BUDAPEST DEEP LEARNING READING SEMINAR

‘WHAT TO DO IF WE DON’T HAVE ENOUGH DATA?’

INTRO

LEVENTE SZABADOS

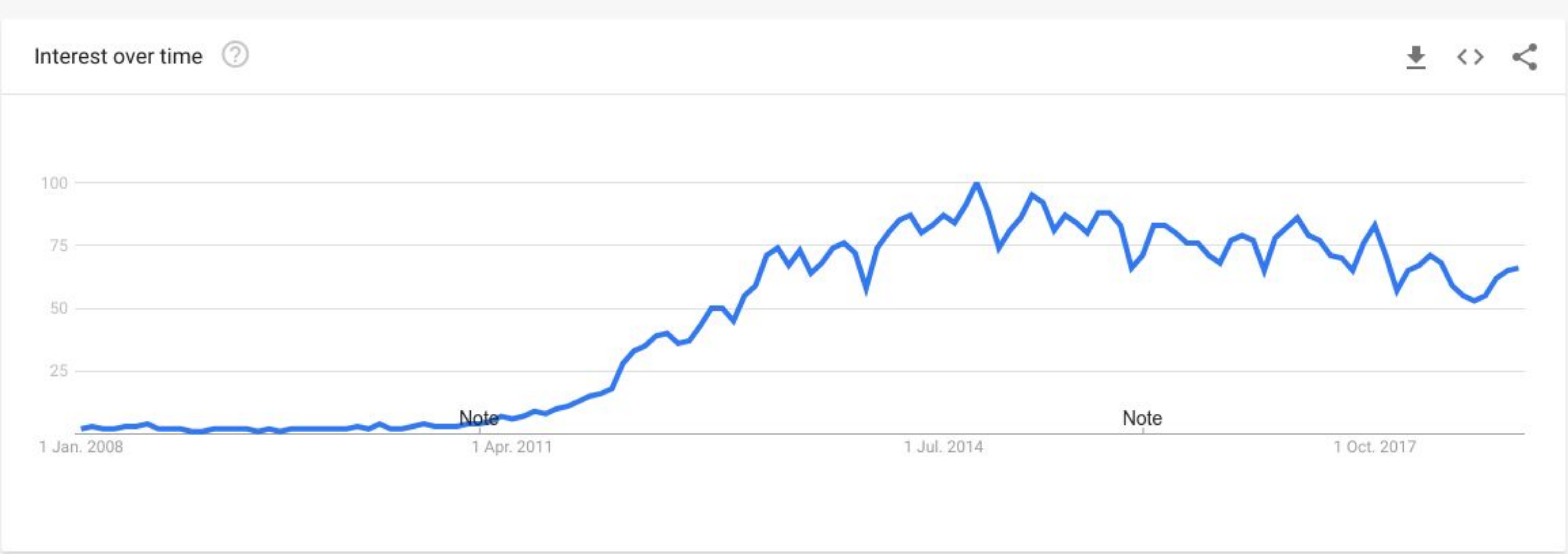
"...originally Buddhist theologian and programmer, senior AI professional, lead of research, lecturer, startupper, ex-CTO

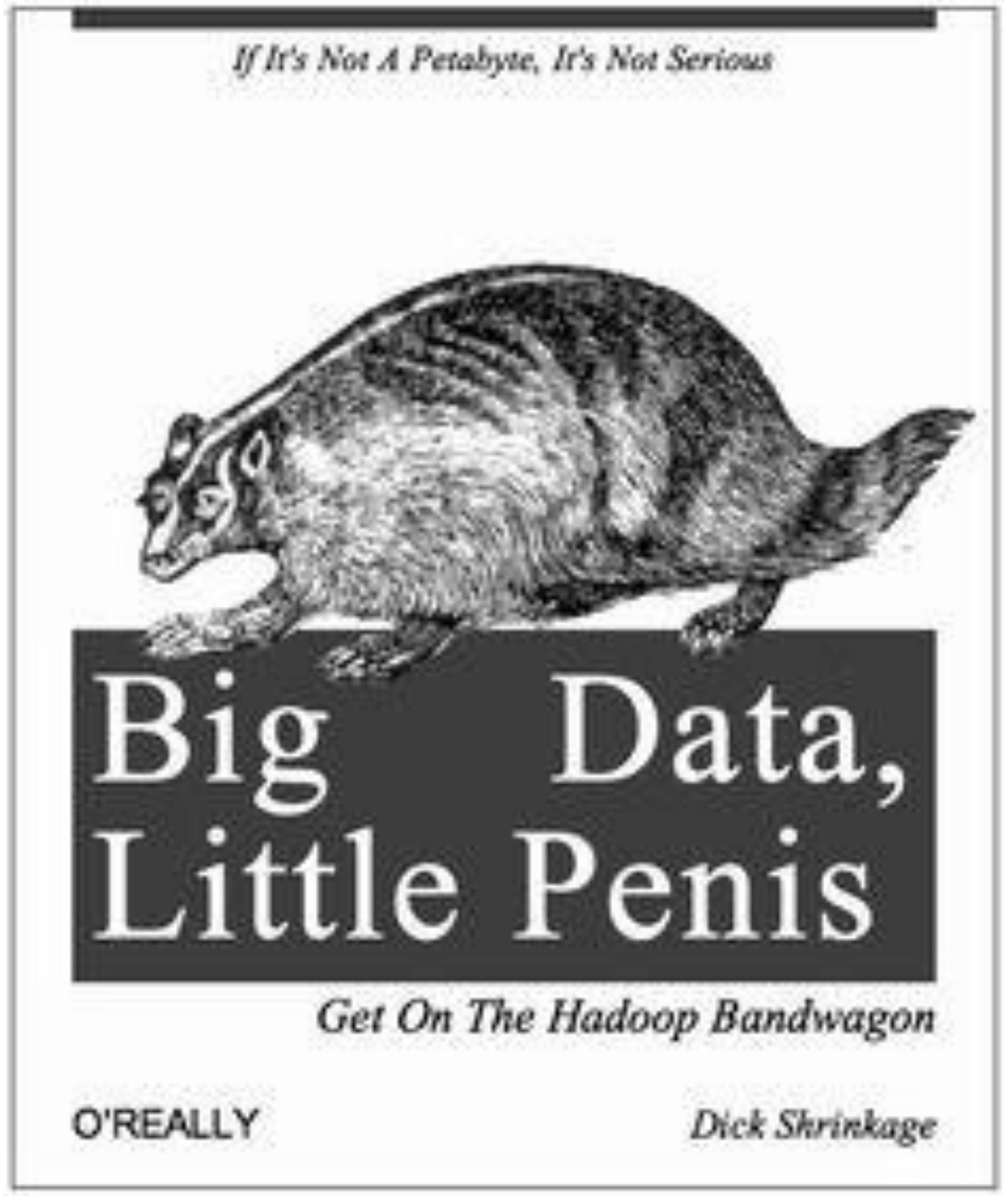
Presently:
Lecturer: Frankfurt School of Finance and Management,
Specialization leader: KÜRT Academy,
Senior Consultant: AI Partners,
Chief organizer: Budapest.AI."

CONTACT

"BIT OVERHYPED?"

BIG DATA IS - NOT EVERYTHING - IS BIG DATA!





DETAILS:
Google trends - Big Data topic 2008-2018
Chris Stucchio: "Don't use HADOOP - Your data isn't that big"

‘NOT ENOGH INFORMATION“

SMALL DATASETS VIOLATE BASIC ASSUMPTIONS

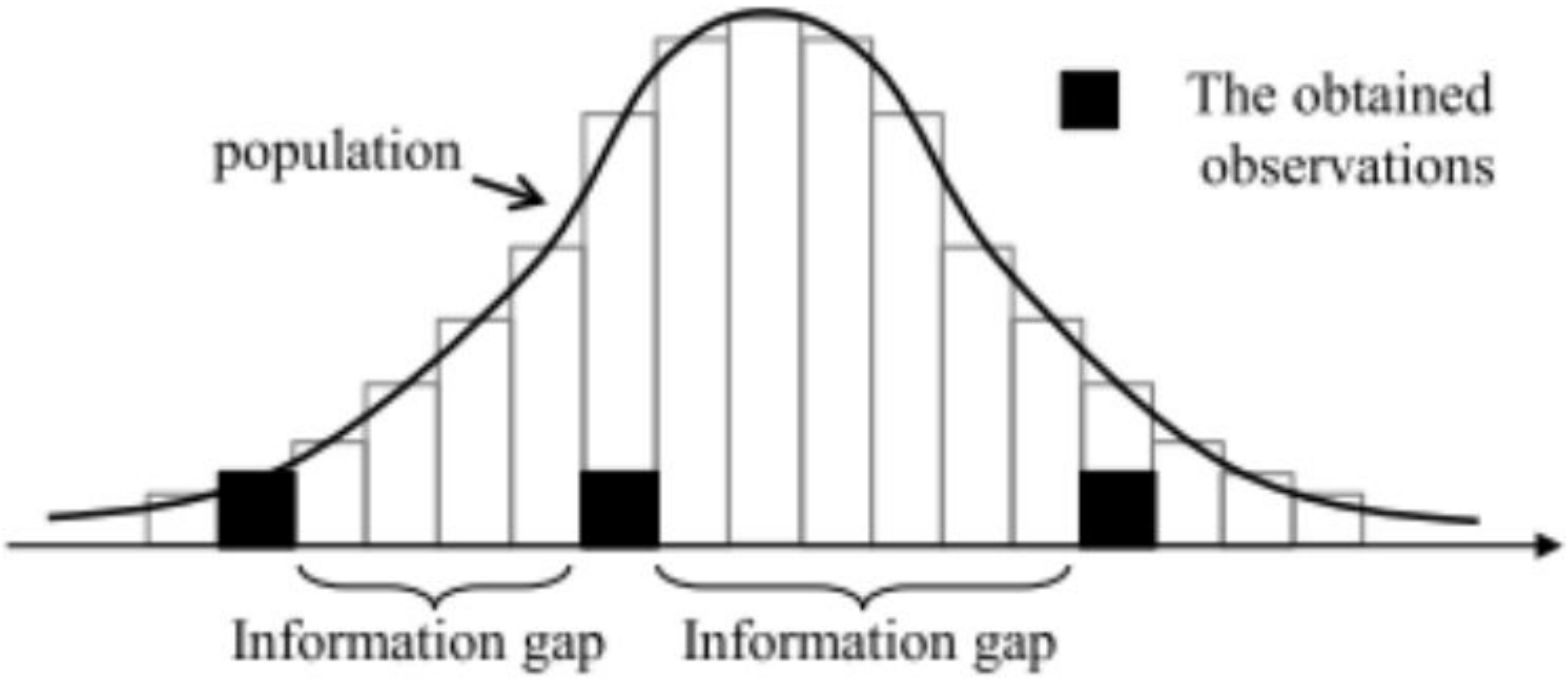
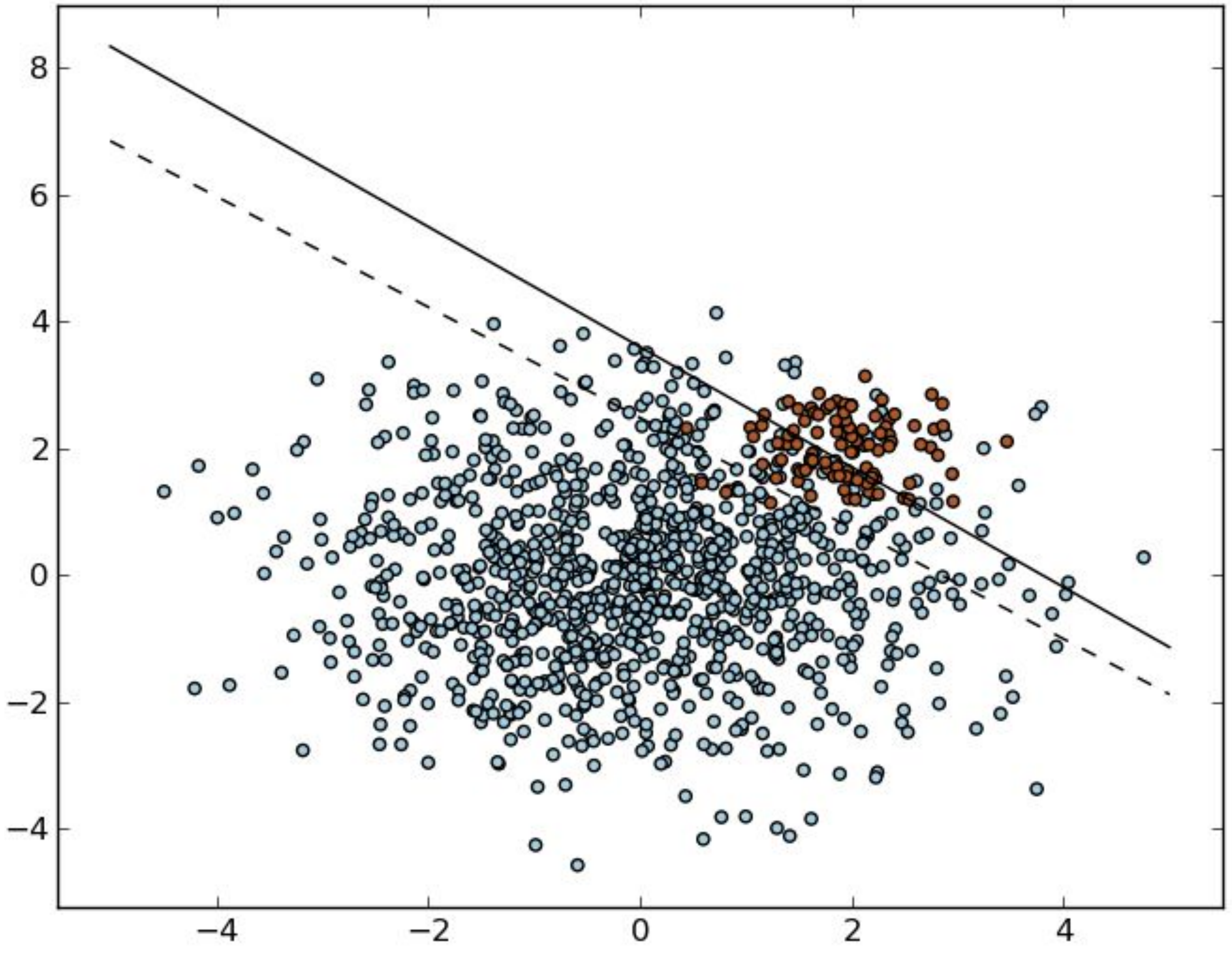


