

# Machine learning in science and society

## From automated science to beneficial artificial intelligence

Christos Dimitrakakis

August 21, 2019

# Introduction

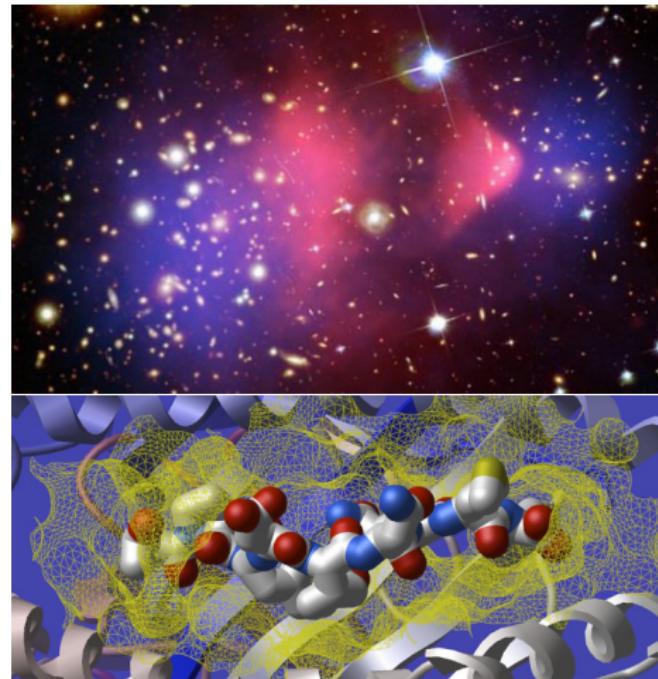
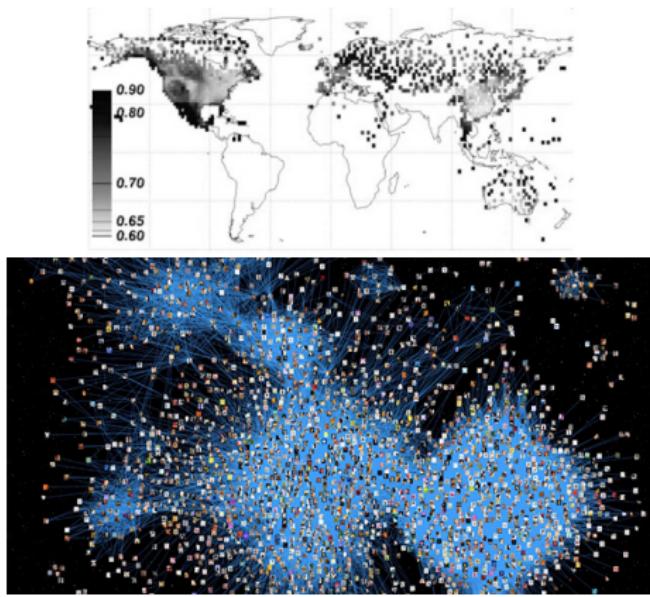
## 1 Introduction to machine learning

- Data analysis, learning and planning
- Experiment design
- Bayesian inference.
- Course overview

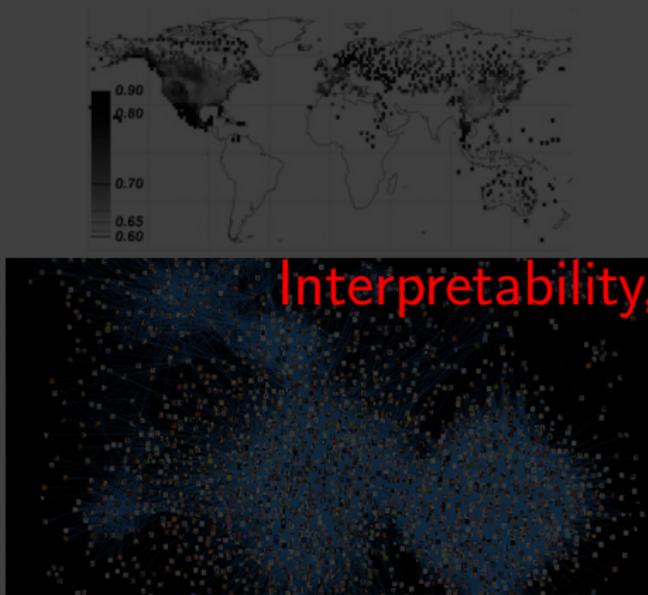
## 2 Nearest neighbours

## 3 Reproducibility

# Scientific applications



# Scientific applications



Interpretability,



Reproducibility

# Pervasive “intelligent” systems



Home assistants



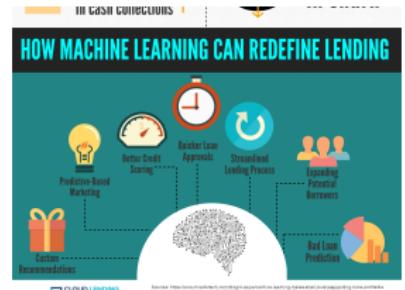
Autonomous vehicles



Web advertising



Ridesharing



Lending



Public policy

# Pervasive “intelligent” systems



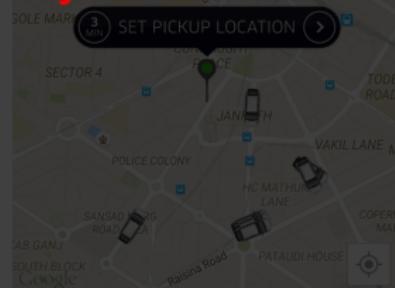
Home assistants



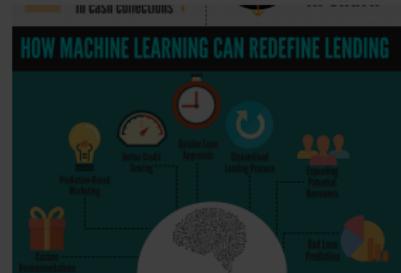
Web advertising



Autonomous vehicles



Ridesharing



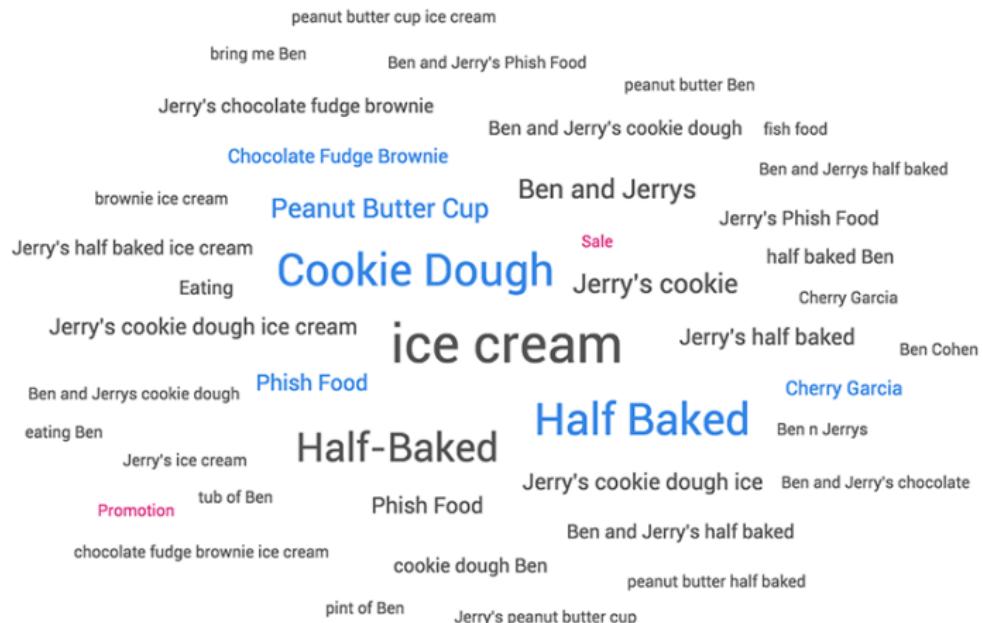
Lending



Public policy

# What can machine learning do?

# Can machines learn from data?



■ Topics ■ Tags ■ Categories

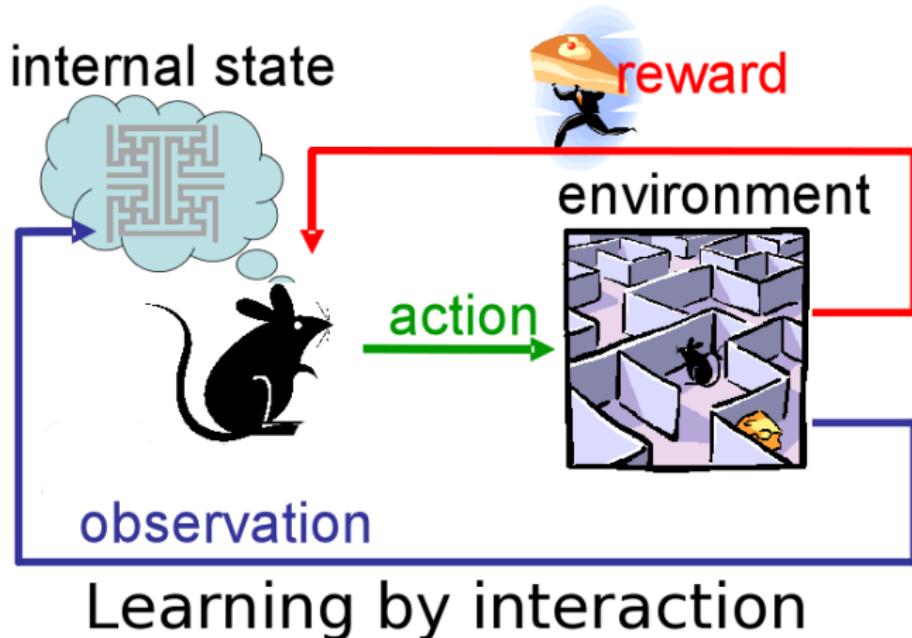
An unsupervised learning problem: topic modelling

# Can machines learn from data?



A supervised learning problem: object recognition

Can machines learn from their mistakes?



