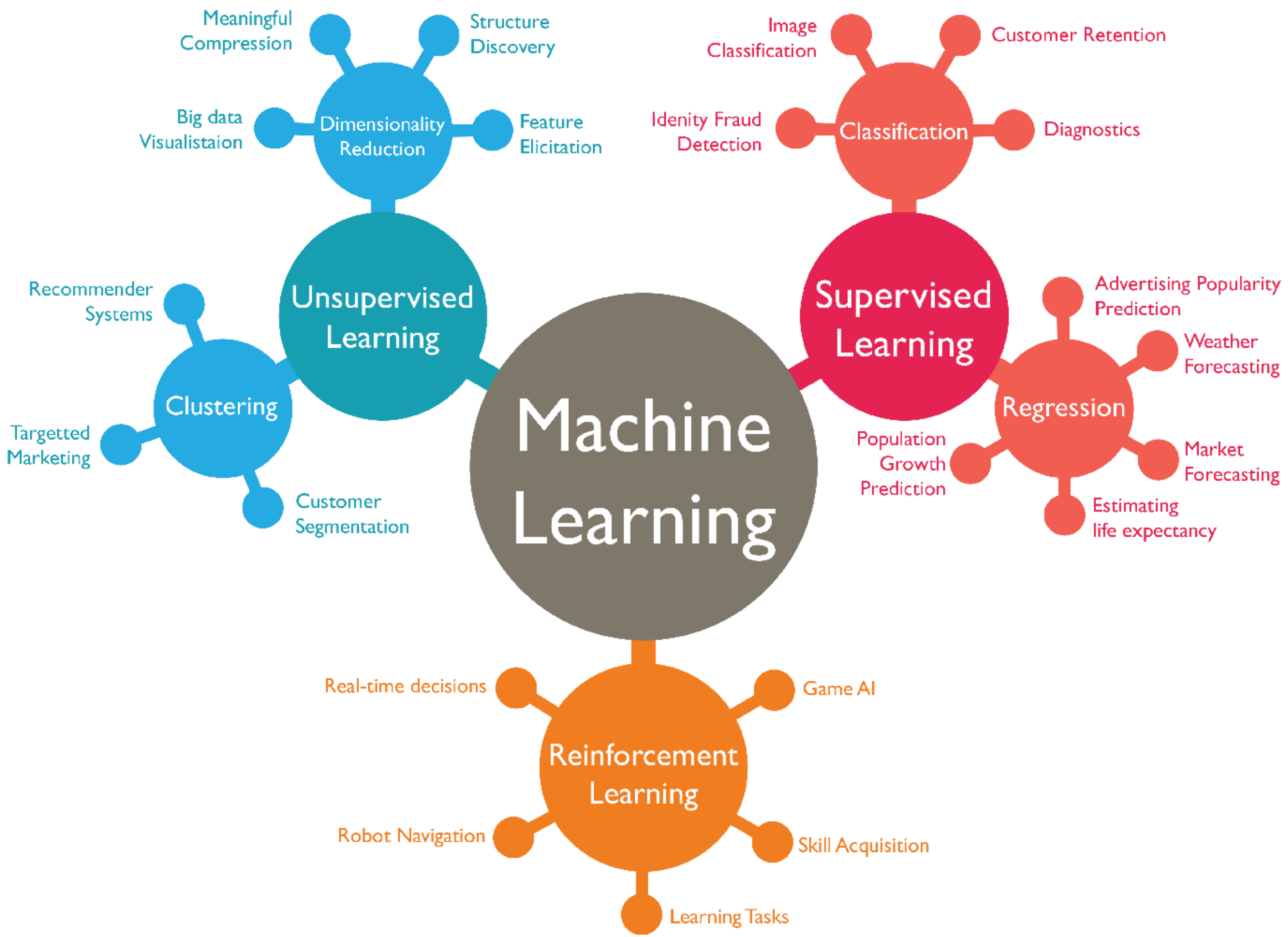
Machine learning

Introduction

Mohammad-Reza A. Dehaqani

dehaqani@ut.ac.ir

Data driven machine learning


Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks

Machine **Perception**

- Build a machine that can recognize patterns:

  - Speech recognition
  - Fingerprint identification
  - OCR (Optical Character Recognition)
  - DNA sequence identification

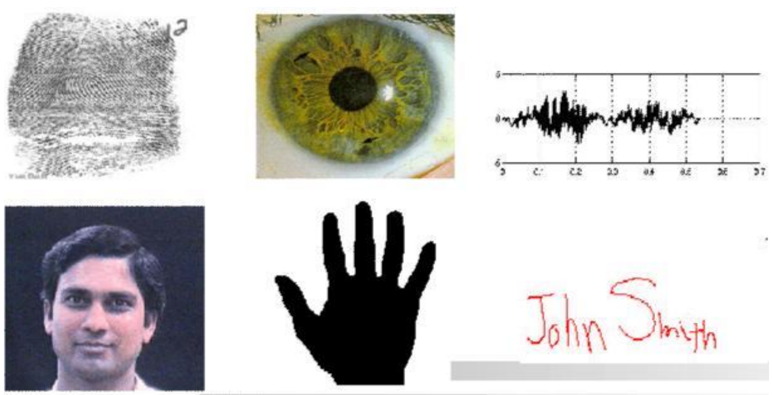
Pattern recognition

- Classify **objects** (instances, examples) into **categories** (classes, labels)
- Has **deep roots** in:
  - probability theory
  - statistics
  - machine learning
  - linear algebra
  - image processing,
  - algorithms
  - ...

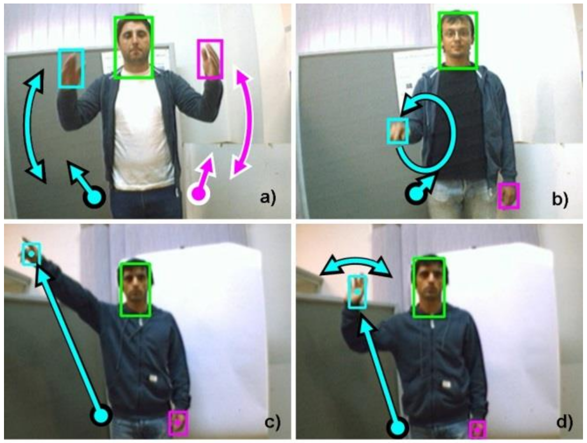
What is a Pattern?

- A pattern could be an object or event.
  - Typically, represented by a vector **x** of numbers

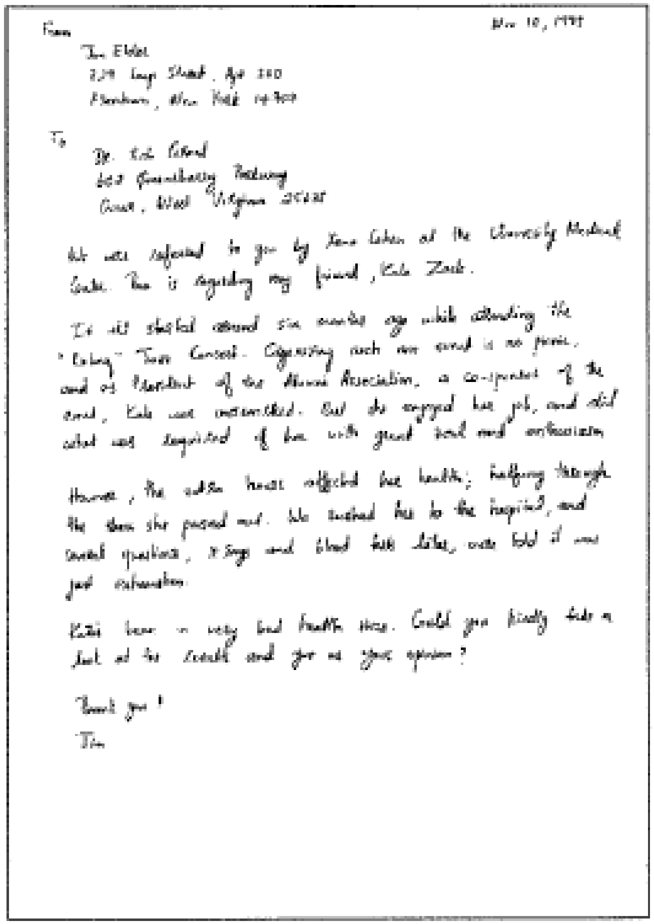
biometric patterns

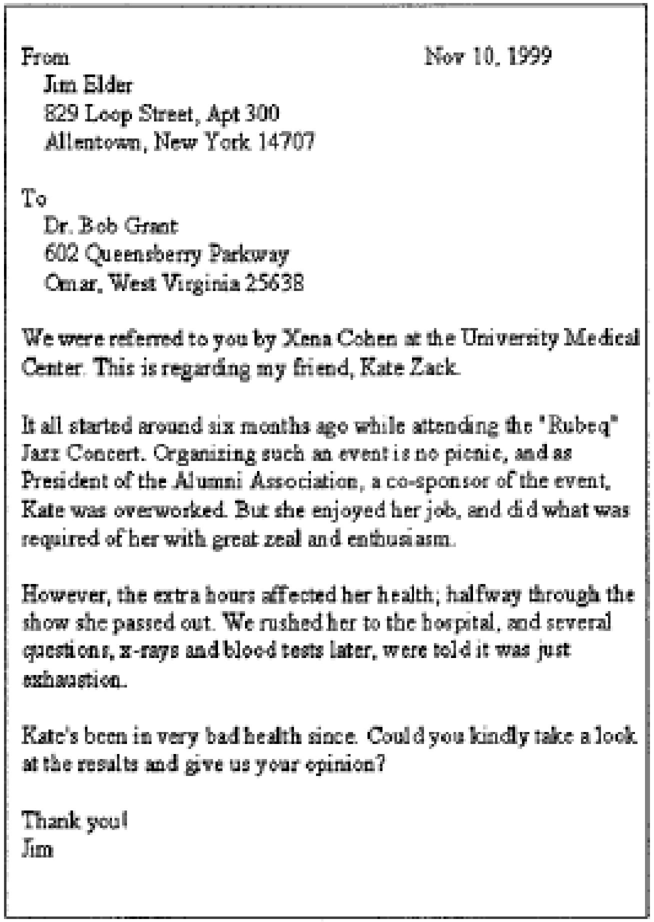


hand gesture patterns



Handwriting Recognition





From Nov 10, 1999
Jim Elder
829 Loop Street, Apt 300
Allentown, New York 14707

To
Dr. Bob Grant
602 Queensberry Parkway
Omar, West Virginia 25638

We were referred to you by Xena Cohen at the University Medical Center. This is regarding my friend, Kate Zack.

