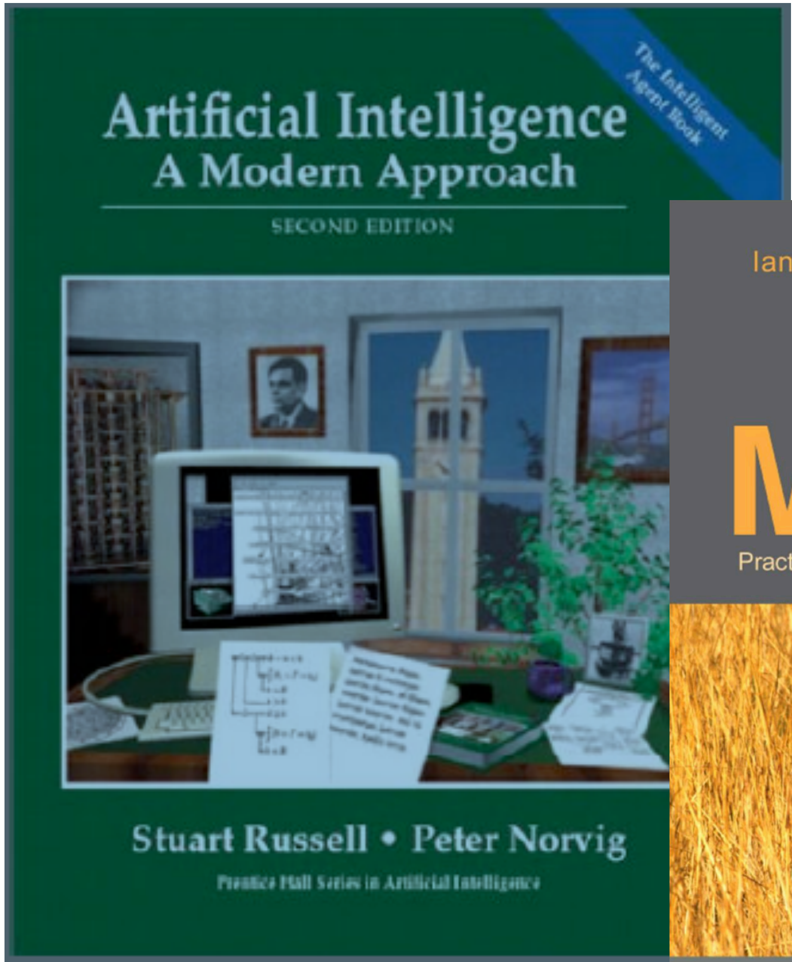
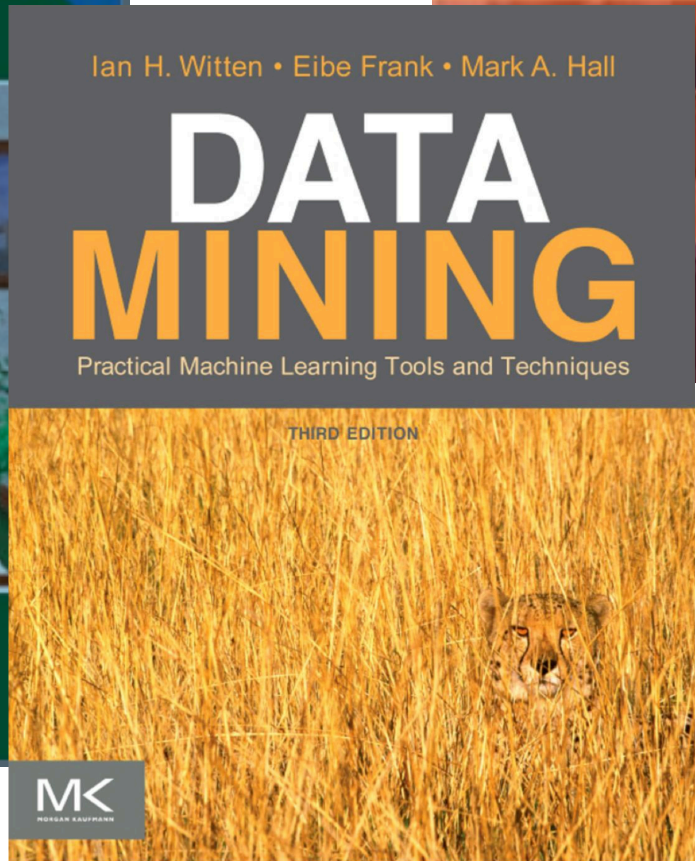
Foundations of Machine Learning and Data Mining

Rainer Marrone, Ralf Möller

Today's slides taken partly from E. ALPAYDIN

Literature


Artificial Intelligence
A Modern Approach
SECOND EDITION
The Intelligent Agent Book
Stuart Russell • Peter Norvig


Ian H. Witten • Eibe Frank • Mark A. Hall
DATA
MINING
Practical Machine Learning Tools and Techniques
THIRD EDITION
MK


Machine Learning

Lab Class and literature

- Thursday, 13:15 – 14:45, ES42 M2589

- Lab Class Fr 9:45-10:30, ES42 M2589
- First Lab Class 11.04.2011,
  Check StudIP for exercise sheets.

