Machine Learning II

Master in Business Analytics and Big Data

Ángel Castellanos

acastellanos@faculty.ie.edu

What is Machine Learning?

[Machine Learning is the] field of study that gives computers the ability to learn without being explicitly programmed.

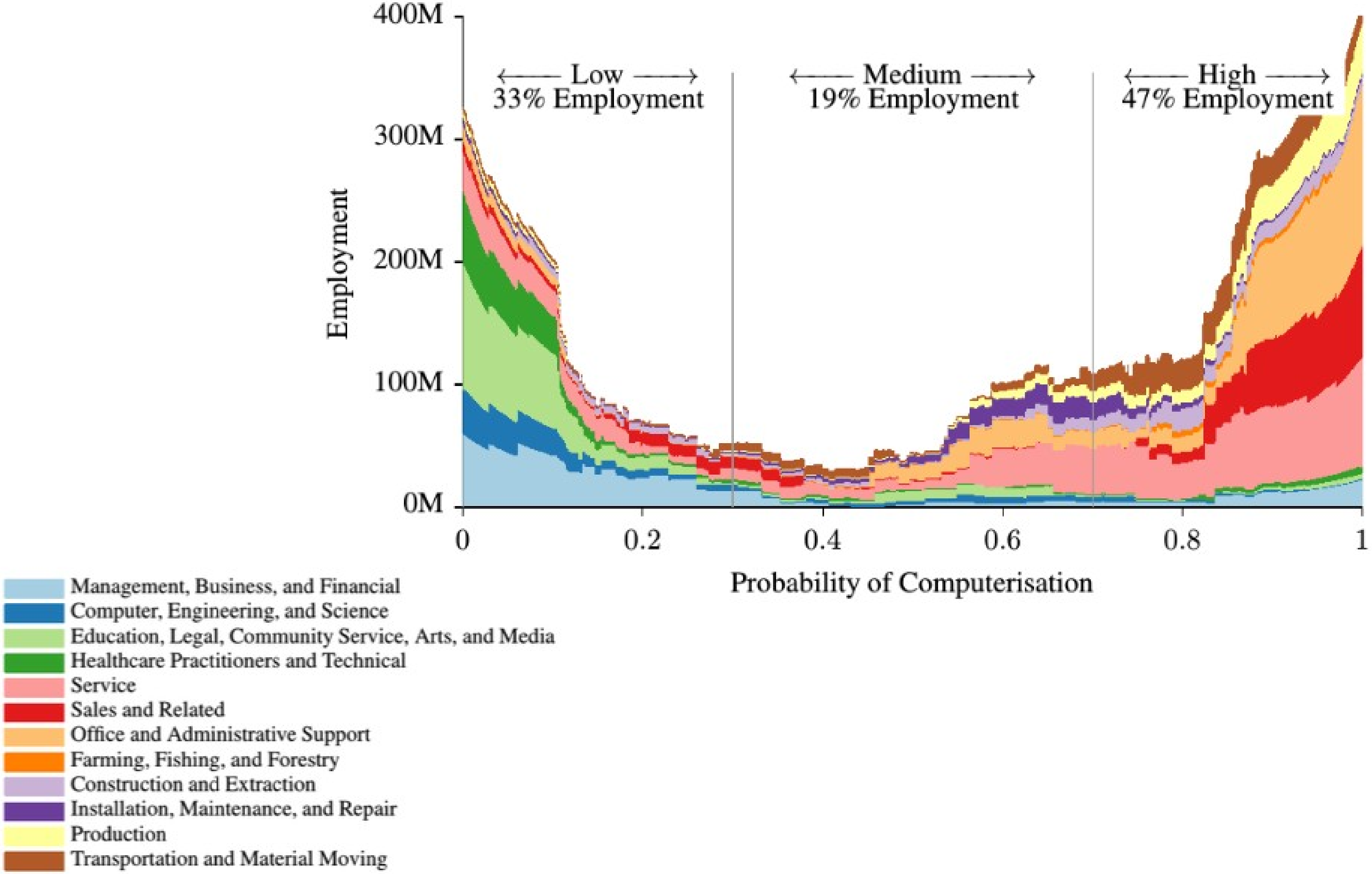
Arthur Samuel, 1959

A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.

Tom Mitchell, 1997

Why is it important?

Automation is everywhere



Machine Learning Algorithms

	Unsupervised	Supervised
Continuous	**Clustering** • K-Means **Dimensionality Reduction** • SVD • PCA	**Regression** • Linear • Polynomial **Regression Trees**
Categorical	**Association Analysis** • A priori • FP-Growth **Hidden Markov Model**	**Classification** • KNN • Decision Trees • Naive-Bayes • SVM

A Few Useful Things to Know about Machine Learning

- Machine learning systems **automatically learn programs from data**,

- Several fine textbooks are available to interested practitioners and researchers. However, much of the “**folk knowledge**” that is needed to successfully develop machine learning applications is not readily available in them.

- So, many **machine learning projects take much longer** than necessary or produce less-than-ideal results

LEARNING =

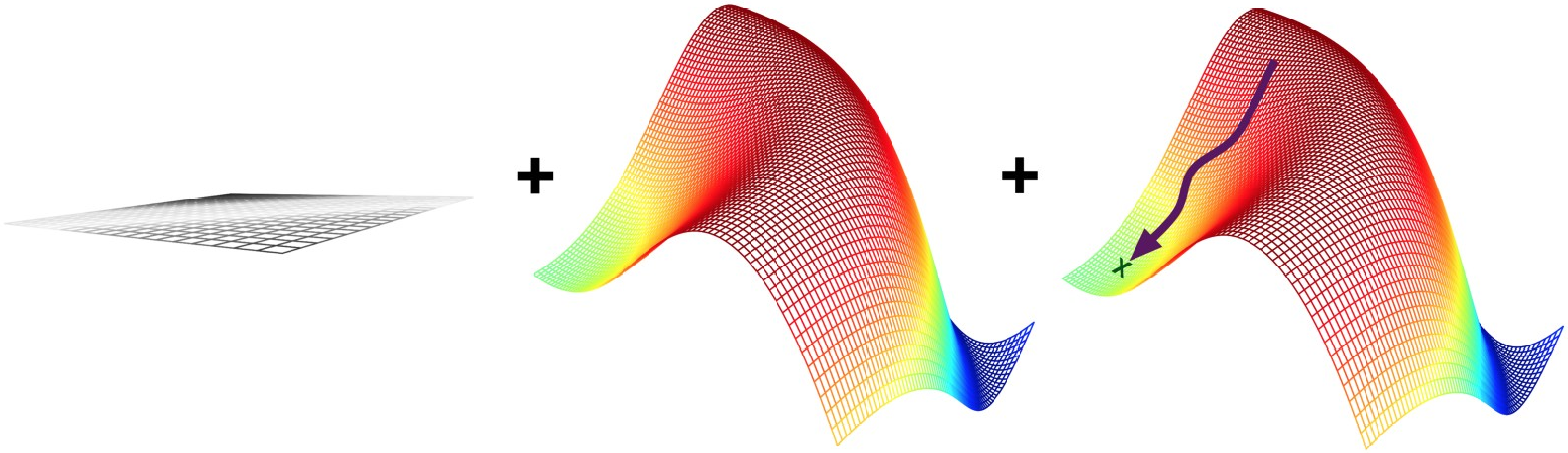
REPRESENTATION + EVALUATION + OPTIMIZATION