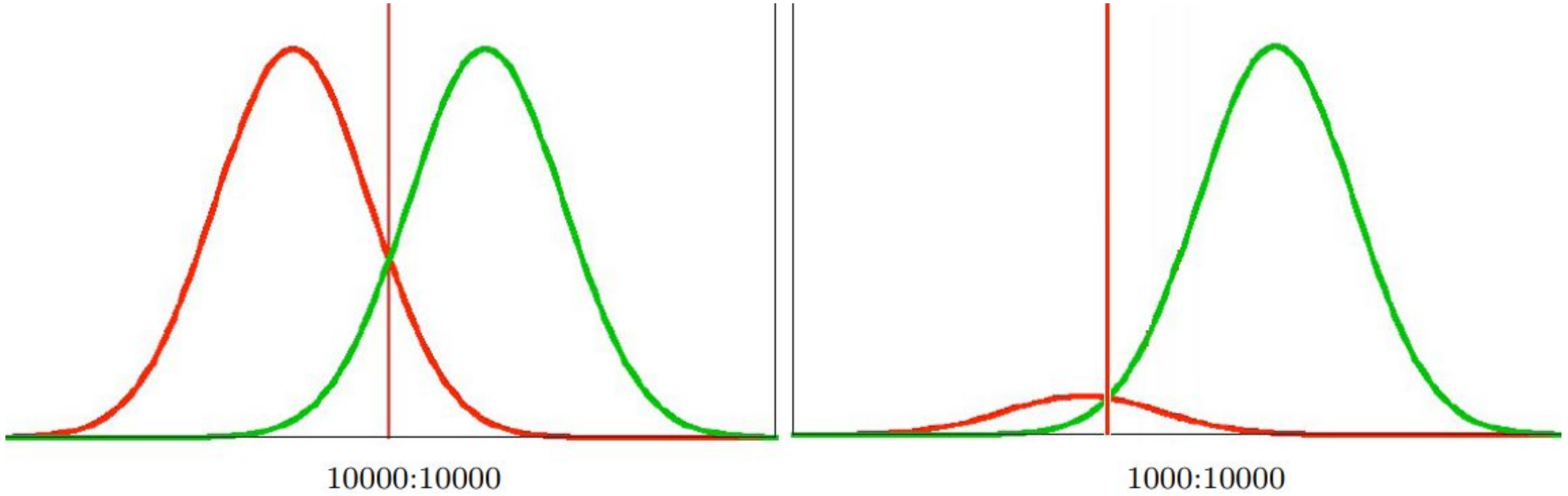
Figure 2. The distribution of a small dataset relative to its population [6]

source:
‘Handling a Small Dataset Problem in Prediction Model by employ Artificial Data Generation Approach: A Review’

"THE BIAS NAMED CLASS IMBALANCE"

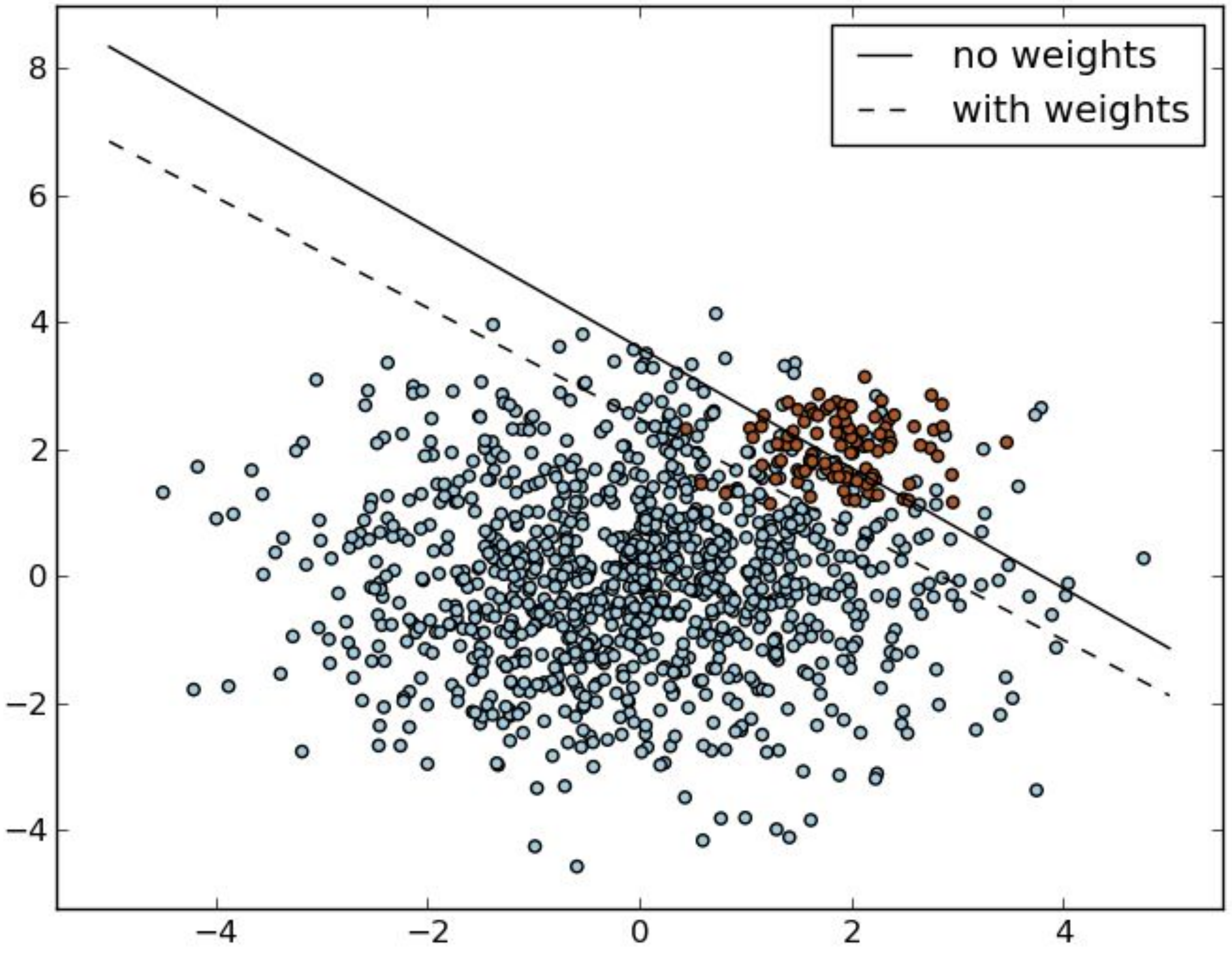
CASE I. - WE DON'T HAVE ENOUGH OF *ONE* THING





source:
"Classification in imbalanced datasets"

SOLUTION 1. - “COST SENSITIVE LEARNING”

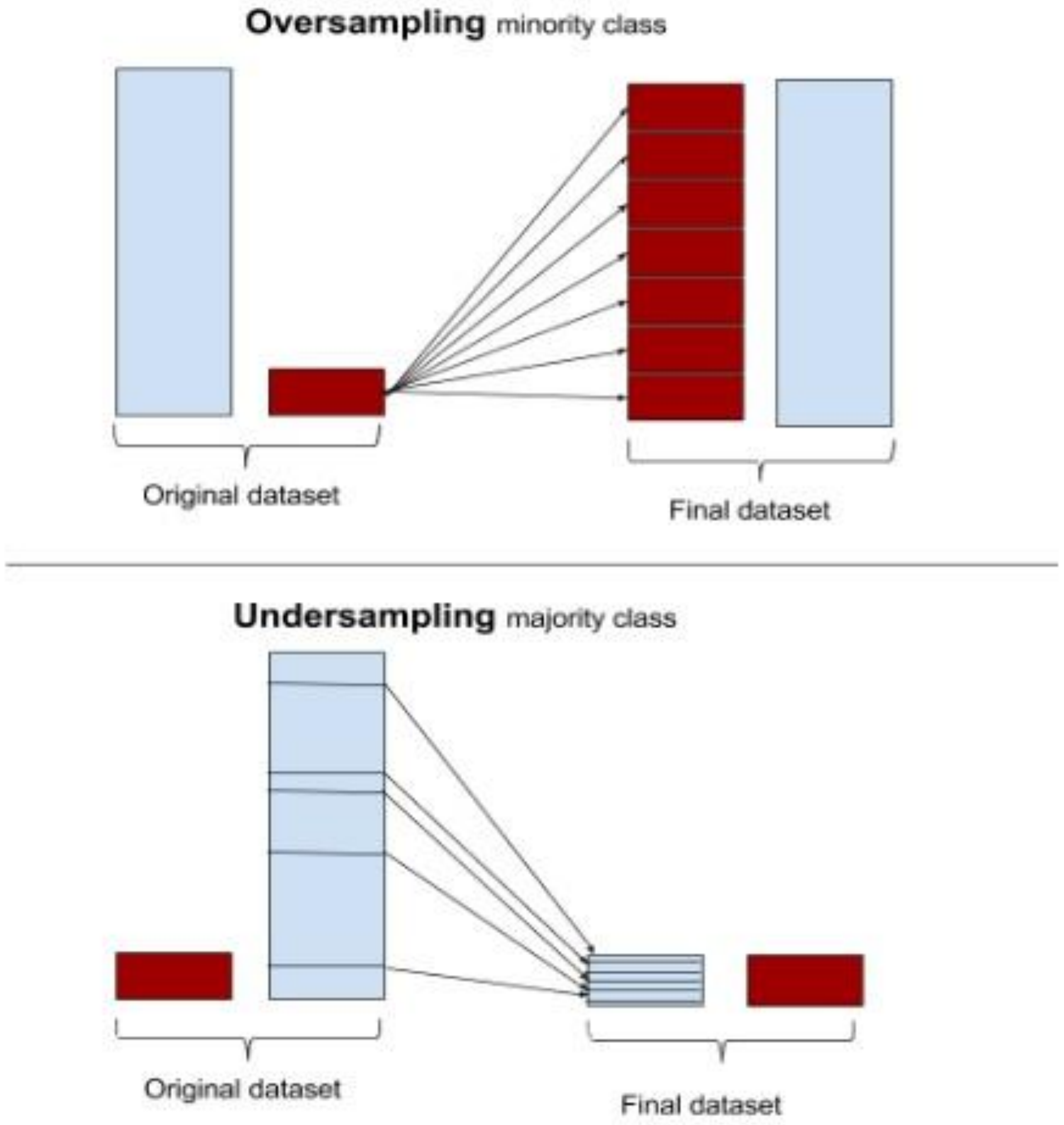


Predicted / Actual	Category-A	Category-B
Category-A	90	0
Category-B	10	0

We can try to **modify our objective** / cost calculation to accomodate the fact, that making an errőr on the minority class is a “more serious issue”.

source:
“Cost sensitive learning and the class imbalance problem”

SOLUTION 2. - “SAMPLING”



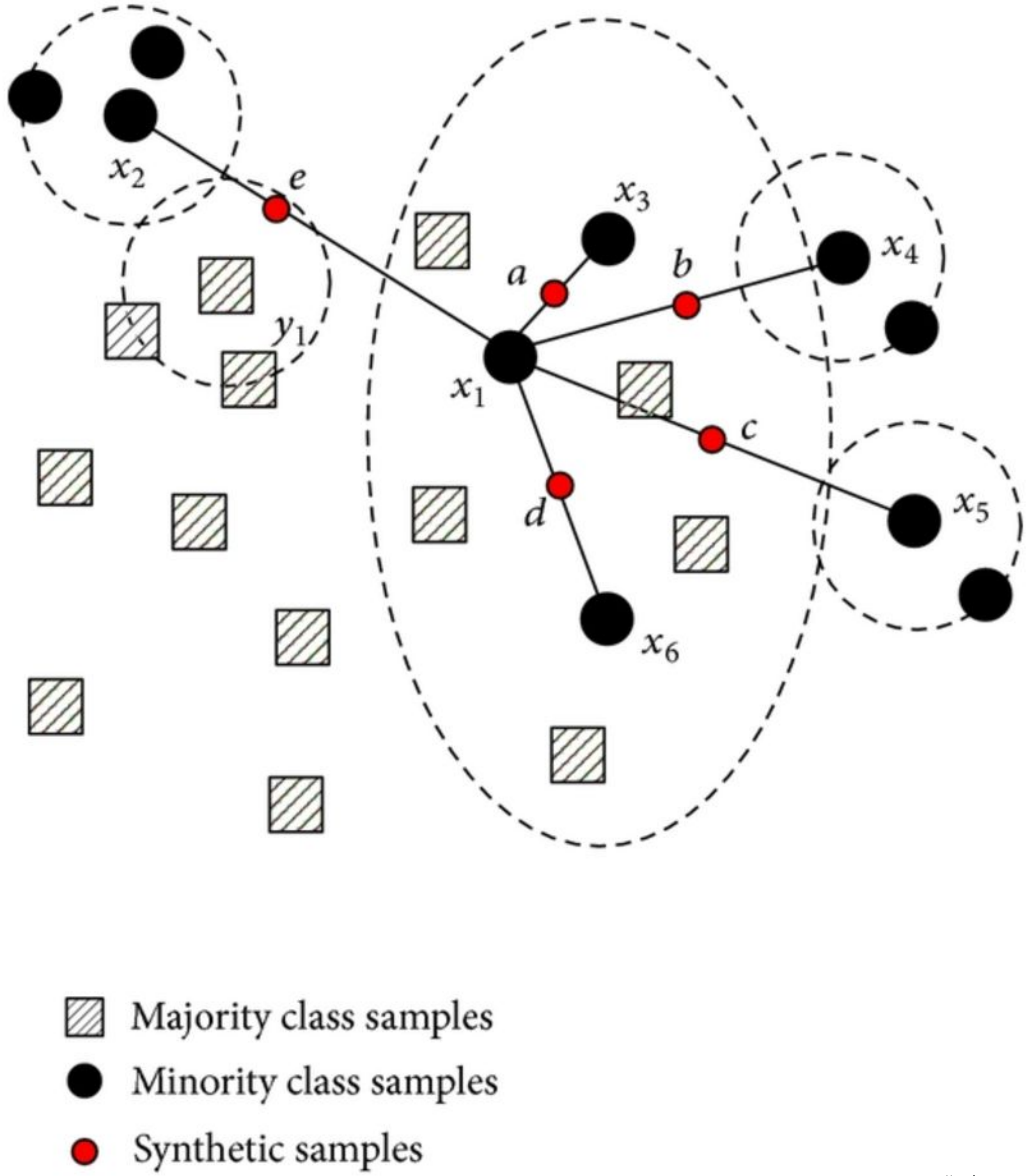
- **Oversampling**:
  - Repeatedly use some of the minority class datapoints
  - Good question is: Which ones?
    - Can we be more intelligent then random choice?