It all started around six months ago while attending the "Rubeq" Jazz Concert. Organizing such an event is no picnic, and as President of the Alumni Association, a co-sponsor of the event, Kate was overworked. But she enjoyed her job, and did what was required of her with great zeal and enthusiasm.

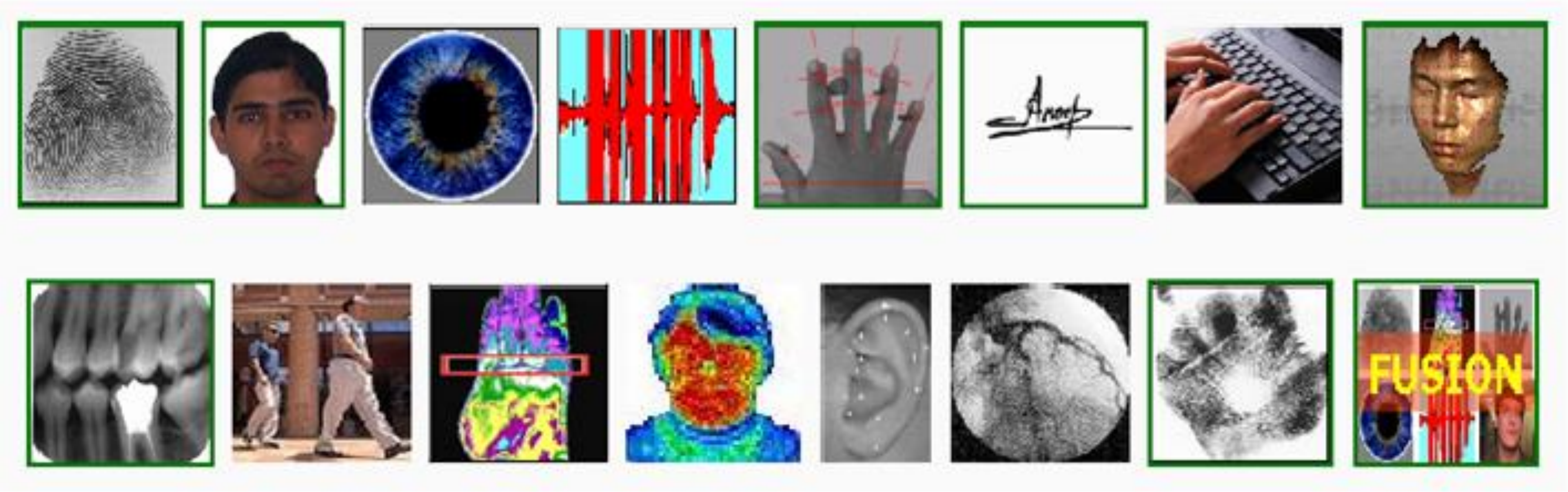
However, the extra hours affected her health; halfway through the show she passed out. We rushed her to the hospital, and several questions, x-rays and blood tests later, were told it was just exhaustion.

Kate's been in very bad health since. Could you kindly take a look at the results and give us your opinion?