Learning algorithms consists of combinations of just three components:

- **Representation**: choosing the set of classifiers that it can possibly learn. This set is called the ***hypothesis space*** of the learner.
  - Linear regression --> Linear function
- **Evaluation**: An evaluation function (also called objective function or scoring function) is needed to distinguish good classifiers from bad ones.
  - What are you trying to improve?
- **Optimization**: Needing a method to search among the classifiers in the language for the highest-scoring one. The choice of optimization technique is key to the efficiency of the learner

LEARNING =

REPRESENTATION + EVALUATION + OPTIMIZATION



Source: https://medium.com/@devnag/machine-learning-representation-evaluation-optimization-fc7b26b38fdb

LEARNING =

REPRESENTATION + EVALUATION + OPTIMIZATION

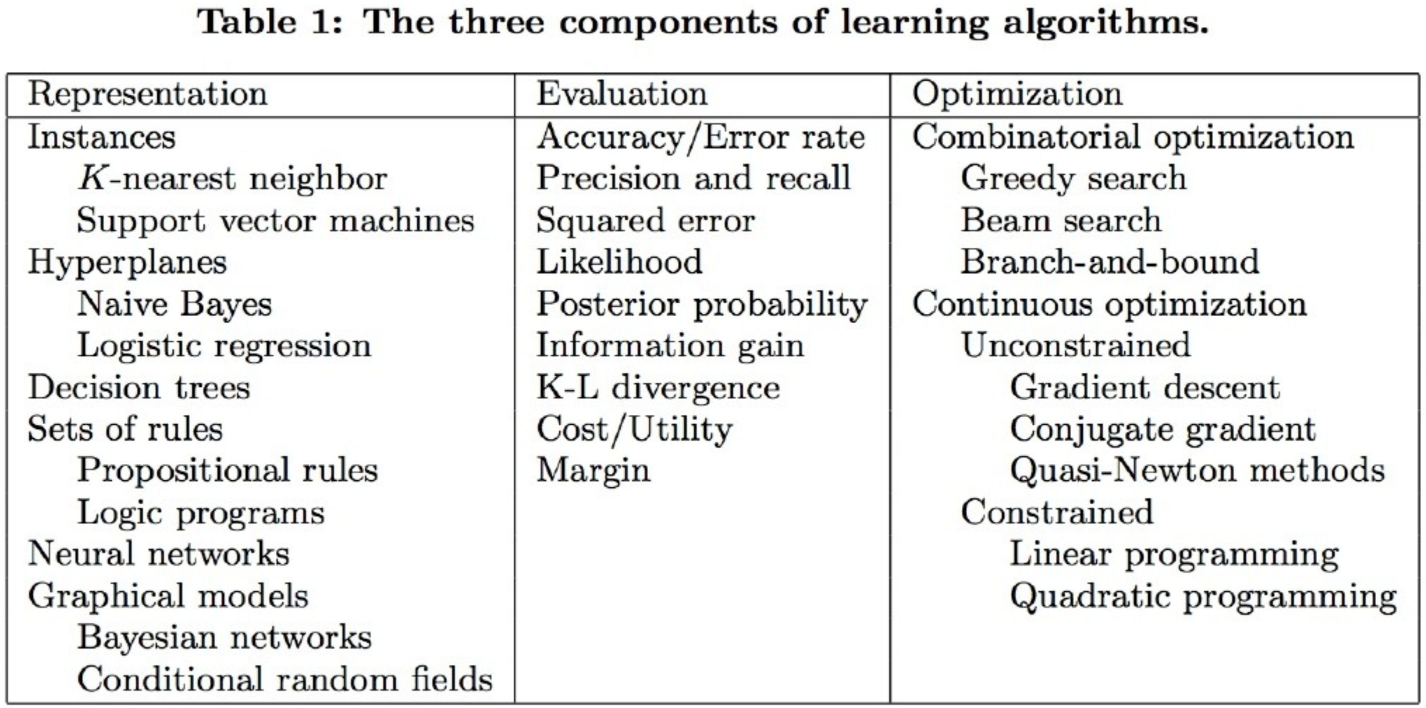


Table 1: The three components of learning algorithms.