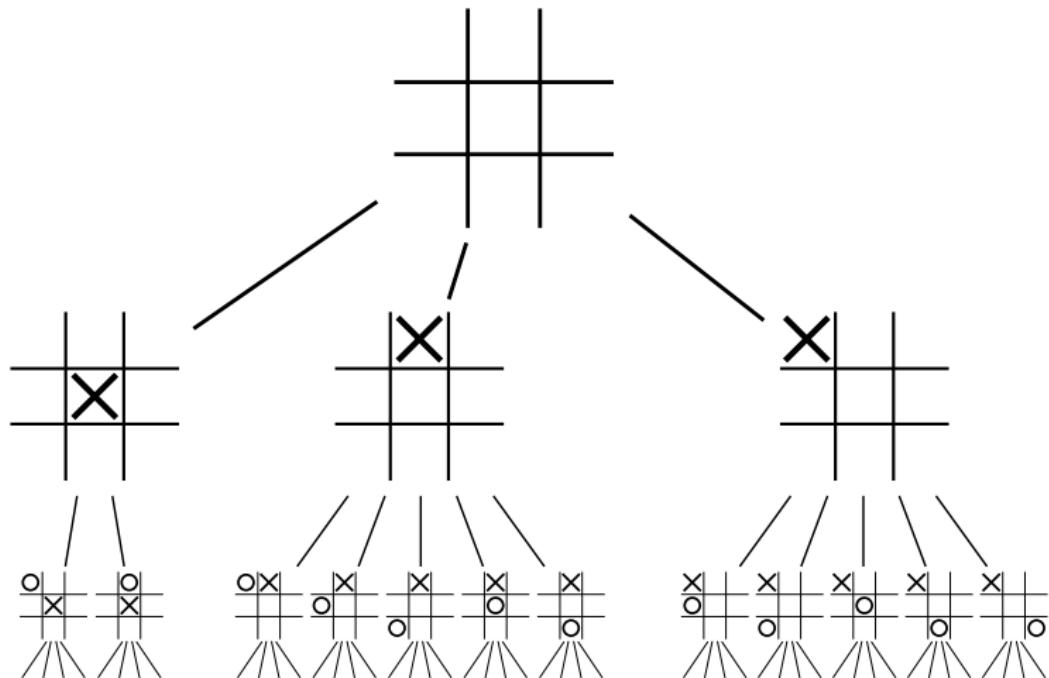
## Reinforcement learning

Take actions  $a_1, \dots, a_t$ , so as to maximise utility  $U = \sum_{t=1}^T r_t$

# Can machines make complex plans?



# Machines can make complex plans!



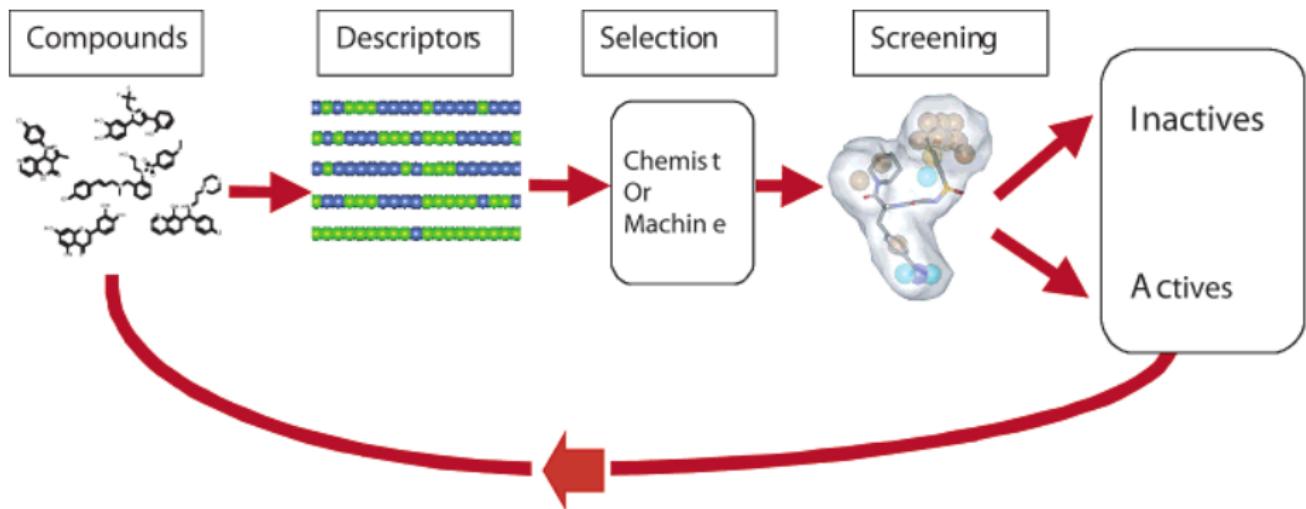
# The scientific process as machine learning



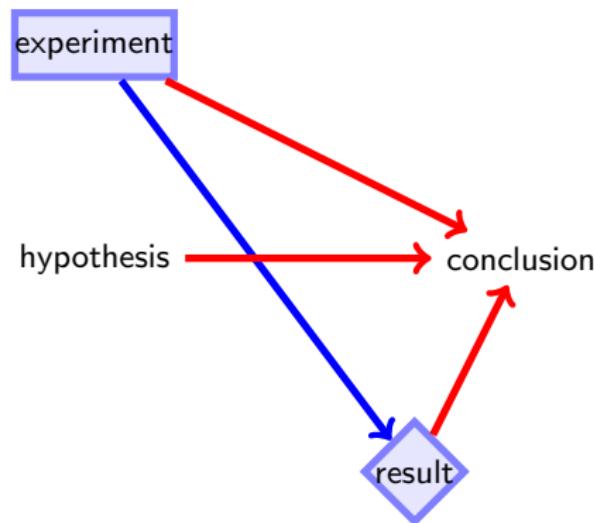
## Adam, the robot scientist



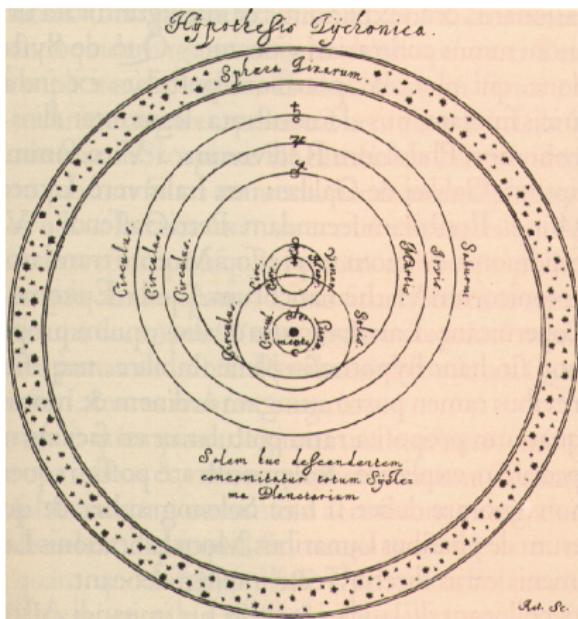
# Drug discovery



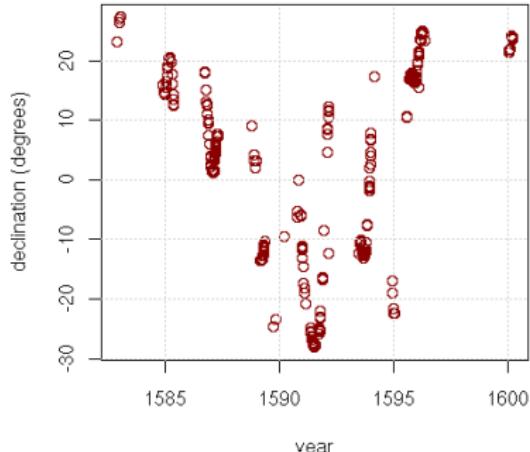
## Drawing conclusions from results



# Tycho Brahe's minute eye measurements



Tycho Brahe's Mars Observations

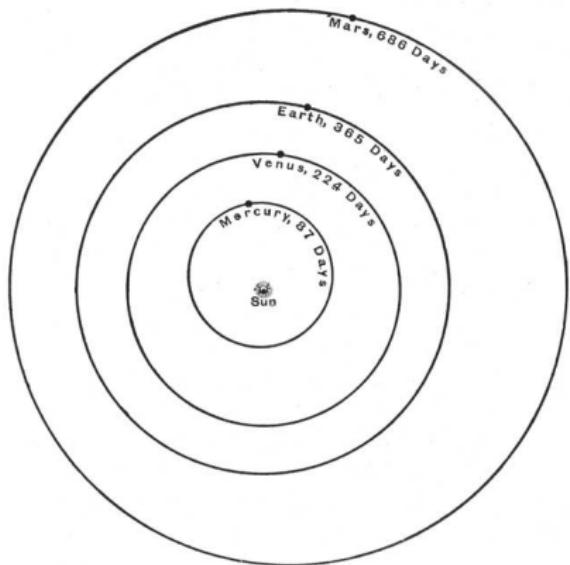


source: Tychonis Brahe Dani Opera Omnia

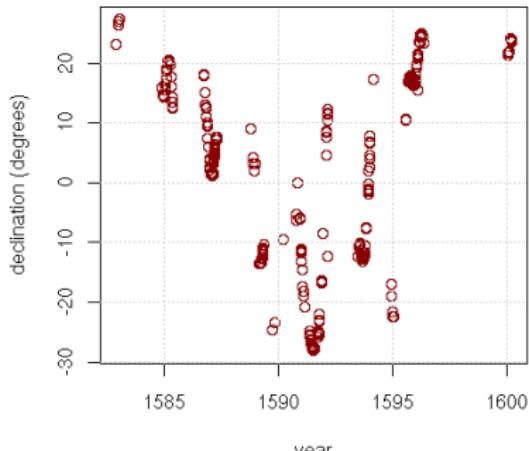
**Figure:** Tycho's measurements of the orbit of Mars and the conclusion about the actual orbits, under the assumption of an earth-centric universe with circular orbits.

- Hypothesis: Earth-centric, Circular orbits
- Conclusion: **Specific** circular orbits

# Johannes Kepler's alternative hypothesis



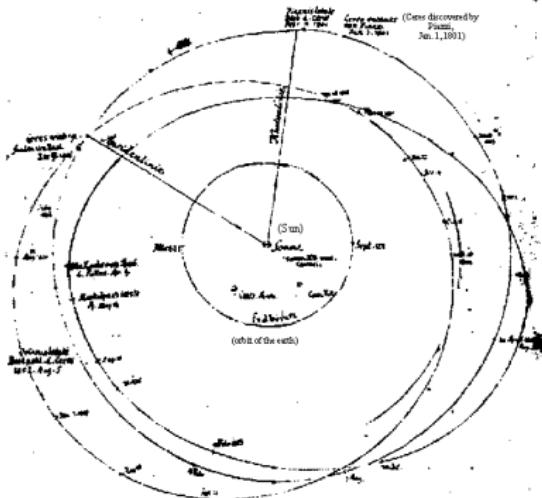
Tycho Brahe's Mars Observations



source: Tychonis Brahe Dani Opera Omnia

- Hypothesis: Circular or elliptic orbits
- Conclusion: Specific elliptic orbits

# 200 years later, Gauss formalised this statistically

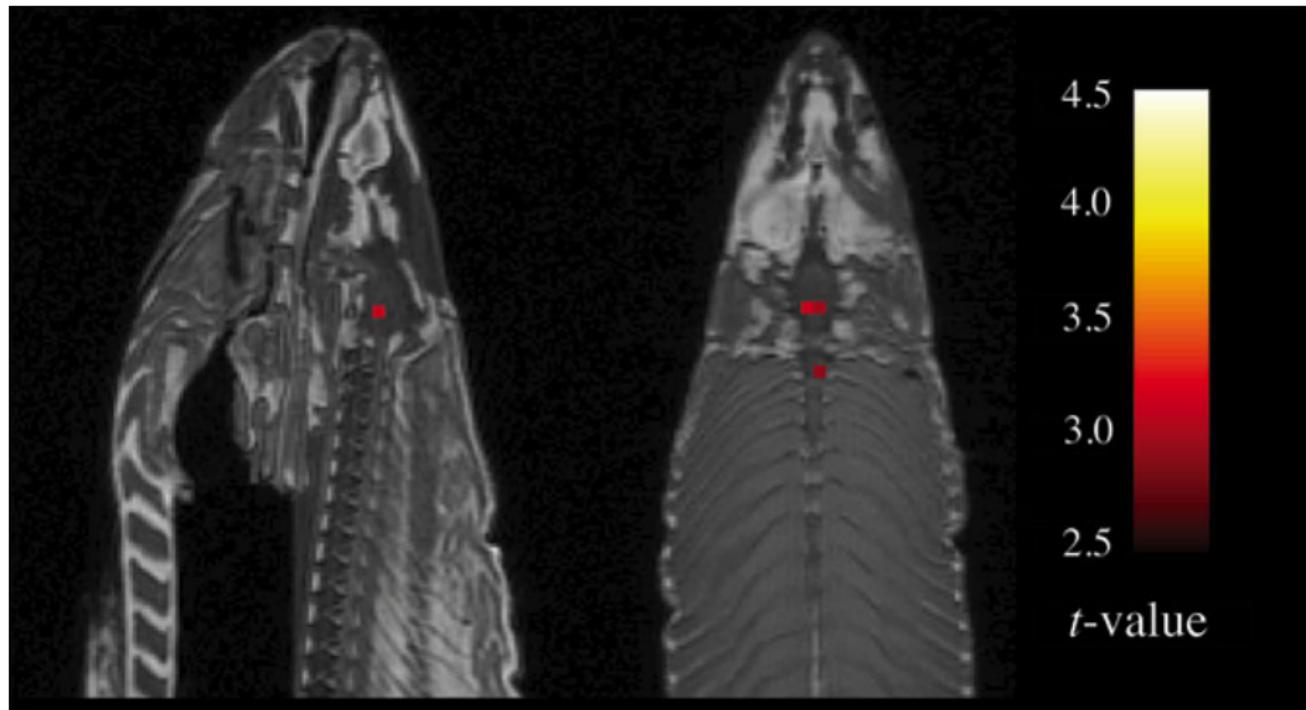


Sketch of the orbits of Ceres and Pallas (nachläß Gauß, Handb. 4). Courtesy of Universitätsbibliothek Göttingen.

