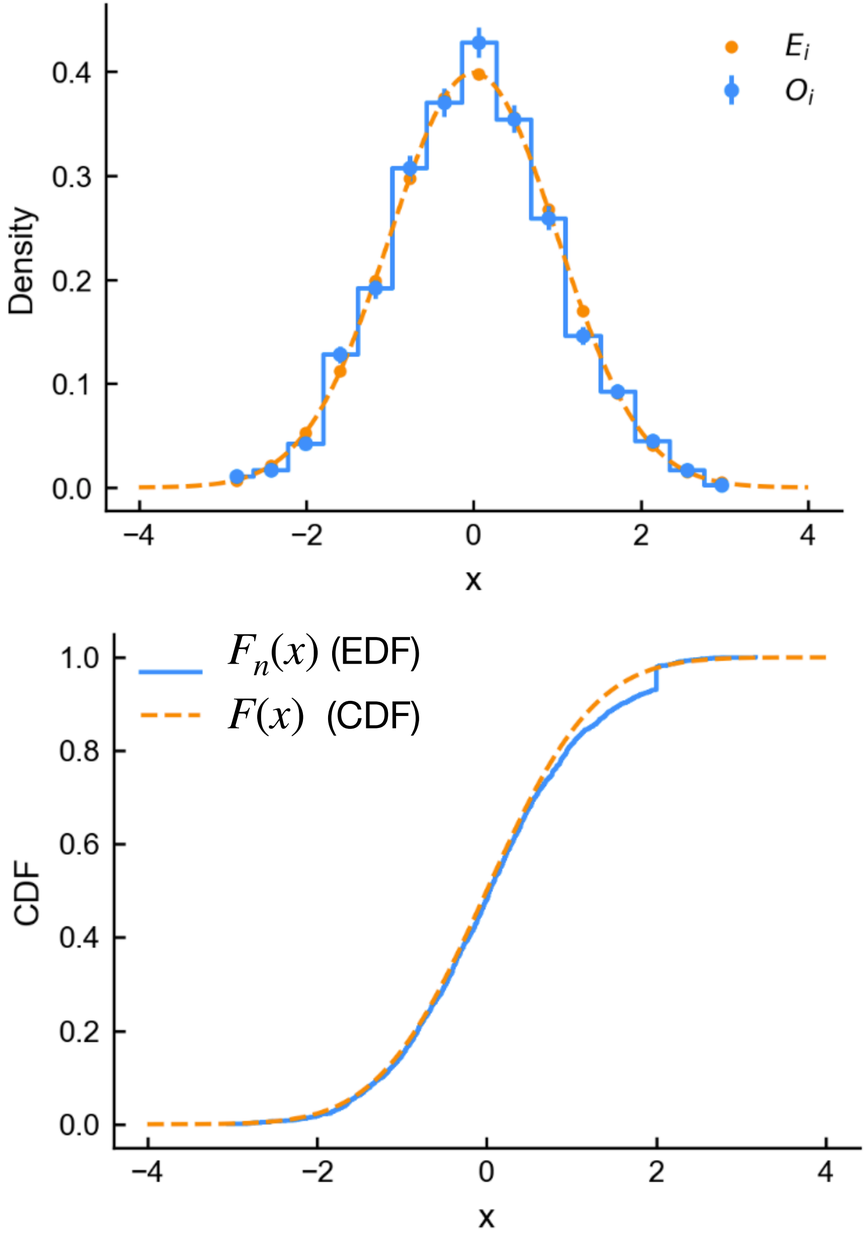
1D methods (a glance)

ECDF based tests
- Kolmogorov-Smirnov
- Cramer-von-Mises
- Anderson-Darling
- …

Likelihood-ratio based tests
- Saturated test
- χ^2 (bin dependent)
- …

Spacing statistics
- Moran
- …

Energy based
- Wasserstein distance
- …



1D methods (a glance)

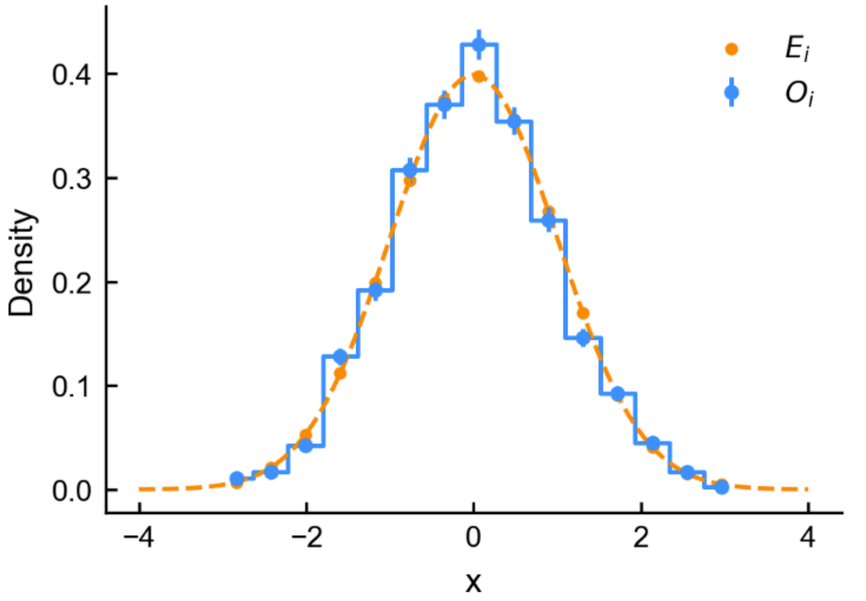
ECDF based tests
- Kolmogorov-Smirnov
- Cramer-von-Mises
- Anderson-Darling
- …

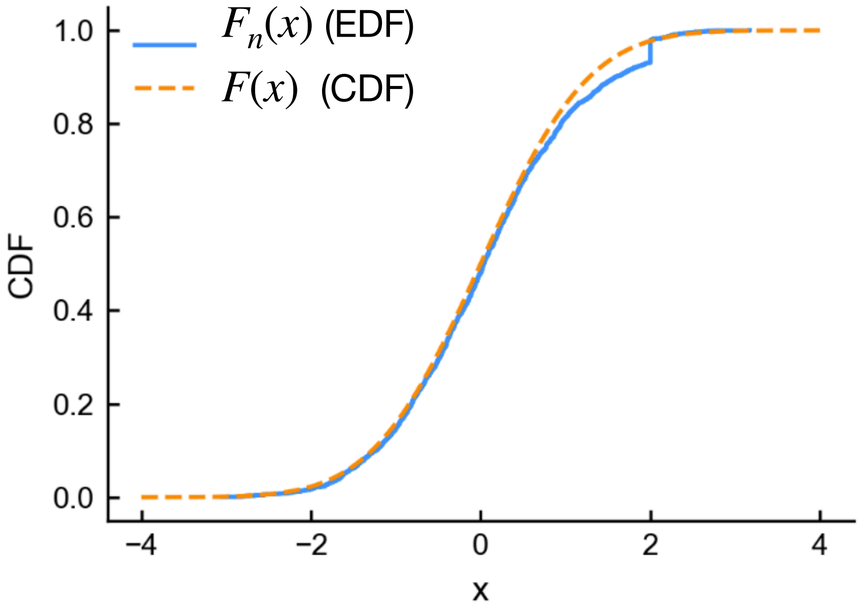
Likelihood-ratio based tests
- Saturated test
- χ^2 (bin dependent)
- …

Spacing statistics
- Moran
- …

Energy based
- Wasserstein distance
- …

Don’t scale to high dimensions!





nD methods

Sliced tests

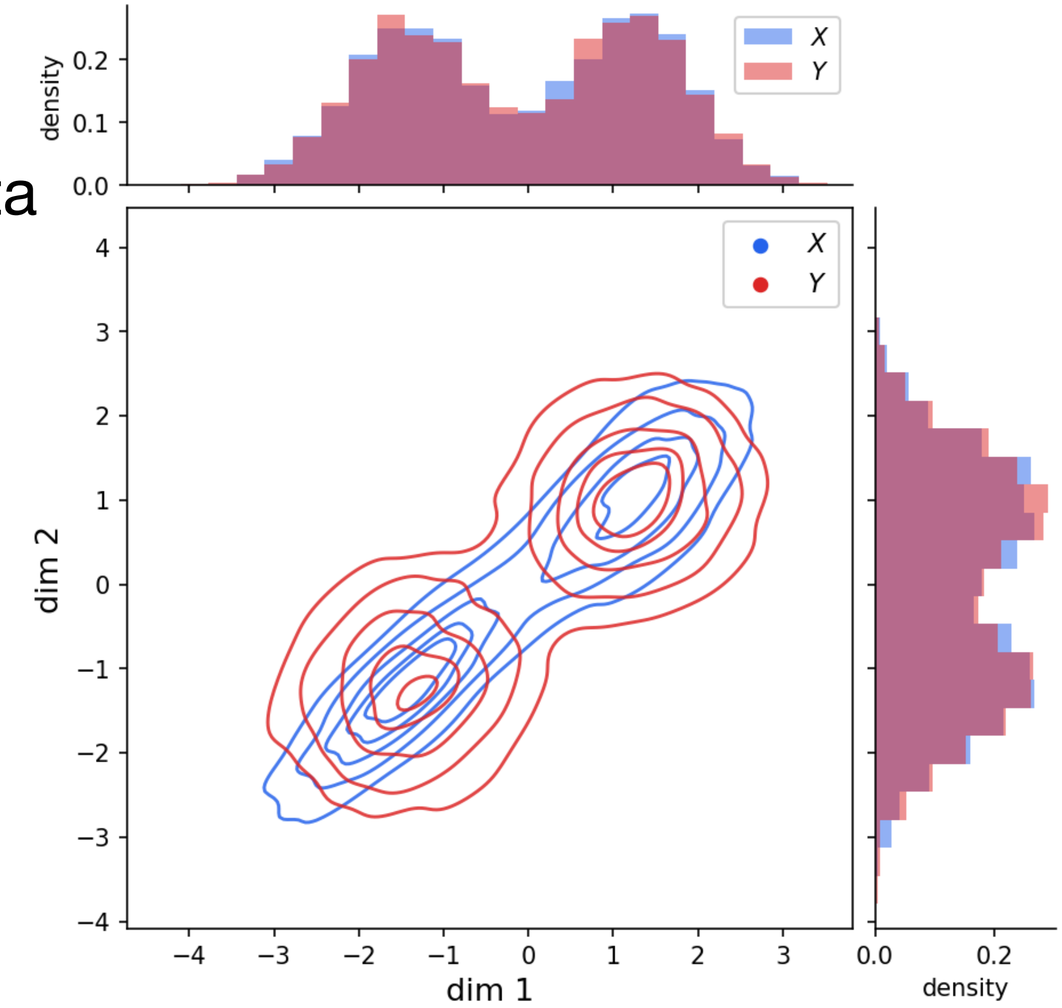
- Perform *n* random 1D projections of the data
- Compute a 1D test over each projection
- Average over the projections

Examples:

- Sliced Wasserstein distance
- Sliced Kolmogorov-Smirnov

In practice: very fast method

Caveat: **Correlations**: whether a sliced test detects them depends entirely on whether the right 1D projections are sampled!



nD methods

Maximum Mean Discrepancy (MMD)

Maximum Mean Discrepancy (MMD) is a **kernel-based** metric used to measure the distance between two probability distributions P and Q over a space $\mathcal{X}$.
It works by:

1. Mapping the distributions into a **reproducing kernel Hilbert space** ($\mathcal{H}$):

$$\phi : \mathcal{X} \to \mathcal{H}$$

2. Calculating their **means** in that space:

$$\mu_{P_X} = \mathbb{E}_{x \sim P_X}[\,\phi(x)], \quad \mu_{P_Y} = \mathbb{E}_{y \sim P_Y}[\,\phi(y)]$$

3. Taking their **difference** between means:

$$\mathrm{MMD}^2(P_X, P_Y; \mathcal{H}) = \left\| \mu_{P_X} - \mu_{P_Y} \right\|_{\mathcal{H}}^2$$

Gretton et al. (Journal of Machine Learning Research, 2012)

nD methods

Maximum Mean Discrepancy (MMD)

Using the kernel trick with kernel function $k(x, x') = \langle \phi(x), \phi(x') \rangle$, the squared MMD can be written as

$$\mathrm{MMD}^2(P_X, P_Y) = \mathbb{E}_{x,x' \sim P_X}[\, k(x, x') \,] + \mathbb{E}_{y,y' \sim P_Y}[\, k(y, y') \,] - 2\, \mathbb{E}_{x \sim P_X,\, y \sim P_Y}[\, k(x, y) \,]$$

Given **finite samples** $\{x_i\}_{i=1}^n \sim P_X$ and $\{y_j\}_{j=1}^m \sim P_Y$, an unbiased estimator is

$$\widehat{\mathrm{MMD}^2} = \frac{1}{n(n-1)} \sum_{i \neq j} k(x_i, x_j) + \frac{1}{m(m-1)} \sum_{i \neq j} k(y_i, y_j) - \frac{2}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} k(x_i, y_j)\,.$$

nD methods

Maximum Mean Discrepancy (MMD)

Strengths: If the kernel is **characteristic**, the mean embedding captures **all moments and all interactions** of the distribution.