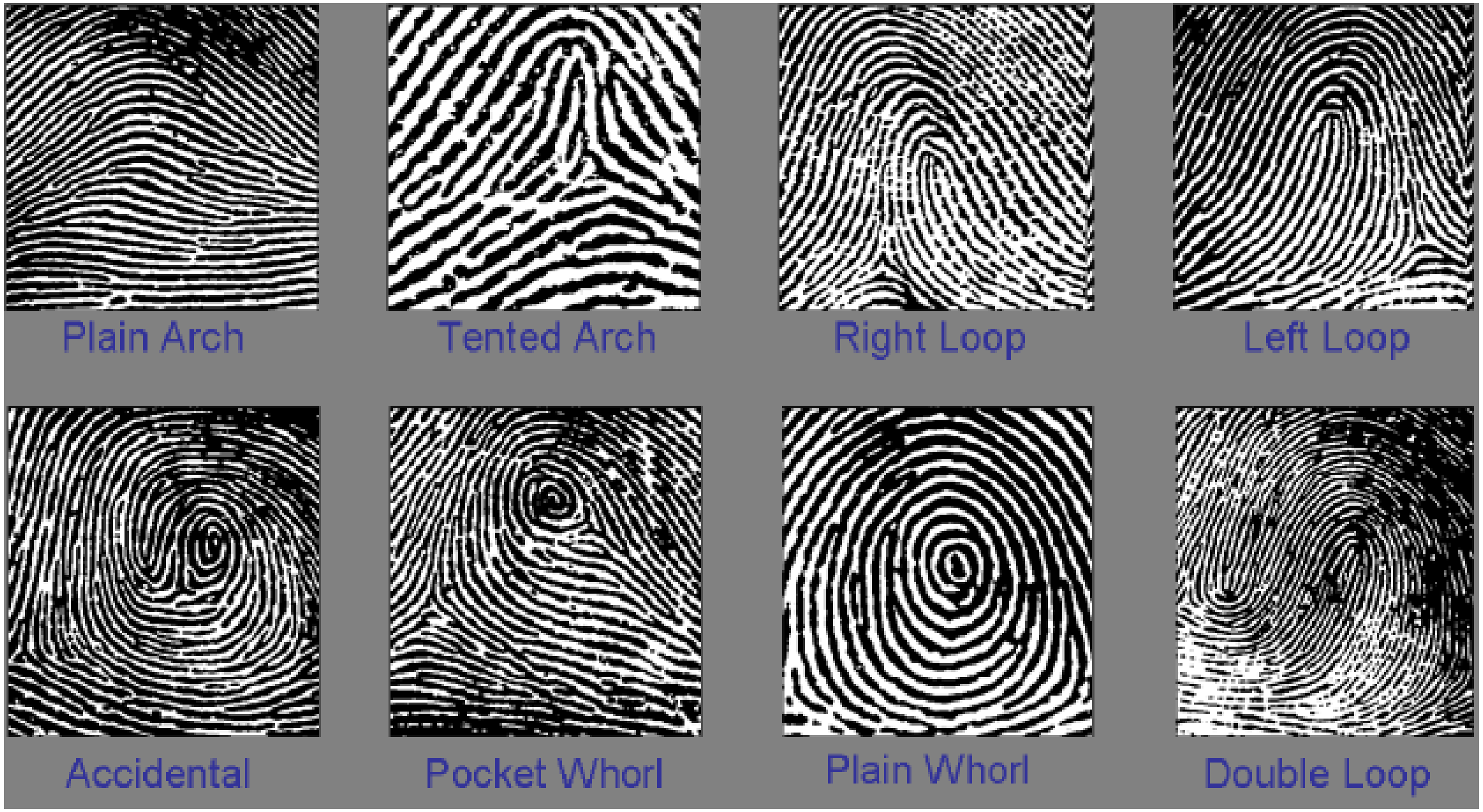
Thank you!
Jim

License Plate Recognition

Biometric Recognition



Fingerprint Classification

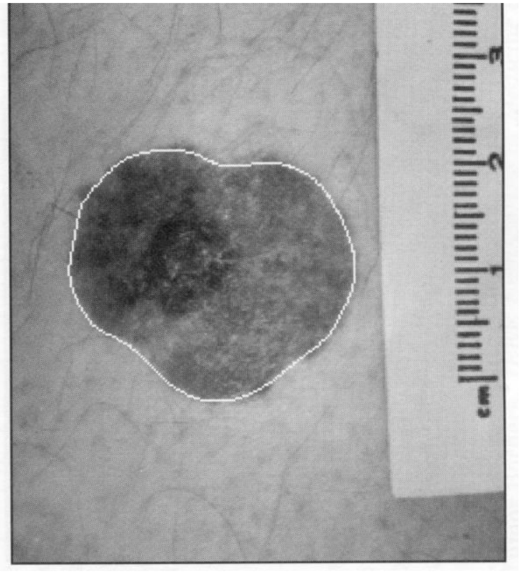


Face Detection

Autonomous Systems

Medical Applications

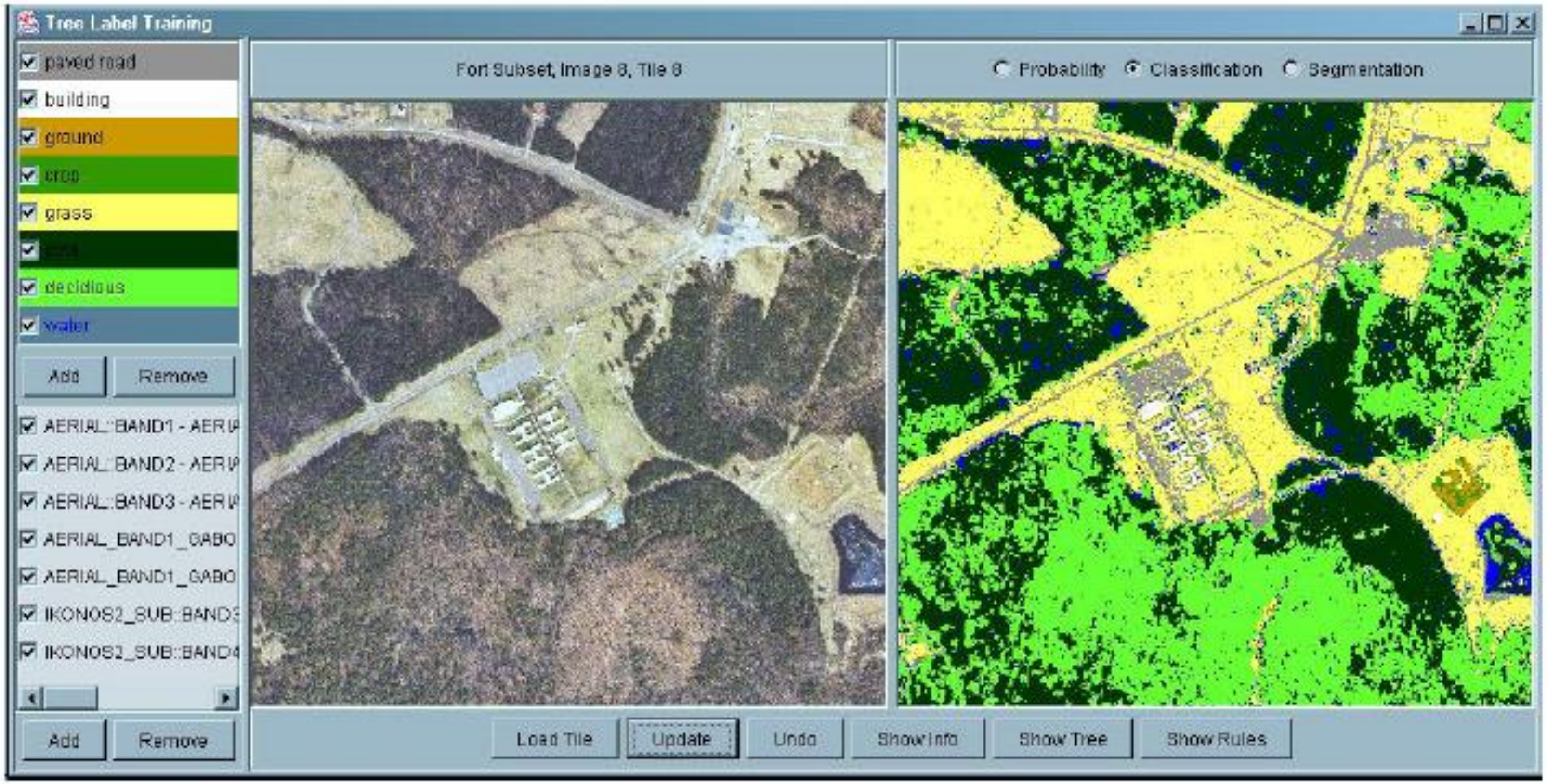
Skin Cancer Detection



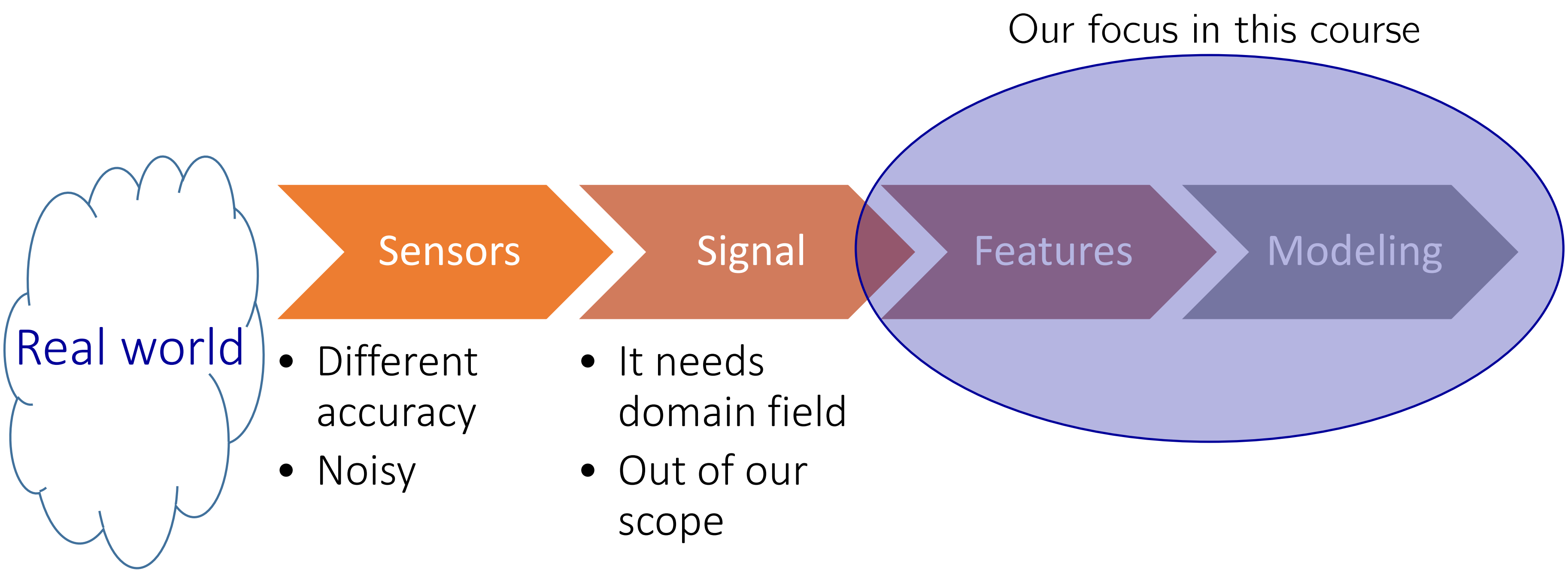
Breast Cancer Detection

Land Classification

(from aerial or satellite images)



Signal vs. feature



Core material

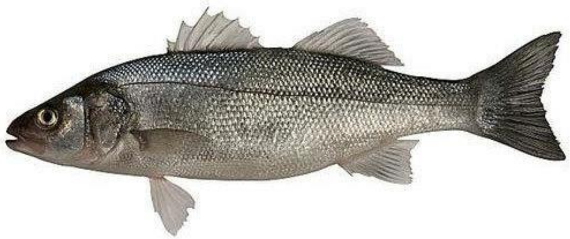
- **Finding patterns** in data; using them to make predictions.
- **Models** and statistics help us understand patterns.
- **Optimization** algorithms "learn" the patterns.
- The most important part of this is the **data**. Data drives everything else.
    - You cannot learn much if you don't have enough data.
- **Machine learning** has changed a lot in the last decade because the internet has made truly vast quantities of **data available**.

An Example

- "Sorting incoming fish on a conveyor according to species using **optical sensing**"

Species

Sea bass



Salmon

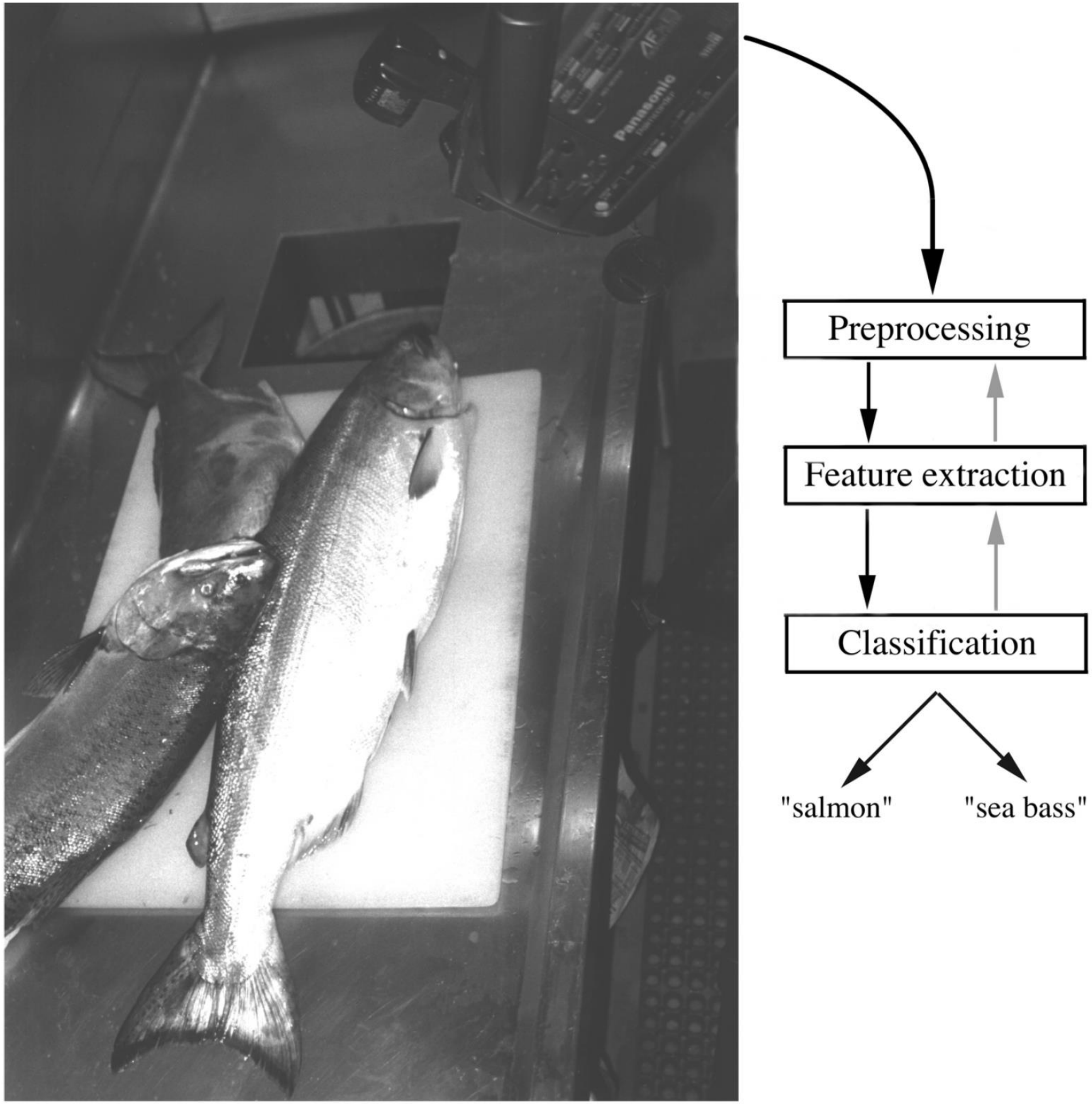
Problem Analysis

- Set up a camera **(sensors)** and take some sample images to extract features
  - Length
  - Lightness
  - Width
  - Number and shape of fins
  - Position of the mouth, etc...