Representation	Evaluation	Optimization
Instances	Accuracy/Error rate	Combinatorial optimization
K-nearest neighbor	Precision and recall	Greedy search
Support vector machines	Squared error	Beam search
Hyperplanes	Likelihood	Branch-and-bound
Naive Bayes	Posterior probability	Continuous optimization
Logistic regression	Information gain	Unconstrained
Decision trees	K-L divergence	Gradient descent
Sets of rules	Cost/Utility	Conjugate gradient
Propositional rules	Margin	Quasi-Newton methods
Logic programs		Constrained
Neural networks		Linear programming
Graphical models		Quadratic programming
Bayesian networks		
Conditional random fields		

Most textbooks are organized by representation, the other components are equally important

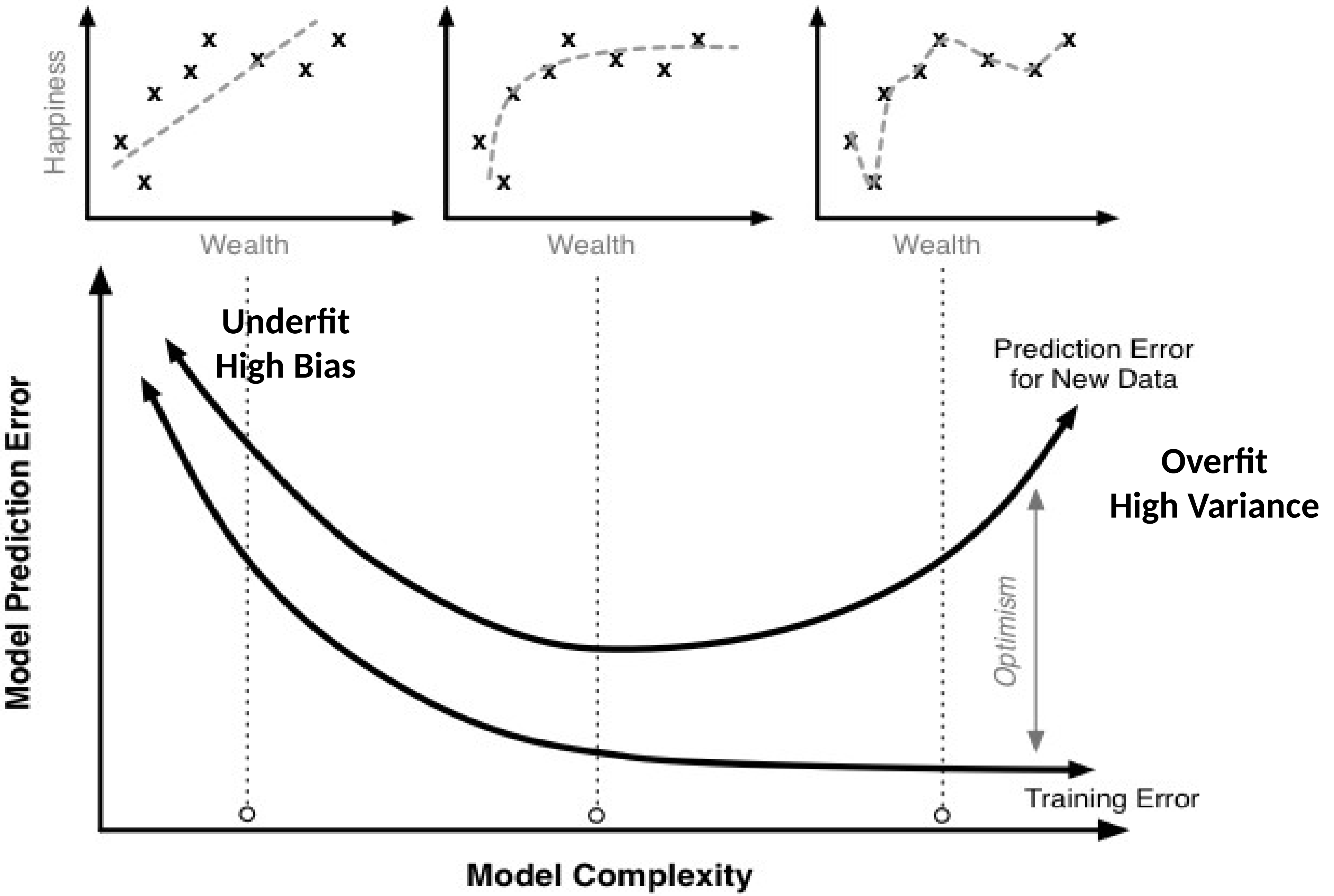
https://homes.cs.washington.edu/~pedrod/papers/cacm12.pdf

IT’S GENERALIZATION THAT COUNTS

- The fundamental goal of machine learning is to **generalize** beyond the examples in the training set. **Avoid overfitting!**

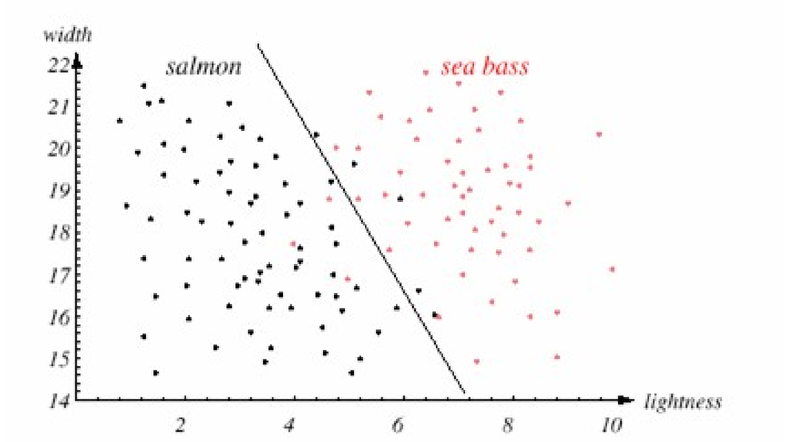
- The most common mistake among machine learning beginners is to test on the training data and have the **illusion of success**.

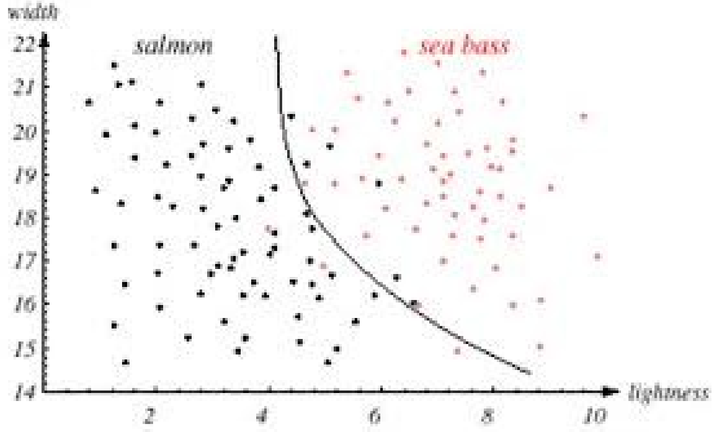
- **Cross-validation:** randomly dividing your training data into (say) ten subsets, holding out each one while training on the rest, testing each learned classifier on the examples it did not see, and averaging the results