Characteristic kernel: positive definite kernel for which the function $P \rightarrow \mu_P = \mathbb{E}_{x\sim P}[\,\phi(x)]$ is *injective* (examples: Gaussian kernels, Laplacian kernels).

Caveat: when working with **finite size samples**, $\widehat{\mathrm{MMD}}^2$ provides an approximation with a **variance** that **increases with dimensionality**. Small distributional discrepancies may be drowned in noise.

In practice: MMD is widely used across scientific domains. It works well with few dimensions.

There is a vast literature on how to improve the estimator or make it more efficient. A few selected works:

Schrab et al. (JMLR, 2023) use multiple scale assumption to mitigate model selection bias

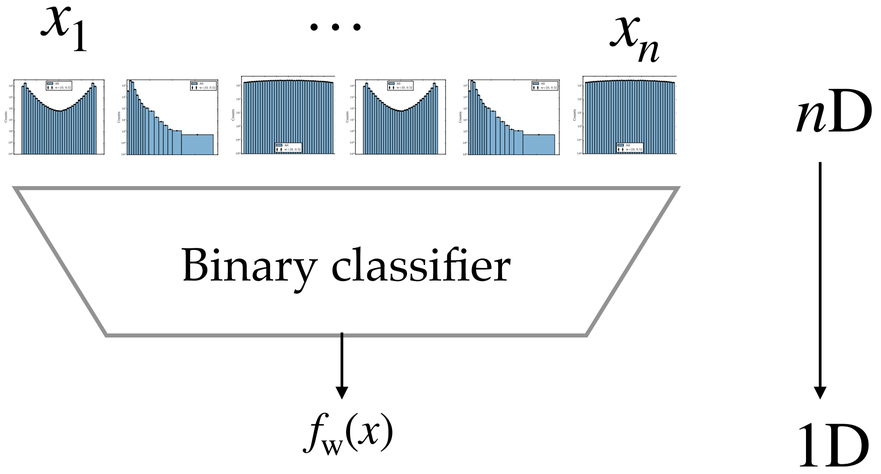
Liu et al. (PMLR, 2020) use neural networks to parametrize the kernel

Chatalic et al. (e-print, 2025) use Nyström approximation for applications at scale (e.g. large sample size)

nD methods

Classifier 2-sample test (C2ST)

- Train a classifier to tell apart two samples, X and Y. [Binary Cross Entropy or similar]

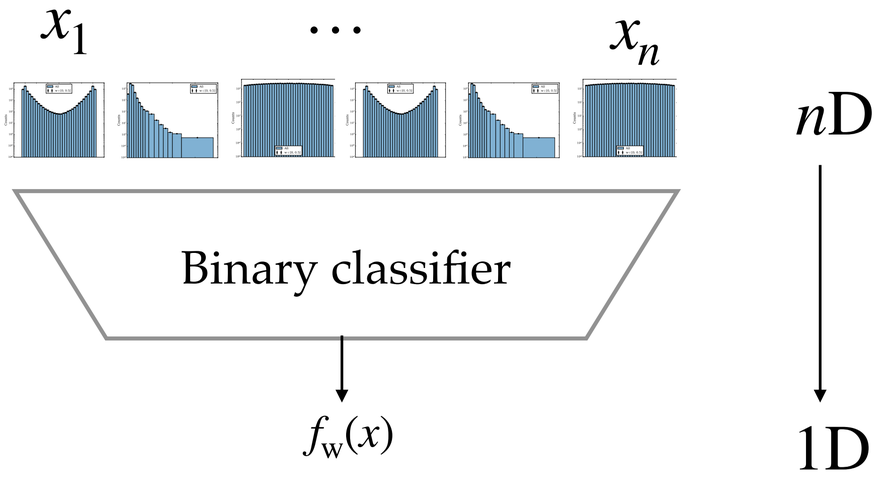


nD methods

Classifier 2-sample test (C2ST)

- Train a classifier to tell apart two samples, X and Y. [Binary Cross Entropy or similar]

- 1D GoF/two-sample test on $f_{\mathrm{w}}(x)$:
  - Classical tests[1]: KS, AD, χ^2, etc.
  - Classification metrics[2]: ACC, AUC, MCE



[1] Friedman (2003)
[2] Charkavarti et al. (2021), Lopez et al. (2017)

nD methods

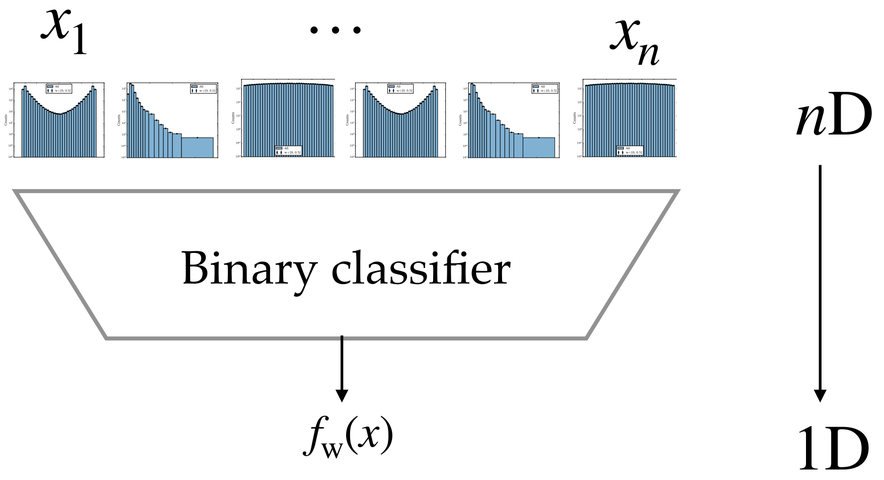
Classifier 2-sample test (C2ST)

- Train a classifier to tell apart two samples, X and Y. [Binary Cross Entropy or similar]

- Likelihood-ratio-test[3]:

$$f_{\mathrm{w}}(x) = \log\left[\frac{n(x\,|\,\mathrm{H_w})}{n(x\,|\,\mathrm{H_0})}\right] \approx \log\left[\frac{n(x\,|\,\mathrm{True})}{n(x\,|\,\mathrm{H_0})}\right]$$

$$t(D) = 2\sum_{x\in D}\left(n(x\,|\,\mathrm{H_0}) - n(x\,|\,\mathrm{H_w}) + \log\frac{n(x\,|\,\mathrm{H_w})}{n(x\,|\,\mathrm{H_0})}\right)$$



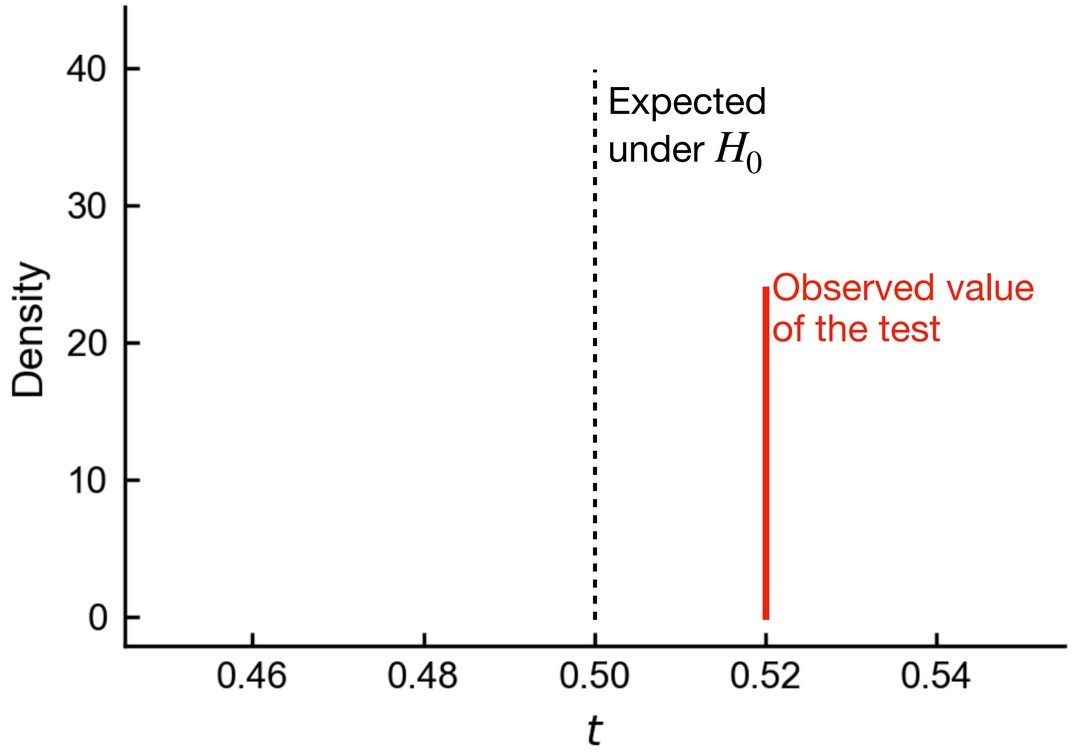
[1] Friedman (2003)
[2] Charkavarti et al. (2021), Lopez et al. (2017)
[3] d'Agnolo, Wulzer (2018), d'Agnolo, Grosso et al. (2021)

CAVEATS

CAVEAT: calibration

- The single value of a test statistic is not sufficient to assess the test ***significance***

Example: Accuracy test

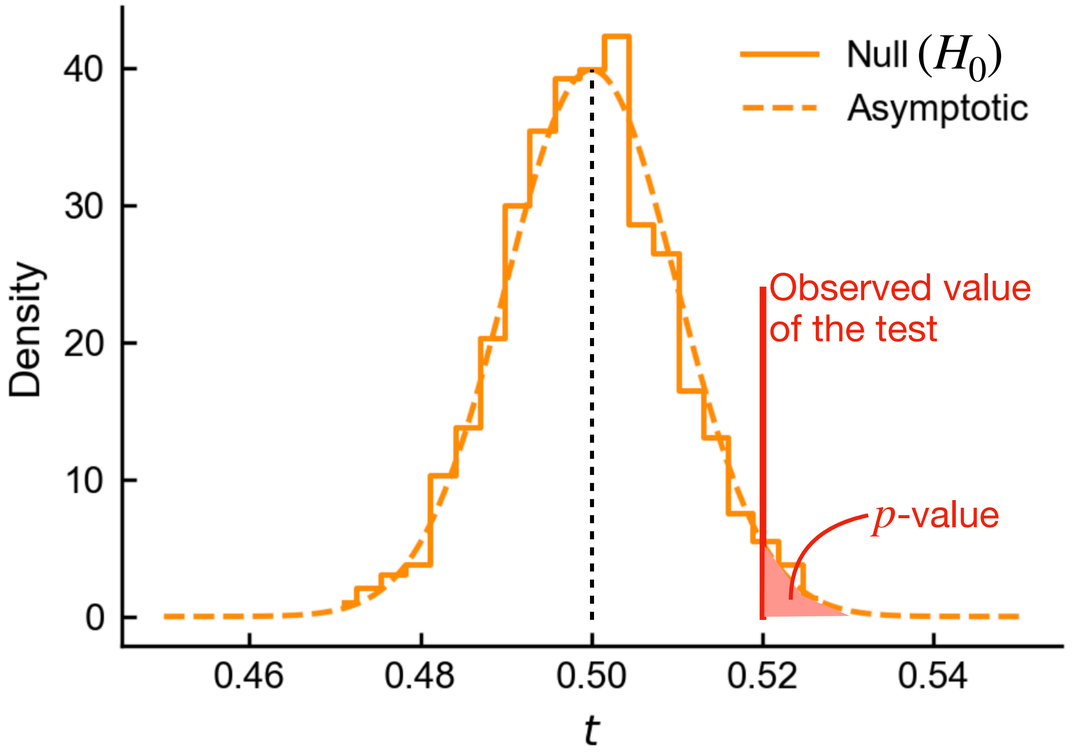


CAVEAT: calibration

- The single value of a test statistic is not sufficient to assess the test ***significance***