Beobachtungen des zu Pallas (§)

	Mittlerer Sonnen- Zeit	Gleiche Auflösung in Zeit	Gleiche Auflösung in Grade	Nordl. Abweich.	Gegentri- chtige Länge	Gemeins. Breite	Ort der Sonne + 20° Logar. d. Distanz §
Jan.							Z
	1 8 43 27,5	16	27 23,25	51 47 45,8	15 27 43,5	1 23 22 55,2	9 11 1 30,9 9,9926156
	3 16 21 27,0	16	27 23,25	51 47 45,8	15 27 43,5	1 23 22 55,2	9 11 1 30,9 9,9926156
	3 8 34 25,3	26 38,4	54 39 26,0	15 46 31,6	15 33 16 55,6	2 59 3 59	9 13 3 26,6 9,9926156
	4 8 10 42,1	29 23	51 55 35 47,3	15 46 31,6	15 33 16 55,6	2 59 3 59	9 13 3 26,6 9,9926156
	10 8 6 25,5	29 23	51 55 35 47,3	15 46 31,6	15 33 16 55,6	2 59 3 59	9 13 3 26,6 9,9926156
	11 8 2 17,5	30 29,73	51 55 35 47,3	15 46 31,6	15 33 16 55,6	2 59 3 59	9 13 3 26,6 9,9926156
	11 8 2 17,5	30 29,73	51 55 35 47,3	15 46 31,6	15 33 16 55,6	2 59 3 59	9 13 3 26,6 9,9926156
	12 8 15 26,5	26 39,9	51 55 35 47,3	15 46 31,6	15 33 16 55,6	2 59 3 59	9 13 3 26,6 9,9926156
	12 8 15 26,5	26 39,9	51 55 35 47,3	15 46 31,6	15 33 16 55,6	2 59 3 59	9 13 3 26,6 9,9926156
	13 7 25 13,3	26 39,9	51 55 35 47,3	15 46 31,6	15 33 16 55,6	2 59 3 59	9 13 3 26,6 9,9926156
	17						.....
	18 7 35 13,3	25 65,	51 55 35 47,3	15 46 31,6	15 33 16 55,6	2 59 3 59	9 13 3 26,6 9,9926156
	19 7 31 28,5	26	6 15,9	51 55 35 47,3	15 46 31,6	1 23 25 59,2	9 30 19 53 6 9,9936067
	21 7 24 2,7	26	10 34,27	51 55 35 47,3	15 46 31,6	1 23 34 21,2	1 49 6 0 10 1 10 40 40,9 9,9921434
	21 7 24 2,7	26	10 34,27	51 55 35 47,3	15 46 31,6	1 23 34 21,2	1 49 6 0 10 1 10 40 40,9 9,9921434
	23 7 16 45,6	27	6 10,9	58 46 43,5	17 9 5,5	1 23 44 55,7	2 28 1 2 21 29 6 9,9931656
	23 7 16 45,6	27	6 10,9	58 46 43,5	17 9 5,5	1 23 44 55,7	2 28 1 2 21 29 6 9,9931656
	28 6 58 51,3	28 26,9	53 13 38,3	17 33 54,1	24 15 25,7	1 21 6,9	8 26 20,9 9,9935942
	30 6 51 52,9	29 45,14	52 27 21,7	17 43 11,0	24 30 9,6	16 16,0	1 20 27 46 2,9 9,9935133
	31 6 48 36,4	17	26 53 33	18,8 17 48 21,5	24 38 7,1	1 10 64,6	10 18 18 28,5 9,9937007
	Febr.						.....
	1 8 44 59,9	29 47,7	53 42 48,0	17 53 16,5	1 24 46 59,2	7 39 9	12 39 9 6 9,9937703
	1 8 44 59,9	29 47,7	53 42 48,0	17 53 16,5	1 24 46 59,2	7 39 9	12 39 9 6 9,9937703
	5 6 31 33,5	33 2,79	53 42 48,0	18 1,0	1 25 45 57,9	6 1 16 49 49 49	1 15 21 15 6 9,9937531
	5 6 31 33,5	33 2,79	53 42 48,0	18 1,0	1 25 45 57,9	6 1 16 49 49 49	1 15 21 15 6 9,9937531
	6 6 31 39,3	34 58,50	53 44 17,9	19 15 23,2	25 53 29,5	3 9 10 19 13 22 10,9 9,9941776	.....
	11 6 11 58,13	37 5,54,54	16 28,11	18 47,58,8	11 26 26 40,0	0 36 3,9	10 22 18 10,9 9,9945823

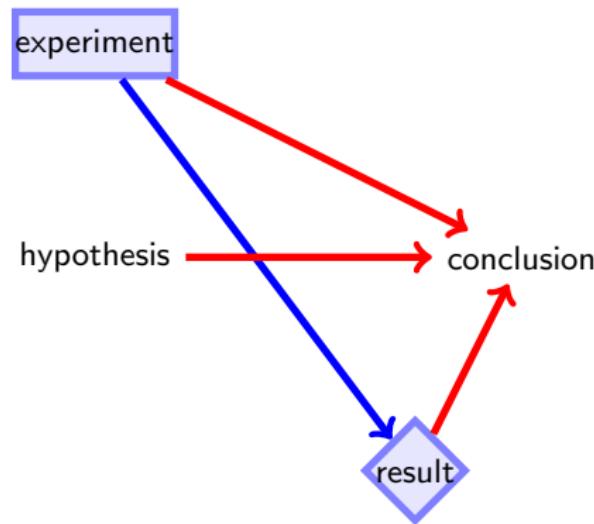
## A warning: The dead salmon mirage



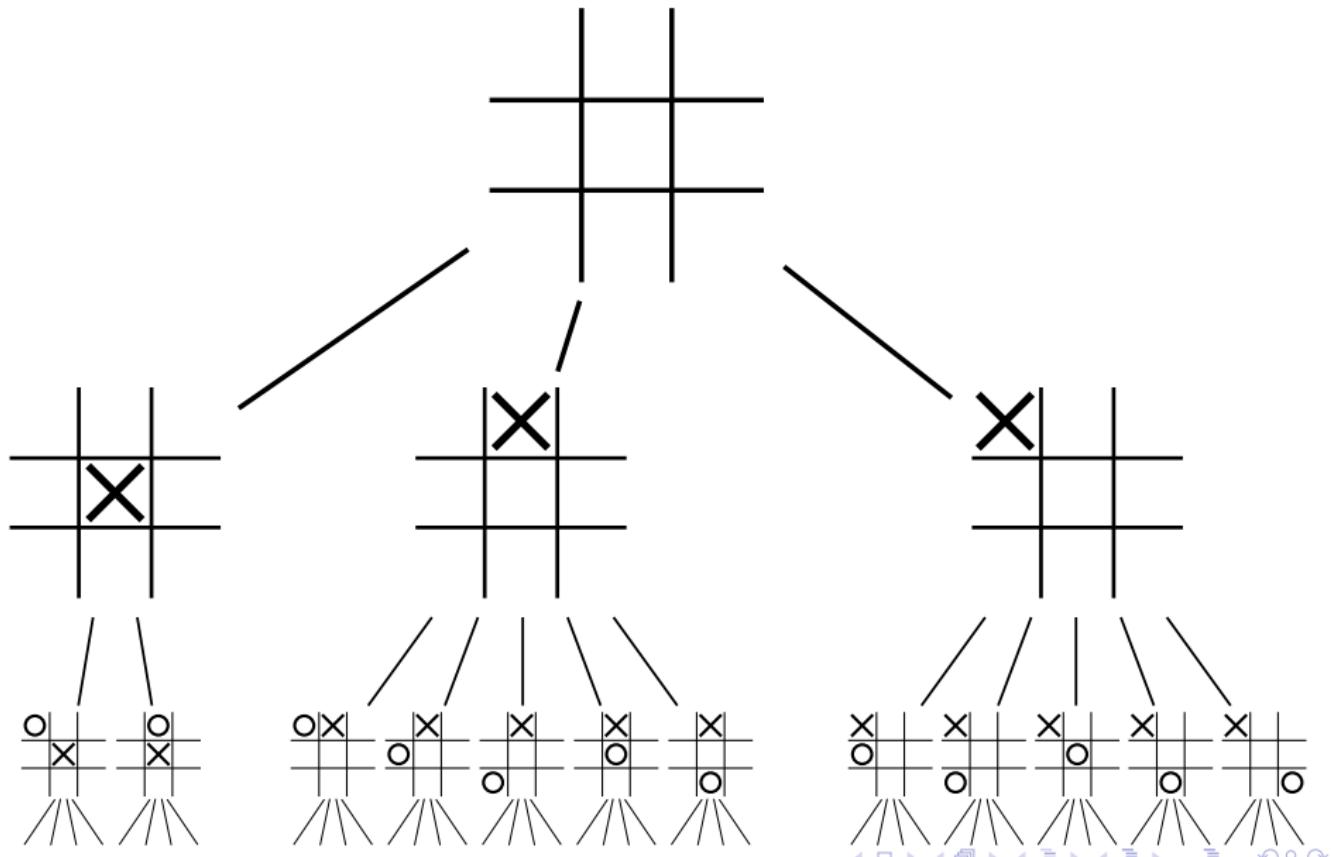
## A simple analysis

`src/reproducibility/mri_analysis.ipynb`

## Planning future experiments



# Planning experiments is like Tic-Tac-Toe



## Eve, another robot scientist



a malaria drug

# Machine learning in practice

## Avoiding pitfalls

- Choosing hypotheses.
- Correctly interpreting conclusions.
- Using a good testing methodology.

## Machine learning in society

- Privacy
- Fairness
- Safety

# Machine learning in practice

## Avoiding pitfalls

- Choosing hypotheses.
- Correctly interpreting conclusions.
- Using a good testing methodology.

## Machine learning in society

- Privacy — Credit risk.
- Fairness
- Safety

# Machine learning in practice

## Avoiding pitfalls

- Choosing hypotheses.
- Correctly interpreting conclusions.
- Using a good testing methodology.

## Machine learning in society

- Privacy — Credit risk.
- Fairness — Job market.
- Safety

# Machine learning in practice

## Avoiding pitfalls

- Choosing hypotheses.
- Correctly interpreting conclusions.
- Using a good testing methodology.

## Machine learning in society

- Privacy — Credit risk.
- Fairness — Job market.
- Safety — Medicine.

# Course structure

## Module structure

- **Activity**-based, hands-on.
- Mini-lectures with short exercises in each class.
- Technical tutorials and labs in alternate week.

## Modules

Three mini-projects.

- Simple decision problems: Credit risk.
- Sequential problems: Medical diagnostics and treatment.

# Technical topics

## Machine learning problems

- Unsupervised learning.
- Supervised learning.
- Reinforcement learning.

## Algorithms and models

- Bayesian inference and graphical models.
- Stochastic optimisation and neural networks.
- Backwards induction and Markov decision processes.