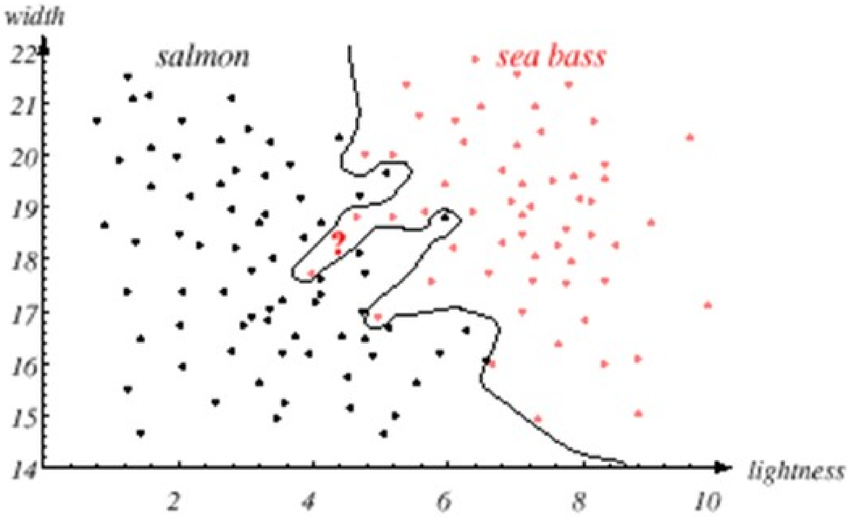


Underfit
High Bias

Overfit
High Variance


width
22
21
20
19
18
17
16
15
14
salmon
sea bass
2
4
6
8
10
lightness


width
22
21
20
19
18
17
16
15
14
salmon
sea bass
2
4
6
8
10
lightness


width
22
21
20
19
18
17
16
15
14
salmon
sea bass
?
2
4
6
8
10
lightness

OVERFITTING HAS MANY FACES

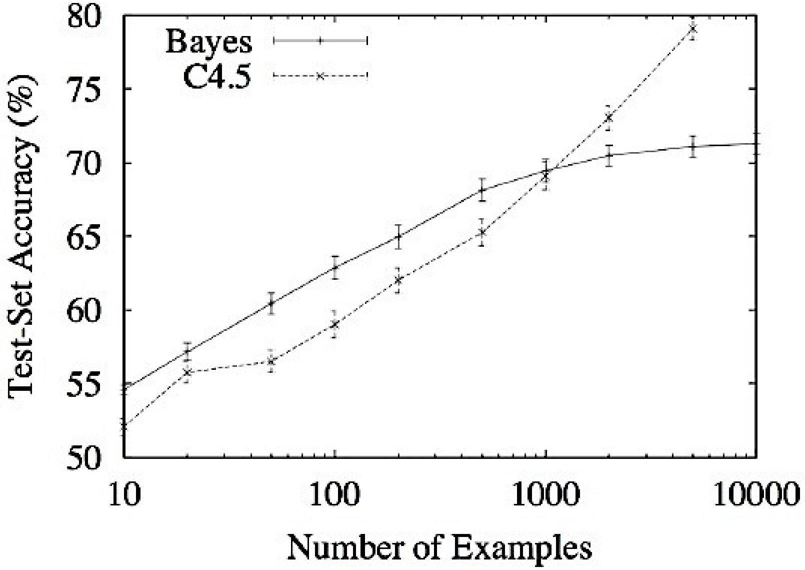
- A ***linear learner*** *has high bias*, because when the frontier between two classes is not a hyperplane the learner is unable to induce it,

- **Decision trees** don’t have this problem because they can represent any Boolean function, but on the other hand they can suffer from high variance: *decision trees learned on different training sets generated by the same phenomenon are often very different, when in fact they should be the same*.

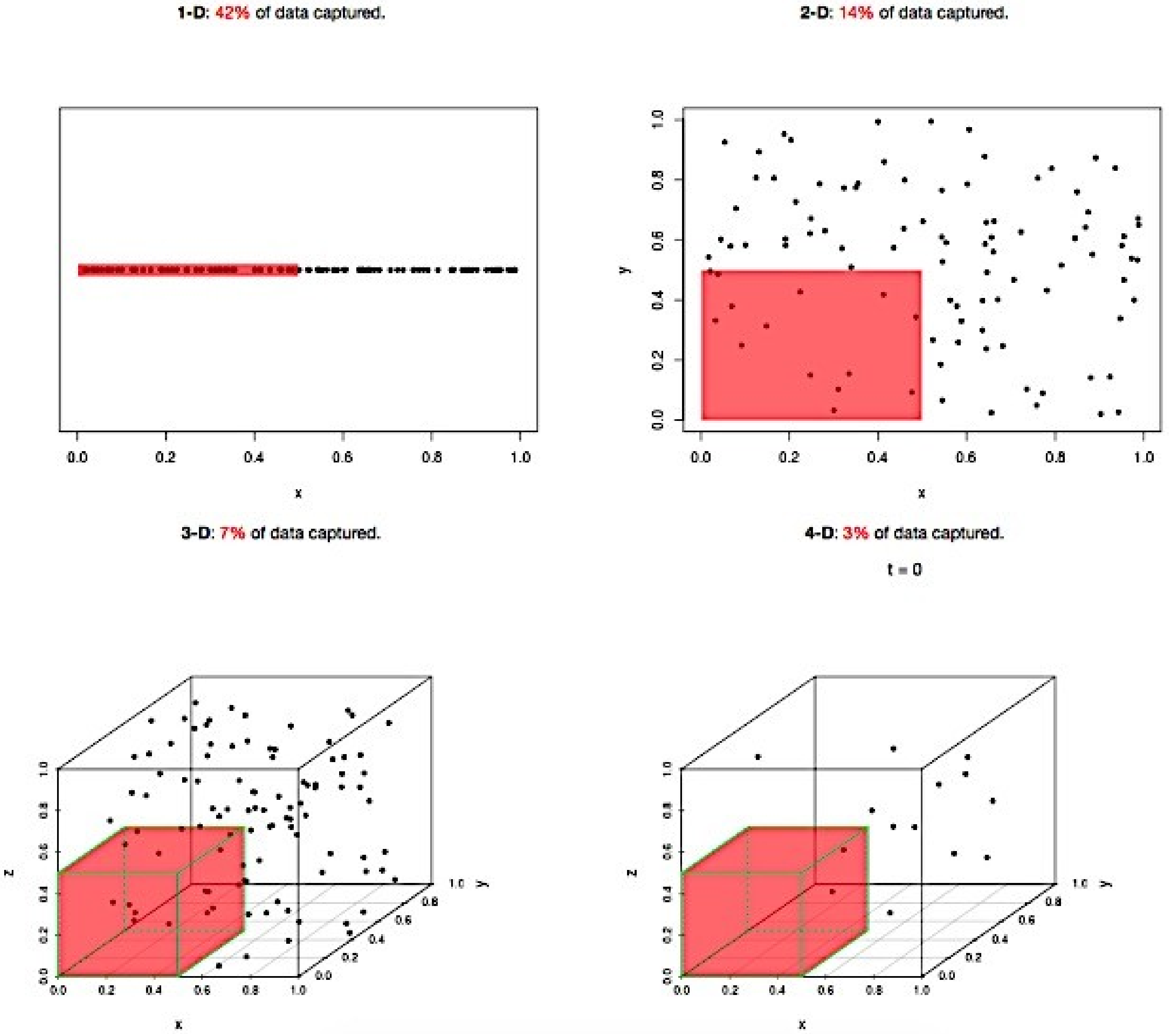
- Thus, contrary to intuition, **a more powerful learner is not necessarily better than a less powerful one**