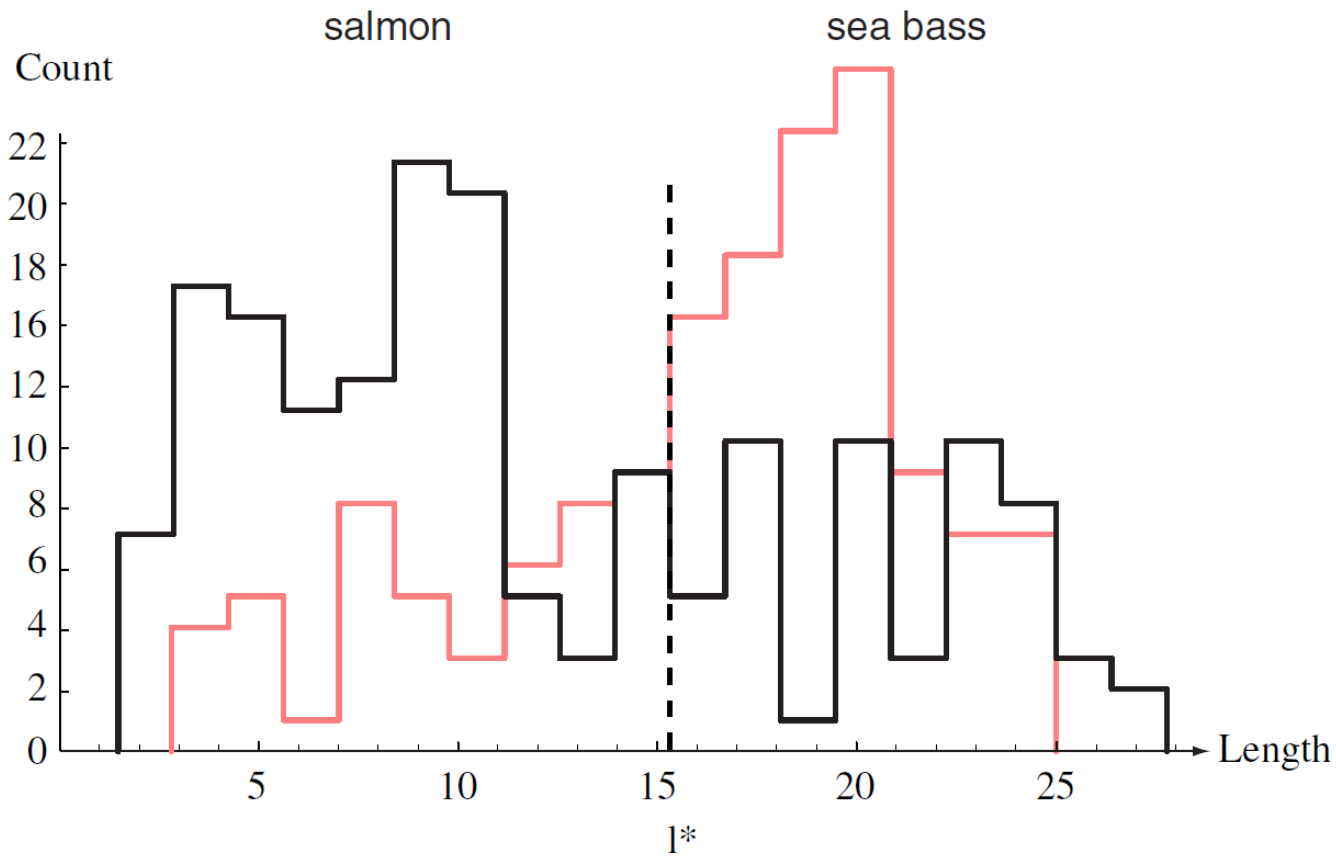
- This is the set of all suggested **features** to explore for use in our classifier!

Preprocessing to obtain features

- Use a **segmentation** operation to isolate fishes from one another and from the background
- Information from a single fish is sent to a feature extractor whose purpose is to **reduce** the data by measuring certain features
- The features are passed to a **classifier**

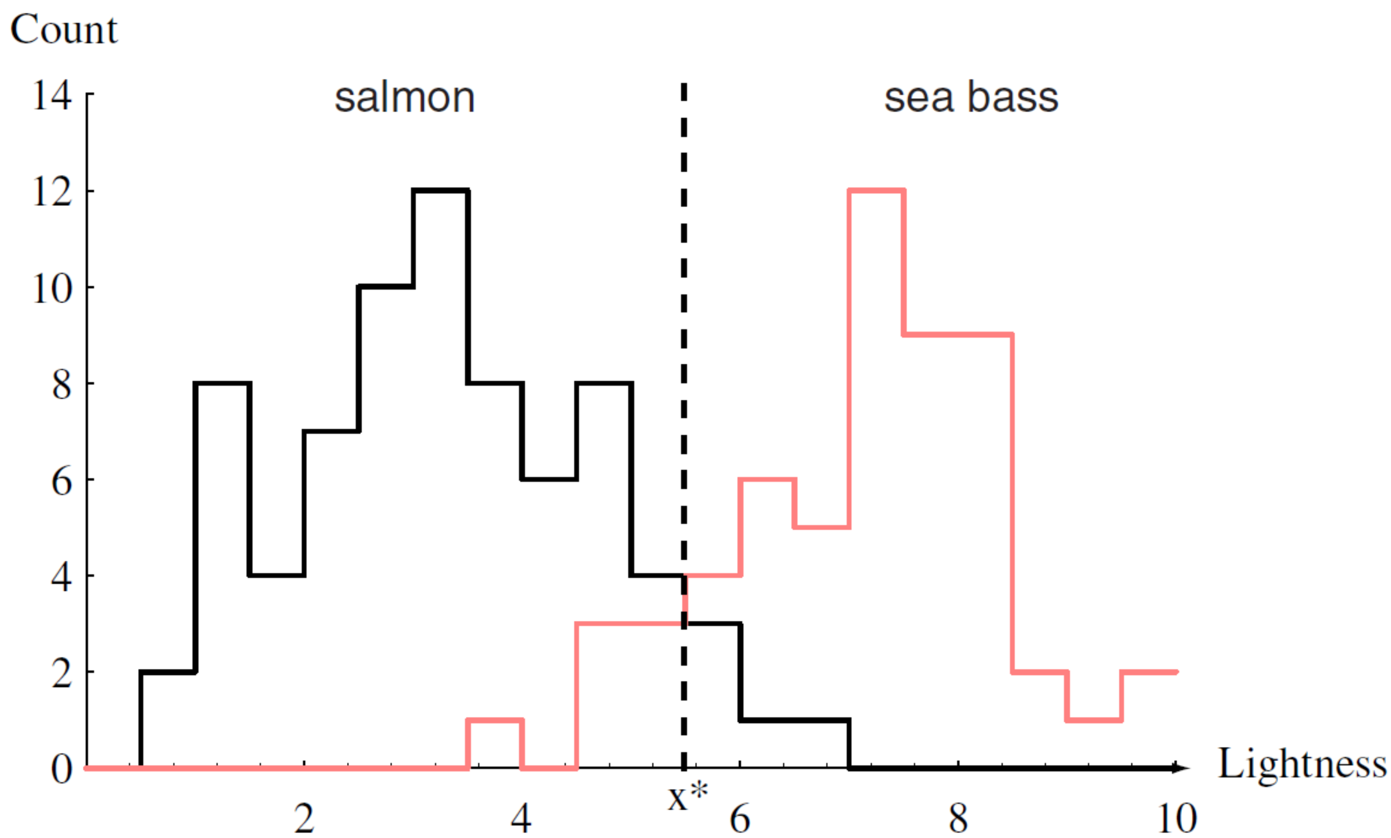

Preprocessing
Feature extraction
Classification
"salmon"
"sea bass"

Length feature for discrimination



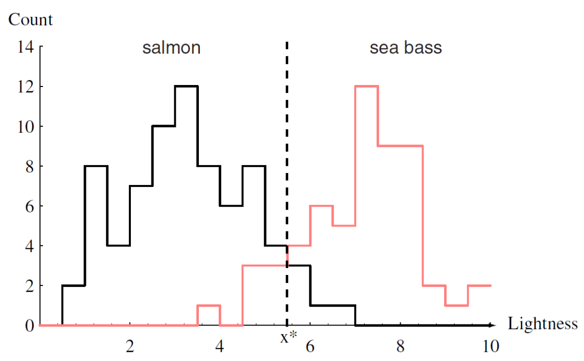
No single **threshold value l*** (**decision boundary**) will serve to unambiguously discriminate between the two categories; **using length alone**, we will have some errors. The value l* marked will lead to the **smallest number of errors**, on average.

Select the lightness as a possible feature



Decision theory;
Threshold decision boundary and cost relationship

Move our **decision boundary** toward smaller values of lightness in order to minimize the **cost** (reduce the number of sea bass that are classified salmon!)