- p***-value:*** *probability of obtaining an outcome of* t *under the null as extreme or more than the observed one*

Example: Accuracy test

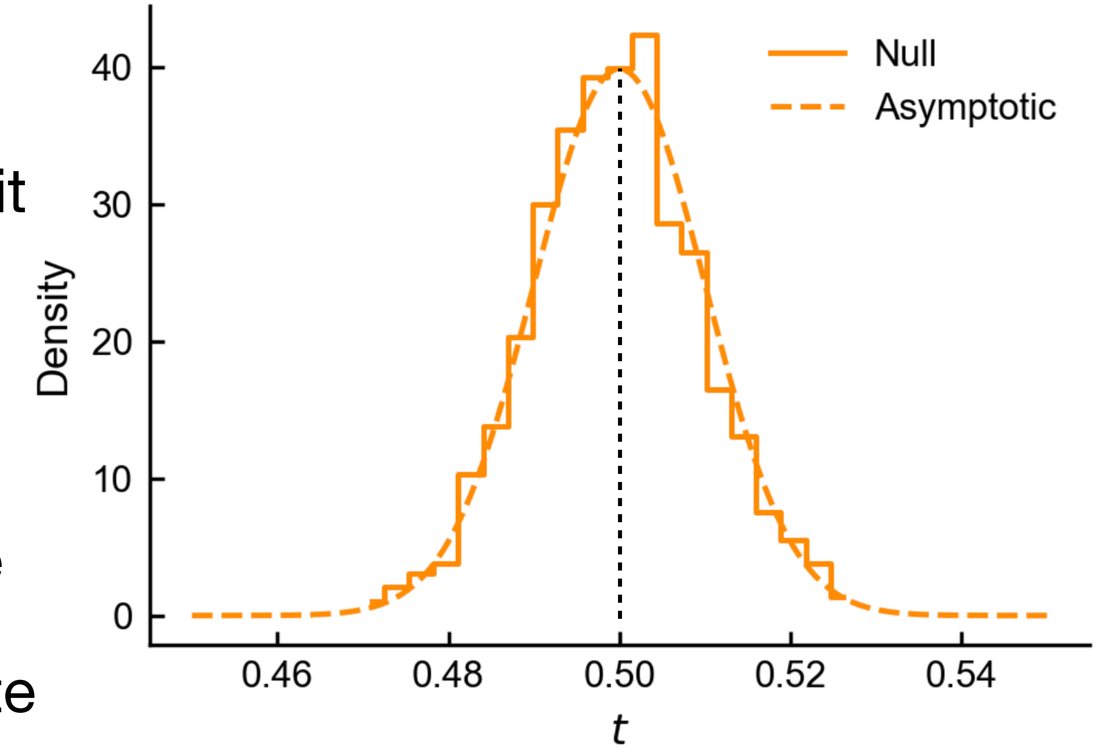


CAVEAT: calibration

How to build the test distribution under the null:

- **Asymptotic formula:** analytic statistical model of the distribution is known from theory (note: if exists it's valid only in the limit of infinite statistics →it must be validated)
- **Bootstrap:** for *n* times: (1) sample from the existing datasets a subset of events; (2) retrain and evaluate.
- **Permutation:** for *n* times: (1) random shuffle of the labels to build two dataset with the same generative model; (2) retrain/recompute and evaluate.

Example: Accuracy test



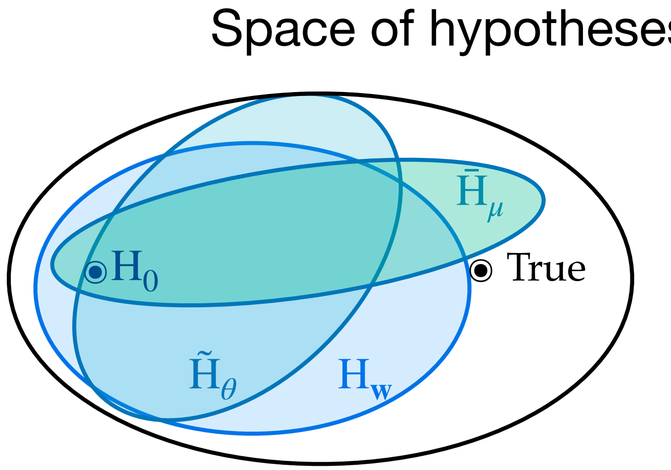
See Charkavarti et al. (2021) for comparison between methods in the context of classifier-based test

CAVEAT: no free lunch

There is no universally optimal statistical test; sensitivity depends on the *assumptions* and *targeted signal*.

Test design matters:

- Choice of test statistic (binning, nearest neighbors, mass window, ML score, …) focuses power on specific families of deviations.
- A test used blindly may fail in ways we don’t notice. Good practice is to identify **what the test is sensitive to and what it will inevitably miss**.



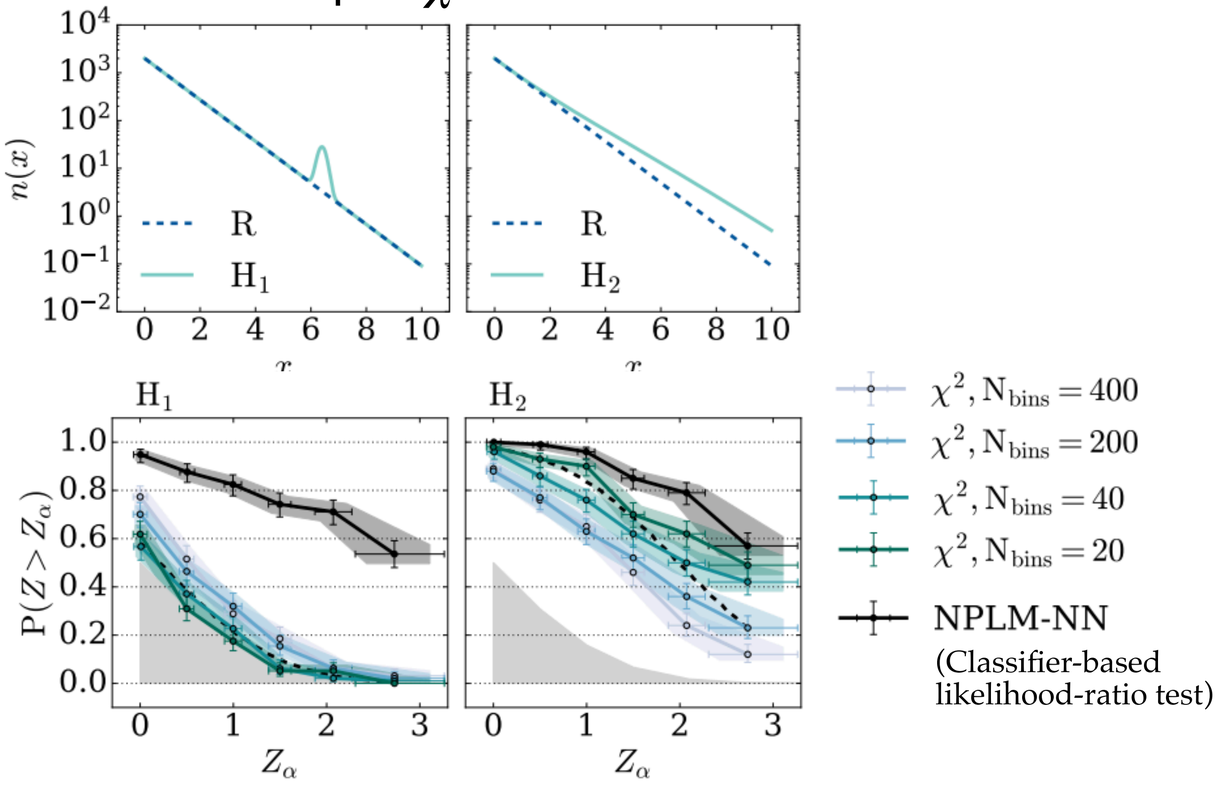
CAVEAT: no free lunch

There is no universally optimal statistical test; sensitivity depends on the *assumptions* and *targeted signal*.

Test design matters:

- Choice of test statistic (binning, nearest neighbors, mass window, ML score, …) focuses power on specific families of deviations.
- A test used blindly may fail in ways we don't notice. Good practice is to identify **what the test is sensitive to and what it will inevitably miss**.

1D example: χ^2 test[1]



[1] Grosso et al. SciPost Phys. 16, 123 (2024)

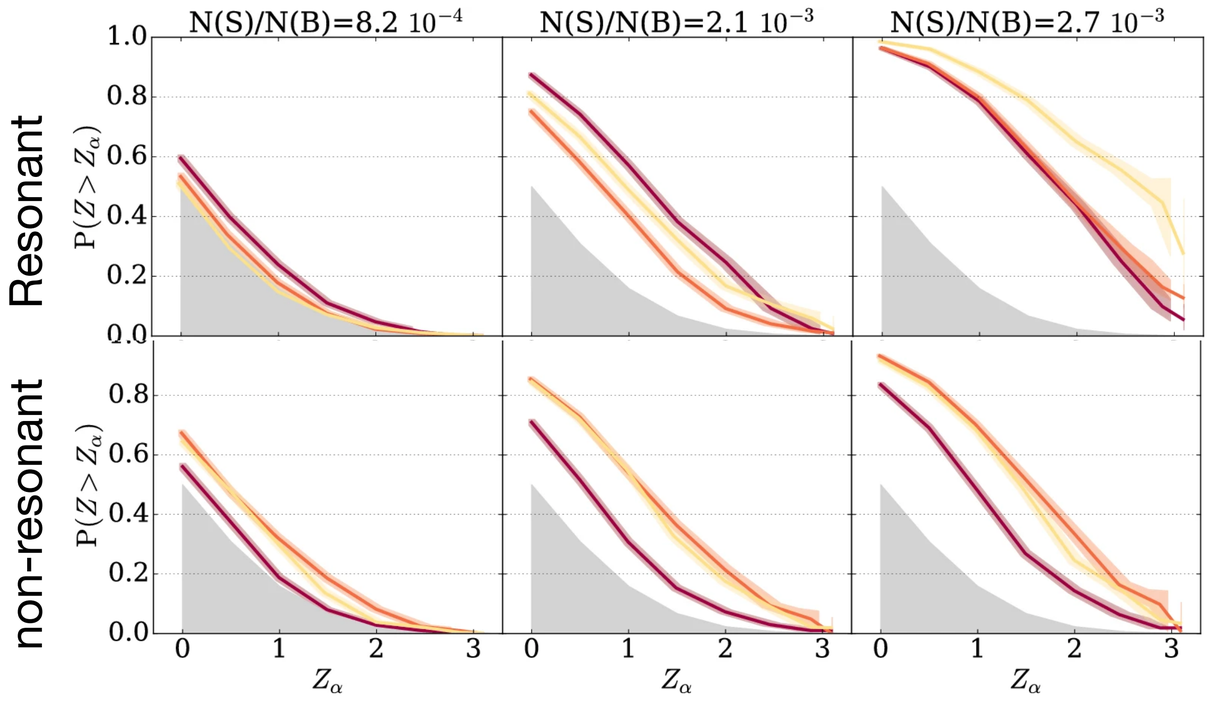
CAVEAT: no free lunch

There is no universally optimal statistical test; sensitivity depends on the *assumptions* and *targeted signal*.

Test design matters:

- Choice of test statistic (binning, nearest neighbors, mass window, ML score, …) focuses power on specific families of deviations.
- A test used blindly may fail in ways we don't notice. Good practice is to identify **what the test is sensitive to and what it will inevitably miss**.