- **Undersampling:**
  - Choose only some of the majority class datapoints
  - Reduces the overall dataset, **not recommended**

source:
Undersampling and oversampling questions

"IF YOU DON'T HAVE IT-CREATE IT""

SOLUTION 3. - DATA SYNTHESIS



- **Create new datapoints! (SMOTE)**

  "First it finds the n-nearest neighbors in the minority class for each of the samples in the class . Then it draws a line between the the neighbors an generates random points on the lines."

- **...and add some noise! (ADASYN)**

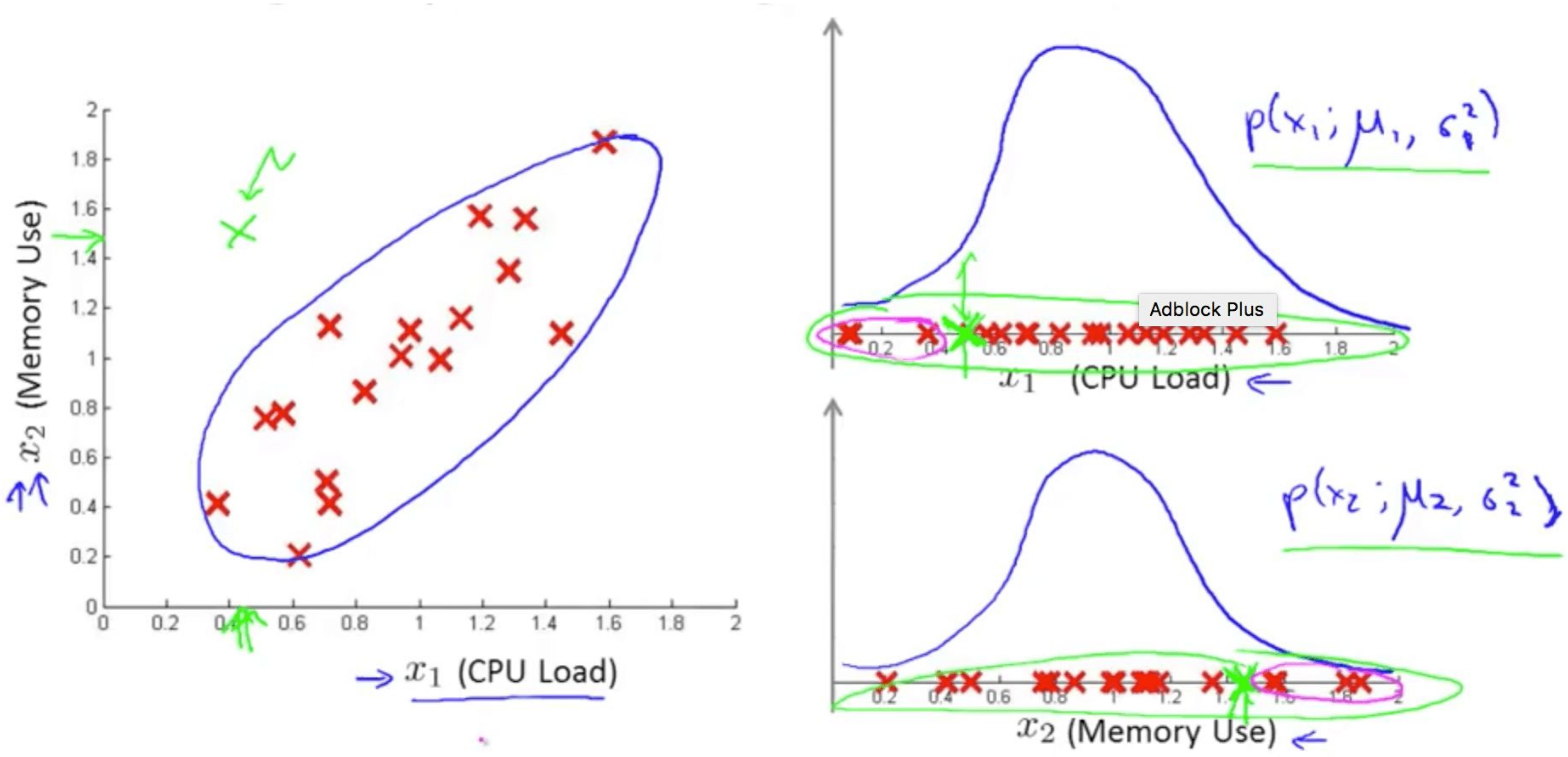
  "After creating those sample it adds a random small values to the points thus making it more realistic. In other words instead of all the sample being linearly correlated to the parent they have a little more variance in them i.e they are bit scattered."

- **...and use clusters! (Cluster Based Oversampling)**

source:
SMOTE and ADASYN
"Clustering and Learning from Imbalanced Data"

“WELL IF YOU LOOK IT THAT WAY“

SOLUTION 4. - RECAST PROBLEM!

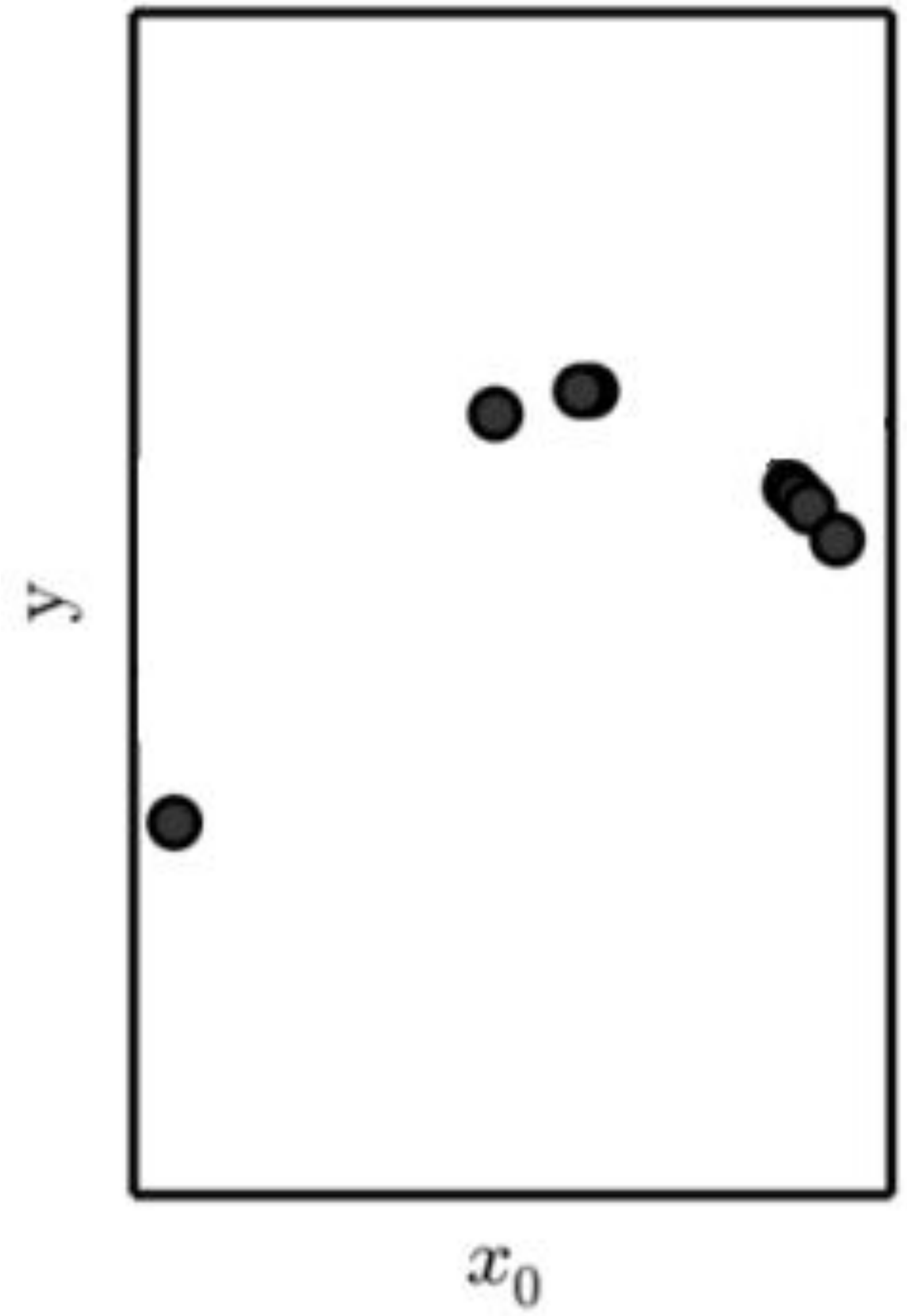


- If the minority class points are so rare, they can be considered “exceptions”, or **“anomalies”**
- There are tools for “one class” classification (eg.: “One class SVM” and “Isolation forests”)
- But if we basically get a good **probabilistic model** of the majority class distribution, we are done.
- This will lead us to **“representation learning”**

source:
Classification based outlier detection techniques
Anomaly Detection using the Multivariate Gaussian Distribution
Ritchie Ng: Anomaly detection