## Further reading

- Bennett et al.<sup>2</sup> describe how the usual uncorrected analysis of fMRI data leads to the conclusion that the dead salmon can reason about human images.
- Bennett et al.<sup>1</sup> discuss how to perform analyses of medical images in a principled way. They also introduce the use of simulations in order to test how well a particular method is going to perform.

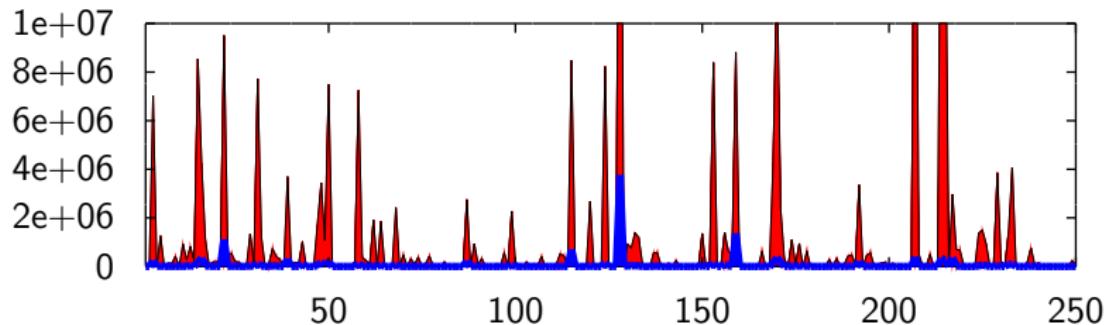
1 Introduction to machine learning

2 Nearest neighbours

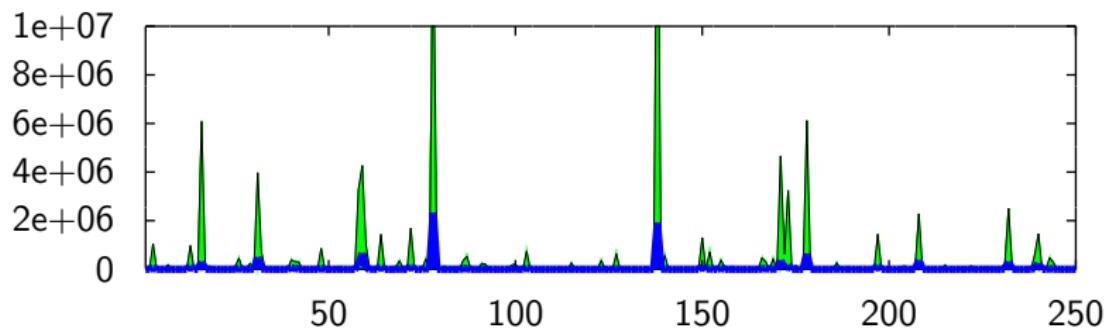
3 Reproducibility

## Discriminating between diseases

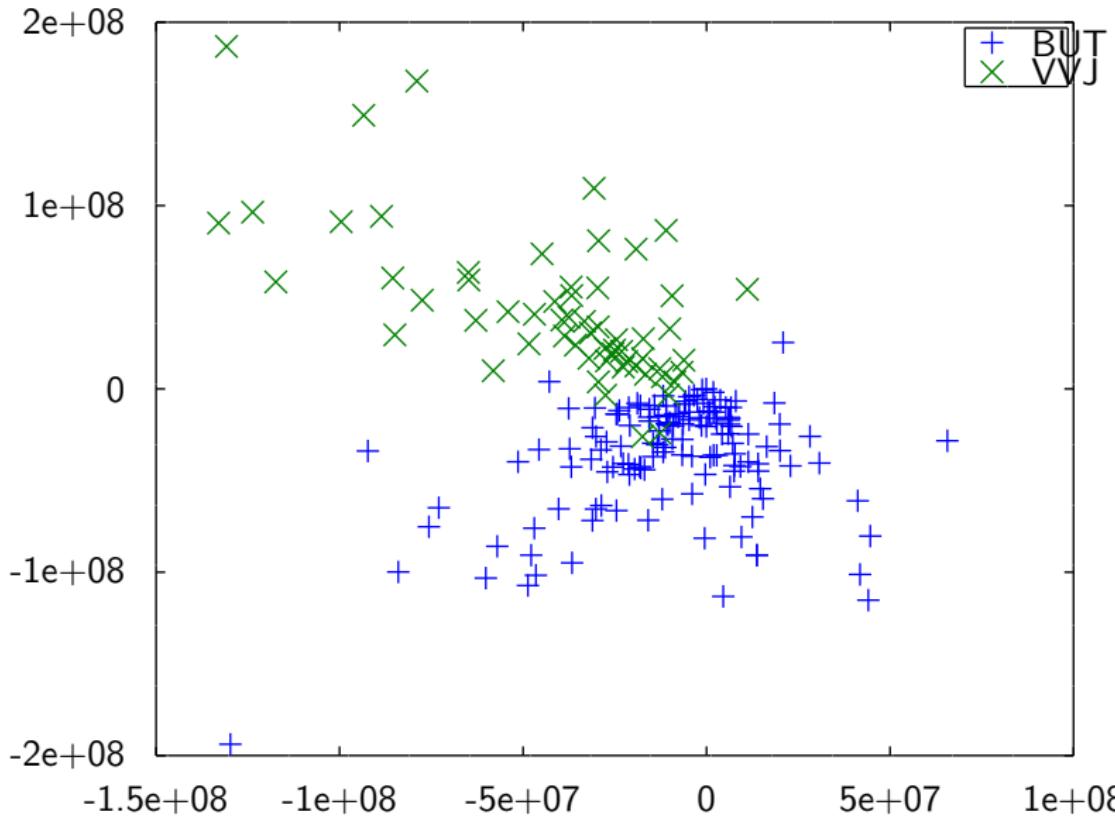
Spectral statistics VVX strain



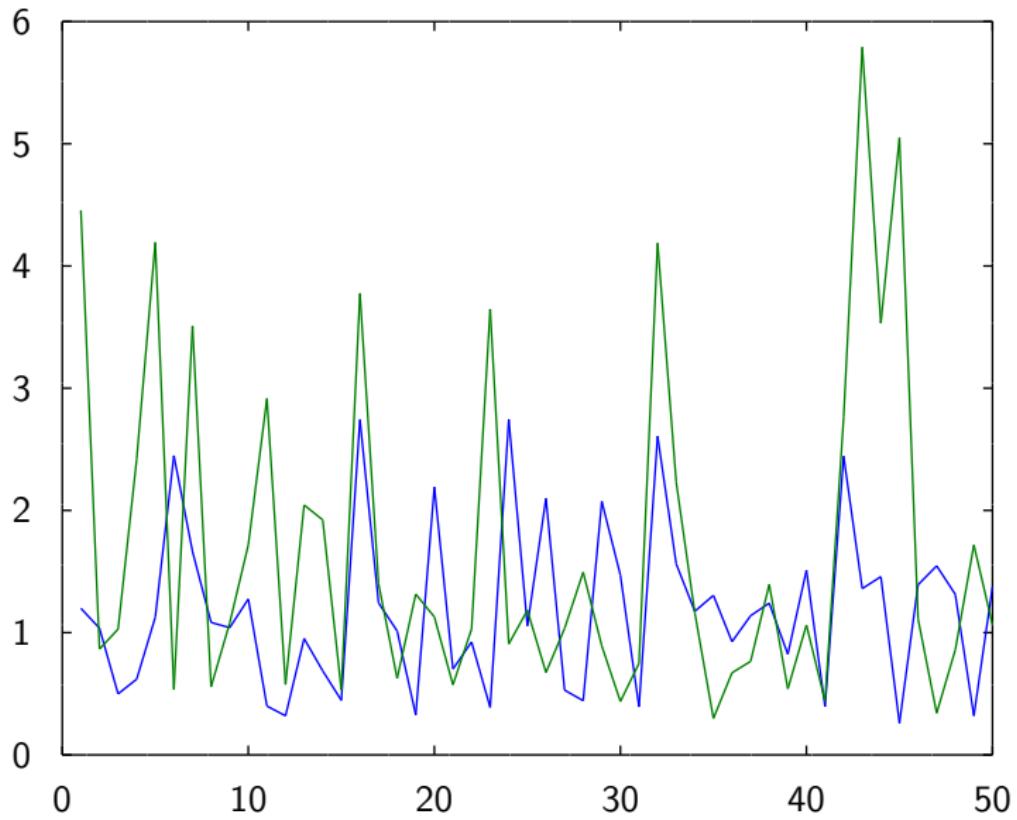
Spectral statistics for BUT



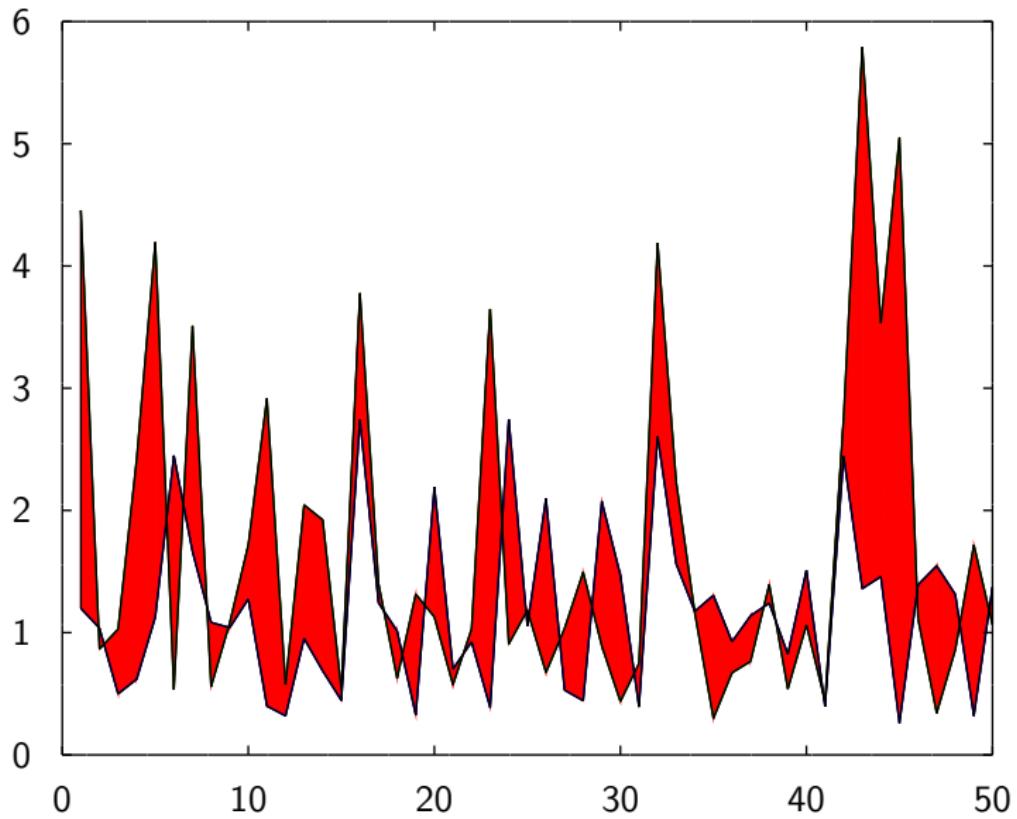
## Nearest neighbour: the hidden secret of machine learning