5D example: Classifier based test[2]



[2] Grosso et al. Eur. Phys. J. C 85, 1074 (2025)

CAVEAT: no free lunch

Example: Ranking generative models in Computer Vision

Rethinking FID: Towards a Better Evaluation Metric for Image Generation

Sadeep Jayasumana Srikumar Ramalingam Andreas Veit Daniel Glasner
Ayan Chakrabarti Sanjiv Kumar
Google Research, New York

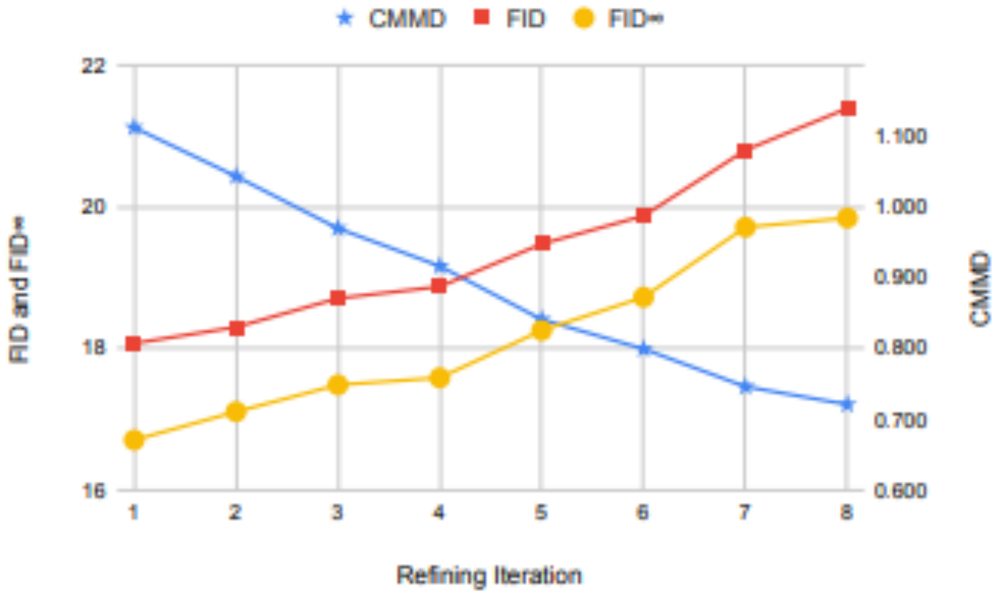
FID (Frechet Inception Distance) has opposite trend than expected!

FID measures location discrepancy, it is not a good proxy of the overall images quality

$$\mathrm{dist}_F^2(P,Q) = \|\boldsymbol{\mu}_P - \boldsymbol{\mu}_Q\|_2^2 + \mathrm{Tr}(\boldsymbol{\Sigma}_P + \boldsymbol{\Sigma}_Q - 2(\boldsymbol{\Sigma}_P\boldsymbol{\Sigma}_Q)^{\frac{1}{2}})$$

https://arxiv.org/pdf/2401.09603

Figure 3. *The quality of the generated image monotonically improves as we progress through Muse's refinement iterations. CMMD correctly identifies the improvements. FID, however, incorrectly indicates a quality degradation (see Figure 4). Prompt: "The Parthenon".*



CAVEAT: no free lunch

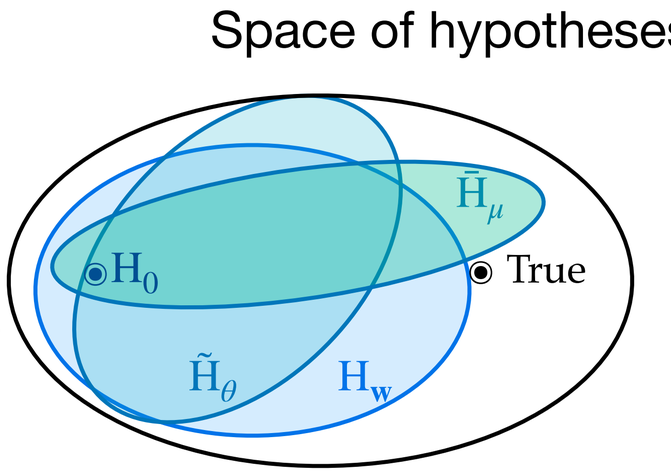
There is no universally optimal statistical test; sensitivity depends on the *assumptions* and *targeted signal*.

Trade-offs:

- Narrowly targeted tests → high sensitivity to specific signals, low robustness to unexpected deviations.
- Broad/unspecific tests → more robust to unknown signals, but diluted sensitivity to particular signatures

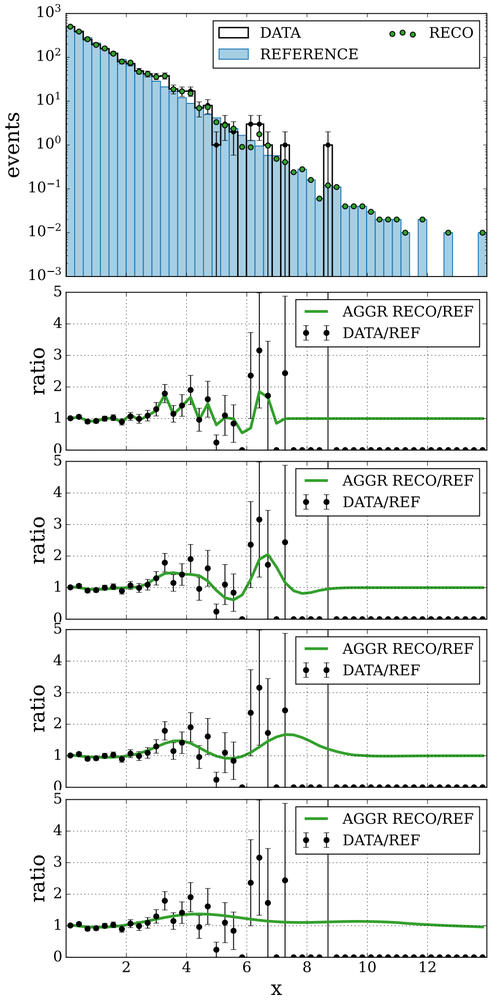
In ML:

- *Model selection*: hyper-parameters/architecture choice poses **hard inductive bias**.
- *Regularization* is a powerful tool to impose **soft inductive bias** on the learning dynamics



CAVEAT: no free lunch

How multiple testing help mitigate this issue



Kernel width

$\sigma = 0.1$

$\sigma = 0.7$

$\sigma = 1.4$

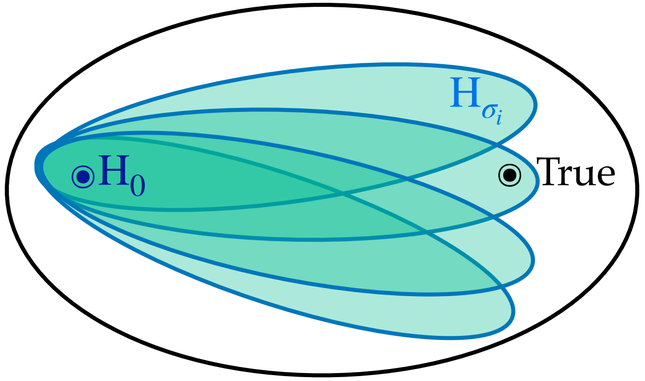
$\sigma = 3.0$

Kernel-based model

$$f_w(x) = \sum_{i=1}^{M} w_i k_\sigma(x, \tilde{x}_i) \approx \log \frac{n(x \mid D)}{n(x \mid \mathrm{H}_0)}$$

$$k_\sigma(x, x') = e^{-\|x - x'\|^2 / 2\sigma^2}$$

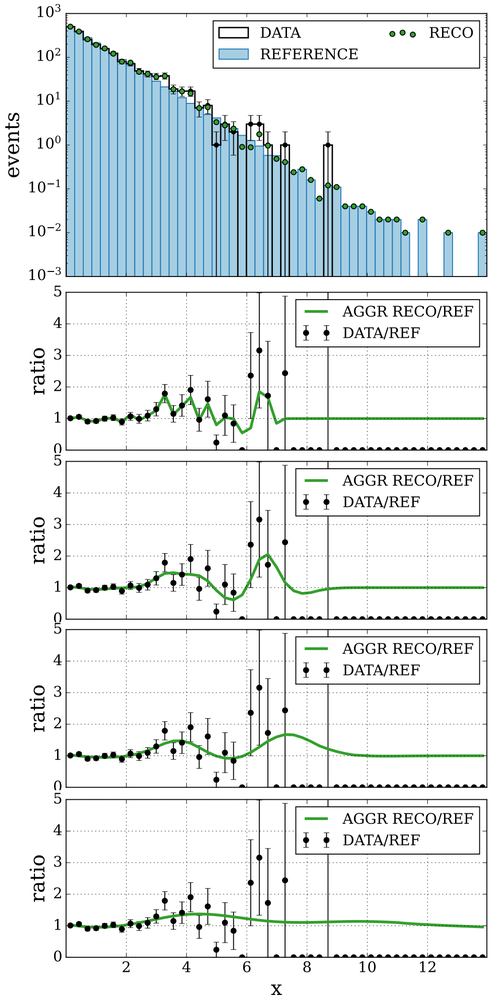
Multiple testing over model hyper-parameters



Grosso and Letizia, Eur. Phys. J. C 85, 4 (2025)

CAVEAT: no free lunch

How multiple testing help mitigate this issue



Kernel width | **Neyman Pearson test** | ***p*-value**

$\sigma = 0.1 \rightarrow t_\sigma(D,R) \rightarrow p_\sigma(D,R)$

$\sigma = 0.7 \rightarrow t_\sigma(D,R) \rightarrow p_\sigma(D,R)$

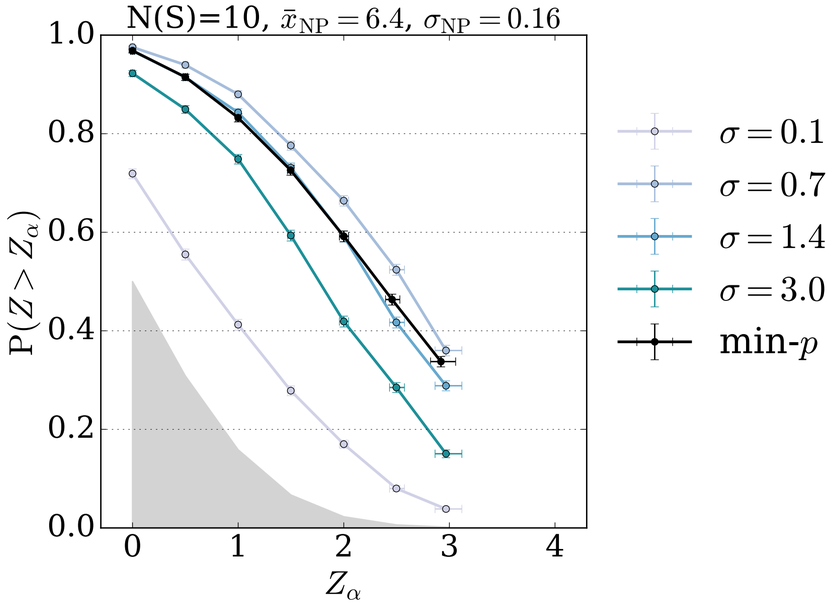
$\sigma = 1.4 \rightarrow t_\sigma(D,R) \rightarrow p_\sigma(D,R)$

$\sigma = 3.0 \rightarrow t_\sigma(D,R) \rightarrow p_\sigma(D,R)$

p-aggregation

$$p_{\min}(D,R) = \min_\sigma p_\sigma(D,R)$$

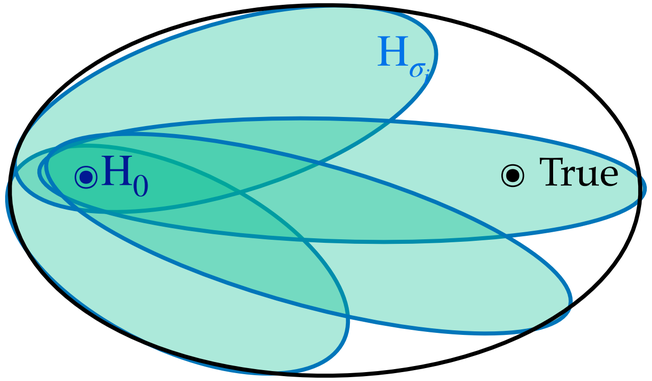
Power curve



Grosso and Letizia, Eur. Phys. J. C 85, 4 (2025)

CAVEAT: no free lunch

How multiple testing help mitigate this issue



Trial factor:

Looking at the data under different lenses increases the chance of discovery, *but also the chance of false positives*!