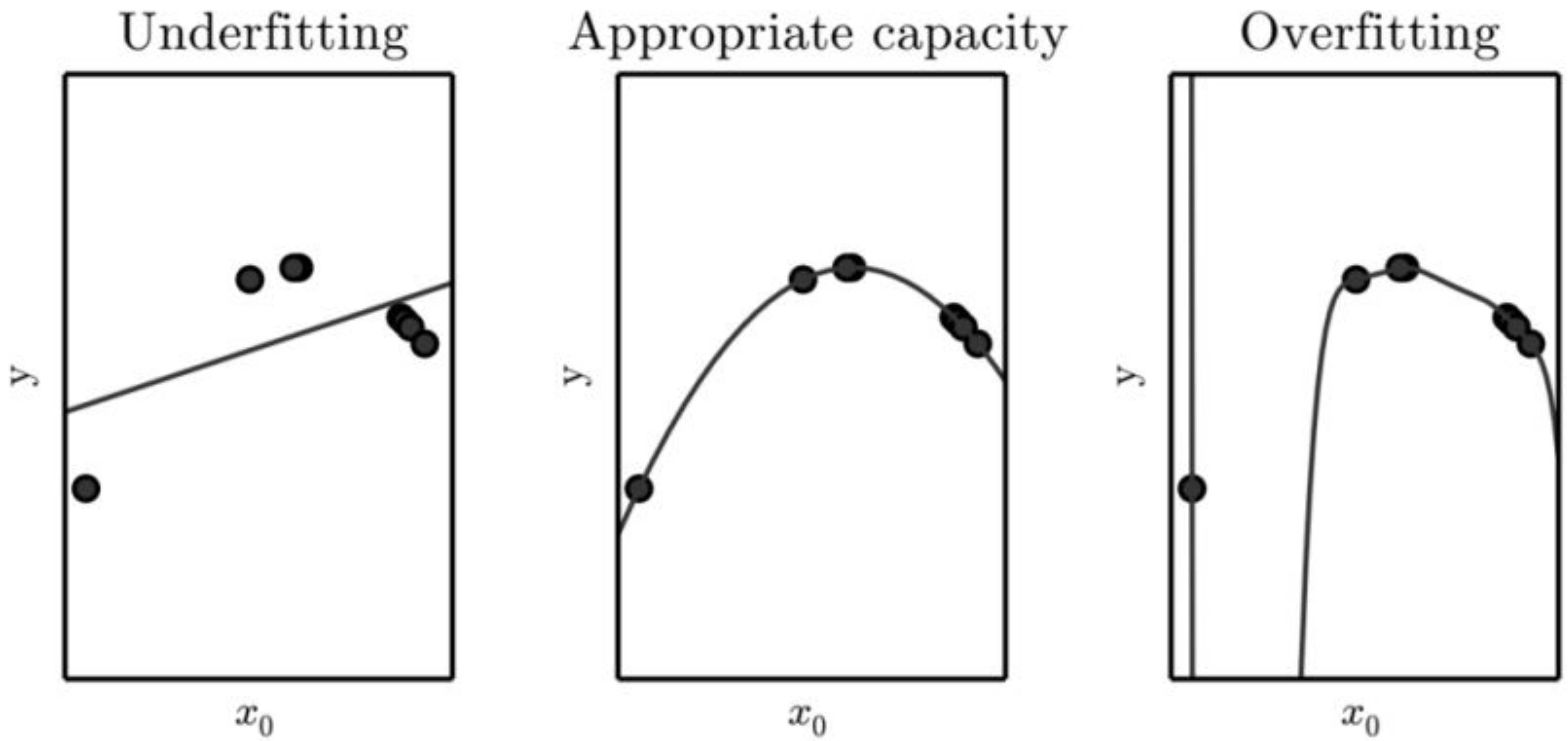
“SIMPLY: NOT ENOUGH DATA

CASE II. - WE DON’T HAVE ENOUGH *ANYTHING*



“NOT ENOUGH IN WHAT SENSE?“

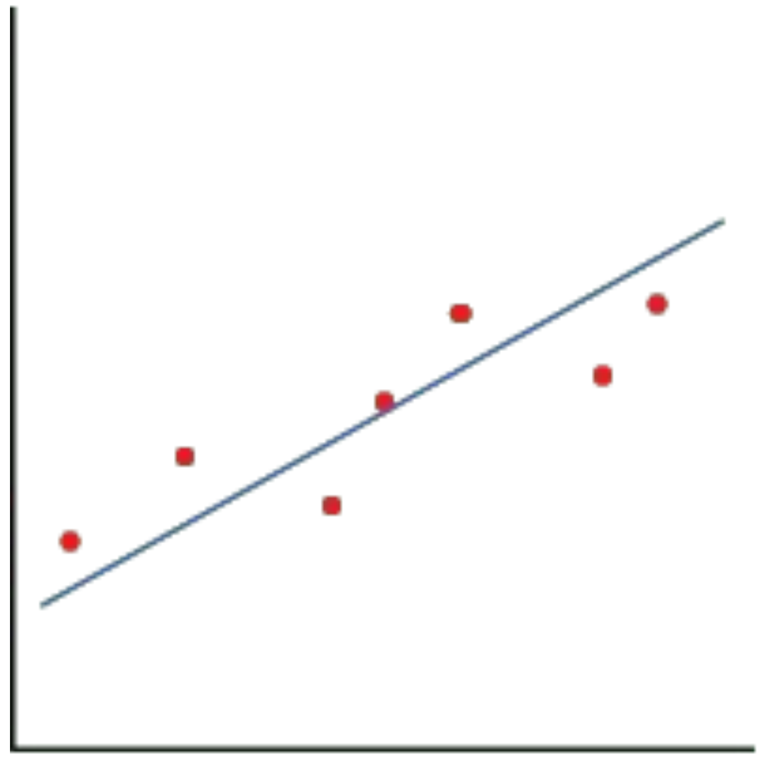
CONNECTION WITH OVERFITTING

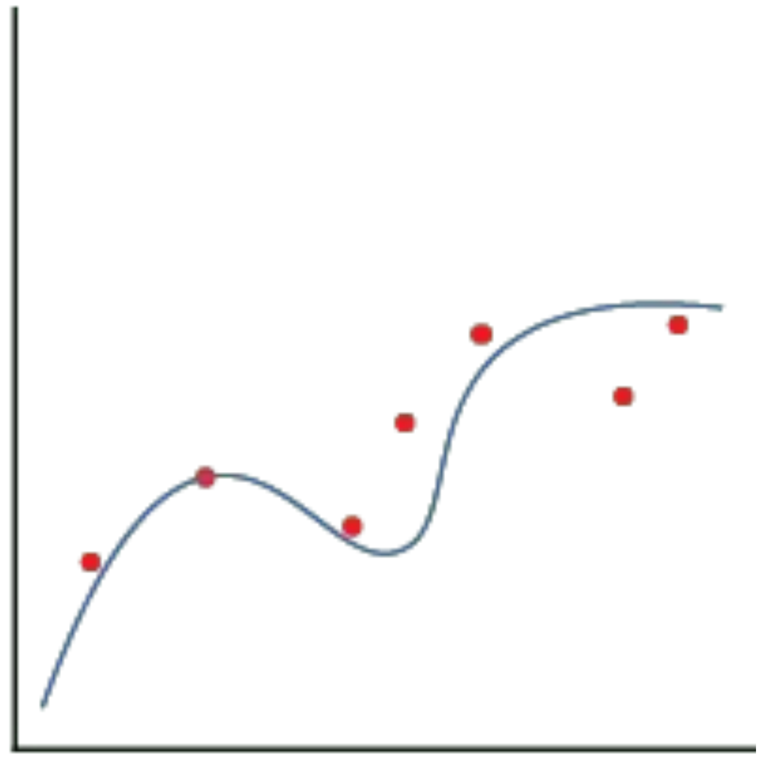


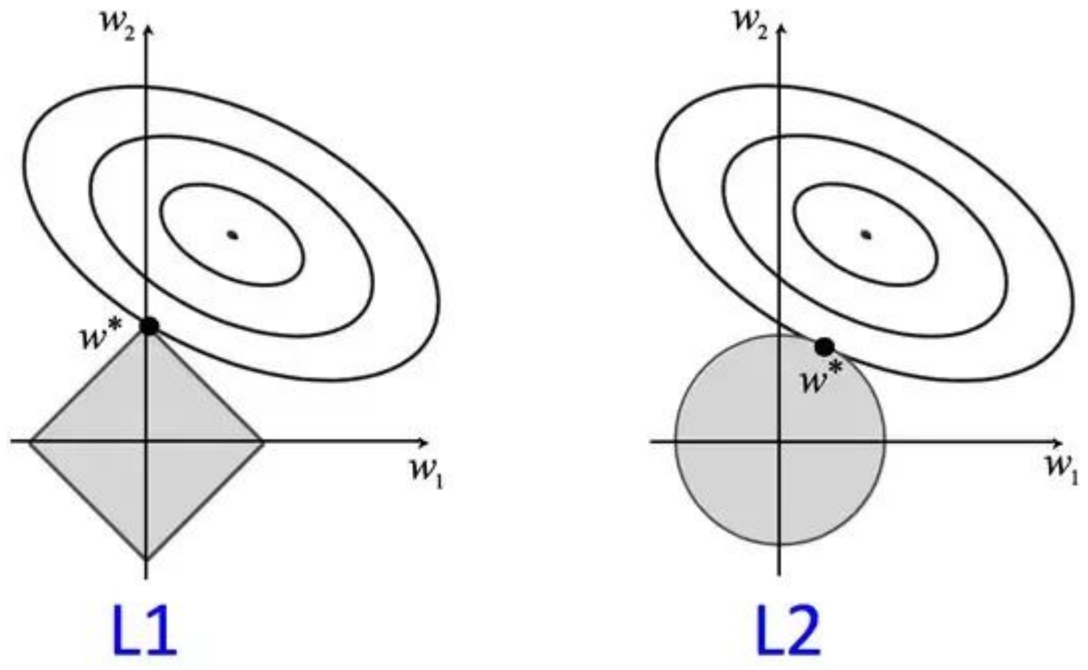
source:
Overfitting - Wikipedia

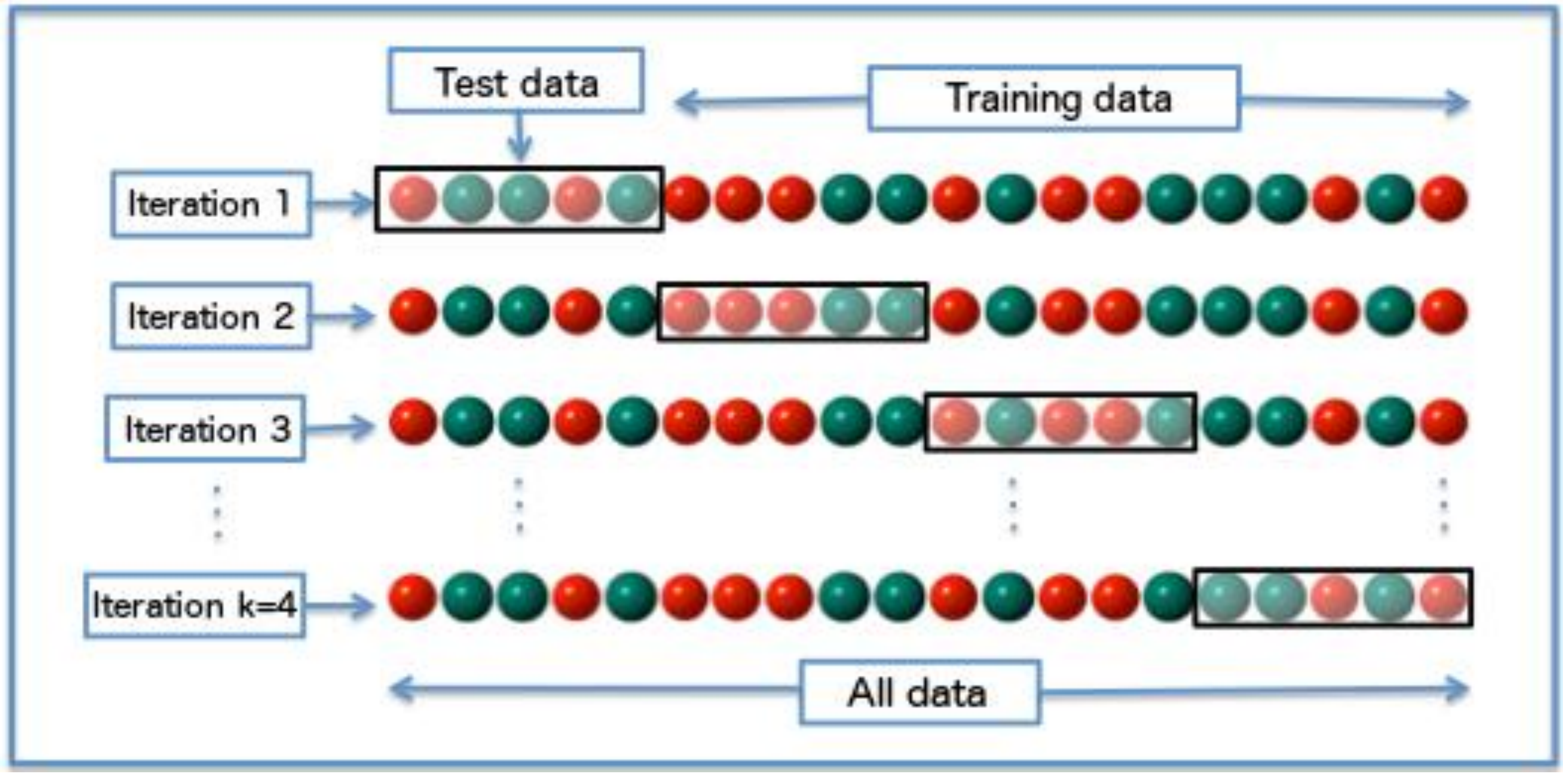
“TRY THE CLASSICS FIRST“

FIRST TRY - CLASSIC METHODS FOR STABILITY





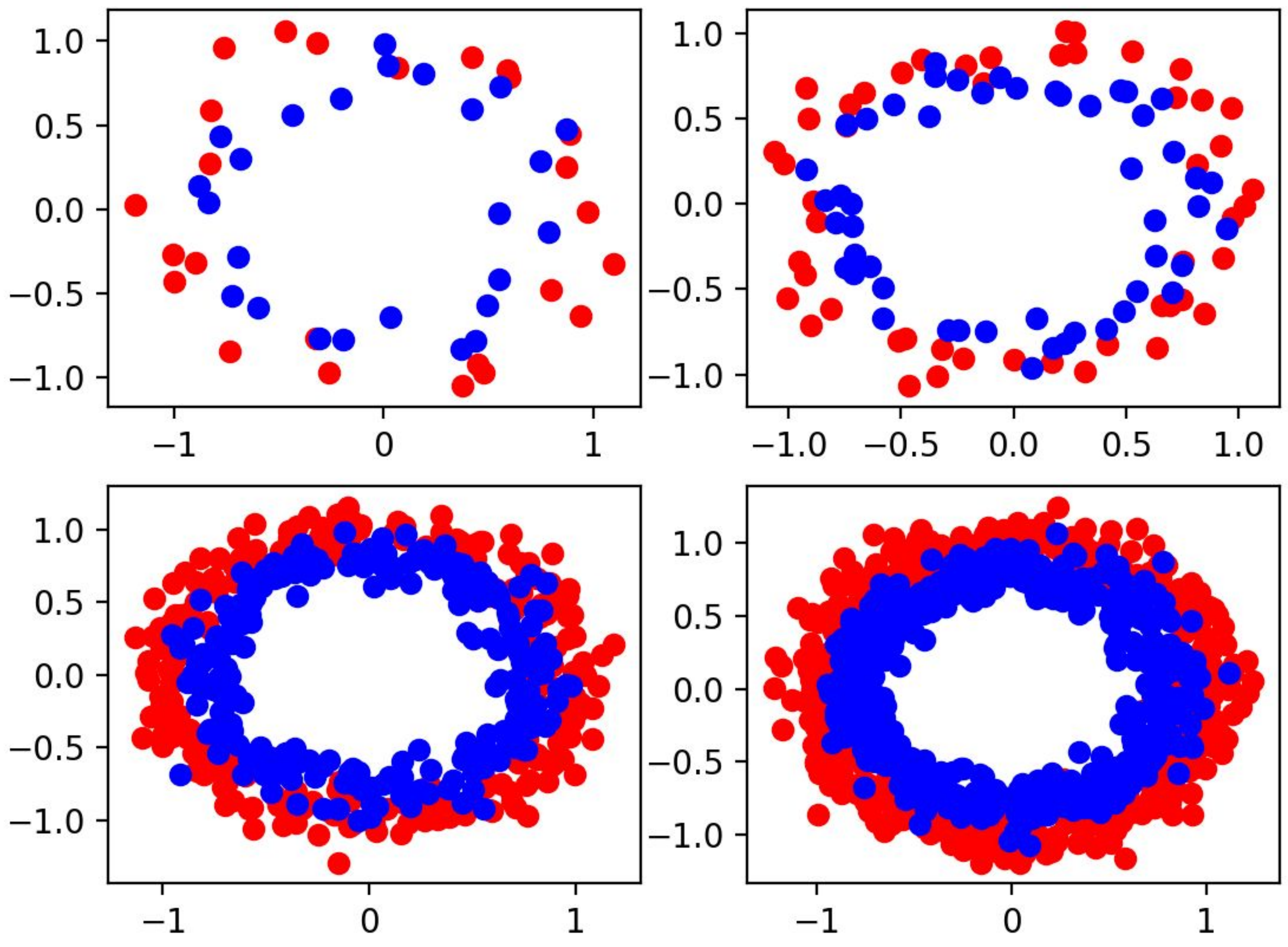


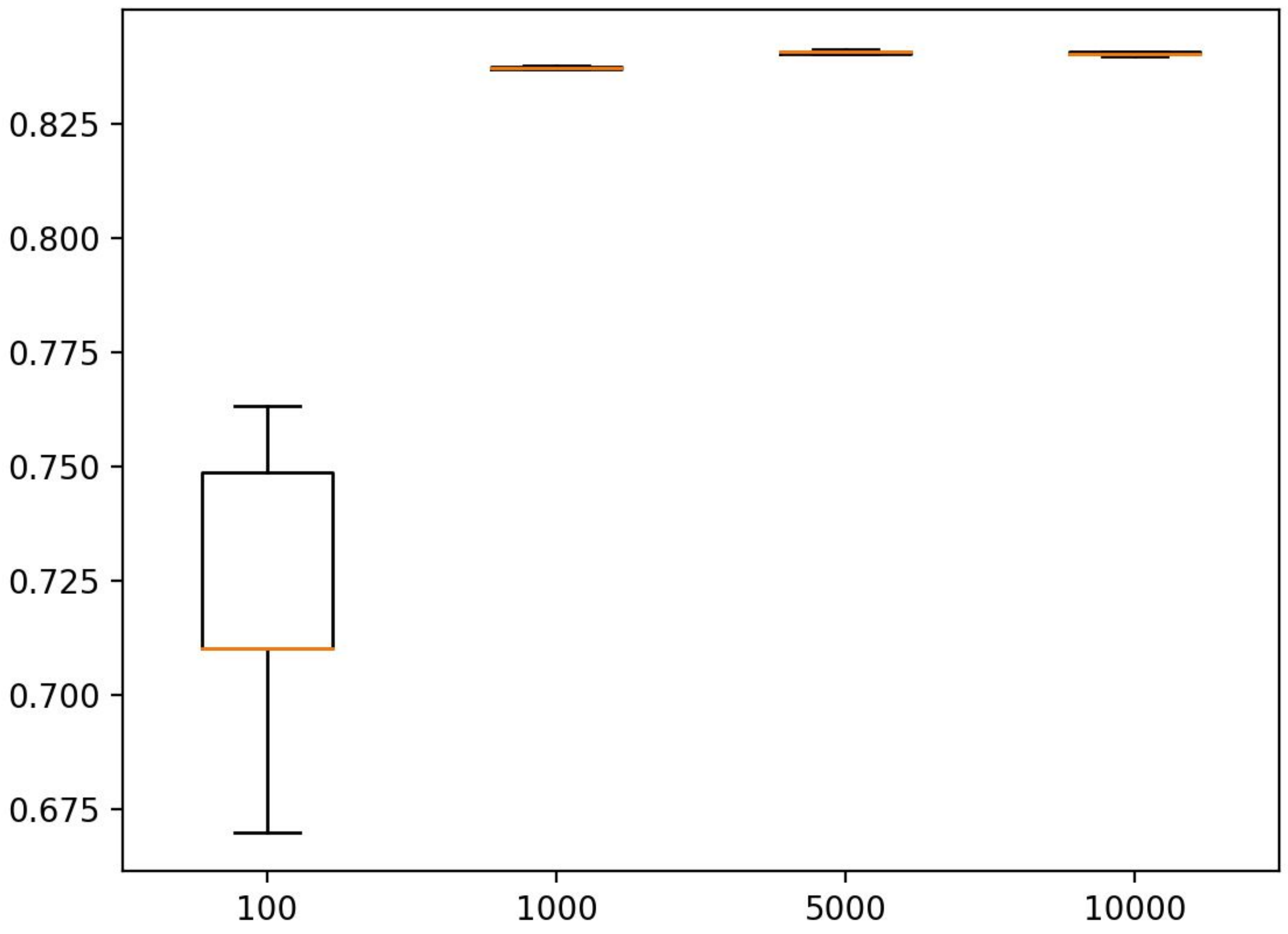


- **Modify the model:**
  - Use a simple model
    - We are often forced to use a complex one since the data itself is complex (dimensions, non-linearity…)