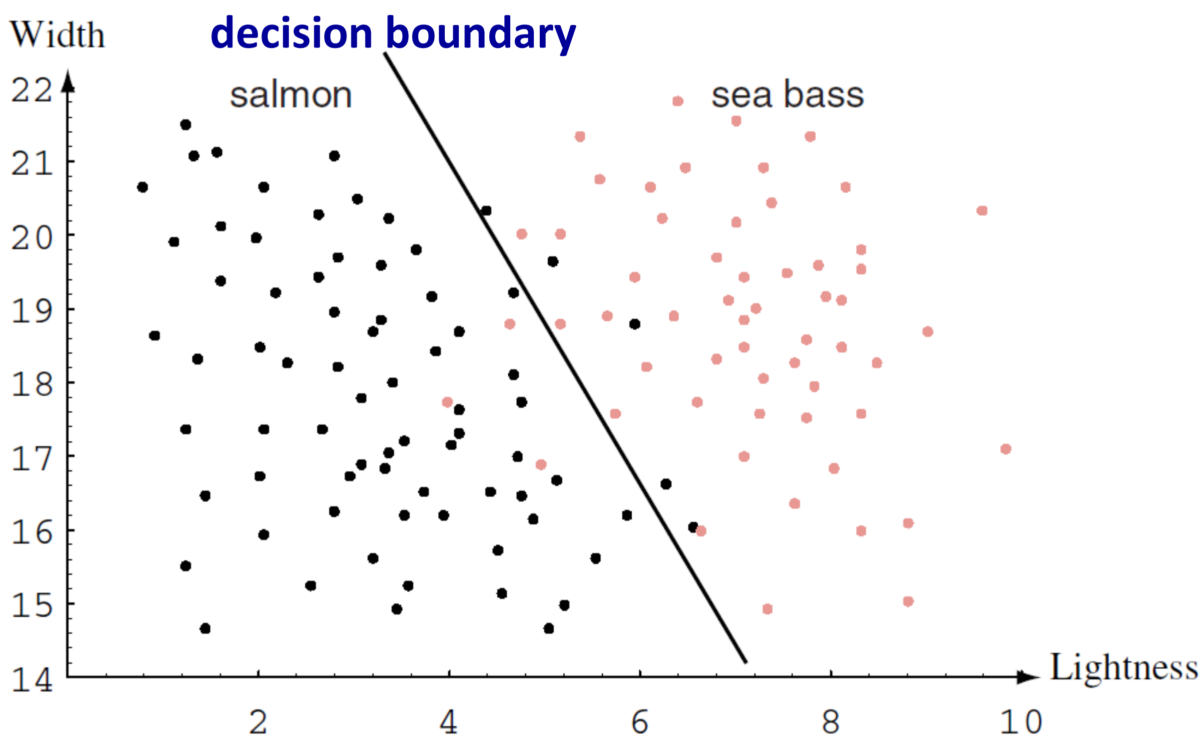


Task of decision theory

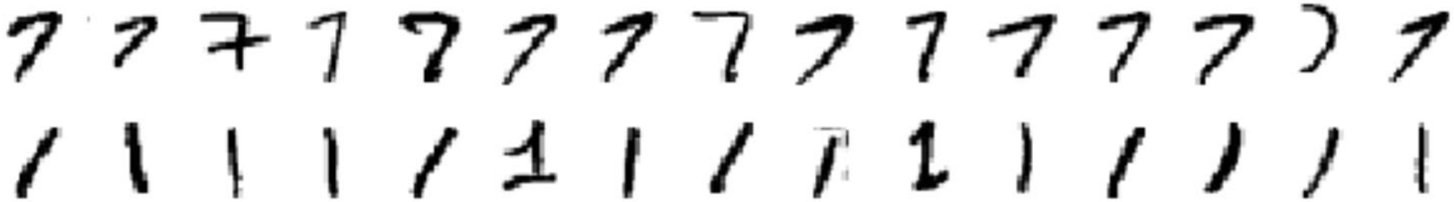
Multiple Features

Adopt the lightness and add the width of the fish

$$\text{Fish} \longrightarrow x^T = [\underbrace{x_1}_{\text{Lightness}},\ \underbrace{x_2}_{\text{Width}}]$$



Representation of digits



$$\begin{array}{|c|c|c|c|}\hline 3 & 3 & 3 & 3 \\ \hline 0 & 0 & 2 & 3 \\ \hline 0 & 0 & 1 & 3 \\ \hline 3 & 3 & 3 & 3 \\ \hline \end{array} \rightarrow \begin{bmatrix} 3 \\ 3 \\ 3 \\ 3 \\ 0 \\ 0 \\ 2 \\ 3 \\ 0 \\ 0 \\ 1 \\ 3 \\ 3 \\ 3 \\ 3 \\ 3 \end{bmatrix}$$

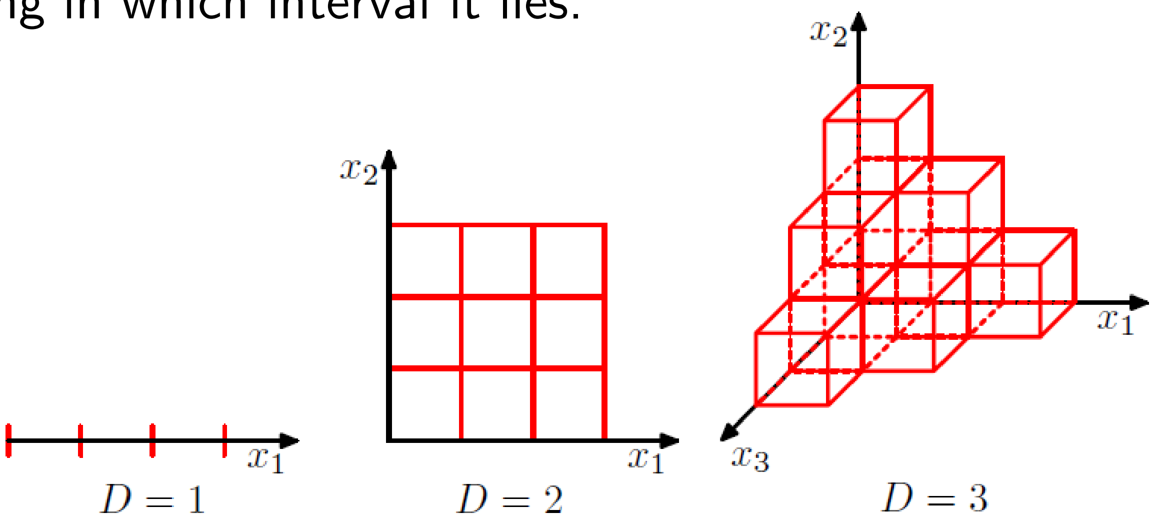
Images are points in 16-dimensional space. Linear decision boundary is a hyperplane

How Many Features?

- Does adding more features always improve performance?
  - It might be difficult and computationally expensive to extract certain features.
  - Correlated features might not improve performance (i.e. redundancy).
  - "Curse" of dimensionality.

Curse of Dimensionality

- Adding too many features can, **paradoxically**, lead to a **worsening** of performance.
  - Divide each of the input features into a number of intervals, so that the value of a feature can be specified approximately by saying in which interval it lies.

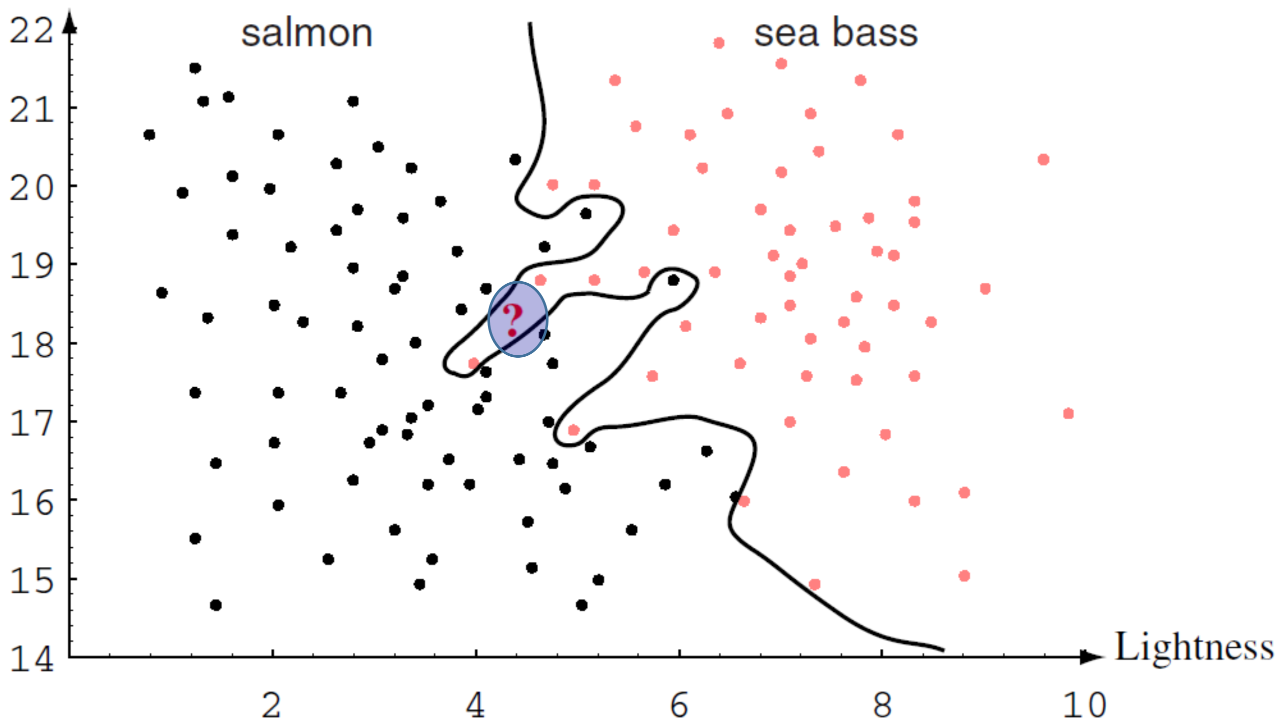


  - If each input feature is divided into **M** divisions, then the total number of cells is $\mathbf{M^d}$ (***d:*** # of features).
  - Since each cell must contain at least one point, the number of needed data grows exponentially with ***d***.

Issue of generalization

This **decision boundary** may lead to perfect classification of our **training samples**, it would lead to poor performance on future patterns (overfitting).