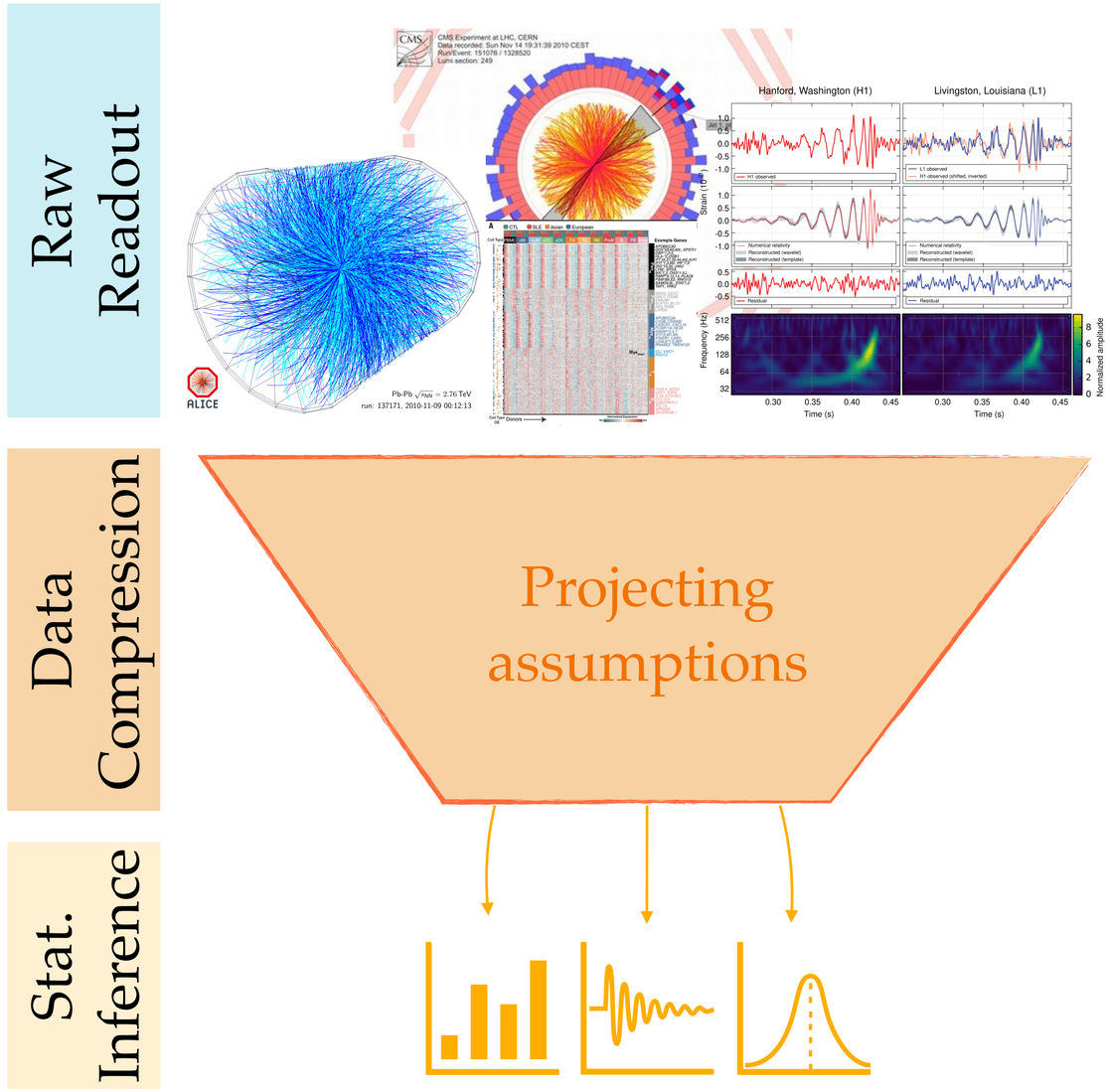
Challenge: define the smallest number of tests that, together, cover the space of hypotheses

Note: As any other test, multiple tests should be **calibrated**! Calibration accounts for the trials factor

Scientific Discovery at scale with two-sample test [a glance to my research]

Curse of dimensionality

- Raw experimental data are high dimensional and often highly structured
- Statistical analysis are only possible after a data compression step which reduces complexity by projecting data in a lower dimensional representation.
- One has to make assumptions about which information is relevant to the downstream task (domain knowledge, current understanding of the field).

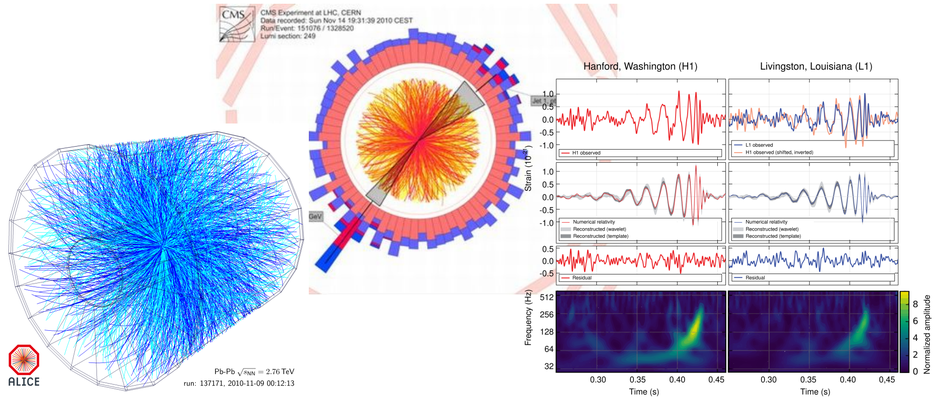


Learning representations

My research focus:

Learn a **better data representation** beforehand (hope for *sufficient statistics*)

Raw Readout



Data Compression

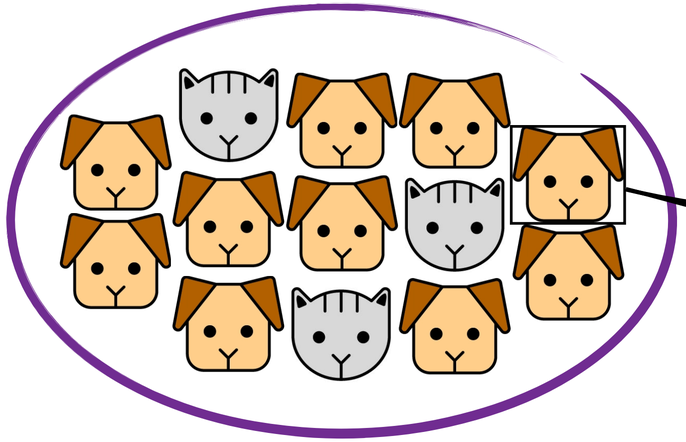
What is a good assumption in absence of a signal models?

Stat. Inference

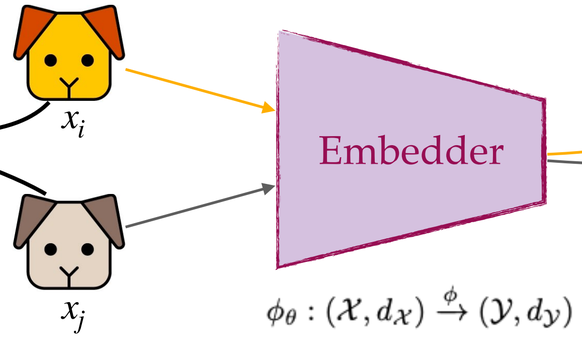
Anomaly Detection

Learn representations for signal discovery

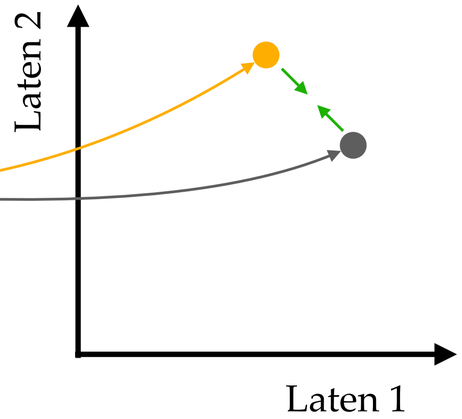
Contrastive Learning: Self-supervised



unlabelled data points



Generate two distorted *views* of the same example



Constraint the organization of the views in the embedding space:

- Views of the same object are pushed together
- Views of different objects are pulled apart

SimCLR loss [2002.05709]:

$$l_{\text{SimCLR}}(x_i, x_j) = -\log \frac{e^{\text{sim}(\phi_\theta(x_i), \phi_\theta(x_j))/\tau}}{\sum_{k=1}^{N} e^{\text{sim}(\phi_\theta(x_i), \phi_\theta(x_k))/\tau}}$$

Cosine similarity:

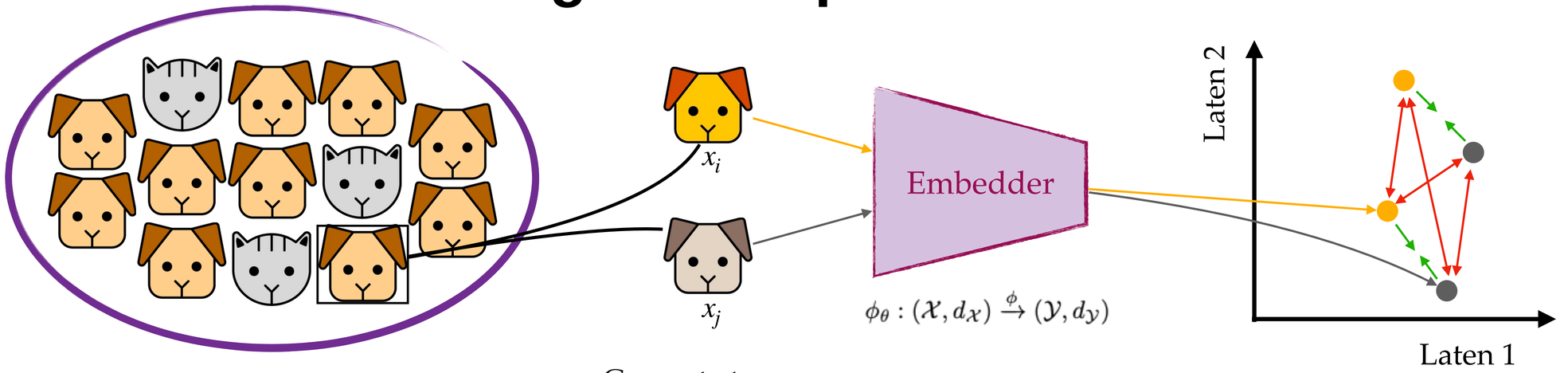
$\text{sim}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{u}^\top \boldsymbol{v} / \|\boldsymbol{u}\| \|\boldsymbol{v}\|$

Exploit **data augmentations** to induce meaningful forms of **implicit bias** in the data **organization**

Loose way to impose **invariance** to transformations

Learn representations for signal discovery

Contrastive Learning: Self-supervised



unlabelled data points

Generate two distorted *views* of the same example

Constraint the organization of the views in the embedding space:

- Views of the same object are pushed together
- Views of different objects are pulled apart

Exploit **data augmentations** to induce meaningful forms of **implicit bias** in the data **organization**

Loose way to impose **invariance** to transformations

SimCLR loss [2002.05709]:

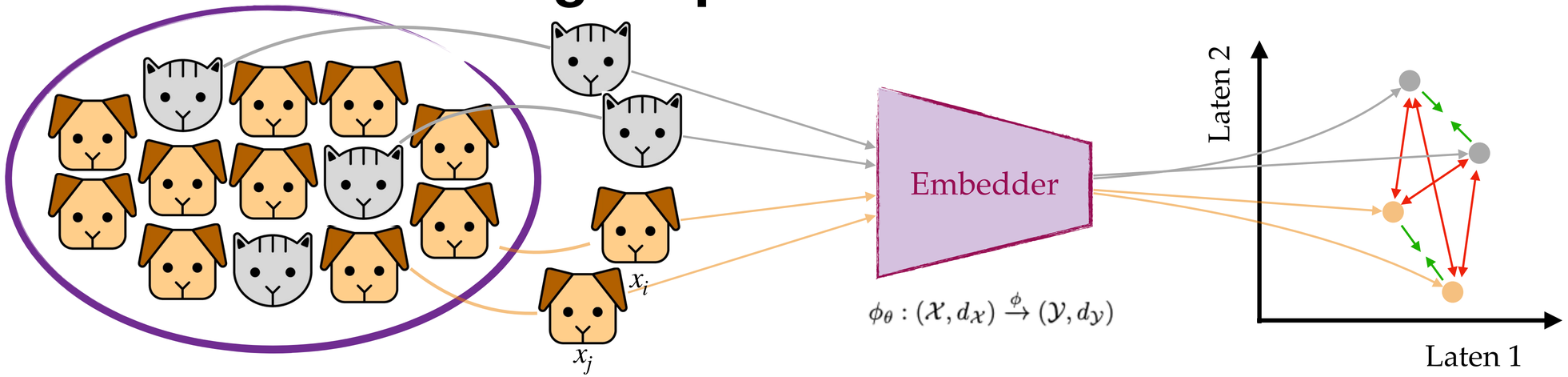
$$l_{\text{SimCLR}}(x_i, x_j) = -\log \frac{e^{\text{sim}(\phi_\theta(x_i), \phi_\theta(x_j))/\tau}}{\sum_{k=1}^{N} e^{\text{sim}(\phi_\theta(x_i), \phi_\theta(x_k))/\tau}}$$