OVERFITTING HAS MANY FACES

- The true classifier is a set of rules, with up to 1000 examples, naive Bayes is more accurate than a rule learner. This happens despite naive Bayes’s false assumption that the frontier is linear! Situations like this are common in machine learning: strong false assumptions can be better than weak true ones, because a learner with the latter needs more data to avoid overfitting



CURSE of DIMENSIONALITY



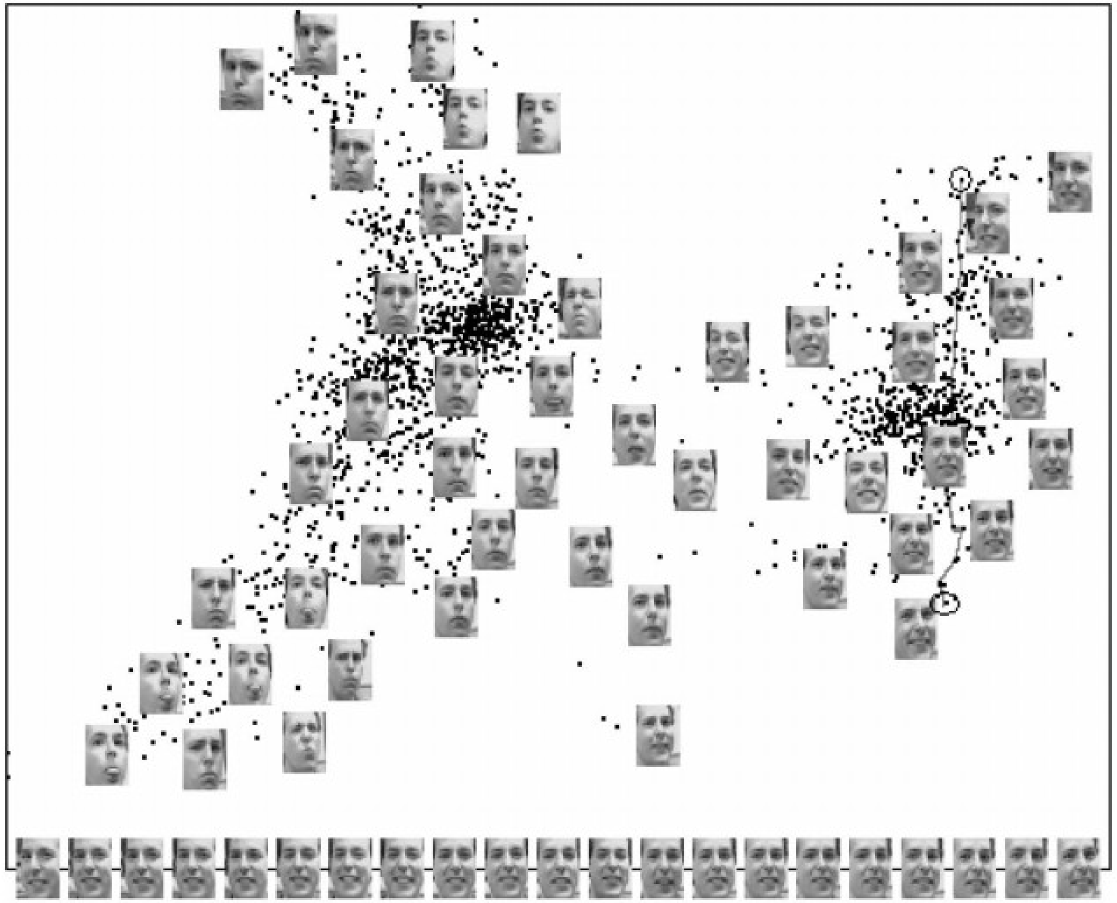
Source: http://www.newsnshit.com/curse-of-dimensionality-interactive-demo/

INTUITION FAILS IN HIGH DIMENSIONS"

- **Curse of dimensionality**: many algorithms that work fine in low dimensions become intractable when the input is high-dimensional.