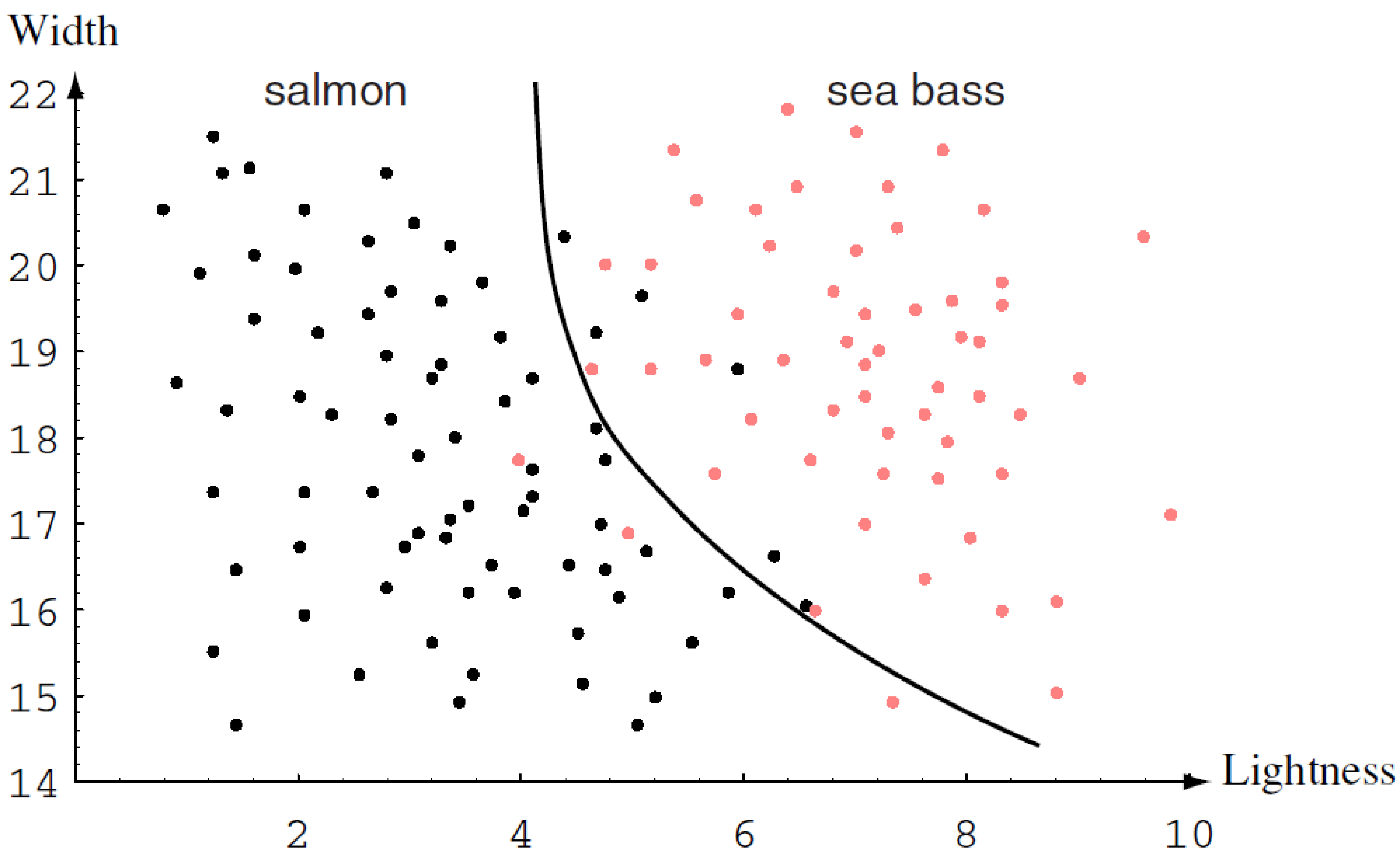
complex models for the fish



the **central aim** of designing a classifier is to correctly classify **novel input**

Optimal tradeoff

The decision boundary shown might represent the optimal tradeoff between **performance on the training set** and **simplicity of classifier**.



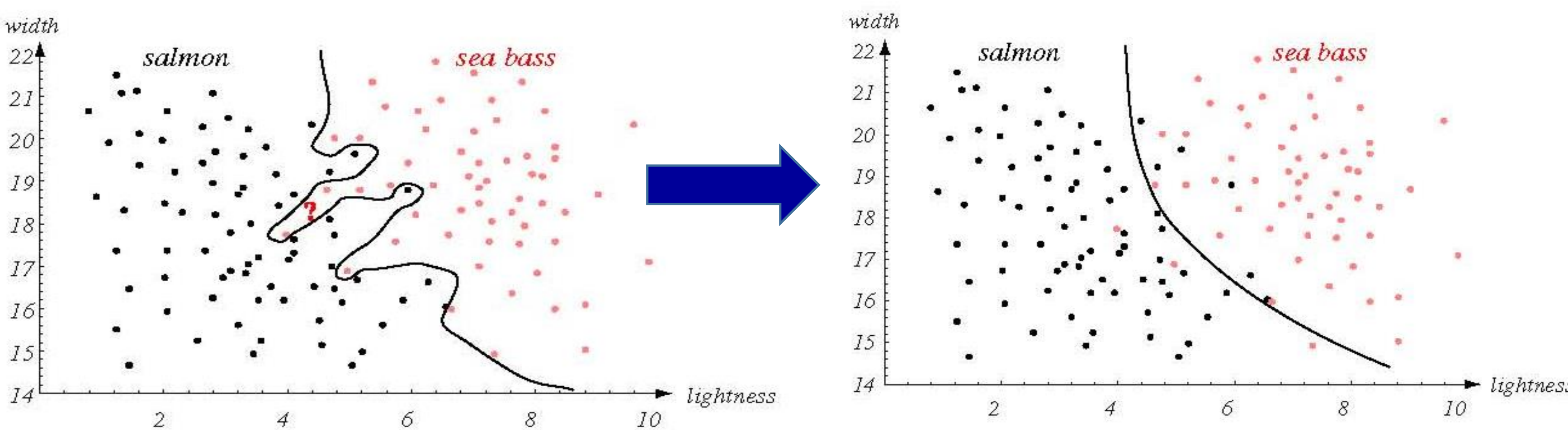
Generalization

- Generalization is defined as the ability of a classifier to produce correct results on **novel** patterns.

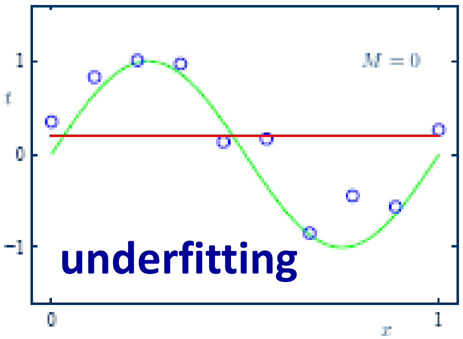
- How can we improve generalization performance ?
  - More training examples (i.e., better model estimates).
  - Simpler models usually yield better performance.

complex model

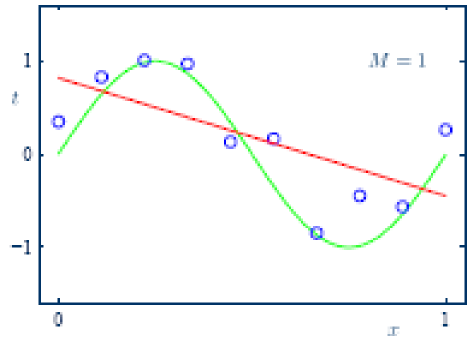
simpler model



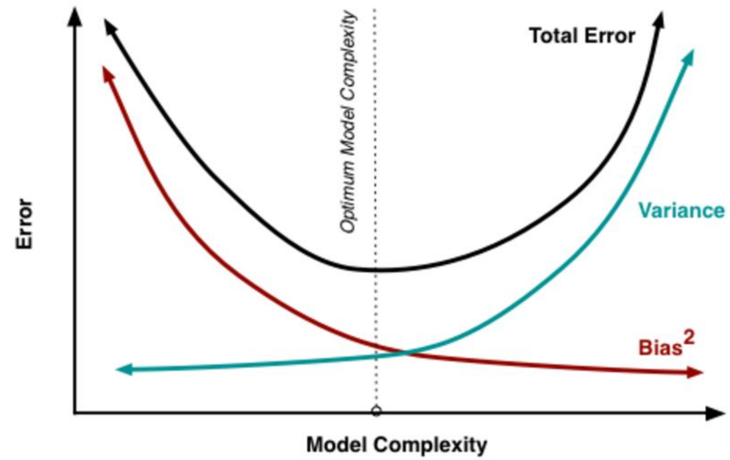
Understanding model complexity (theory of learning): The bias–variance tradeoff

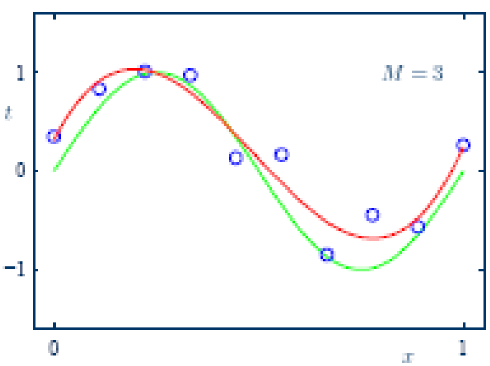


(a) 0'th order polynomial

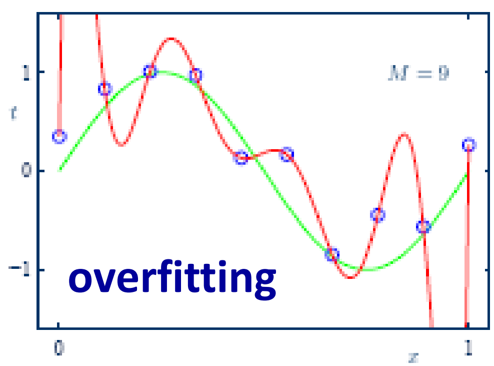


(b) 1'st order polynomial





(c) 3'rd order polynomial

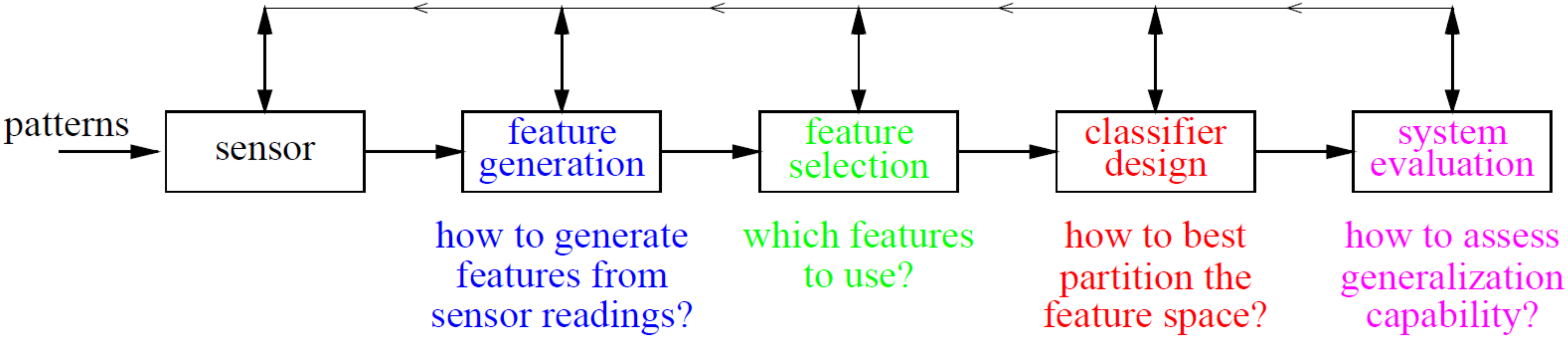


(d) 9'th order polynomial

bias–variance tradeoff is the property of a set of predictive models whereby models with a lower bias in parameter estimation have a higher variance of the parameter estimates across samples, and vice versa]