Cosine similarity:

$\text{sim}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{u}^\top \boldsymbol{v} / \|\boldsymbol{u}\| \|\boldsymbol{v}\|$

Learn representations for signal discovery

Contrastive Learning: Supervised



labelled data points

Generate two *views* of the same class

Constraint the organization of the views in the embedding space:

- Views of the same object are pushed together
- Views of different objects are pulled apart

SimCLR loss [2002.05709]:

$$l_{\text{SimCLR}}(x_i, x_j) = -\log \frac{e^{\text{sim}(\phi_\theta(x_i), \phi_\theta(x_j))/\tau}}{\sum_{k=1}^{N} e^{\text{sim}(\phi_\theta(x_i), \phi_\theta(x_k))/\tau}}$$

Cosine similarity:

$\text{sim}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{u}^\top \boldsymbol{v} / \|\boldsymbol{u}\| \|\boldsymbol{v}\|$

Exploit **data augmentations** to induce meaningful forms of **implicit bias** in the data **organization**

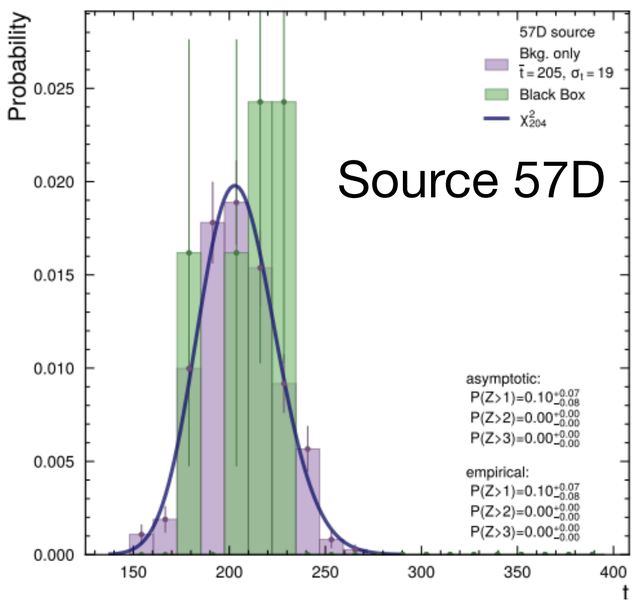
Loose way to impose **invariance** to transformations

Learn representations for signal discovery

Contrastive Learning: Supervised

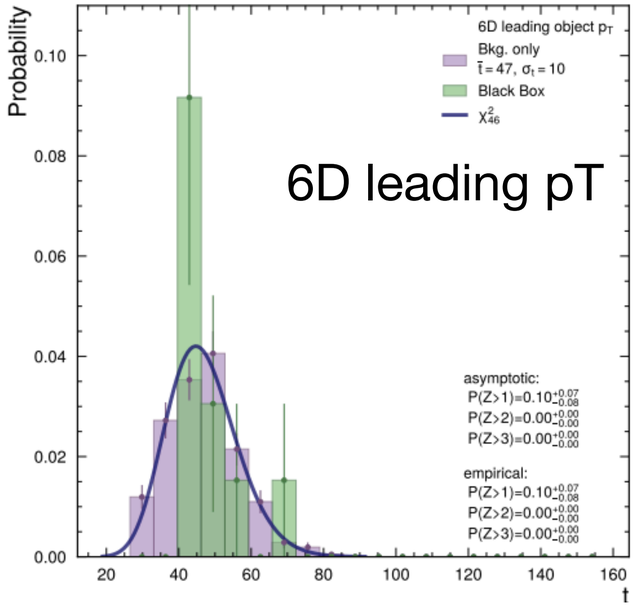
Black box detection

Original representation



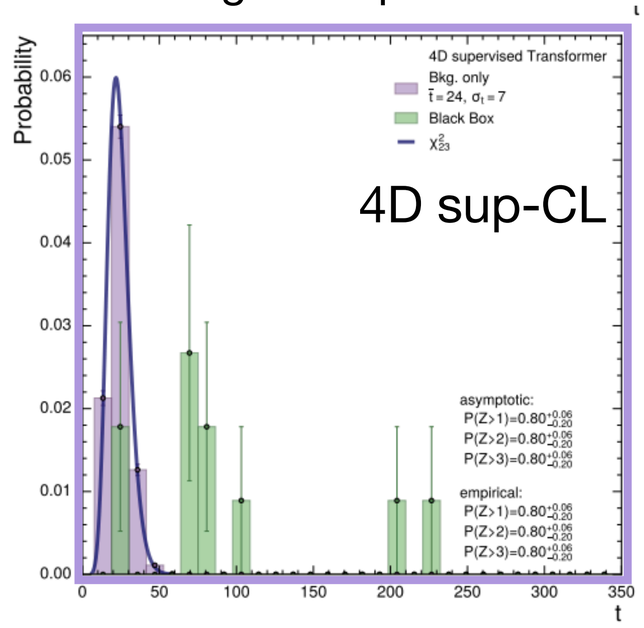
LRT

A physics motivated reduction (not optimized on the signal)



LRT

Supervised CL, using labelled simulations of background processes



LRT

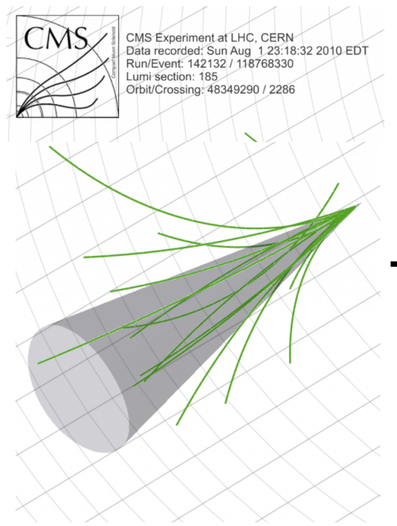
Experiments performed using ADC2021 dataset [2107.02157]:
19 particle physics objects, 3 components (p_T, η, ϕ)

From Metzger et al. (Phys. Rev. D 112, 072011)

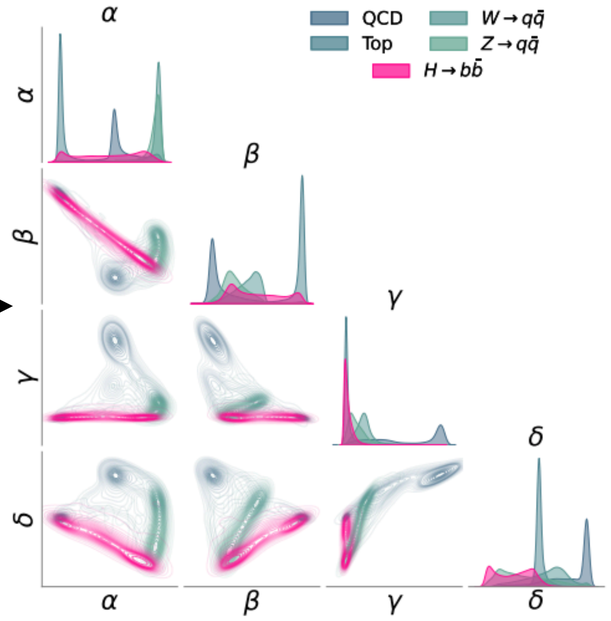
Learn representations for signal discovery

Contrastive Learning: Supervised

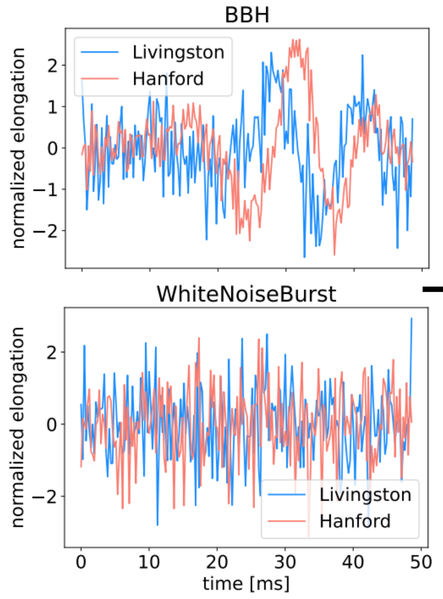
LHC jets (point cloud)



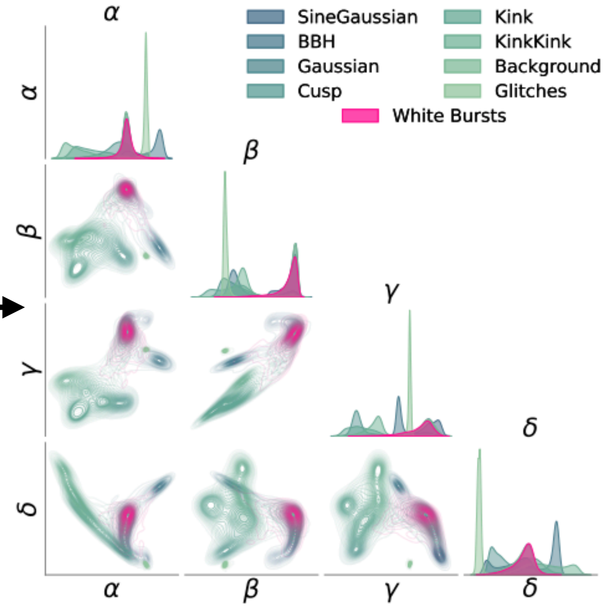
4D organized latent representation



LIGO gravitational waves (time series)



4D organized latent representation



From: Bright-Thonney et al. (NeurIPS 2025)

Conclusions

Open problems

ROBUSTNESS:

- Is the anomaly a real novelty or the product of domain shifts?
- Can we distinguish among anomalies and how?

INTERPRETABILITY & EXPLAINABILITY:

- A generative model ranks better than another: do we understand why? Can we point at the failure source?
- I detection was made with 3sigma significance, can we tell which new phenomenon is behind that?

EFFICIENCY AT SCALE:

- How do we look at billions of data?
- How can we test fast? Online applications

Take home message

Choose your method according to your problem settings!

- Exploit available **knowledge**
- Identify **priors** on the problems
- Identify **assumptions** behind the methods
- Identify **limitations**/failure modes of the strategy (when you should stop trusting the result)
- **Efficiency** matters! Pick the methods that better fits the scale of your problem (not necessarily the fanciest one :))

More readings

Good overviews/comparison between methods:

- How good are your fits? Unbinned multivariate goodness-of-fit tests in high energy physics William *JINST 5 (2010), P09004*
- Machine learning and multivariate goodness of fit Weisser and Williams *pre-print:1612.07186*
- Goodness of fit by Neyman-Pearson testing Grosso, Letizia, Pierini, Wulzer. *SciPost Phys. 16 (2024) 5, 123*
- Refereeing the referees: evaluating two-sample tests for validating generators in precision sciences Grossi, Letizia, Torre *Mach.Learn.Sci.Tech. 6 (2025) 1, 015052*

Backup slides

Maximum Likelihood Ratio test

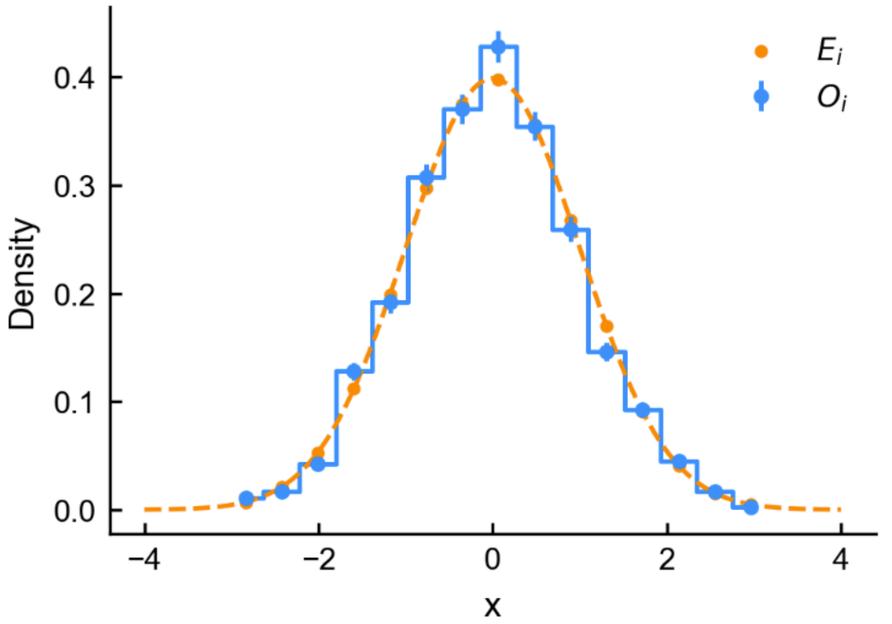
The quantity used to perform maximum-likelihood estimation (MLE) for the free parameters w of a given model H_w can be used as a goodness-of-fit test:

$$t(\mathcal{D}) = \max_{\mathbf{w}} \left[2 \log \frac{\mathcal{L}(\mathcal{D} \,|\mathrm{H}_\mathbf{w})}{\mathcal{L}(\mathcal{D} \,|\mathrm{H}_\mathbf{0})} \right]$$