- There is an effect that partly counteracts the curse, which might be called the ***“blessing of non-uniformity”***. In some applications examples are not spread uniformly throughout the instance space, but are concentrated on or near a lower-dimensional manifold

INTUITION FAILS IN HIGH DIMENSIONS



Source: https://ieeexplore.ieee.org/document/1388260

FEATURE ENGINEERING IS THE KEY

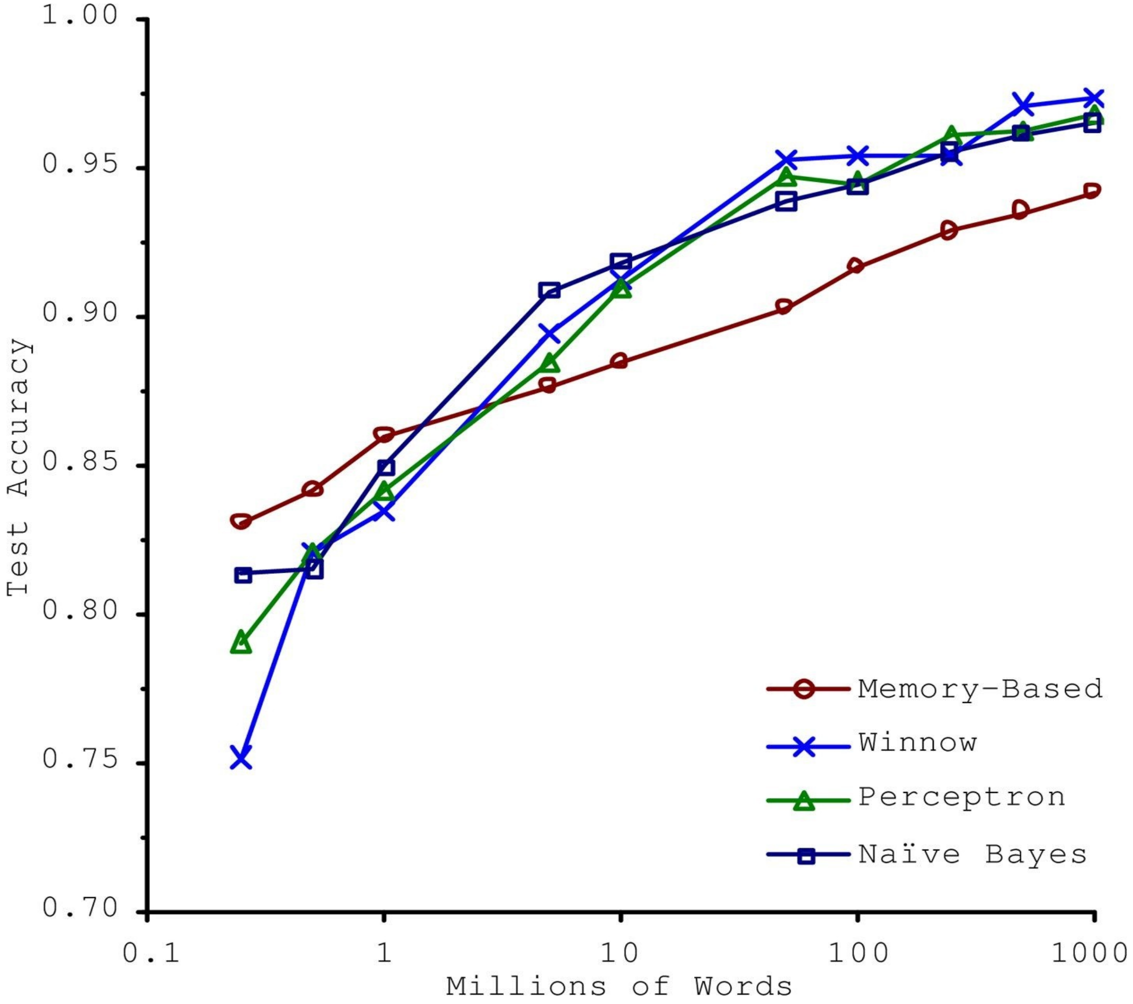
- Some machine learning projects succeed and some fail. What makes the difference? the most important factor is the **features** used.

- Often,the raw data is not in a form that is amenable to learning, but you can construct features from it.

- machine learning is not a one-shot process of building a data set and running a learner, but rather an **iterative process** of running the learner, analyzing the results, modifying the data and/or the learner, and repeating

MORE DATA BEATS A CLEVERER ALGORITHM

-



MORE DATA BEATS A CLEVERER ALGORITHM

- As a rule, it pays to try the **simplest learners first** (e.g., naive Bayes before logistic regression, k-nearest neighbor before support vector machines). More sophisticated learners are seductive, but they are usually harder to use, because they have more knobs you need to turn to get good results, and because their internals are more opaque