  - Use special models (eg. SUFTware)
- **Modify the objective:**
  - Add regularization term (Capacity control)
  - Use “max margin” objective (like in SVMs)
- **Modify the training:**
  - Use crossvalidation for getting a bit more out of the data
  - Use PU Learning

"THE MORE THE MERRYER"

HOW MUCH IS "ENOUGH" FOR A SMALL NEURAL NET?



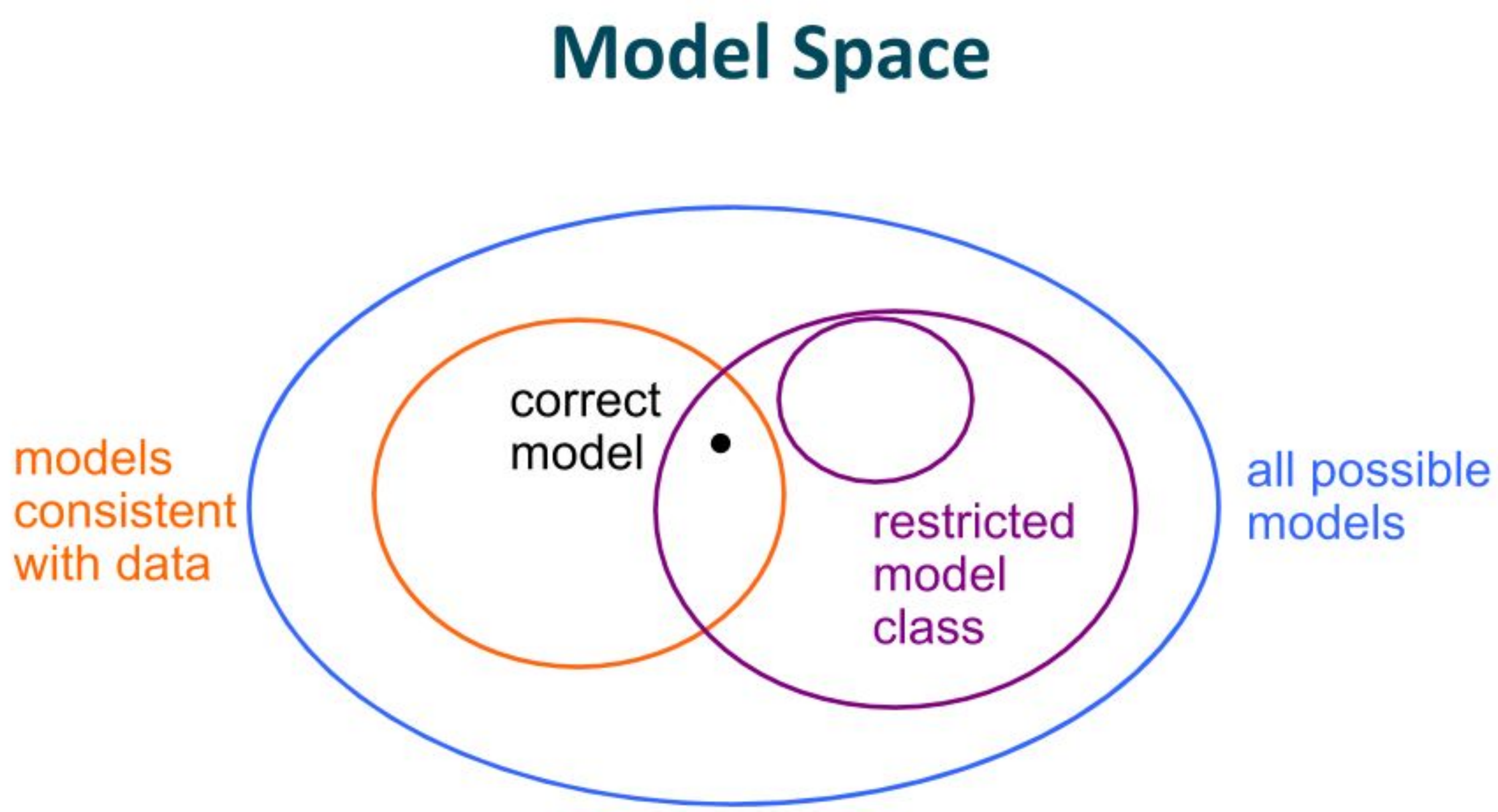


"...two inputs, 25 nodes in the hidden layer, and one output..."

source:
Jason Brownlee: "Impact of dataset size on deep learning model skill and performance estimates"

"THE MORE THE MERRYER"

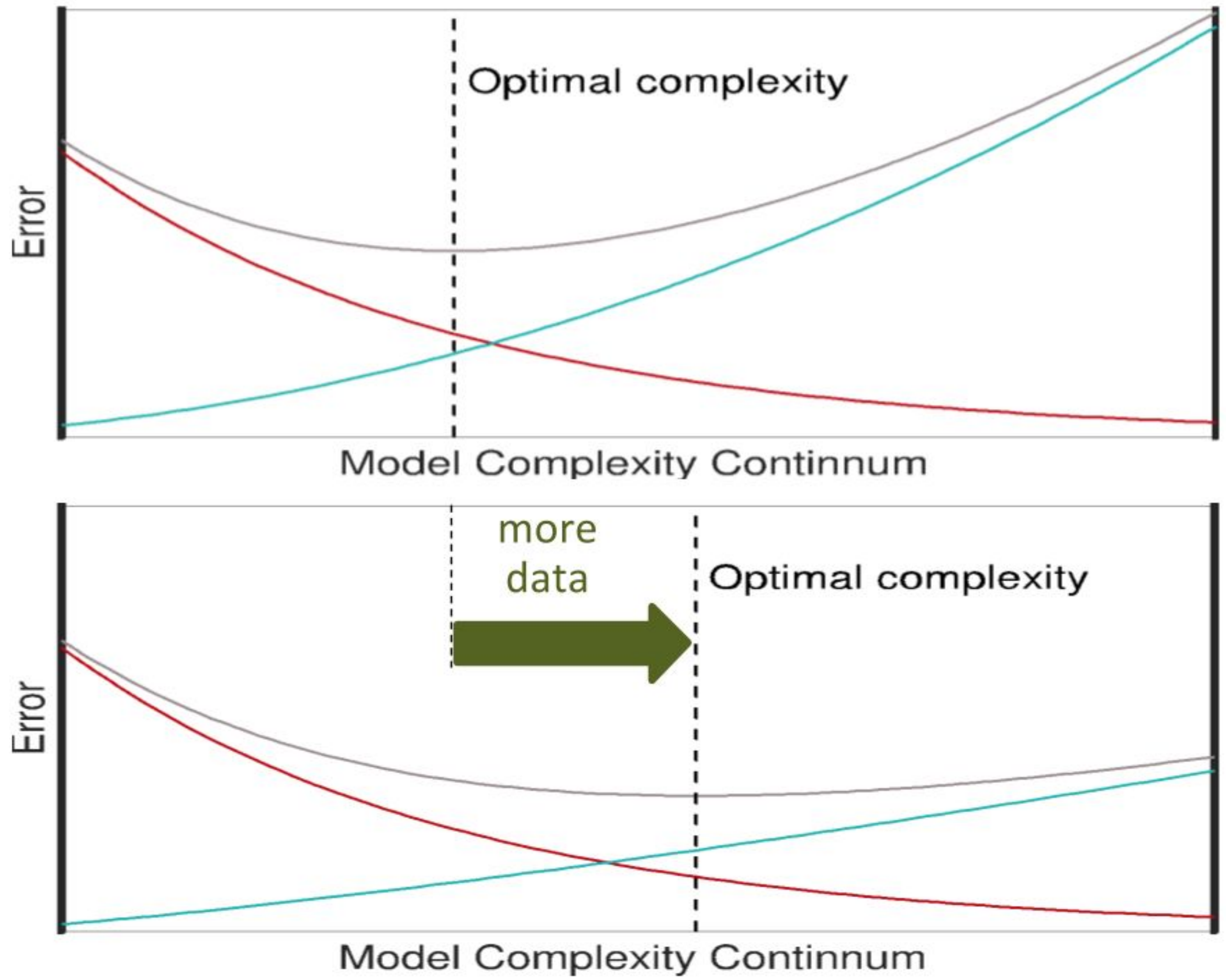
REMARK: ADDING MORE DATA ACTS AS "REGULARIZER"