## Comparing spectral data



## Comparing spectral data



# The nearest neighbour algorithm

---

## Algorithm 1 $k$ -NN Classify

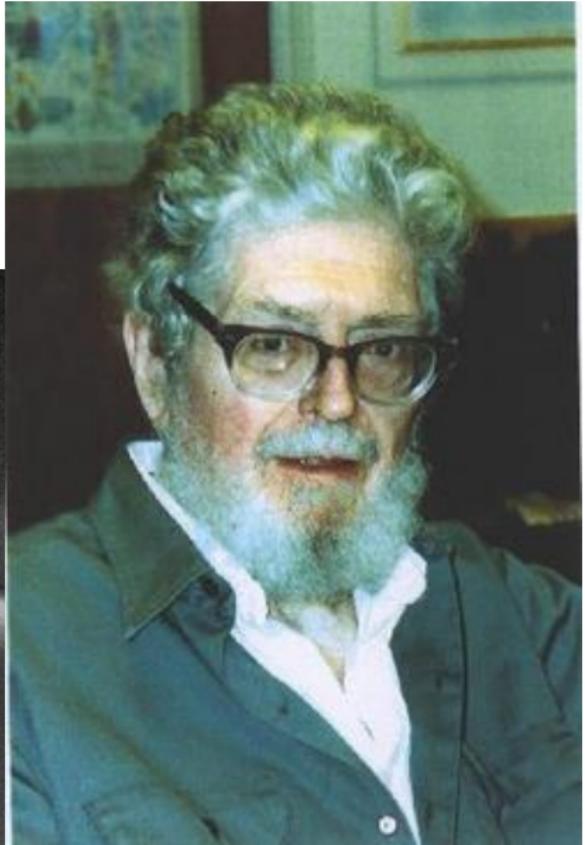
---

- 1: **Input** Data  $D = \{(x_1, y_1), \dots, (x_T, y_T)\}$ ,  $k \geq 1$ ,  $d : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}_+$ , new point  $x \in \mathcal{X}$
  - 2:  $D = \text{Sort}(D, d)$  % Sort  $D$  so that  $d(x, x_i) \leq d(x, x_{i+1})$ .
  - 3:  $p_y = \sum_{i=1}^k \mathbb{I}\{y_i = y\} / k$  for  $y \in \mathcal{Y}$ .
  - 4: **Return**  $p \triangleq (p_1, \dots, p_k)$
- 

## Algorithm parameters

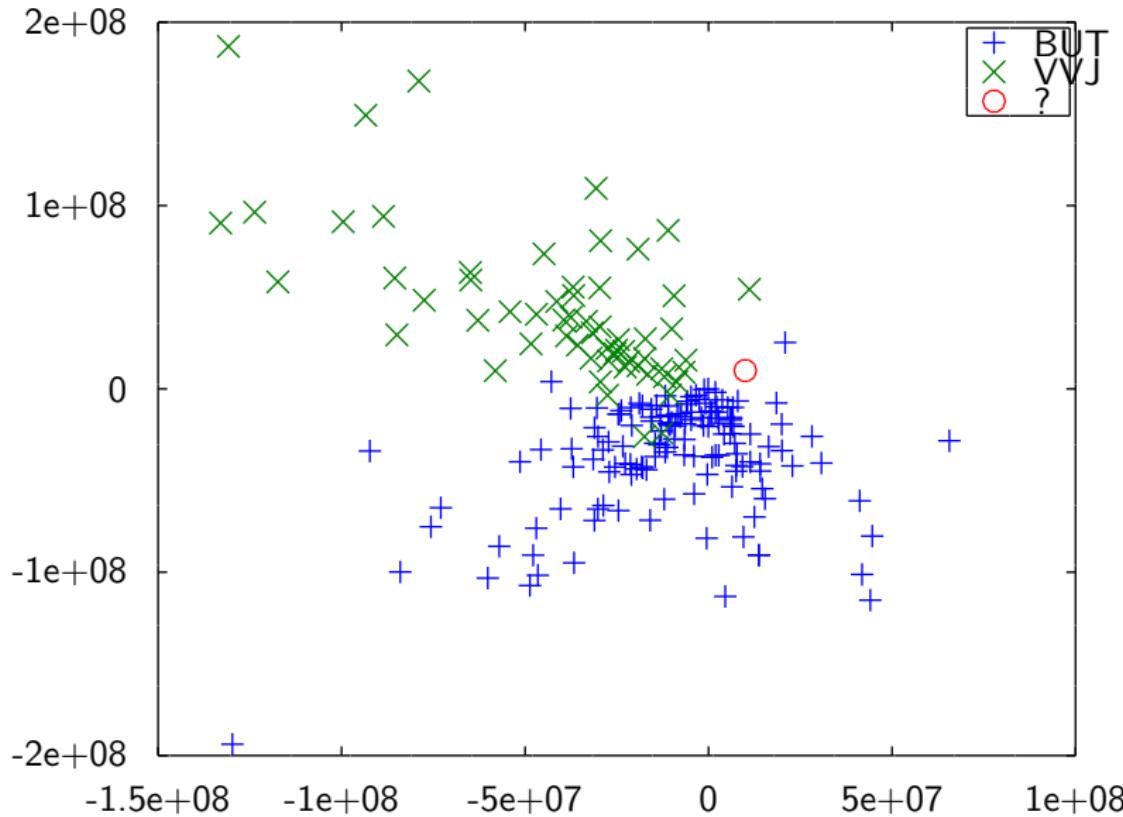
- Neighbourhood  $k \geq 1$ .
- Distance  $d : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}_+$ .

What does the algorithm output when  $k = T$ ?

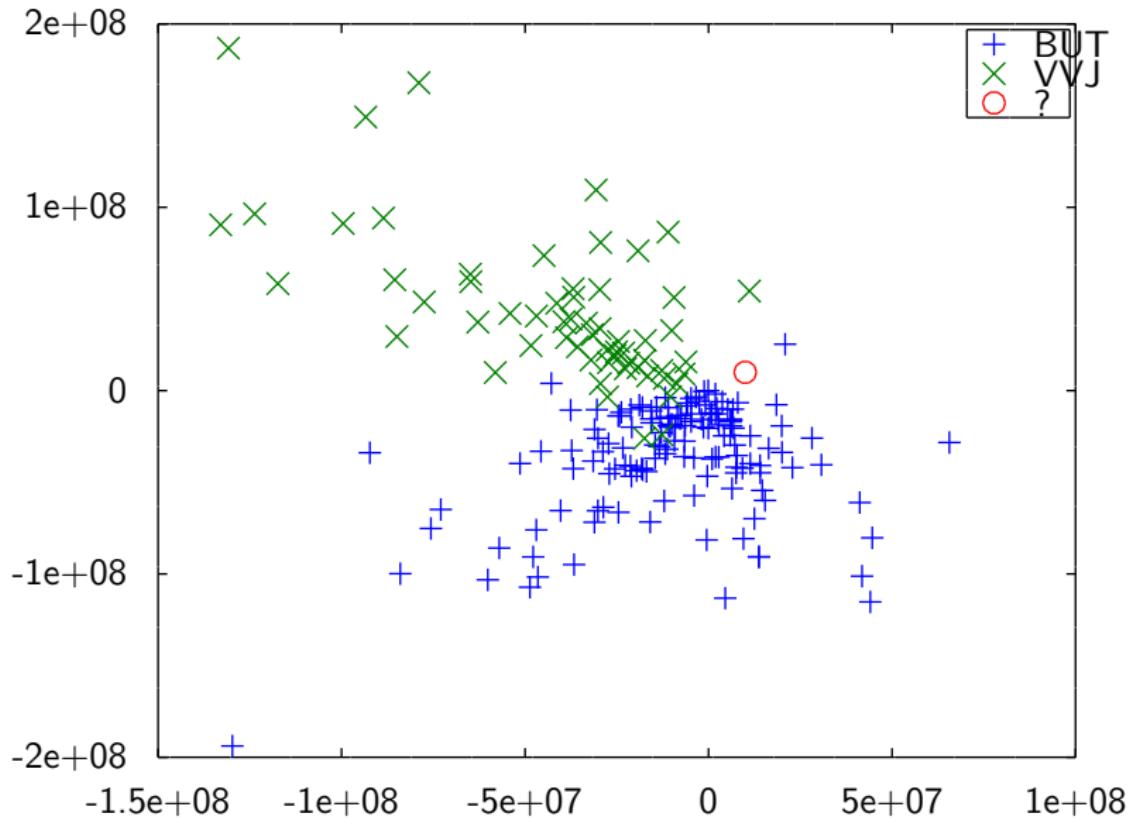


**Figure:** The nearest neighbours algorithm was introduced by Fix and Hodges Jr<sup>3</sup>, who also proved consistency properties.

## Nearest neighbour: What type is the new bacterium?



## Nearest neighbour: What type is the new bacterium?



What if it's a completely different strain?

## Separating the model from the classification policy

- The k-NN algorithm returns a model giving class probabilities for new data points.
- Deciding a class given the model

$$\pi(a | x) = \mathbb{I}\{p_a \geq p_y \forall y\}, \quad p = \text{k-NN}(D, k, d, x)$$

# Hands on with Python console

- `src/decision-problems/knn-classify.py`
- `src/decision-problems/KNN.ipynb`

## Discussion: Shortcomings of $k$ -nearest neighbour

- Choice of  $k$
- Choice of metric  $d$ .
- Representation of uncertainty.
- Scaling with large amounts of data.
- Meaning of label probabilities.

# Learning outcomes

## Understanding

- How kNN works
- The effect of hyperparameters  $k, d$  for nearest neighbour.
- The use of kNN to classify new data.

## Skills

- Use a standard kNN class in python
- Optimise kNN hyperparameters in an unbiased manner.
- Calculate probabilities of class labels using kNN.

## Reflection

- When is kNN a good model?
- How can we deal with large amounts of data?
- How can we best represent uncertainty?

## 1 Introduction to machine learning

## 2 Nearest neighbours

## 3 Reproducibility

- The human as an algorithm
- Algorithmic sensitivity
- Beyond the data you have: simulation and replication

## Computational reproducibility: Can the study be repeated?

Can we, from the available information and data, exactly reproduce the reported methods and results?

- jupyter notebooks
- svn, git or mercurial version control systems

## Scientific reproducibility: Is the conclusion correct?

Can we, from the available information and a **new** set of data, reproduce the conclusions of the original study?

When publishing results about a **new method**, computational reproducibility is essential for scientific reproducibility.

# RealClear Politics

[Polls](#) [Election 2018](#) [Video](#) [Writers](#) [More](#)

Poll	Date	Sample	MoE	Clinton (D)	Trump (R)	Spread
<b>Final Results</b>	--	--	--	48.2	46.1	<b>Clinton +2.1</b>
<b>RCP Average</b>	<b>11/1 - 11/7</b>	--	--	<b>46.8</b>	<b>43.6</b>	<b>Clinton +3.2</b>
Bloomberg	11/4 - 11/6	799 LV	3.5	46	43	Clinton +3
IBD/TIPP Tracking	11/4 - 11/7	1107 LV	3.1	43	42	Clinton +1
Economist/YouGov	11/4 - 11/7	3669 LV	-	49	45	Clinton +4
LA Times/USC Tracking	11/1 - 11/7	2935 LV	4.5	44	47	Trump +3
ABC/Wash Post Tracking	11/3 - 11/6	2220 LV	2.5	49	46	Clinton +3
FOX News	11/3 - 11/6	1295 LV	2.5	48	44	Clinton +4
Monmouth	11/3 - 11/6	748 LV	3.6	50	44	Clinton +6
NBC News/Wall St. Jrnal	11/3 - 11/5	1282 LV	2.7	48	43	Clinton +5
CBS News	11/2 - 11/6	1426 LV	3.0	47	43	Clinton +4
Reuters/Ipsos	11/2 - 11/6	2196 LV	2.3	44	39	Clinton +5

All General Election: Trump vs. Clinton Polling Data



RCP POLL AVERAGE

General Election: Trump vs.  
Clinton

46.8	Clinton (D) +3.2
43.6	Trump (R)



## The principle of independent evaluation

Data used for estimation cannot be used for evaluation.

## Data Collection



**Figure:** The decision process in classification.



Figure: The decision process in classification.