Underfitting vs. overfitting

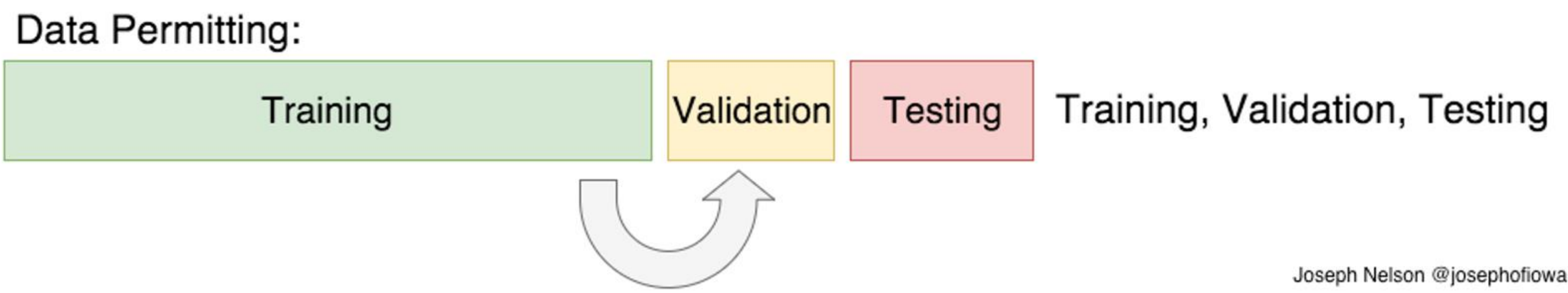
- The **bias** error: (**underfitting**)
  - From **erroneous assumptions** in the learning algorithm.
  - High bias can cause an algorithm to **miss the relevant relations** between features and target outputs.
- The **variance** error: (**overfitting**).
  - From **sensitivity** to **small fluctuations** in the training set.
  - High variance can cause an algorithm to **model the random noise** in the training data, rather than the intended outputs



- Feat. Gen.: Want to **reduce sensitivity to noise** and reduce complexity but **retain important Information**
  - Big sensor information into small number of features
- Feat. Sel.: Want to **reduce complexity** and reduce **redundancy** but retain important information
  - Select small set of features that **separates classes**
- Classif. Des.: Want **small generalization error** and fast training and classification (i.e. low complexity)
- Sys. Eval.: Want to accurately **estimate classier's generalization error**
- Some stages might be combined
- Feedback loops

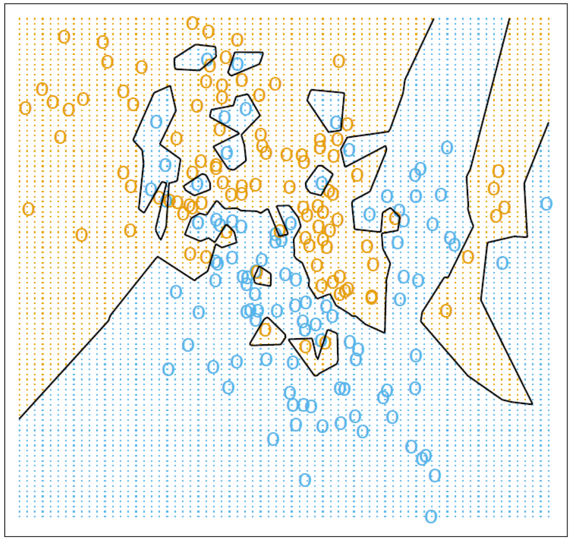
Now we have 3 sets:

- **training** set used to learn model weights
- **validation** set used to tune hyperparameters, choose among different models
- **test set** used as FINAL evaluation of model. Keep in a vault. Run ONCE, at the very end.