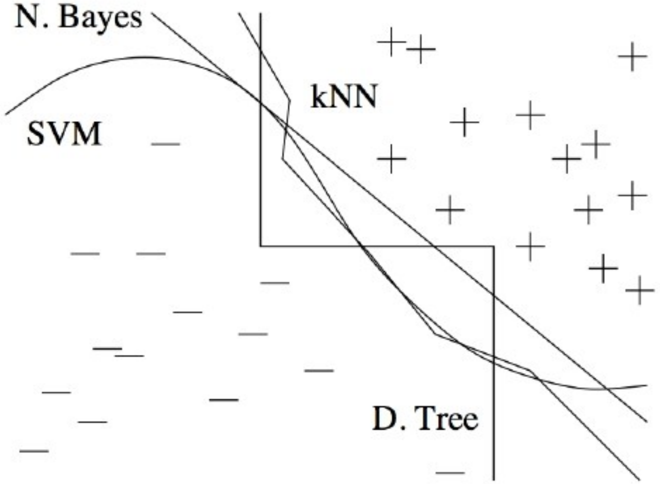


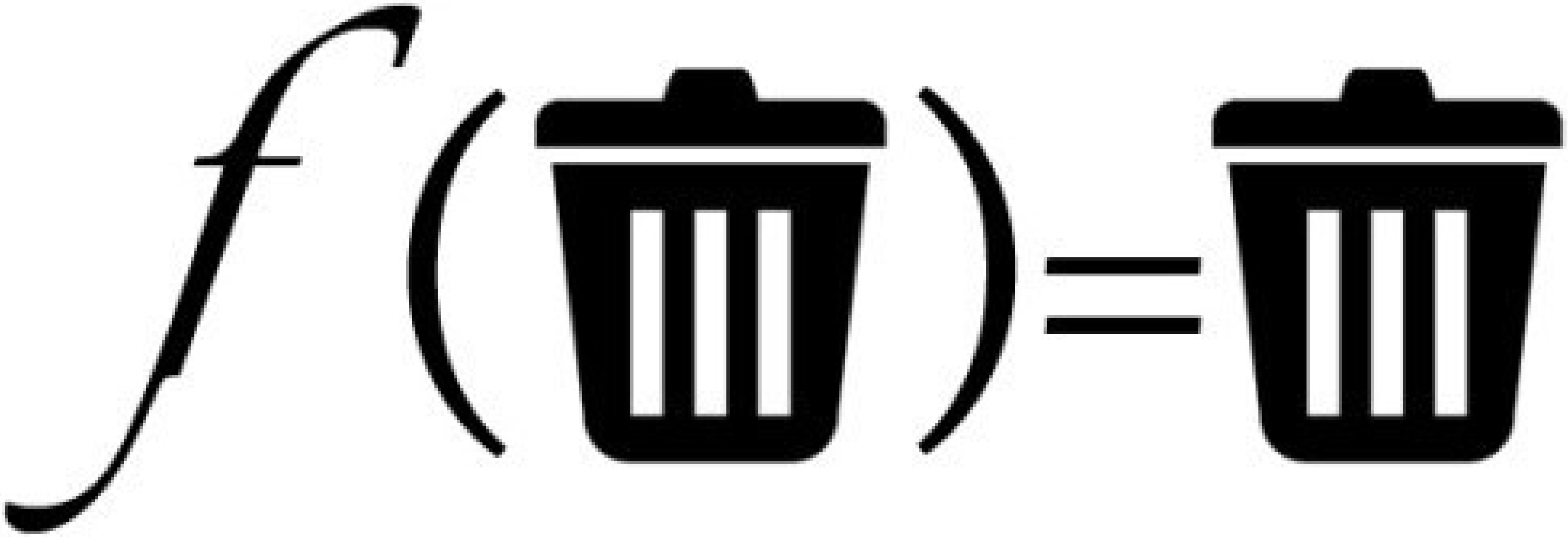
Figure 3: Very different frontiers can yield similar class predictions. (+ and − are training examples of two classes.)

DATA ALONE IS NOT ENOUGH

- Every learner must **embody some knowledge** or assumptions beyond the data it's given.
- One of the key criteria for choosing a representation is which kinds of knowledge are easily expressed in it:
  - If we have a lot of knowledge about what makes examples similar in our domain, *instance-based methods* may be a good choice.
  - If we have knowledge about probabilistic dependencies, *graphical models* are a good fit.
  - And if we have knowledge about what kinds of preconditions are required by each class, "IF . . . THEN . . ." *rules* may be the the best option.

DATA ALONE IS NOT ENOUGH

GIGO PRINCIPLE



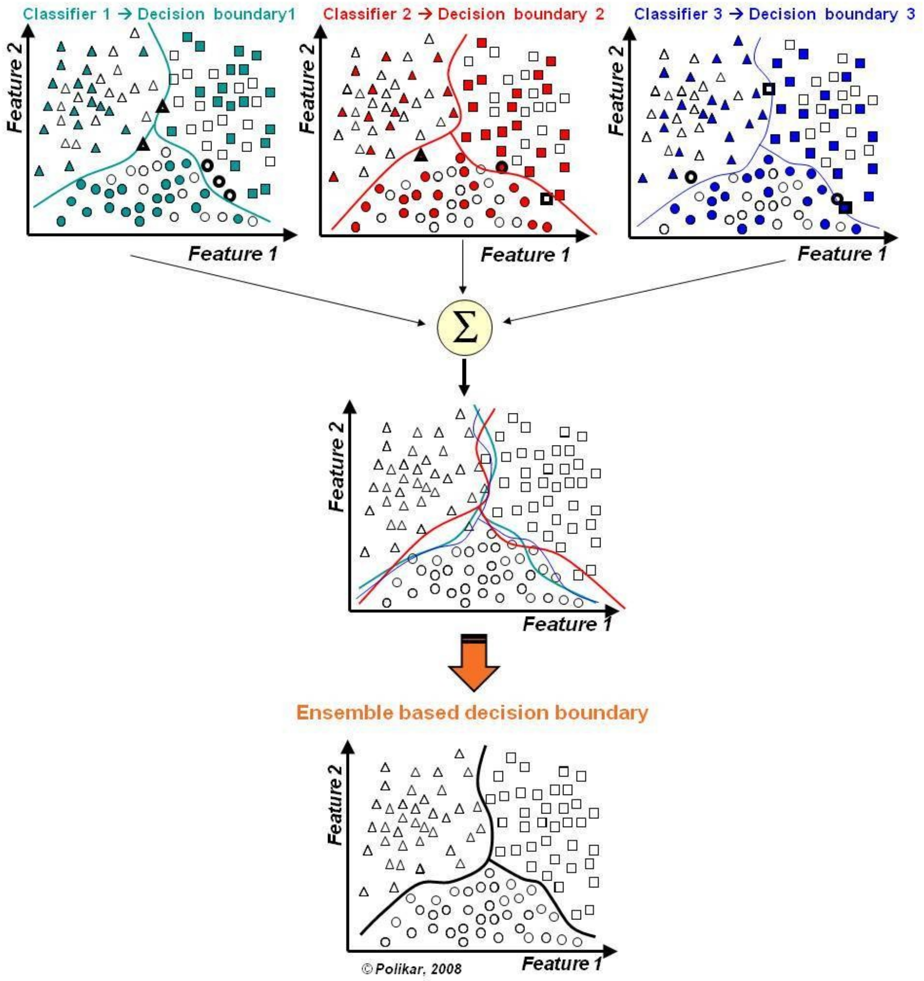
LEARN MANY MODELS, NOT JUST ONE

- Before, everyone had their favorite learner, with some reasons to believe in its superiority. Most effort went into trying many variations of it and selecting the best one.

- **No Free Lunch Theorem**: There is no single best model that works best for all problems

- **Ensemble:** If instead of selecting the best variation found, we combine many variations, the results are better.