Algorithm, hyperparameters

**Figure:** The decision process in classification.

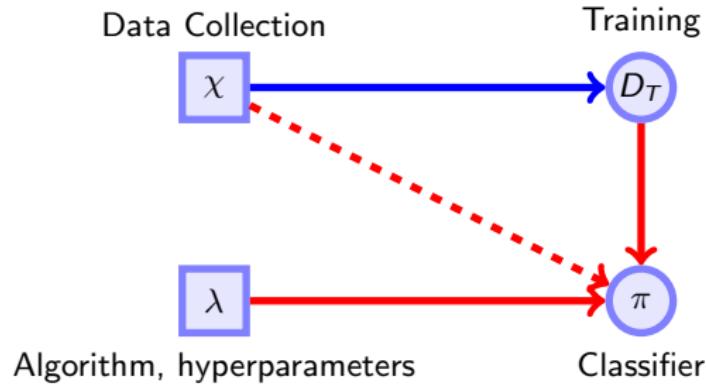


Figure: The decision process in classification.

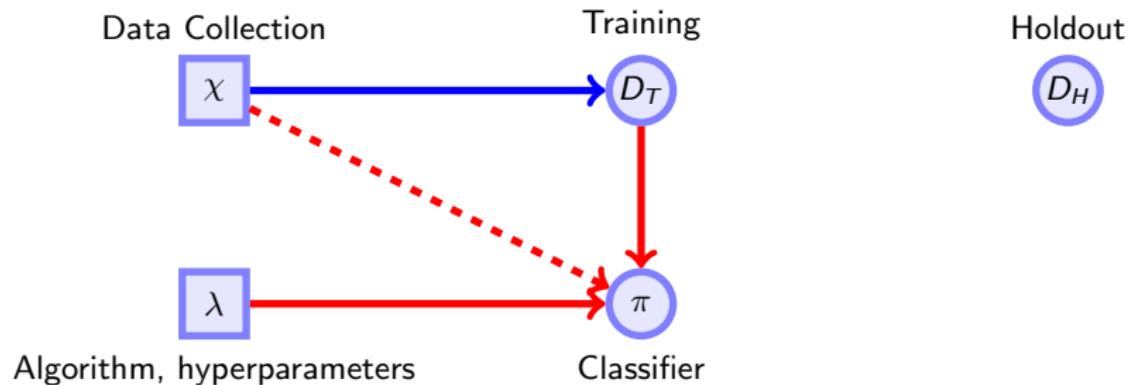


Figure: The decision process in classification.

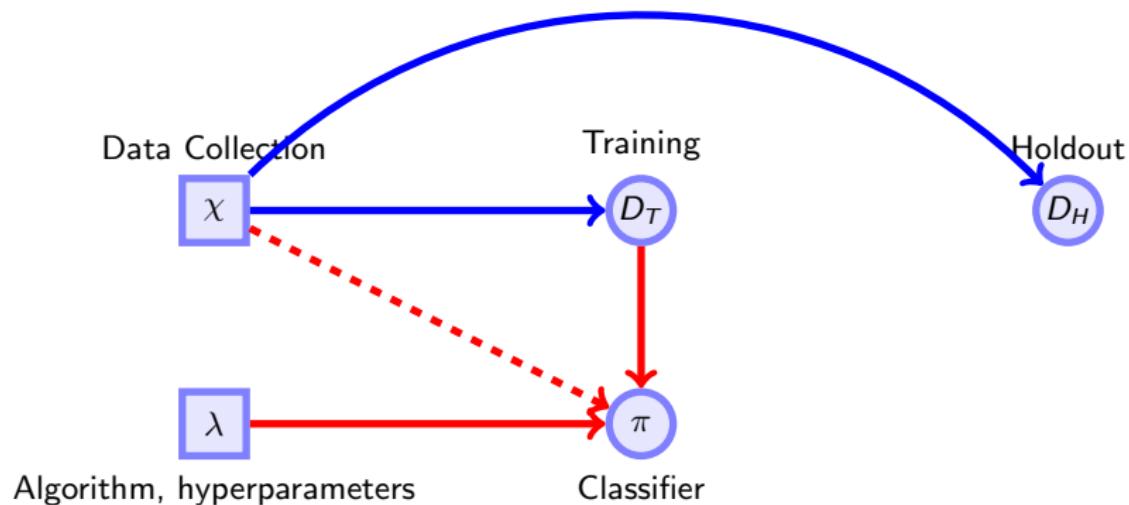


Figure: The decision process in classification.

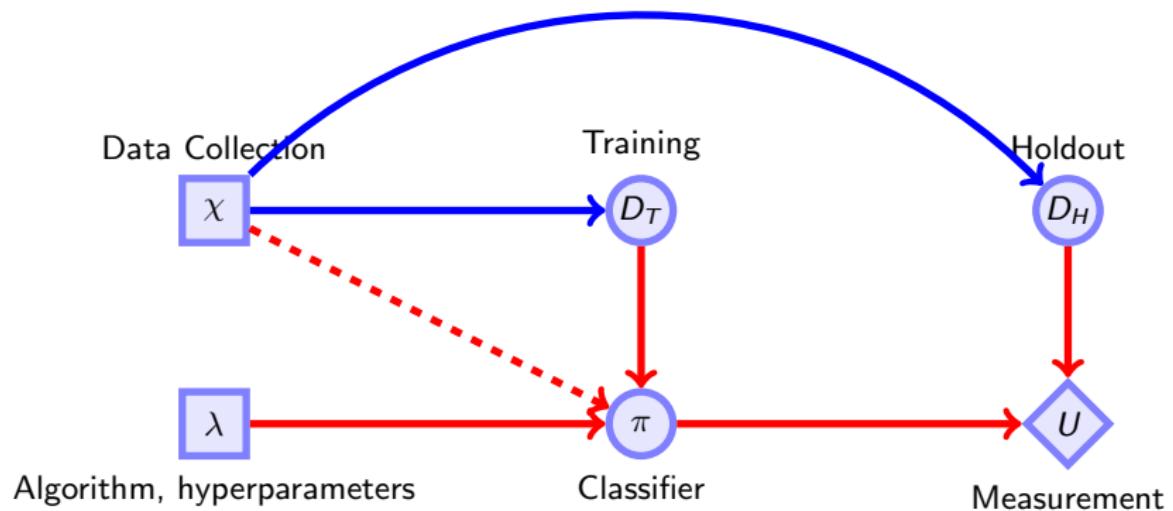
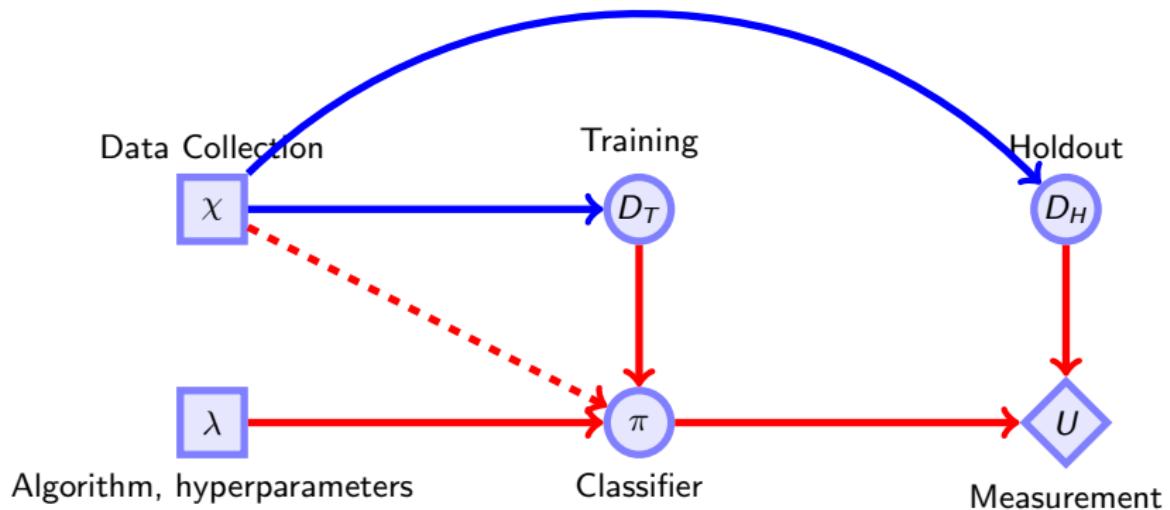


Figure: The decision process in classification.



**Figure:** The decision process in classification.

## Classification accuracy

$$\mathbb{E}_x[U(\pi)] = \sum_{x,y} \underbrace{\mathbb{P}_x(x,y)}_{\text{Data probability}} \overbrace{\pi(a=y | x)}^{\text{Decision probability}}$$

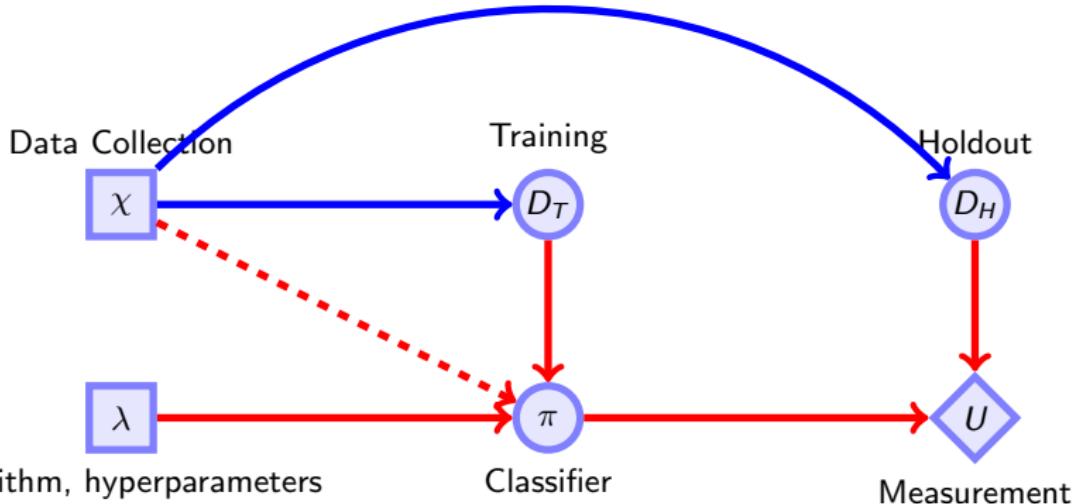


Figure: The decision process in classification.