Restricting model class can help

Or it can hurt

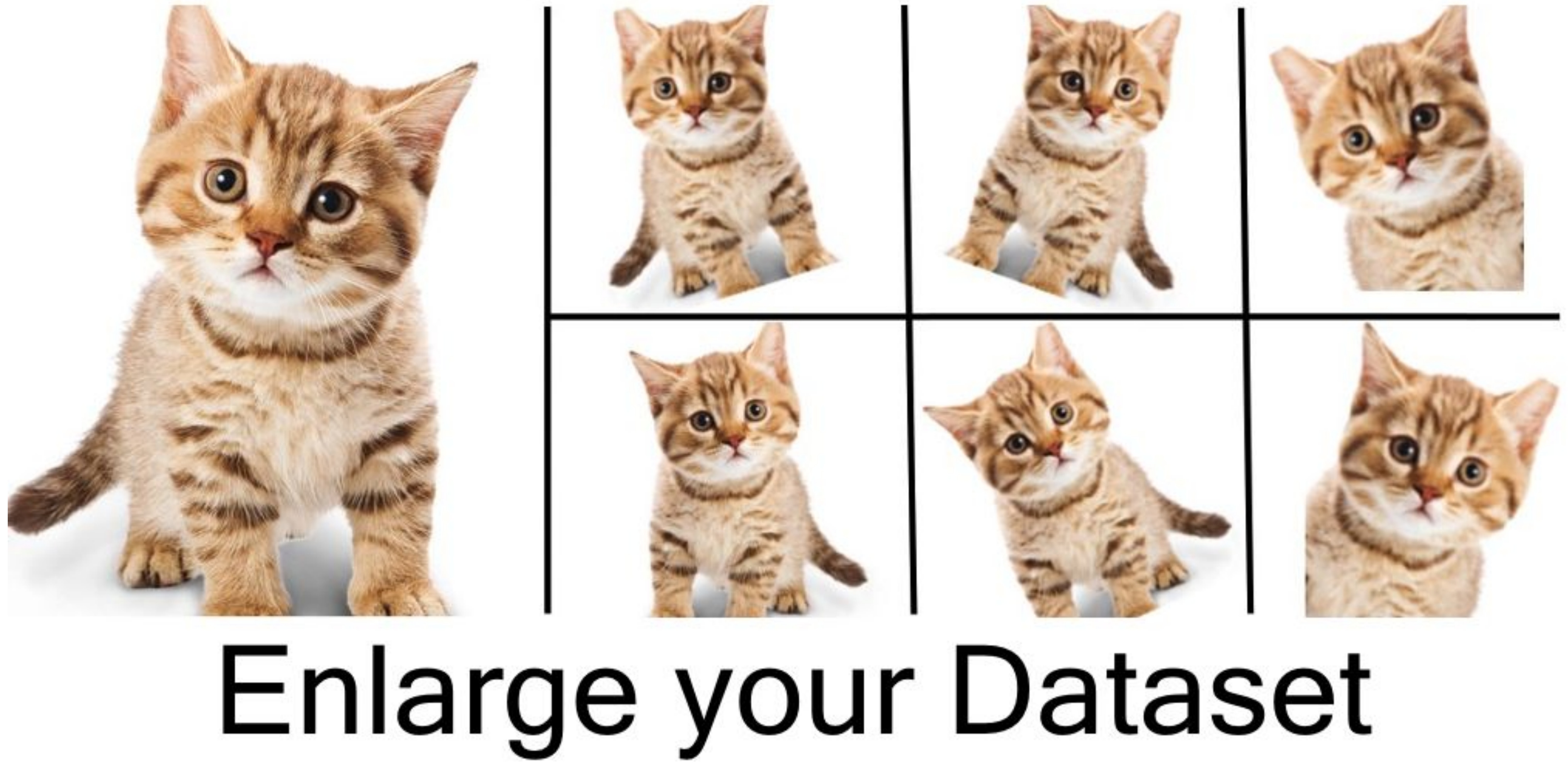
Depends on whether restrictions are domain appropriate

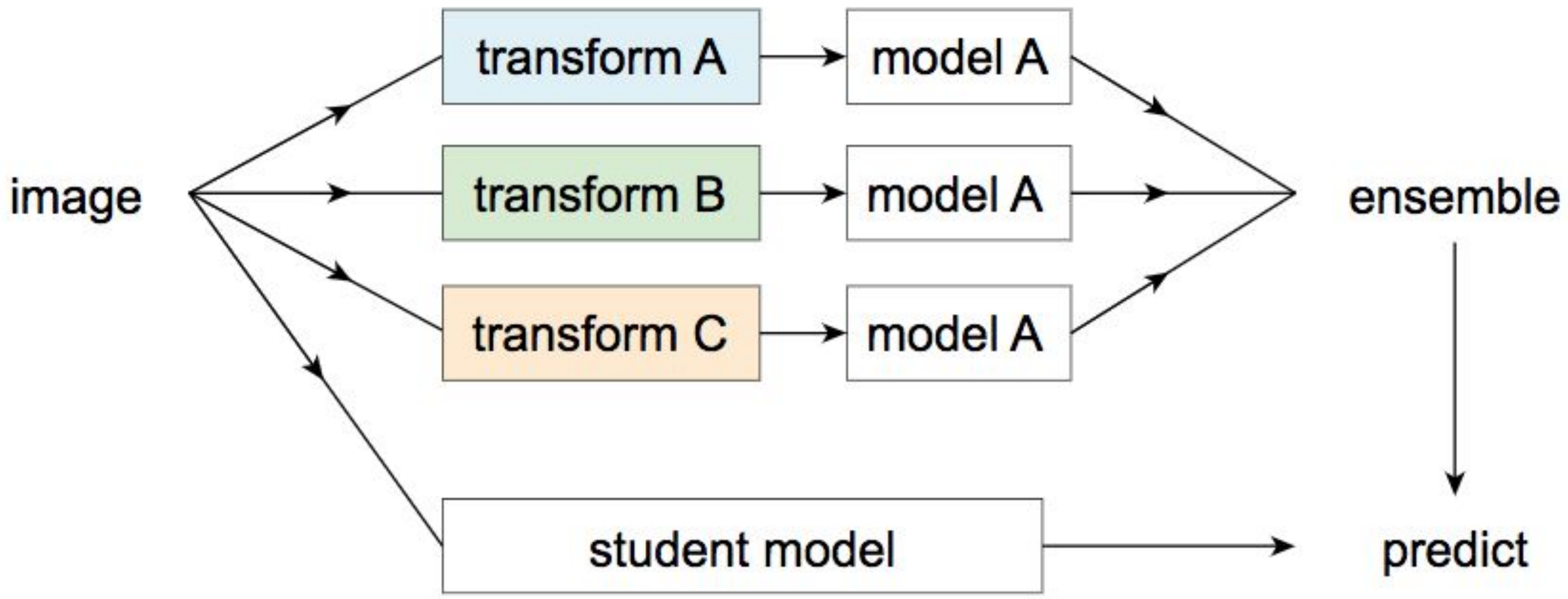


source:
"Lecture series of Michael C. Mozer at DeepLearn2017 Bilbao"

"THE MORE THE MERRIER"

GET MORE "DATA" 1. - GENERATE OR AUGMENT



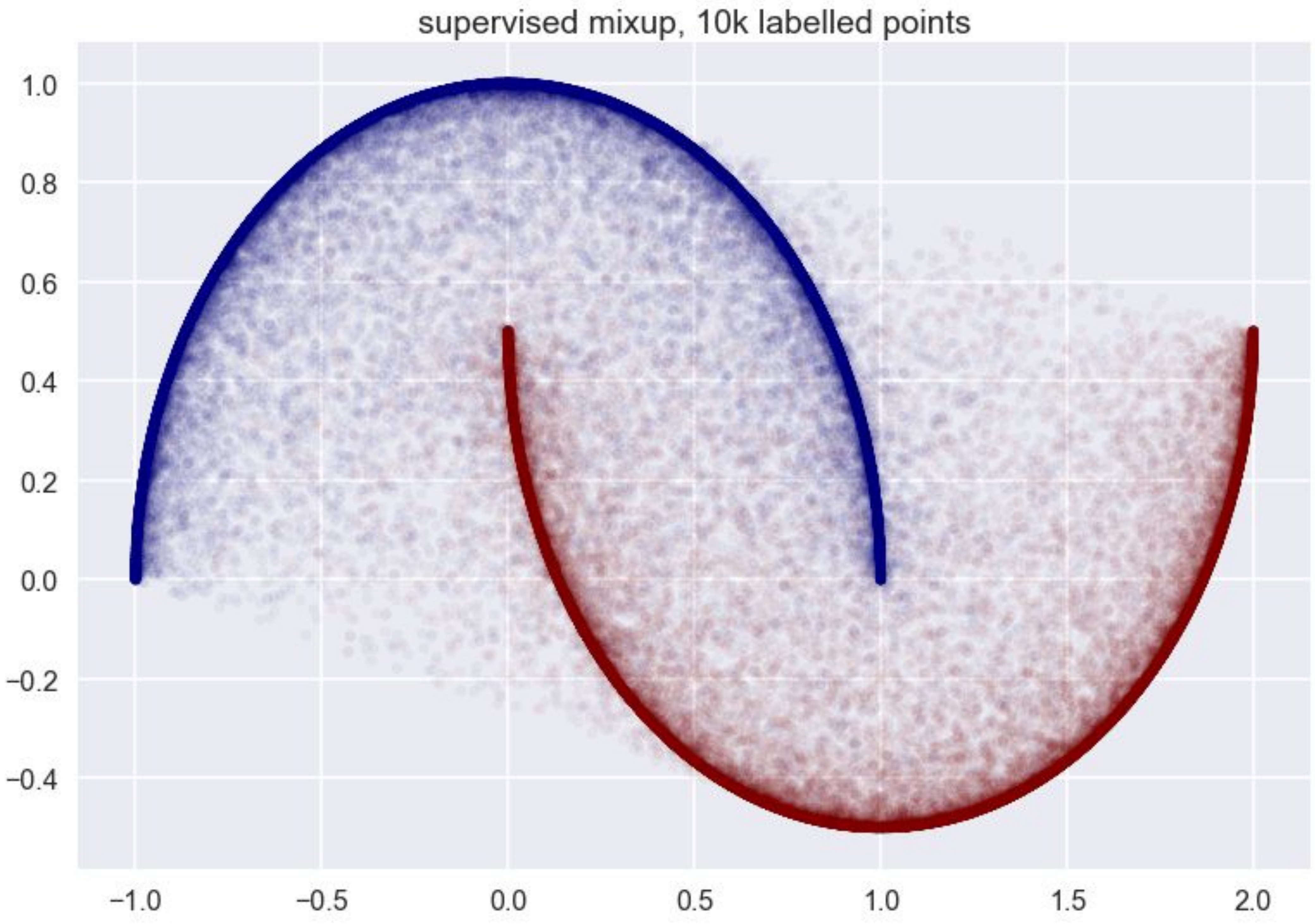


- **Data augmentation:**
  - Use simple operations to modify the data
    - Images: rotate, mirror, crop,...
    - **MUST be realistic for the domain distribution**
- **"Self labeling":**
  - Transform data, train subclassifiers, use them on new data, add predictively labelled data to original.
- **Weak supervision:**
  - Can be, that labels wil be noisy - crowdsourcing

source:
"Data augmentation - How to use Deep Learning when zou have limited data?"
"Data distillation: Towards omni-supervised learning"
"A brief introduction to weakly supervised learning"
"The Quiet Semi-Supervised Revolution"

"LEARN THE DISTRIBUTION!"

GET MORE "DATA" 1.1 - "MIXUP"



The idea of "Mixup":

Contribution Motivated by these issues, we introduce a simple and data-agnostic data augmentation routine, termed *mixup* (Section 2). In a nutshell, *mixup* constructs virtual training examples

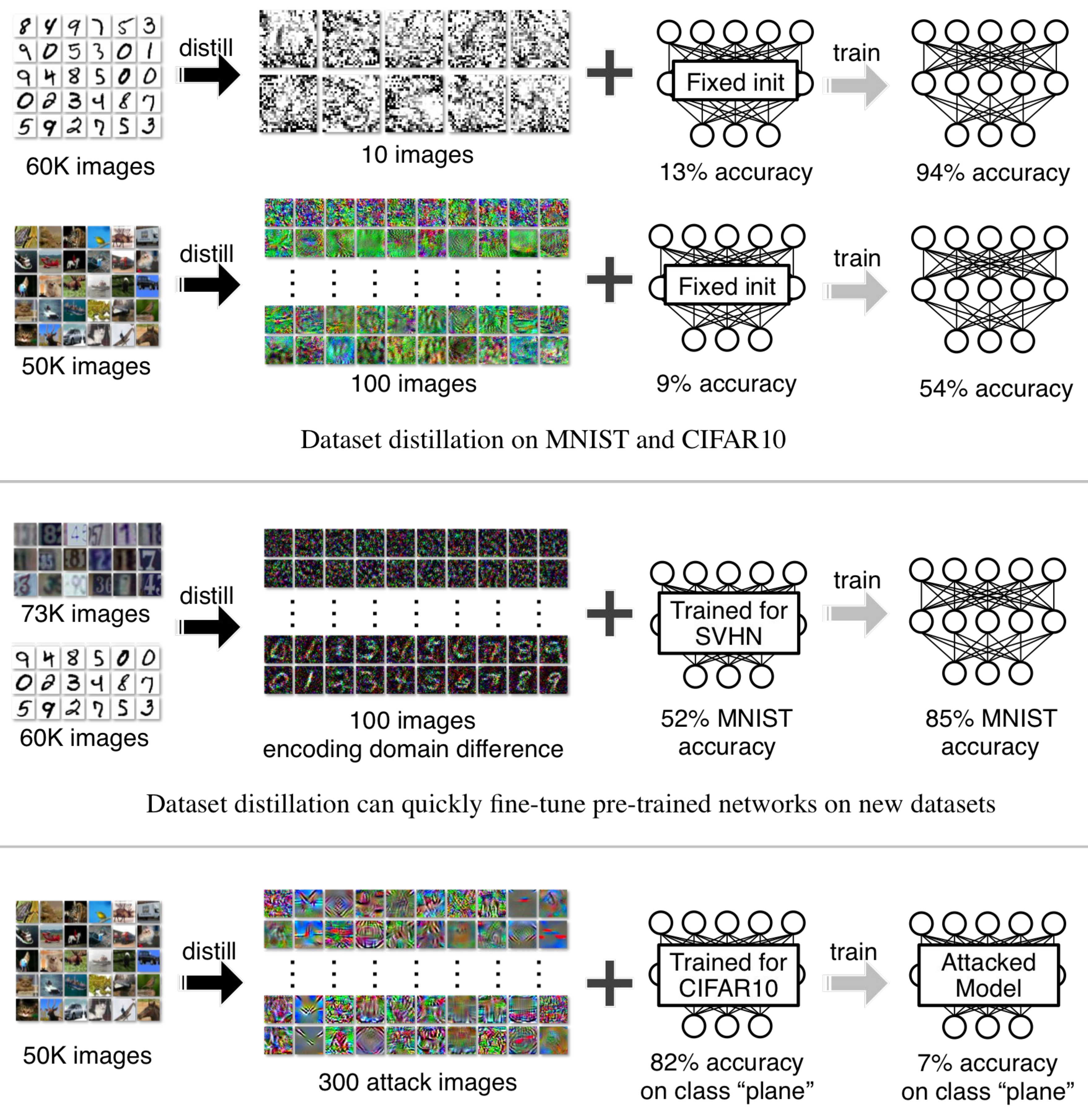
$$\tilde{x} = \lambda x_i + (1 - \lambda) x_j,$$
$$\tilde{y} = \lambda y_i + (1 - \lambda) y_j,$$

where (x_i, y_i) and (x_j, y_j) are two examples drawn at random from our training data, and $\lambda \in [0, 1]$. Therefore, *mixup* extends the training distribution by incorporating the prior knowledge that linear interpolations of feature vectors should lead to linear interpolations of the associated targets. *mixup* can be implemented in a few lines of code, and introduces minimal computation overhead.