The test is a good test if the model H_w is a **flexible but regularized composite hypothesis**

- Binned version: H_w =“saturated model”[1]
  $E(\mathrm{H}_\mathrm{w})_i = O_i,\ \forall i = 1, \ldots, n$

$$t(D) = \chi^2_{\mathrm{Sat}} = 2 \sum_{i=1}^{n} \left(E_i - O_i + O_i \log \frac{O_i}{E_i} \right)$$



$E_i = i$-th bin expected frequency under H_0

$O_i =$ i-th bin observed frequency

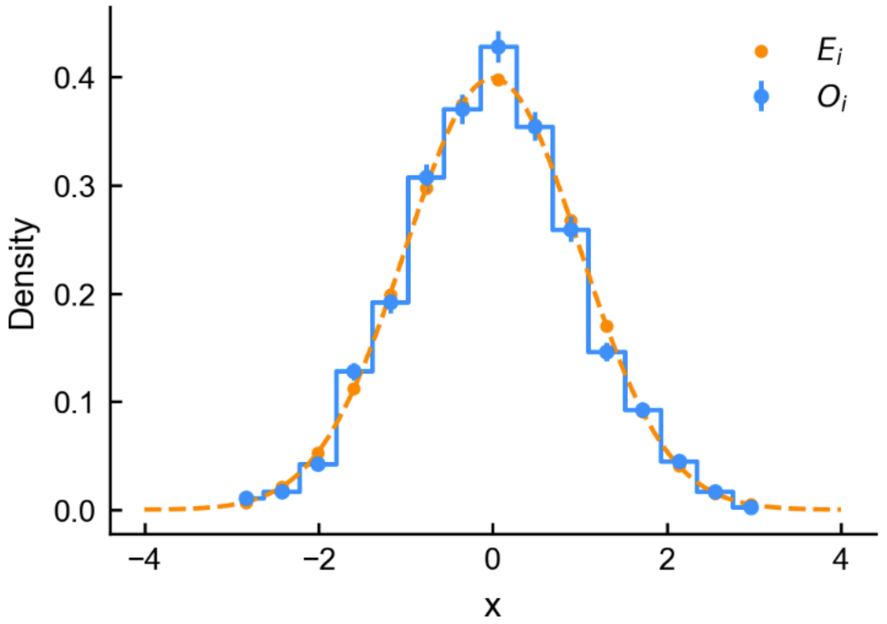
$O_i \sim \mathrm{Poisson}(E_i)$

[1]Baker & Cousins, Nucl.Instrum.Meth. 221 (1984)

χ^2 test

Gaussian limit of the binned likelihood-ratio test

- The most known χ^2 tests:
  - Pearson χ^2: $\chi^2_{\text{Pearson}} = \sum_{i=1}^{m} \frac{(O_i - E_i)^2}{E_i}$
  - Neyman χ^2: $\chi^2_{\text{Neyman}} = \sum_{i=1}^{m} \frac{(O_i - E_i)^2}{O_i}$



$E_i = i$-th bin expected frequency under H_0

$O_i = i$-th bin observed frequency

$O_i \sim \text{Poisson}(E_i)$

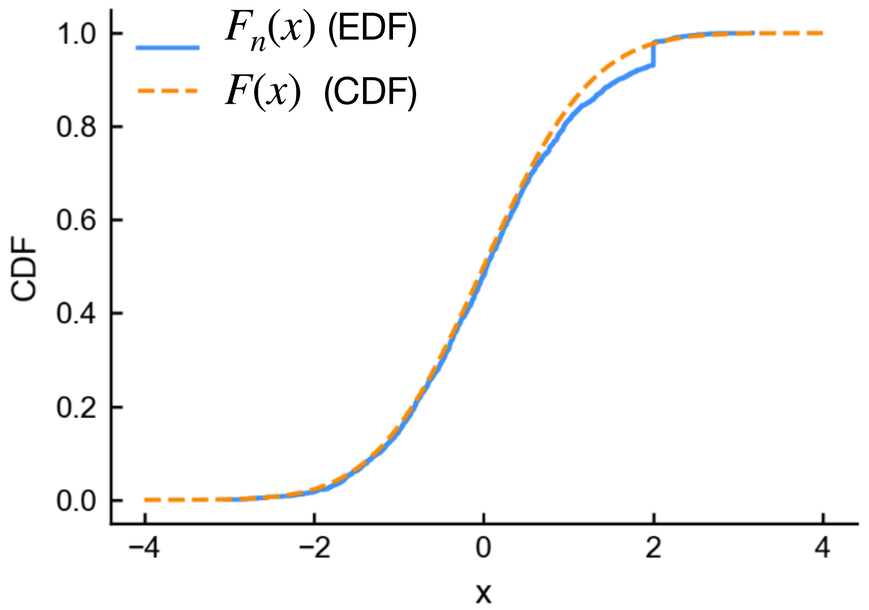
Caveats:

- Both Pearson and Neyman tests are biased for small frequencies!
- The binned nature of the test limits applicability to few dimensions
- Number of bins and binning choice determine the outcome

Empirical Distribution Function Tests (EDF statistics)

Based on the comparison of the expected Cumulative Distribution Function (CDF) with the empirical one (EDF) estimated from the data sample of size n.

- In general more powerful than a χ^2 test
- Caveat: It can only be applied to continuous distributions



Empirical Distribution Function Tests (EDF statistics)

Tests in this family differ for the metric chosen to measure the distance.

Some of the most known tests:

- Kolmogorov-Smirnov (1933): $\mathrm{KS}_n(D) = \sup_{x \in D} |F(x) - F_n(x)|$

More importance to the bulk

- Cramer-von-Mises (1928–1930): $\mathrm{CvM} = n \int_{-\infty}^{+\infty} \left(F(x) - F_n(x)\right)^2 dF(x)$
- Anderson-Darling (1952): $\mathrm{AD} = n \int_{-\infty}^{+\infty} \frac{\left(F(x) - F_n(x)\right)^2}{F(x)\left(1 - F(x)\right)} dF(x)$

More weight to the tails

$F(x)$ = CDF

$F_n(x)$ = EDF

Classifier based tests

Calibration

- **Train-test split:**
  - Train the classifier using one part of the data
  - Perform a test on the classifier output using the other part of the data.
  - This guarantees that the distribution of the test statistics under H_0 is what expected.
  - However, throwing away part of the data for training reduces the power of the test. This can be mitigated by performing cross-validation.???

- **Train and test all:**
  - Train the classifier and compute a test on all the data
  - Build a distribution of the test under H_0 by retraining multiple times the classifier using anomaly-free data $D \sim \mathrm{H}_0$ [random permutation / pseudo-experiments].

Metrics for comparison

P-value

For applications that aim at discovery, a typical way to evaluate an anomaly detection method is the p-value or the corresponding Z-score:

Z-score (standardized parametrization of the p-value):

$$Z_p = \Phi^{-1}(1-p)$$

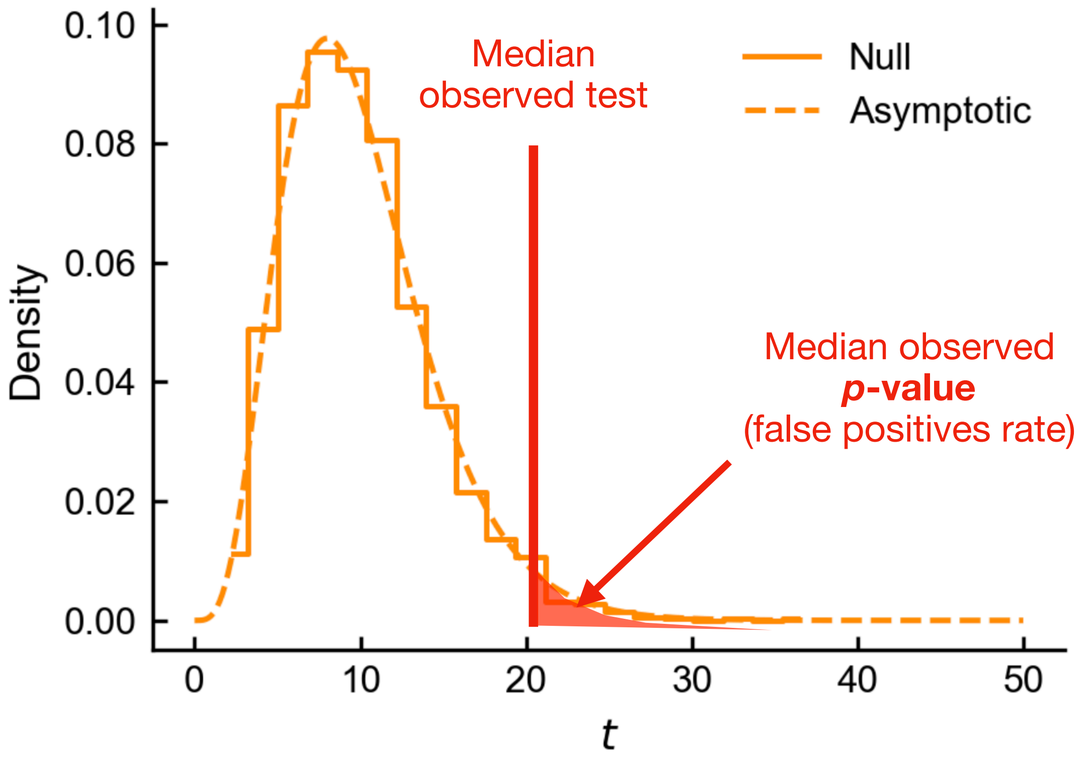
Typical values in fundamental physics:

$Z = 3\sigma \rightarrow p$-value $\approx 1.35 \cdot 10^{-3}$ (evidence)

$Z = 5\sigma \rightarrow p$-value $\approx 2.9 \cdot 10^{-7}$ (discovery)

In other domains (biology, computer-science, …):

$Z = 1.6\sigma \leftarrow p$-value ≈ 0.05



Metrics for comparison

Power

When one wants to evaluate the performance against a *specific alternative hypothesis*, the **power** provides an informative metric.