## Classification accuracy

$$\mathbb{E}_{D_H} U(\pi) = \sum_{(x,y) \in D_H} \pi(a = y | x) / |D_H|.$$

# The human as an algorithm.

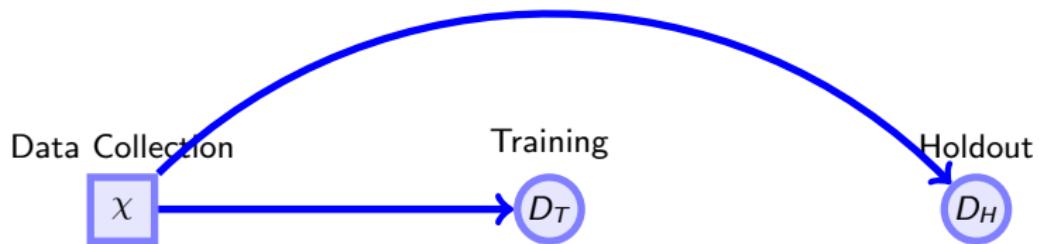
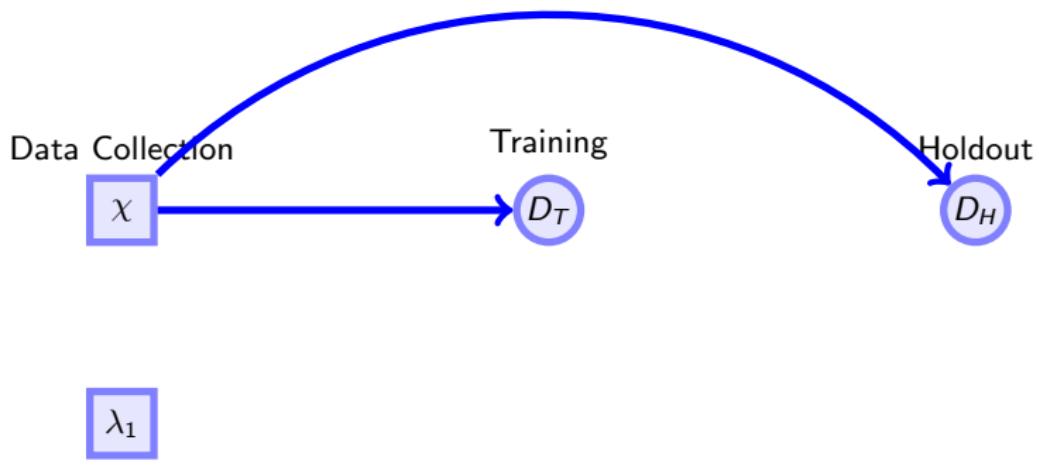