LEARN MANY MODELS, NOT JUST ONE


Classifier 1 → Decision boundary1
Classifier 2 → Decision boundary 2
Classifier 3 → Decision boundary 3
Feature 2
Feature 1
Σ
Ensemble based decision boundary
©Polikar, 2008

Source: https://www.datasciencecentral.com/profiles/blogs/want-to-win-at-kaggle-pay-attention-to-your-ensembles

TAKE-HOME POINTS

- Be aware of overfitting
- Feature-Engineer the Curse of Dimensionality
- More and Better Data
- Combine data with expertise
- Ensemble your models

Syllabus, Goals, Practices and Assignments

PRACTICE → Feature Engineering

PRACTICE → Performance Metrics

PRACTICE → Tree-based Methods

Nearest Methods

PRACTICE → Naïve Bayes

Discriminant Analysis

PRACTICE → Dimensionality Reduction

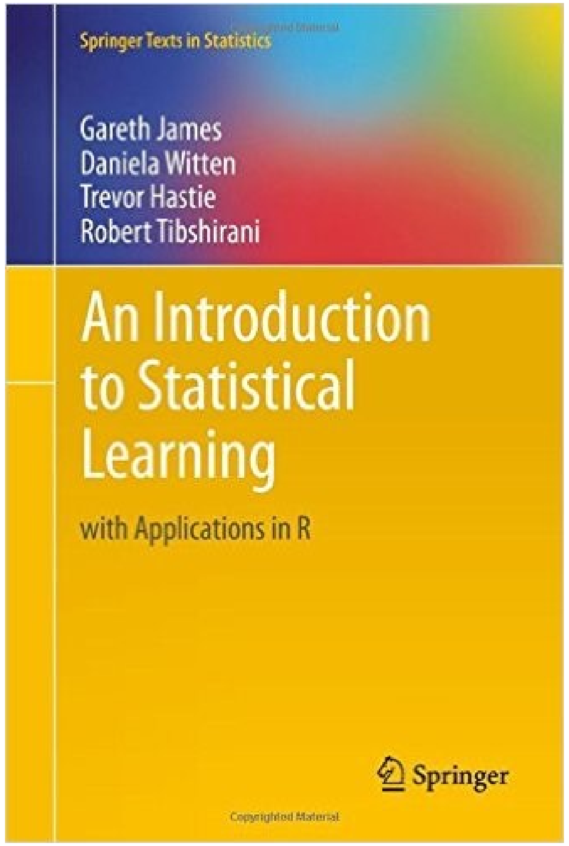
PRACTICE → Support Vector Machines

1st Assignment (25%)

2nd and 3rd Assignment (25%)

EXAM (40%)

Books



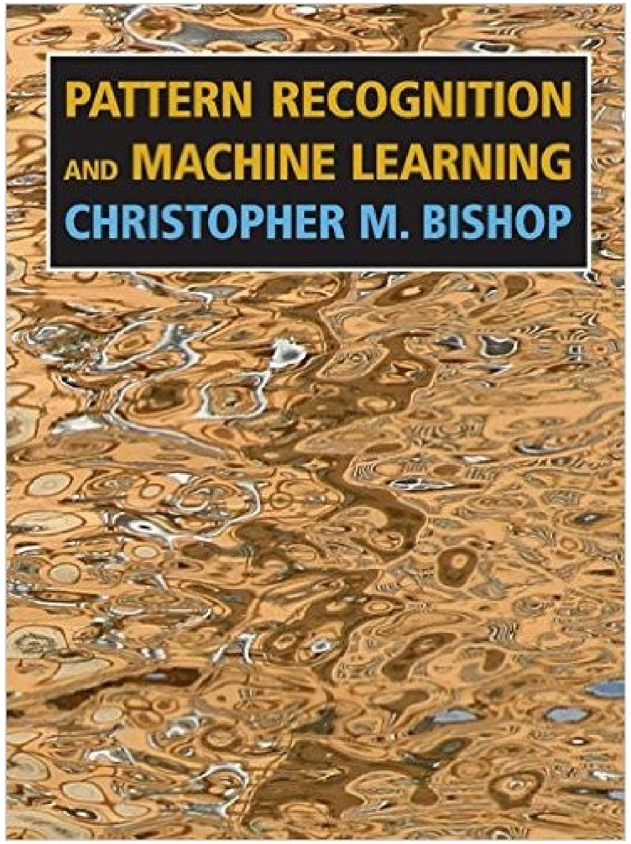
Springer Texts in Statistics

Gareth James
Daniela Witten
Trevor Hastie
Robert Tibshirani

An Introduction to Statistical Learning

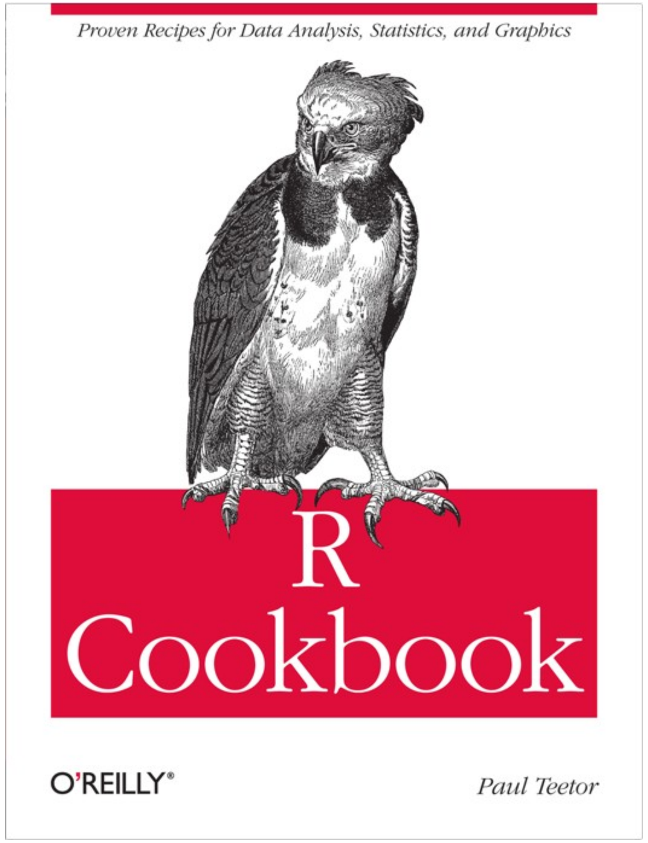
with Applications in R

Springer



PATTERN RECOGNITION AND MACHINE LEARNING

CHRISTOPHER M. BISHOP



Proven Recipes for Data Analysis, Statistics, and Graphics

R Cookbook

O'REILLY®

Paul Teetor

Need Help?

Ofrecido por Google en: English català galego euskara