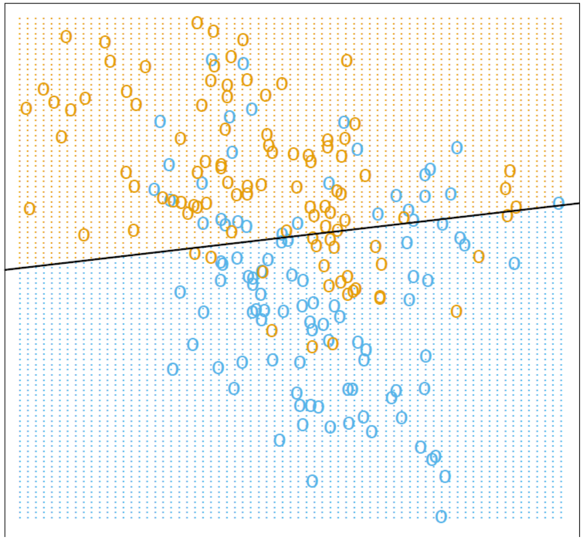


Classifier examples

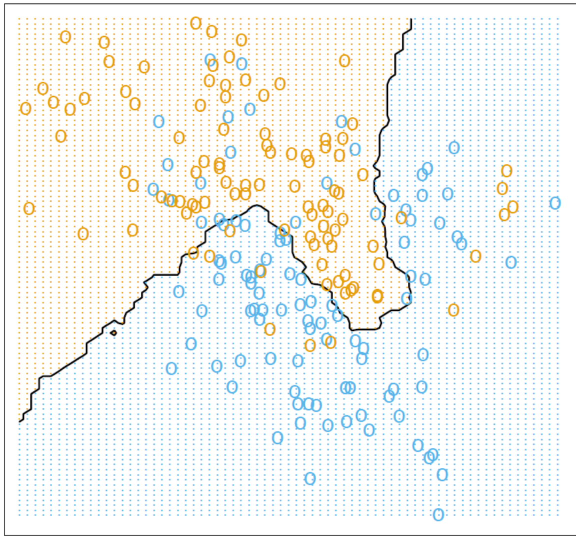
nearest neighbor classifier,



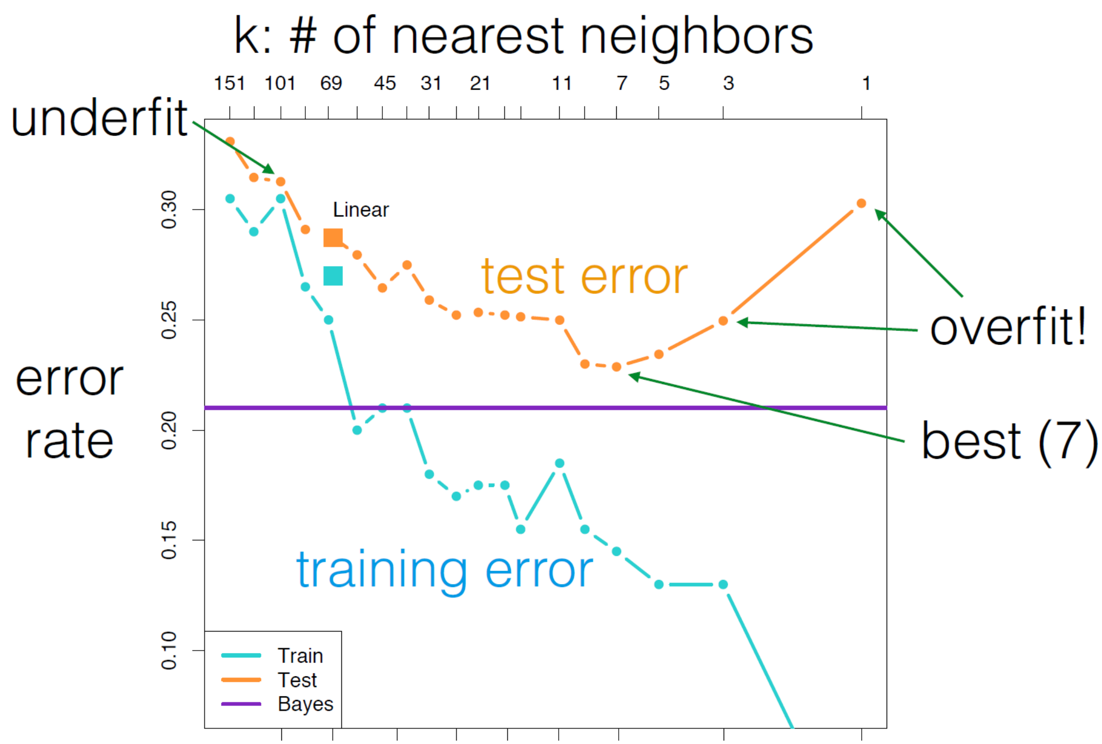
linear classifier



15-nearest neighbor classifier,

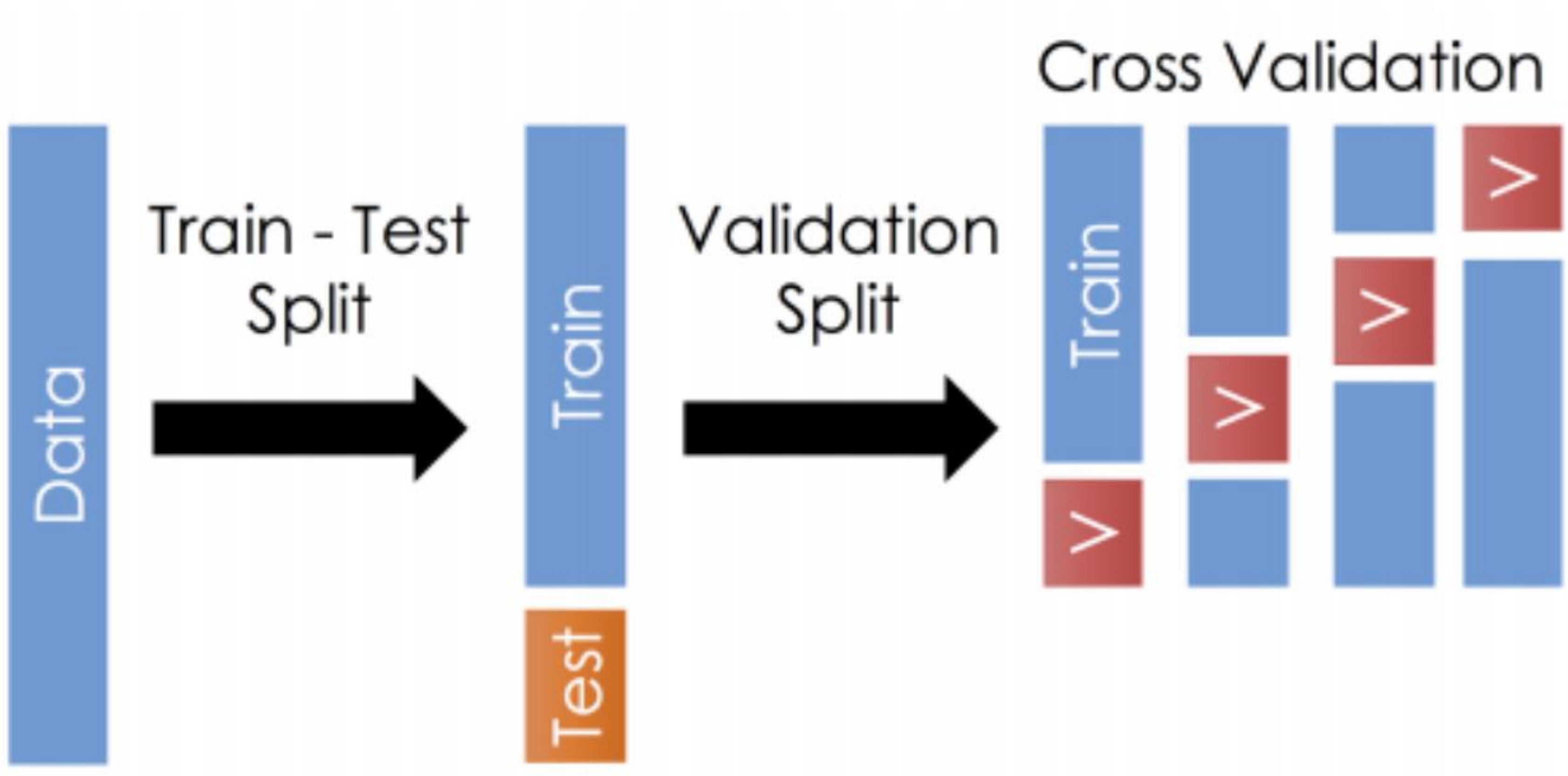


Hyperparameters



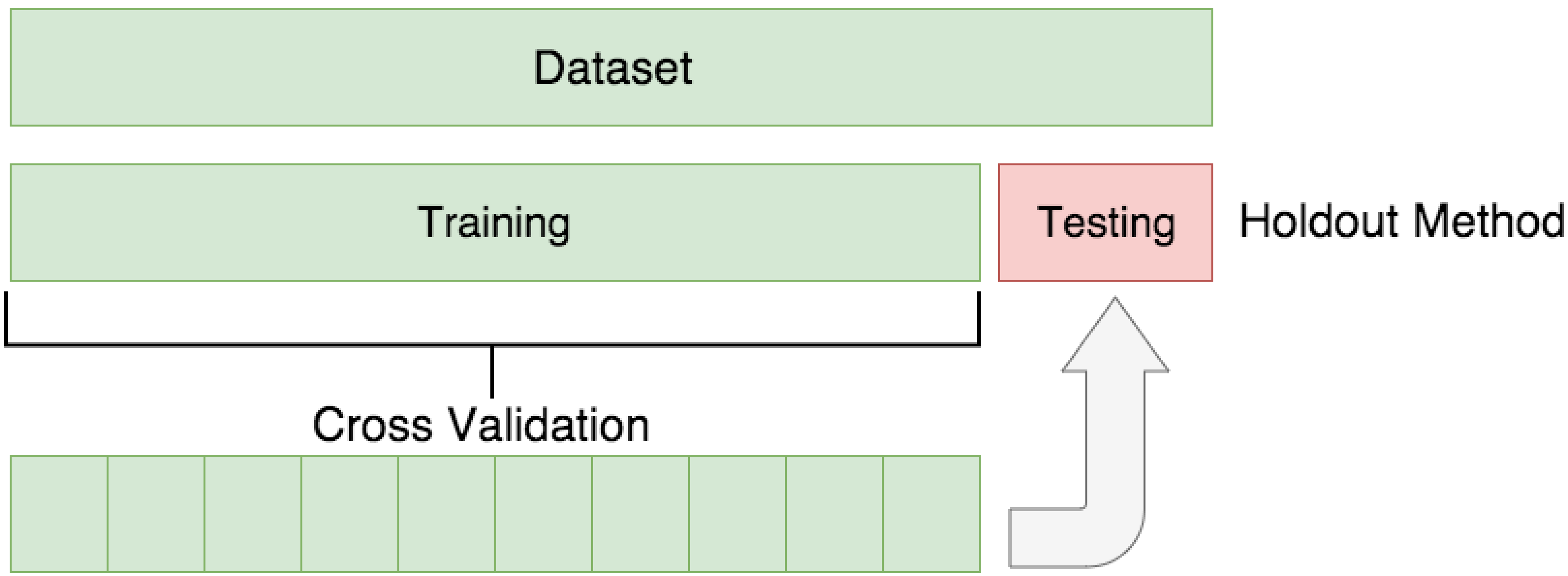
- Most ML algorithms have a few hyperparameters that control over/underfitting, e.g. k in k-nearest neighbors.
- We select them by validation

Cross validation



Holdout methods

- **K-Fold Cross Validation**
- **Leave P-out Cross Validation**
- **Leave One-out Cross Validation**



Post Processing

- Feature extraction
  - Discriminative features
  - Invariant features with respect to translation, rotation and scale.
- Classification
  - Use a feature vector provided by a feature extractor to assign the object to a category
- Post Processing
  - Exploit context input dependent information other than from the target pattern itself to improve performance

Feature Choice

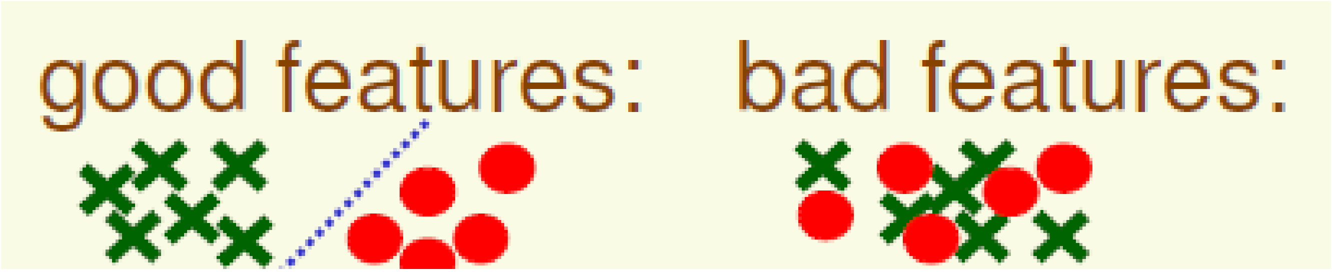
Depends on the characteristics of the problem domain.

- **Simple** to extract
- **Invariant** to irrelevant transformation
- **Insensitive** to noise.

Missing Features problem

"Quality" of Features

- How to choose a good set of features?
  - **Discriminative** features



  - **Invariant** features (e.g., invariant to geometric transformations such as translation, rotation and scale)

Missing Features

- Certain features might be missing (e.g., due to occlusion).
- How should we train the classifier with missing features ?
- How should the classifier make the best decision with missing features ?

Cost of miss-classifications

- Fish classification: two possible classification errors:

  (1) Deciding the fish was a sea bass when it was a salmon.

  (2) Deciding the fish was a salmon when it was a sea bass.

- Are both errors equally important ?

Cost of miss-classifications

- Suppose that:
  - Customers who buy salmon will object vigorously if they see sea bass in their cans. (false alarm; type I error)
  - Customers who buy sea bass will not be unhappy if they occasionally see some expensive salmon in their cans. (missed called; type II error)

- How does this knowledge affect our decision?

Computational Complexity

- What is the trade-off between **computational ease (or complexity)** and **performance**?

- (How an algorithm **scales** as a function of the number of features, patterns or categories?)

Time Complexity of Algorithms

- Big-Theta
  - The function g(n) is Θ(f(n)) iff there exist two real positive constants $c_1 > 0$ and $c_2 > 0$ and a positive integer n_0 such that:

$$c_1 f(n) \geq g(n) \geq c_2 f(n) \text{ for all } n \geq n_0$$

- Big-Oh
  - Upper bounds of complexity
- Big-Omega
  - Lower bound (g(n) $\geq c$f(n) for all $n \geq n_0$)

Ascending order of complexity

$1 \leftarrow \log\log n \leftarrow \log n \leftarrow n \leftarrow n\log n \leftarrow n^{\alpha};\ _{1<\alpha<2} \leftarrow n^2 \leftarrow n^3$

$\leftarrow n^m;\ _{m>3} \leftarrow 2^n \ldots$

Running time T(n)	Complexity O(n)
$n^2 + 100\,n + 1$	$O(n^2)$
$0.001n^3 + n^2 + 1$	$O(n^3)$
$23\,n$	$O(n)$
$100000\,n^2 + 10000\,n$	$O(n^2)$
2^{3+n}	$O(2^n)$ as $2^{3+n} \equiv 2^3 \cdot 2^n$
$2 \cdot 3^n$	$O(3^n)$