Figure: Selecting algorithms and hyperparameters through holdouts

# The human as an algorithm.



Algorithm, hyperparameters

Figure: Selecting algorithms and hyperparameters through holdouts

# The human as an algorithm.

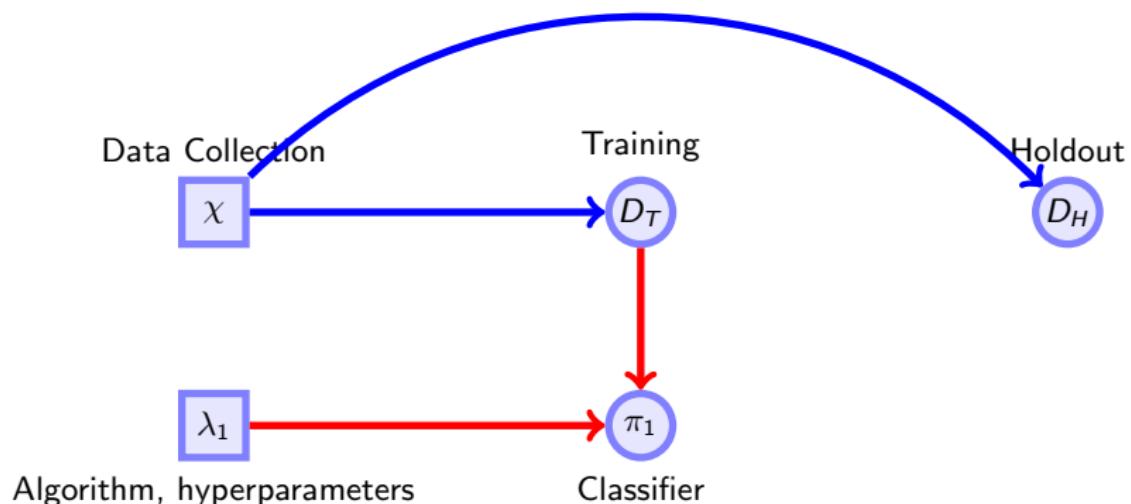


Figure: Selecting algorithms and hyperparameters through holdouts

# The human as an algorithm.

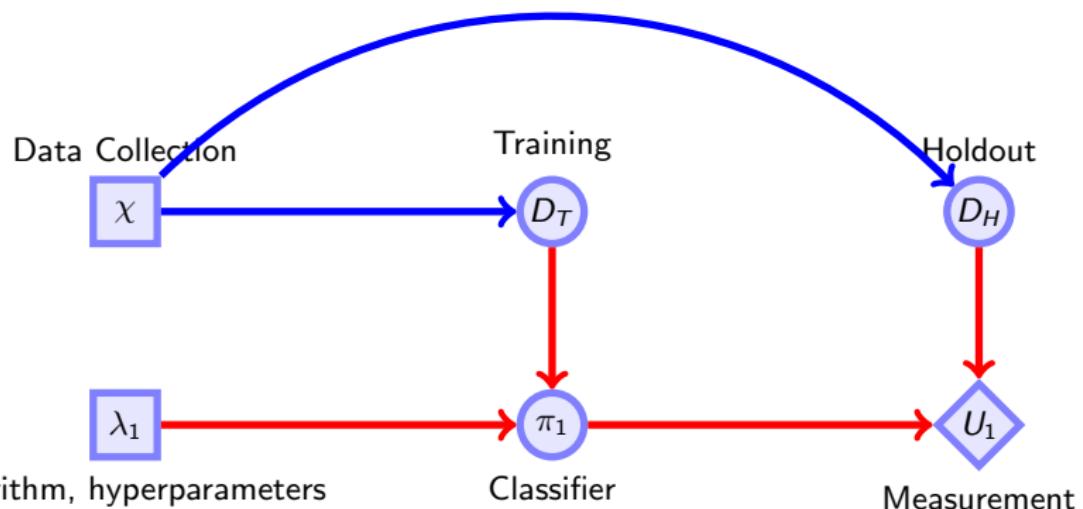


Figure: Selecting algorithms and hyperparameters through holdouts

# The human as an algorithm.

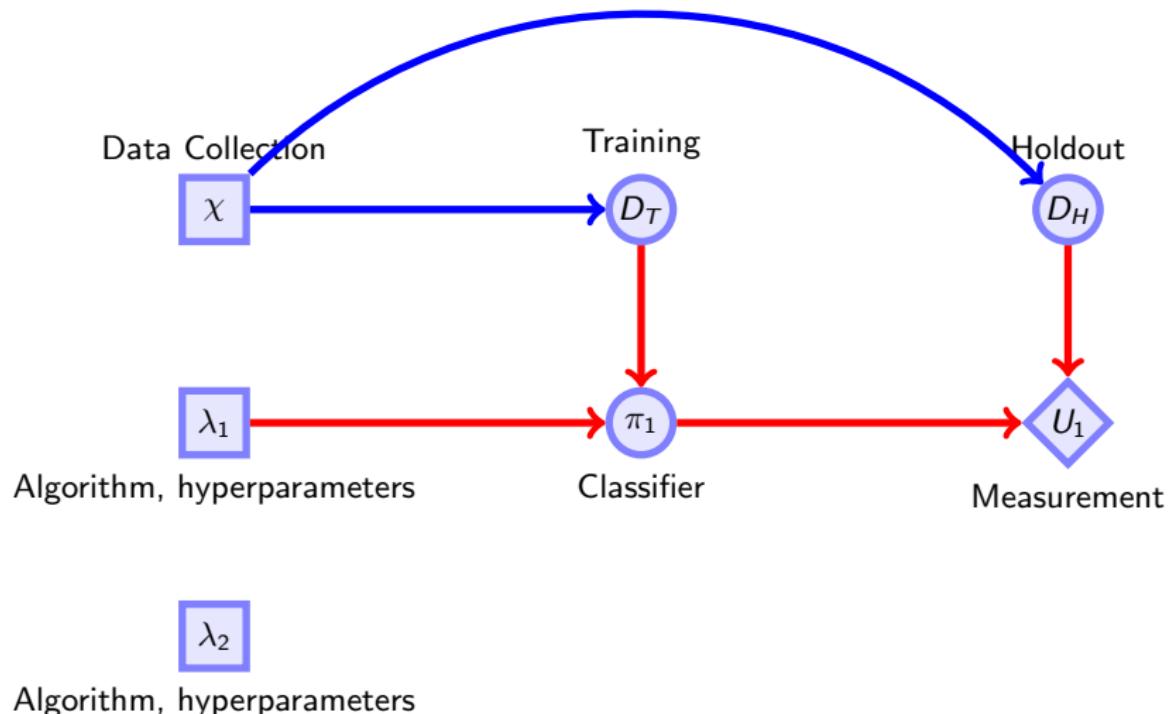


Figure: Selecting algorithms and hyperparameters through holdouts

# The human as an algorithm.

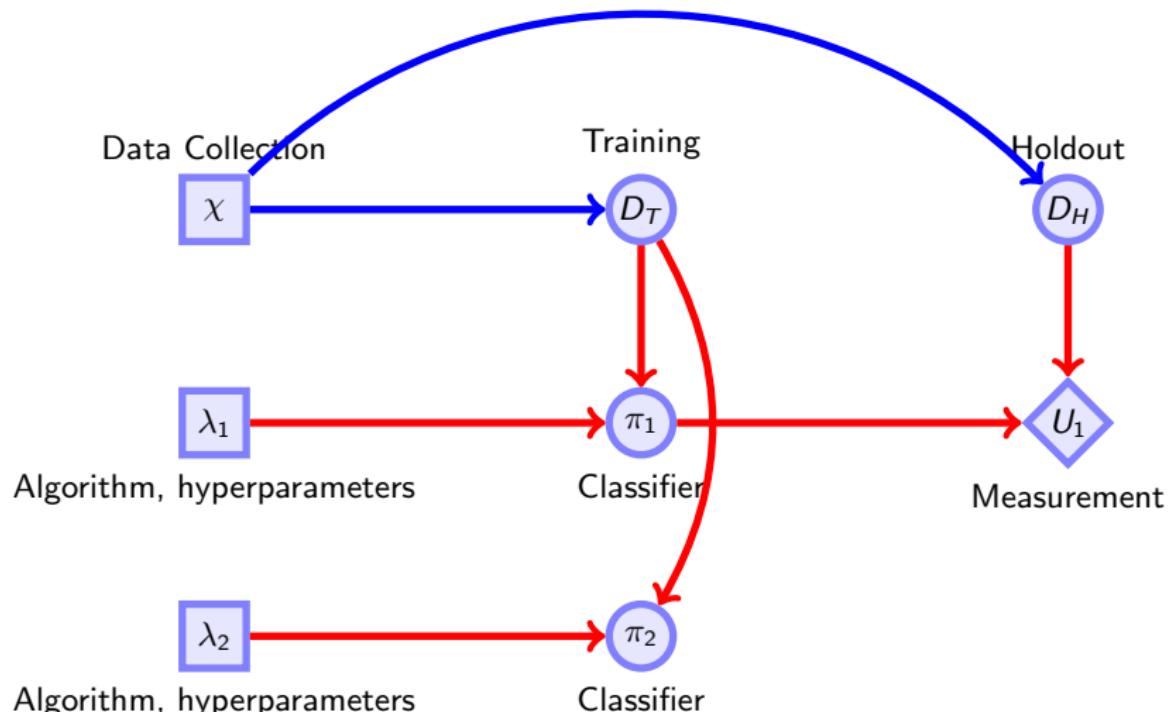


Figure: Selecting algorithms and hyperparameters through holdouts

# The human as an algorithm.

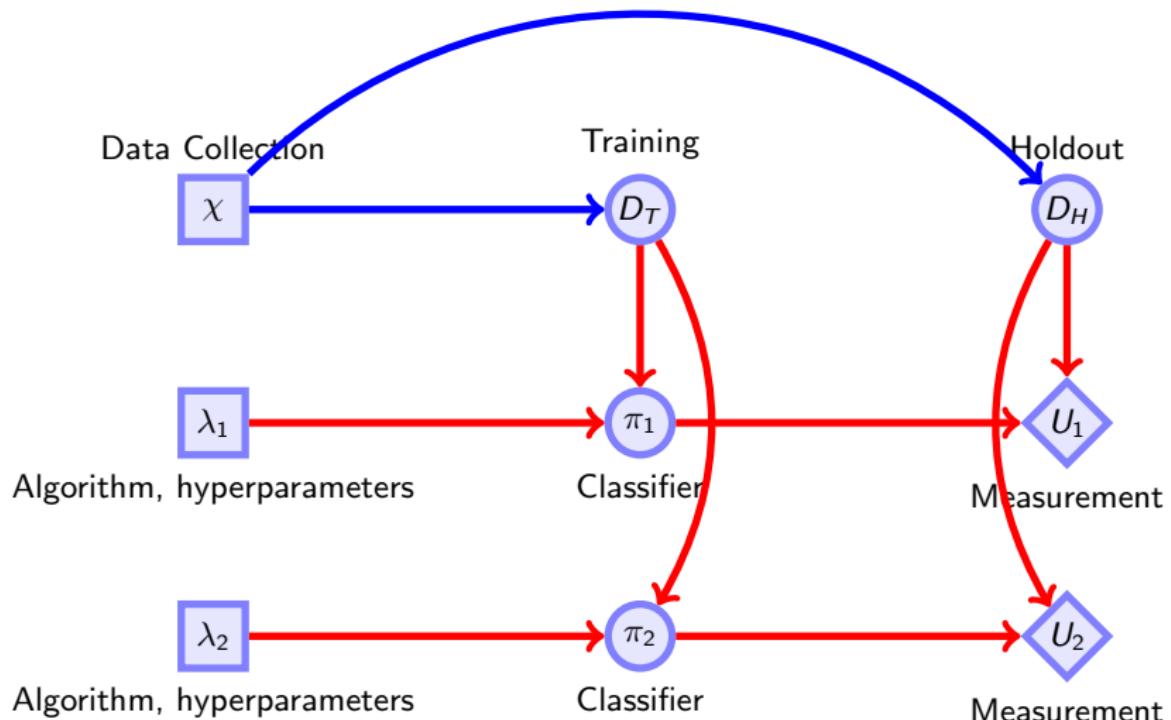


Figure: Selecting algorithms and hyperparameters through holdouts

## Holdout sets

- Original data  $D$ , e.g.  $D = (x_1, \dots, x_T)$ .
- Training data  $D_T \subset D$ , e.g.  $D_T = x_1, \dots, x_n$ ,  $n < T$ .
- Holdout data  $D_H = D \setminus D_T$ , used to measure the quality of the result.
- Algorithm  $\lambda$  with hyperparameters  $\phi$ .
- Get algorithm output  $\pi = \lambda(D_T, \phi)$ .
- Calculate quality of output  $U(\pi, D_H)$

## Holdout and test sets for unbiased algorithm comparison

### Algorithm 2 Unbiased adaptive evaluation through data partitioning

Partition data into  $D_T, D_H, D^*$ .

**for**  $\lambda \in \Lambda$  **do**

**for**  $\phi \in \Phi_\lambda$  **do**

$\pi_{\phi, \lambda} = \lambda(D_T, \phi)$ .

**end for**

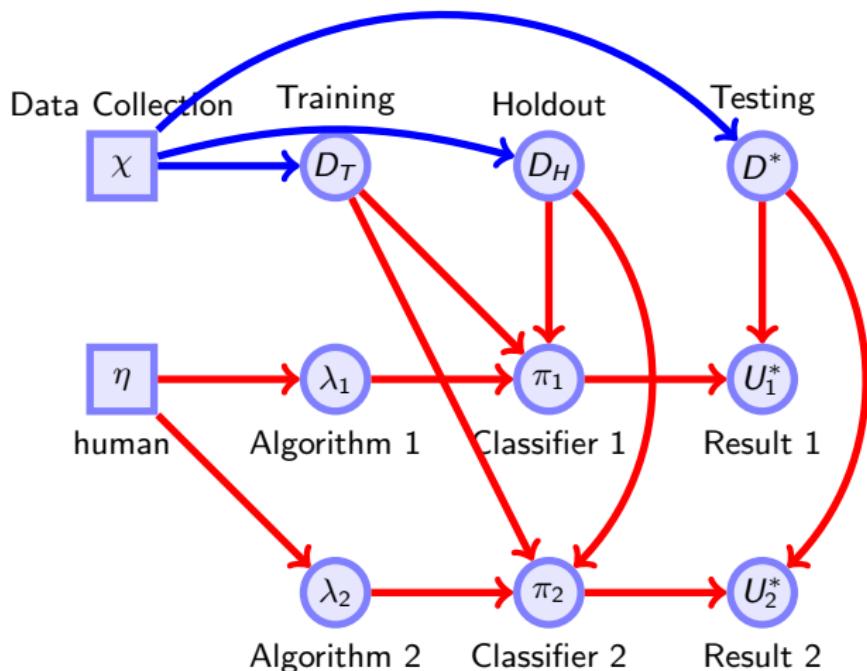
    Get  $\pi_\lambda^*$  maximising  $U(\pi_{\phi, \lambda}, D_H)$ .

$u_\lambda = U(\pi_\lambda^*, D^*)$ .

**end for**

$\lambda^* = \arg \max_\lambda u_\lambda$ .

## Final performance measurement



# Independent data sets

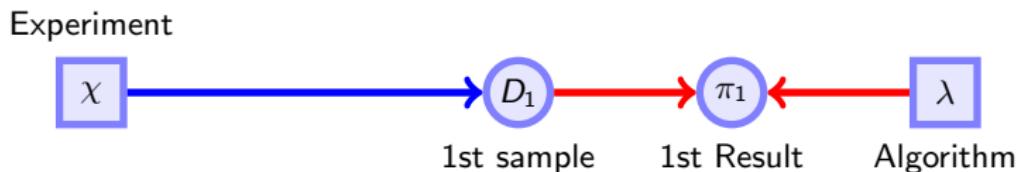


Figure: Multiple samples

## Independent data sets

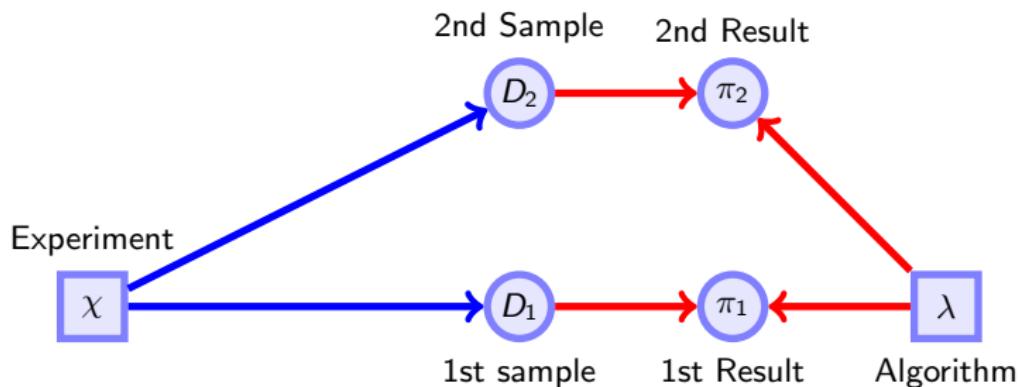


Figure: Multiple samples

## Bootstrap samples

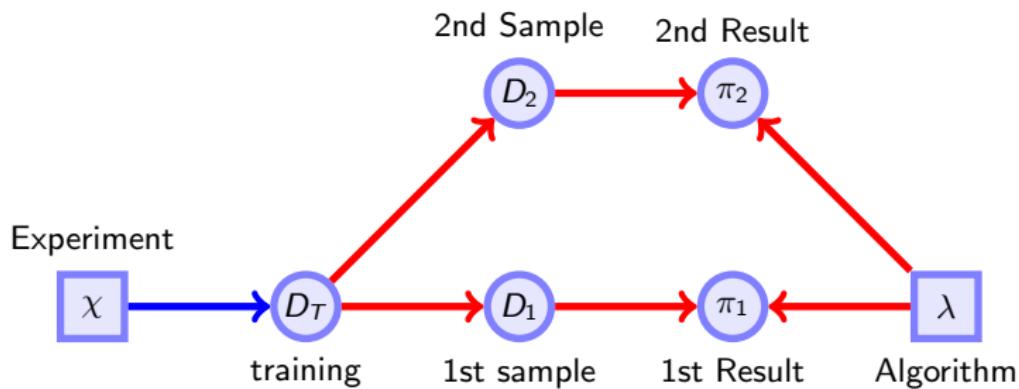


Figure: Bootstrap replicates of a single sample

## Bootstrapping

Bootstrapping is a general technique that can be used to:

- Estimate the sensitivity of  $\lambda$  to the data  $x$ .
- Obtain a distribution of estimates  $\pi$  from  $\lambda$  and the data  $x$ .
- When estimating the performance of an algorithm on a small dataset  $D^*$ , use bootstrap samples of  $D^*$ .

## Bootstrapping

- 1 **Input** Training data  $D$ , number of samples  $k$ .
- 2 **For**  $i = 1, \dots, k$
- 3      $D^{(i)} = \text{Bootstrap}(D)$
- 4 **return**  $\{D^{(i)} \mid i = 1, \dots, k\}$ .

where  $\text{Bootstrap}(D)$  samples with replacement  $|D|$  points from  $D_T$ .

# Cross-validation

## $k$ -fold Cross-Validation

- 1** **Input** Training data  $D_T$ , number of folds  $k$ , algorithm  $\lambda$ , measurement function  $U$
- 2** Create the partition  $D^{(1)}, \dots, D^{(k)}$  so that  $\bigcup_{i=1}^k D^{(k)} = D$ .
- 3** Define  $D_T^{(i)} = D \setminus D^{(i)}$
- 4**  $\pi_i = \lambda(D_T^{(i)})$
- 5** **For**  $i = 1, \dots, k$ :
- 6**     $\pi_i = \lambda(D^{(i)})$
- 7**     $u_i = U(\pi_i)$
- 8** **return**  $\{y_1, \dots, y_k\}$ .

# Simulation

## Steps for a simulation pre-study

- 1 Define a data-generating process as close to the original dataset as possible.
- 2 Collect data according to your protocol.
- 3 Run the intended analysis.
- 4 See if the results are reasonable, or if you need more power.

# Simulation

## Simulation study

- 1 Create a simulation that allows you to collect data similar to the real one.
- 2 Collect data from the simulation and analyse it according to your protocol.
- 3 If the results are not as expected, alter the protocol or the simulation. In which cases do you get good results?
- 4 Finally, use the best-performing method as the protocol.

# Independent replication

## Replication study

- 1 Reinterpret the original hypothesis and experiment.
- 2 Collect data according to the original protocol, **unless flawed**.
- 3 Run the analysis again, **unless flawed**.
- 4 See if the conclusions are in agreement.

## Learning outcomes

### Understanding

- What is a hold-out set, cross-validation and bootstrapping.
- The idea of not reusing data input to an algorithm to evaluate it.
- The fact that algorithms can be implemented by both humans and machines.

### Skills

- Use git and notebooks to document your work.
- Use hold-out sets or cross-validation to compare parameters/algorithms in Python.
- Use bootstrapping to get estimates of uncertainty in Python.

### Reflection

- What is a good use case for cross-validation over hold-out sets?
- When is it a good idea to use bootstrapping?
- How can we use the above techniques to avoid the false discovery problem?
- Can these techniques fully replace independent replication?

- [1] Craig M. Bennett, George L. Wolford, and Michael B. Miller. The principled control of false positives in neuroimaging. *Social cognitive and affective neuroscience*, 4(4):417–22, 2009. URL <https://pdfs.semanticscholar.org/19c3/d8b67564d0e287a43b1e7e0f496eb1e8a945.pdf>.
- [2] Craig M Bennett, Abigail A Baird, Michael B Miller, and George L Wolford. Journal of serendipitous and unexpected results. *Journal of Serendipitous and Unexpected Results (jsur.org)-Vol. 1(1):1–5*, 2012. URL <https://teenspecies.github.io/pdfs/NeuralCorrelates.pdf>.
- [3] Evelyn Fix and Joseph L Hodges Jr. Discriminatory analysis-nonparametric discrimination: consistency properties. Technical report, California Univ Berkeley, 1951.