Approximates a whole distribution!

source:
Mixup: Beyond empirical risk minimization
Mixup: Data-dependent Data Augmentation (analysis by inFERENCe)"

SIDENOTE: DATASET DISTILLATION



Dataset distillation on MNIST and CIFAR10

Dataset distillation can quickly fine-tune pre-trained networks on new datasets

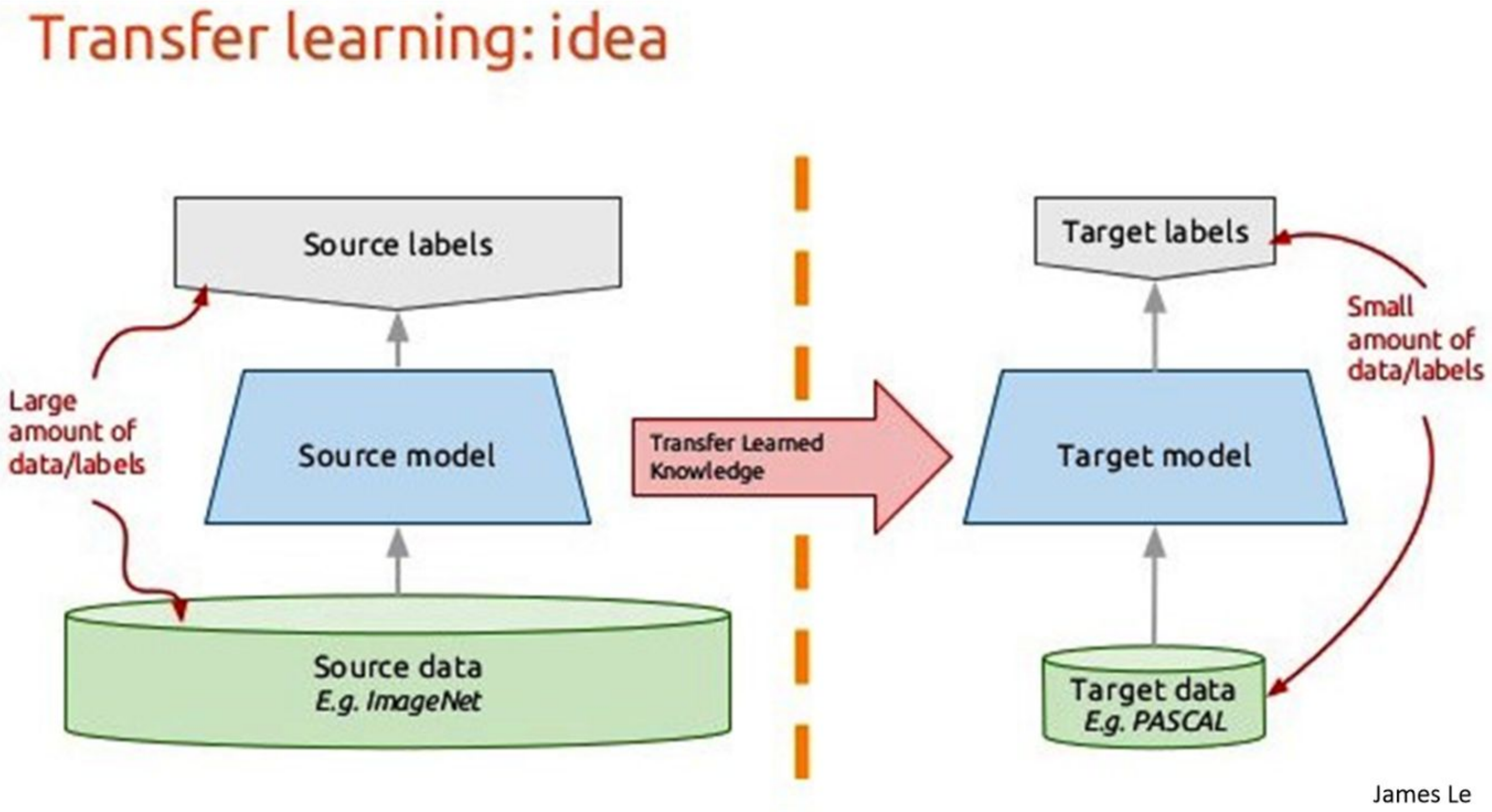
Dataset distillation can maliciously attack classifier networks

"...The idea is to synthesize a small number of data points that do not need to come from the correct data distribution, but will, when given to the learning algorithm as training data, approximate the model trained on the original data. For example, we show that it is possible to compress **60, 000 MNIST training images into just 10 synthetic distilled images (one per class) and achieve close to original performan**ce with only a few steps of gradient descent, given a particular fixed network initialization"

WTF????

source:
Dataset Dstillation

GET MORE “DATA” 2. - TRANSFER IT! (COMPRESSED)



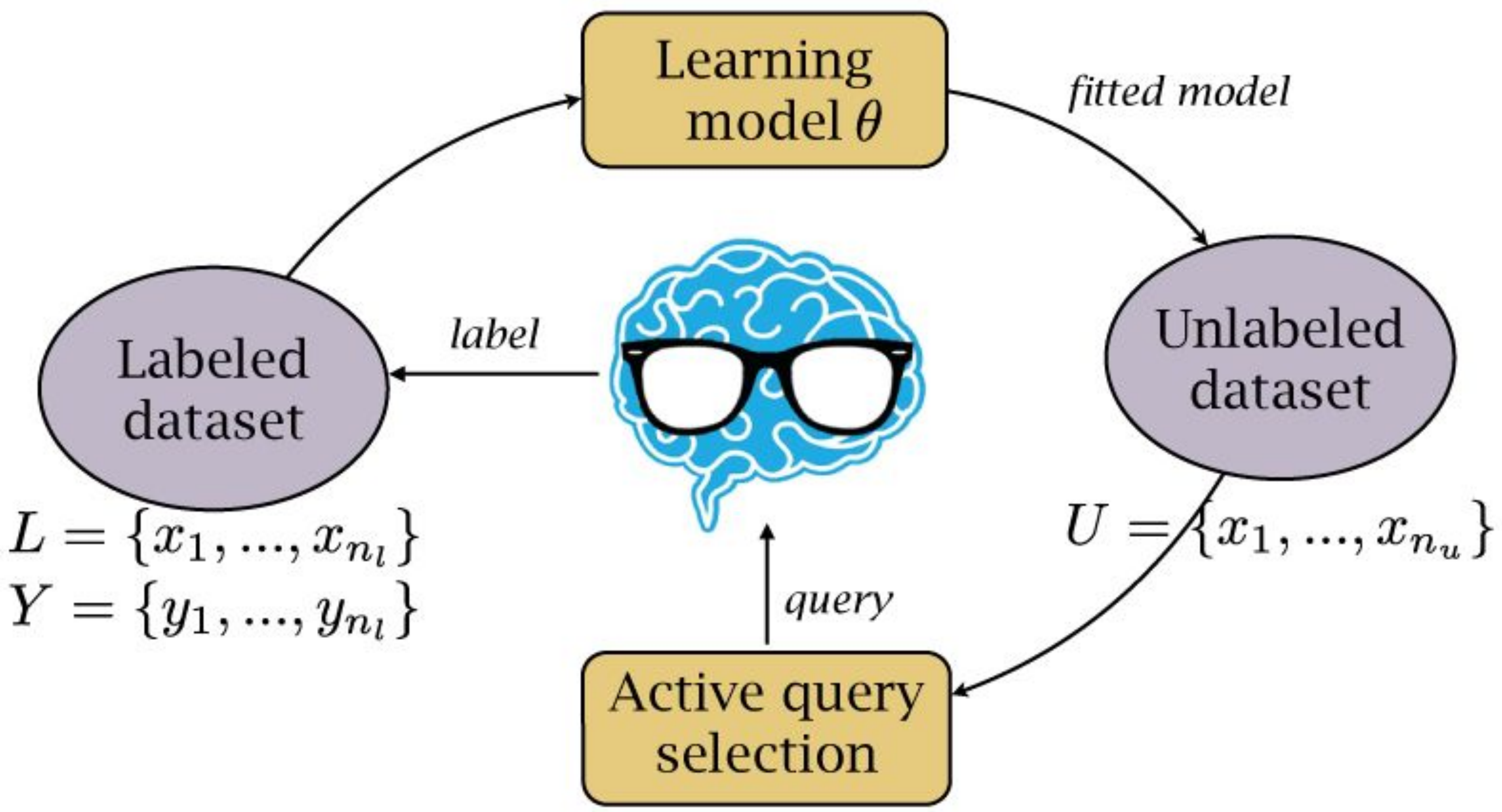
- **Transfer learning!**
  - A **HUGE** topic in itself (with more and more spohisticated methods for preventing **“catastrophic forgetting”**)
  - **We have to see, that models are “storing” data, albeit compressed.**
  - There are **plenty of pre-trained models available, USE THEM!**

- What model to “transfer”?
  - Notion of “learning a whole representation space” (see eg.: Mixup method)
  - **GANs or VAEs** are generally strong candidates (+ few labeled data case)

source:
“Transfer Learning - Machine Learning's Next Frontier”
“”Data Augmentation in Emotion Classification Using Generative Adversarial Networks”

“PLEASE?“

GET MORE “DATA” 3. - ASK FOR IT! :-)



- **Crowdsource!**
  - Amazon Mechanical Turk
  - or CrowdFlower.
- **Design a learning loop!**
  - Continuous, **Online learning**
  - There are key points worth asking for (margin, adversarial examples)
    -> **Active learning**
    -> **Building Models via Comparisons**

source:
“Adversarial sampiing for active learning”
“Atacking machine learning with adversarial examples”
“ModAL - Active learning with Keras”

MEASUREMENT VS BUSINESS RISK
- THE FALSE FOCUS ON ACCURACY

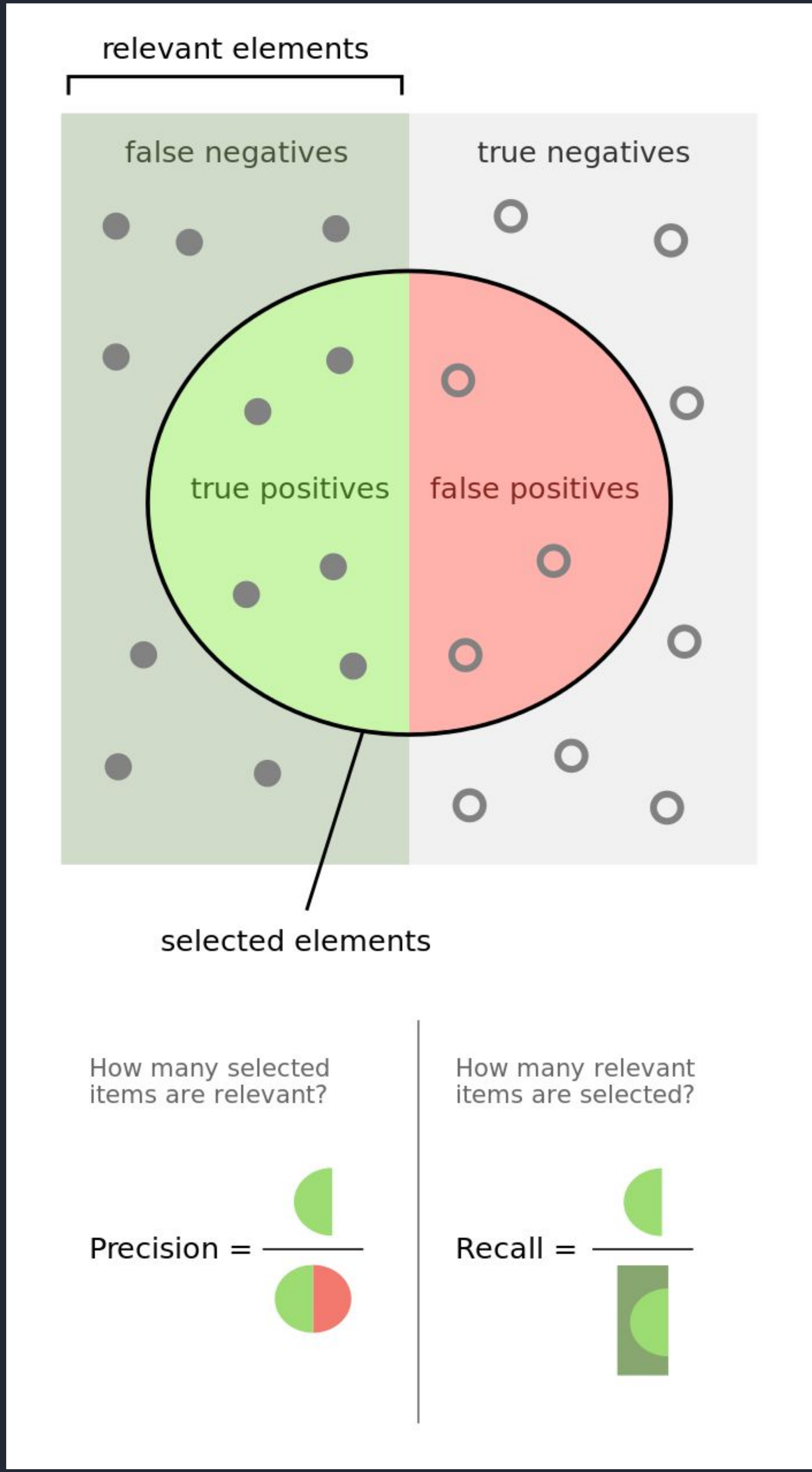
“I HAVE 90% ACCURACY!”