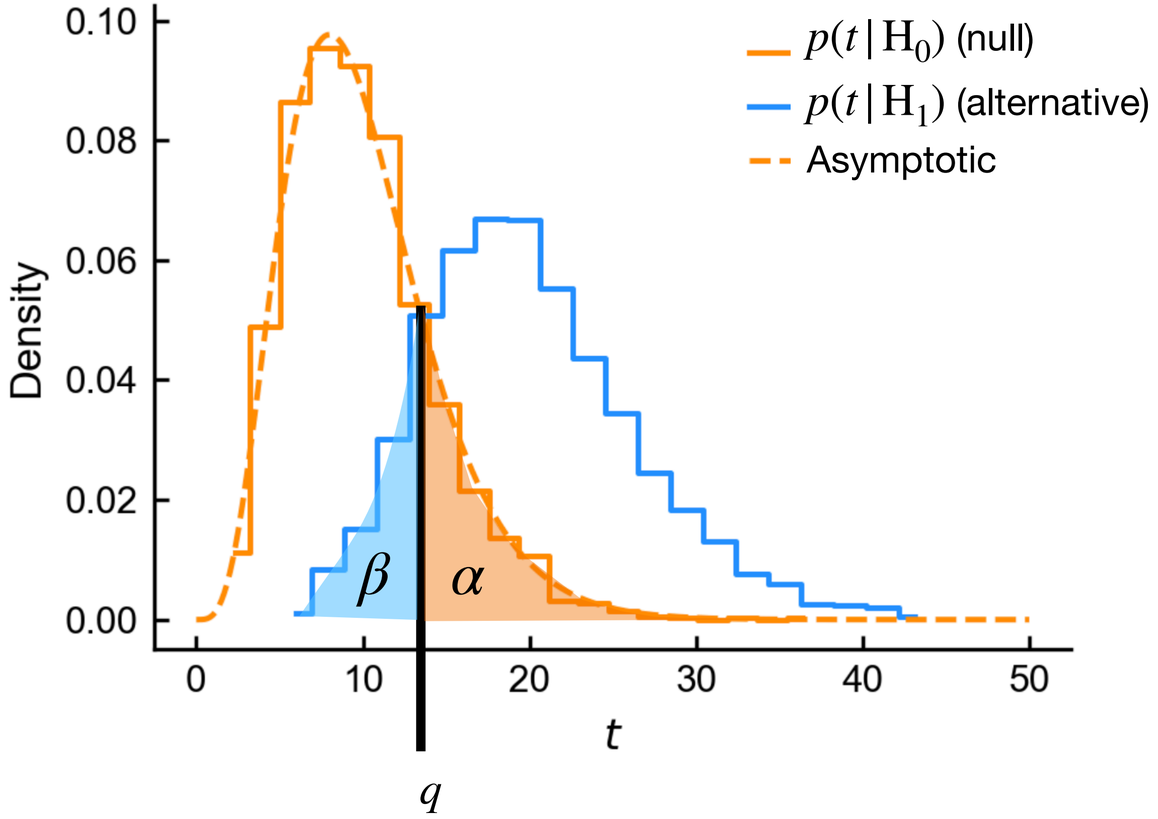
Let's introduce two basic notions:

- Type I error (false positive rate)

$$\alpha = \int_q^\infty p(t\,|\,\mathrm{H}_0)\,dt$$

- Type II error (false negative rate)

$$\beta = 1 - \int_q^\infty p(t\,|\,\mathrm{H}_1)\,dt$$



[1] https://root.cern.ch/doc/master/classTEfficiency.html

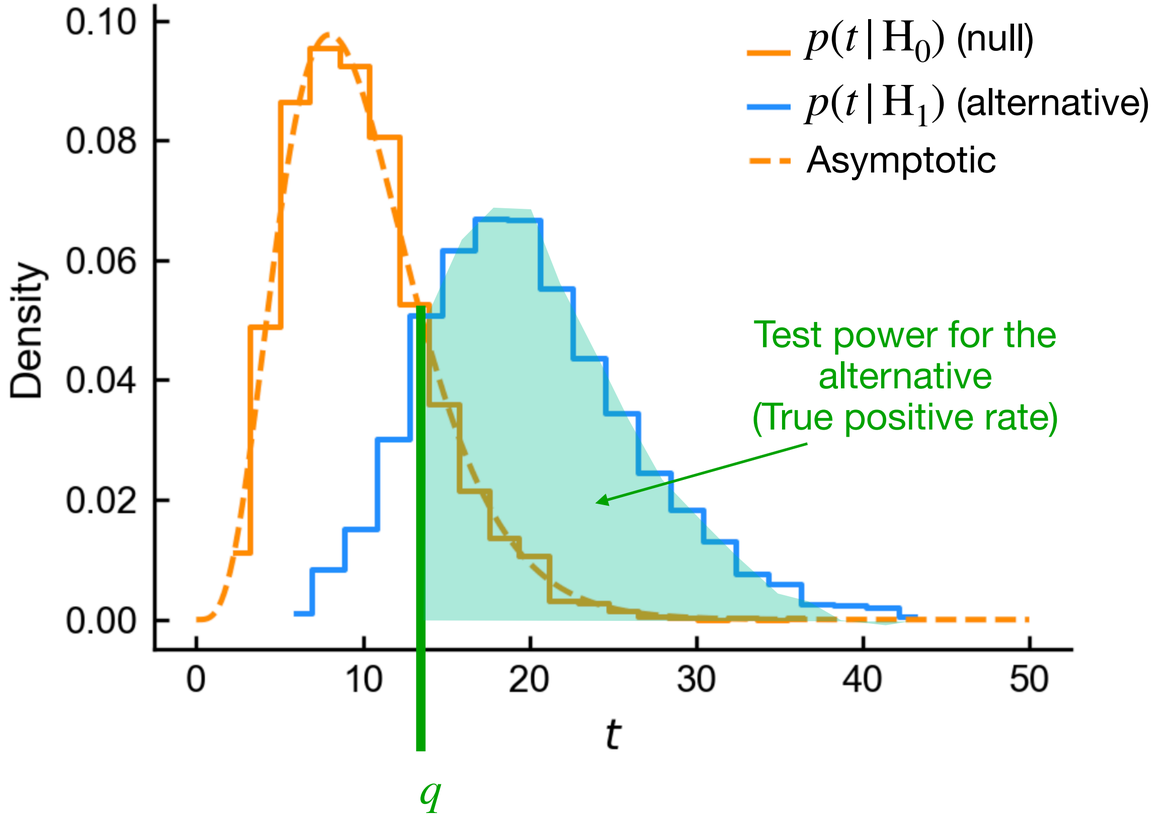
Metrics for comparison

Power

When one wants to evaluate the performance against a *specific alternative hypothesis*, the **power** provides an informative metric.

- Power: $1 - \beta = \int_q^\infty p(t\,|\,\mathrm{H}_1)\,dt$

- Note: to estimate the error on the power: use methods designed for efficiencies[1] (example Clopper-Pearson)



[1] https://root.cern.ch/doc/master/classTEfficiency.html

Metrics for comparison

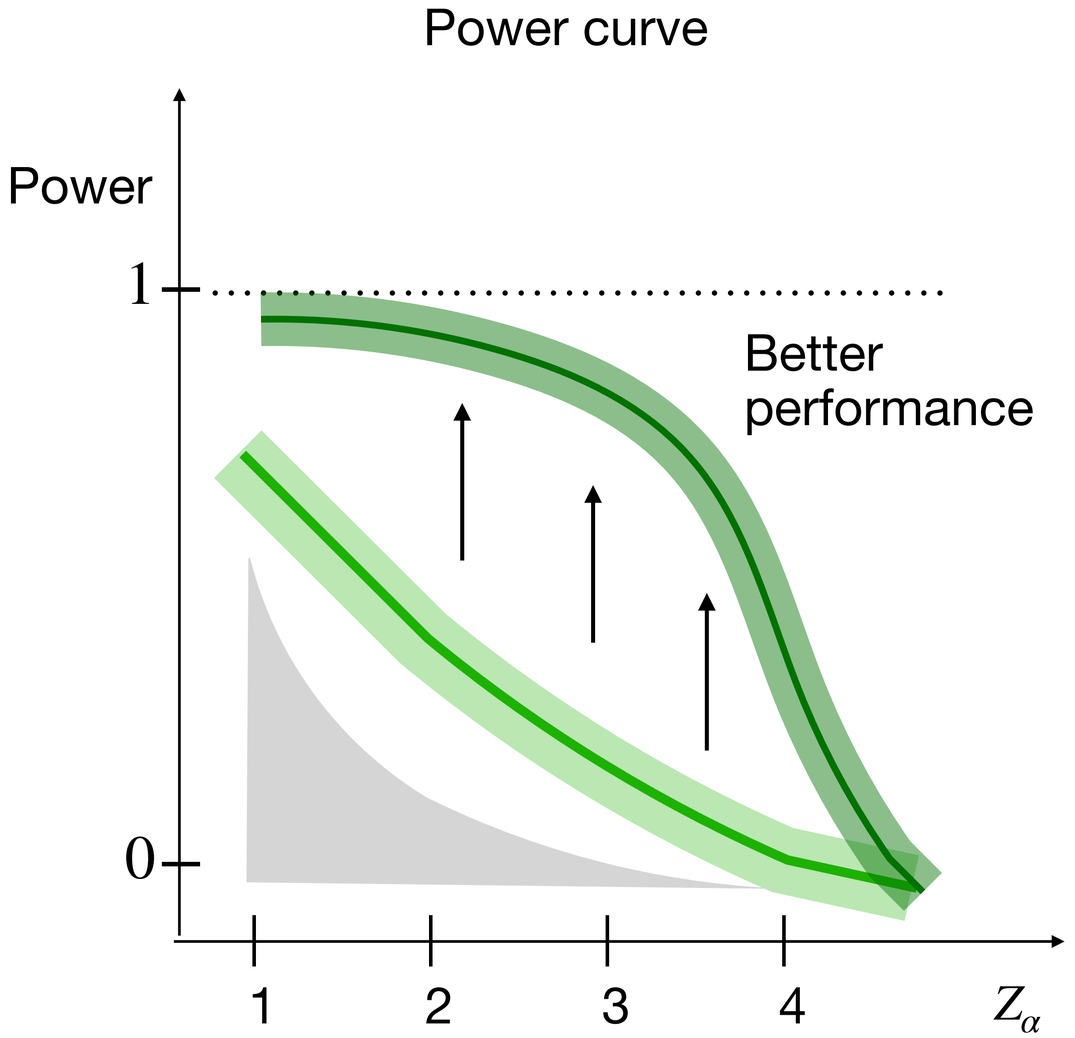
Power Curve

When one wants to evaluate the performance against a *specific alternative hypothesis*, the **power** provides an informative metric.

- Power: $1 - \beta = \int_q^\infty p(t\,|\,\mathrm{H}_1)\,dt$

- Note: to estimate the error on the power: use methods designed for efficiencies[1] (example Clopper-Pearson)

- We can scan q and map values of β vs. α into a sensitivity curve.
- To facilitate the interpretation, it is useful to map:
  - $\alpha \to Z_\alpha$
  - $\beta \to 1 - \beta$ (e.g. power)

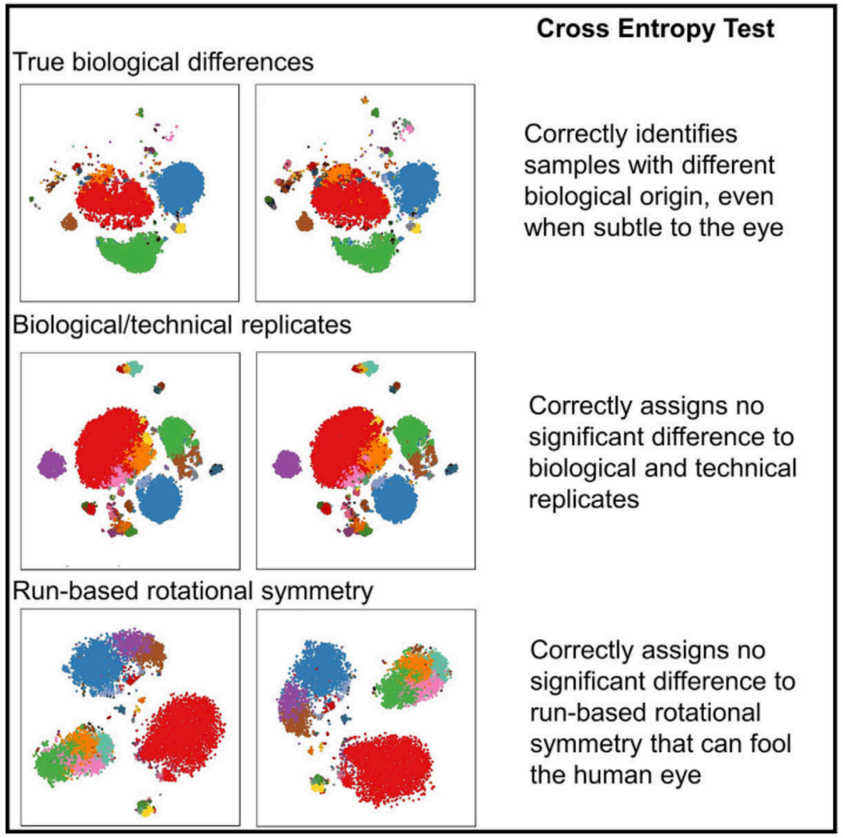


Cell Reports
Methods

Article

A cross entropy test allows quantitative statistical comparison of t-SNE and UMAP representations

Graphical abstract



Authors

Carlos P. Roca, Oliver T. Burton, Julika Neumann, ..., Rafael V. Veiga, Stéphanie Humblet-Baron, Adrian Liston

Correspondence

al989@cam.ac.uk

In brief

Dimensionality-reduction tools such as t-SNE and UMAP allow visualizations of single-cell datasets. Roca et al. develop and validate the cross entropy test for robust comparison of dimensionality-reduced datasets in flow cytometry, mass cytometry, and single-cell sequencing. The test allows statistical significance assessment and quantification of differences.

https://doi.org/10.1016/j.crmeth.2022.100390