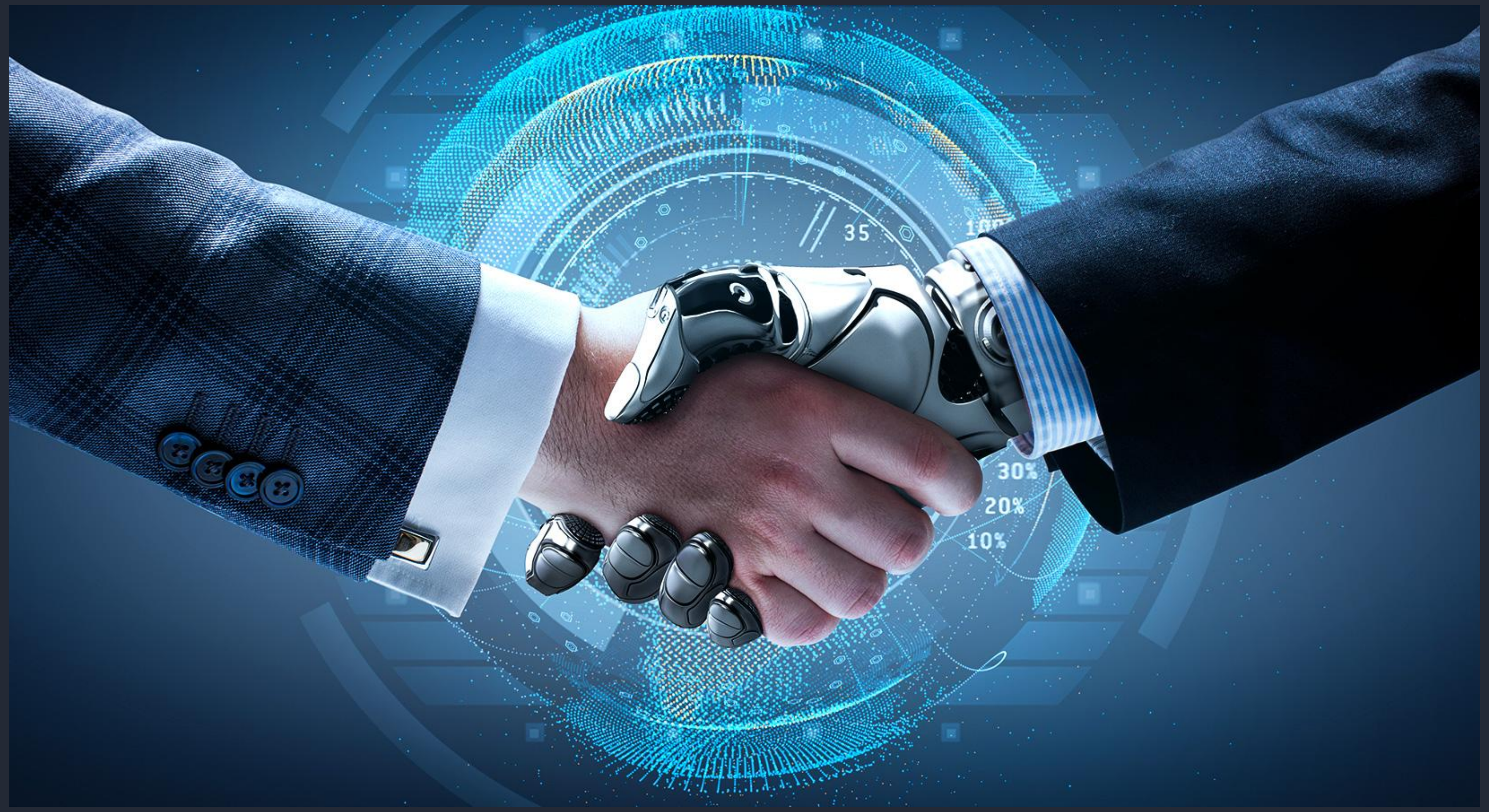
CLASSIFICATION RISK:

“Your cancer predictions are 90% accurate.
We have 10 dead people.”

SOLUTION:

Substitution of huiman expertise is not the way!
Think in cooperative systems!





DON'T REPLACE, AUGMENT!

COOPERATIVE SYSTEMS ARE MINIMIZING RISK

[Artificial intelligence VS Augmented intelligence]

WHAT IF WE DON’T NEED THAT MUCH DATA AT ALL?

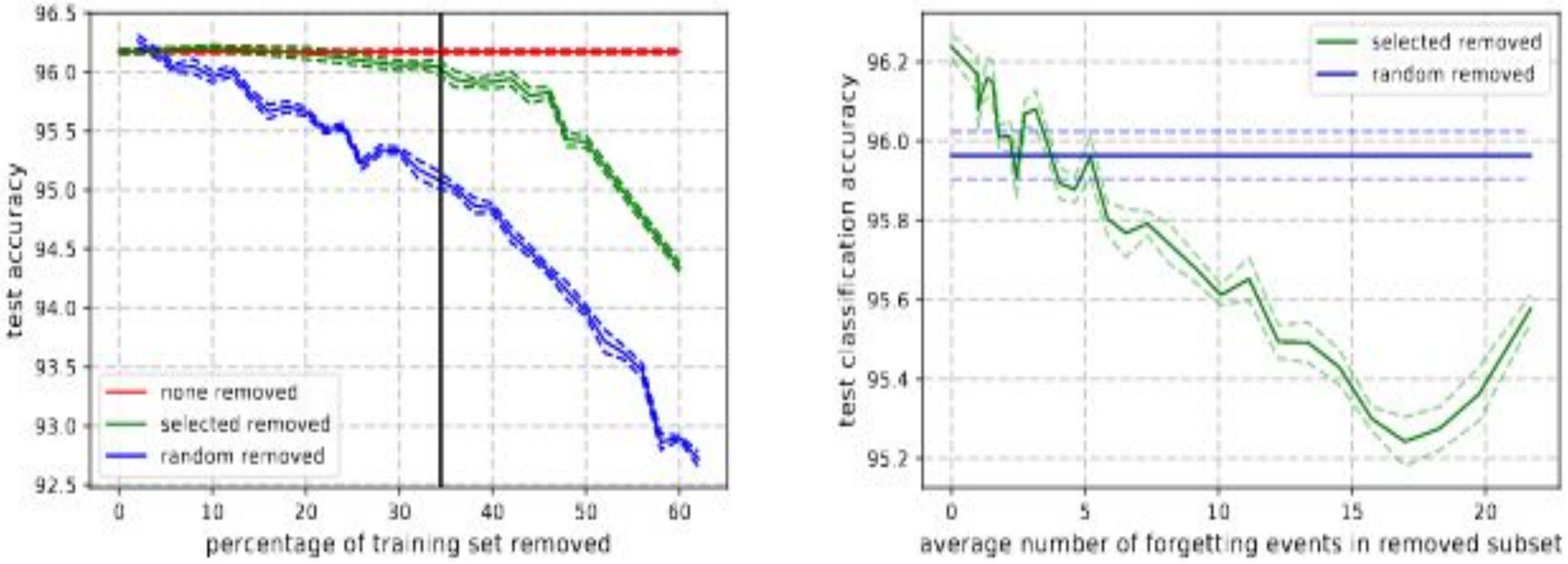
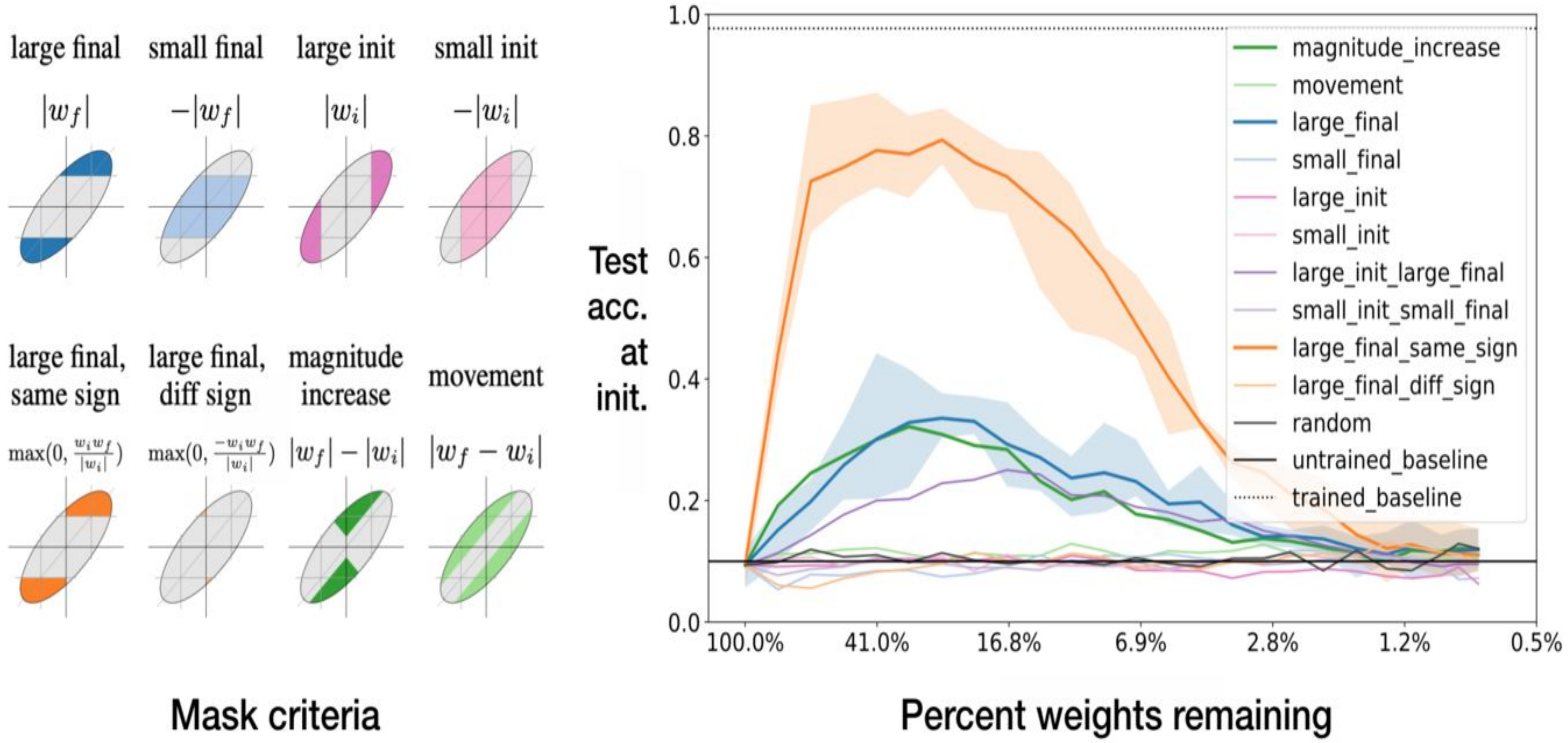


Figure 5: *Left* Generalization performance on *CIFAR-10* of ResNet18 where increasingly larger subsets of the training set are removed (mean +/- std error of 5 seeds). When the removed examples are selected at random, performance drops very fast. Selecting the examples according to our ordering can reduce the training set significantly without affecting generalization. The vertical line indicates the point at which all unforgettable examples are removed from the training set. *Right* Difference in generalization performance when contiguous chunks of 5000 increasingly forgotten examples are removed from the training set. Most important examples tend to be those that are forgotten the most.

- "A **forgetting event** happens when the neural network makes a misclassification (of a sample) at time *t+1*, having already made an accurate classification at time *t*,
- "...find that **91.7% of MNIST, comprise of unforgettable examples**."
- "**Unforgettable examples, ... encode mostly redundant information ...** removing the most unforgettable examples.
- **On CIFAR-10, 30% of the dataset can be removed without affecting test accuracy"**

source:
“An Empirical Study of Example Forgetting during Deep Neural Network Learning”
‘Seven Myths in Machine Learning Research’

WHAT IF WE DON’T NEED THAT BIG MODELS AT ALL?



- There are winning “lottery tickets”, that is: subnetworks with high performace **on initialization!**
- Seems like much of the **performance of large networks comes from these subnetworks**
- If we prune large networks, keeping these “winners”, **performace can** even increase (or **not decrease much**)
- The subnets can be found by only keeping those weights that **move away from zero during training**

source:
‘The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks’
‘Deconstructing Lottery Tickets: Zeros, Signs, and the Supermask’

THANKS FOR THE ATTENTION!

LET'S CONTINUE!

PRESENTATION

COMMUNITY

MYSELF