Why "Learn" ?

- Machine learning is programming computers to optimize a *performance criterion* using example data or past experience.
- There is no need to "learn" to calculate payroll
- Learning is used when:
  - Human expertise does not exist (navigating on planet X),
  - Humans are unable to explain their expertise (speech recognition)
  - Solution changes in time (routing on a computer network)
  - Solution needs to be adapted to particular cases (user biometrics)

What We Talk About When We Talk About “Learning”

- Learning general models from data of particular examples
- Data is cheap and abundant (data warehouses, data marts); knowledge is expensive and scarce.
- Example in retail: Customer transactions to consumer behavior:

  People who bought “Da Vinci Code” also bought “The Five People You Meet in Heaven” (www.amazon.com)

- Build a model that is *a good and useful approximation* to the data.

Data Mining

Application of machine learning methods to large databases is called “Data mining”.

- Retail: Market basket analysis, Customer relationship management (CRM)
- Finance: Credit scoring, fraud detection
- Manufacturing: Optimization, troubleshooting
- Medicine: Medical diagnosis
- Telecommunications: Quality of service optimization
- Bioinformatics: Motifs, alignment
- Web mining: Search engines
- ...

What is Machine Learning?

- Optimize a performance criterion using example data or past experience.
- Role of Statistics: Building mathematical models, core task is inference from a sample
- Role of Computer science: Efficient algorithms to
  - Solve the optimization problem
  - Representing and evaluating the model for inference

Sample of ML Applications

- Learning Associations
- Supervised Learning
  - Classification
  - Regression
- Unsupervised Learning
- Reinforcement Learning

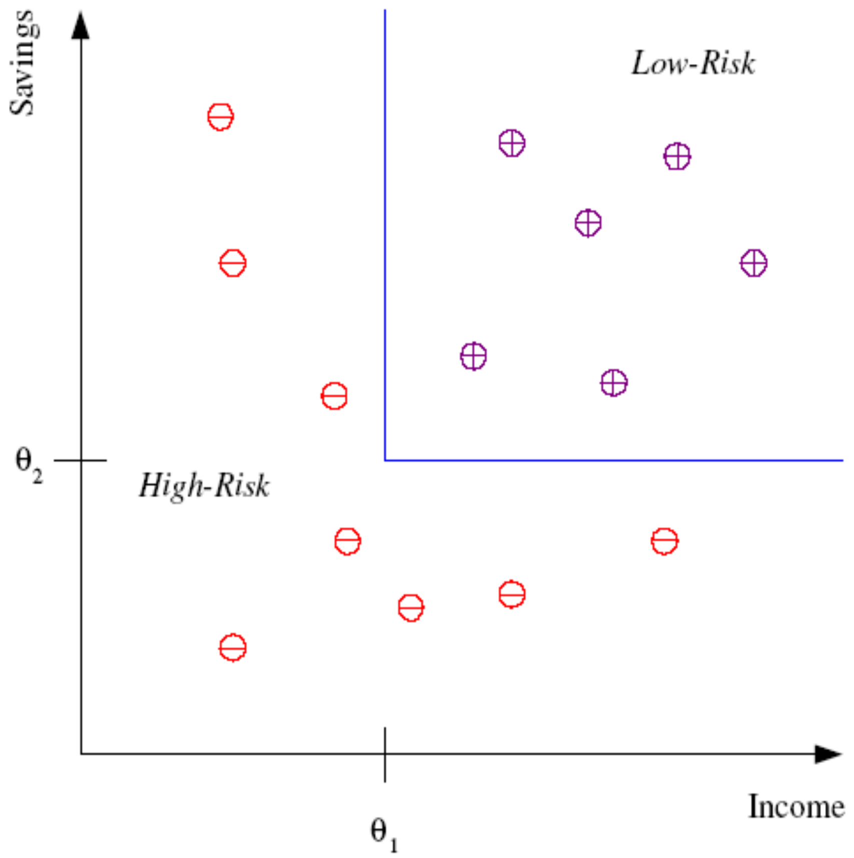
Learning Associations

- Basket analysis:
  $P(Y \mid X)$ probability that somebody who buys X also buys Y where X and Y are products/services.

  Example: $P(\text{chips} \mid \text{beer}) = 0.7$
- If we know more about customers or make a distinction among them:
  - $P(Y \mid X, D)$
    where D is the customer profile (age, gender, martial status, ...)
  - In case of a Web portal, items correspond to links to be shown/prepared/downloaded in advance

Classification

- Example: Credit scoring
- Differentiating between low-risk and high-risk customers from their *income* and *savings*



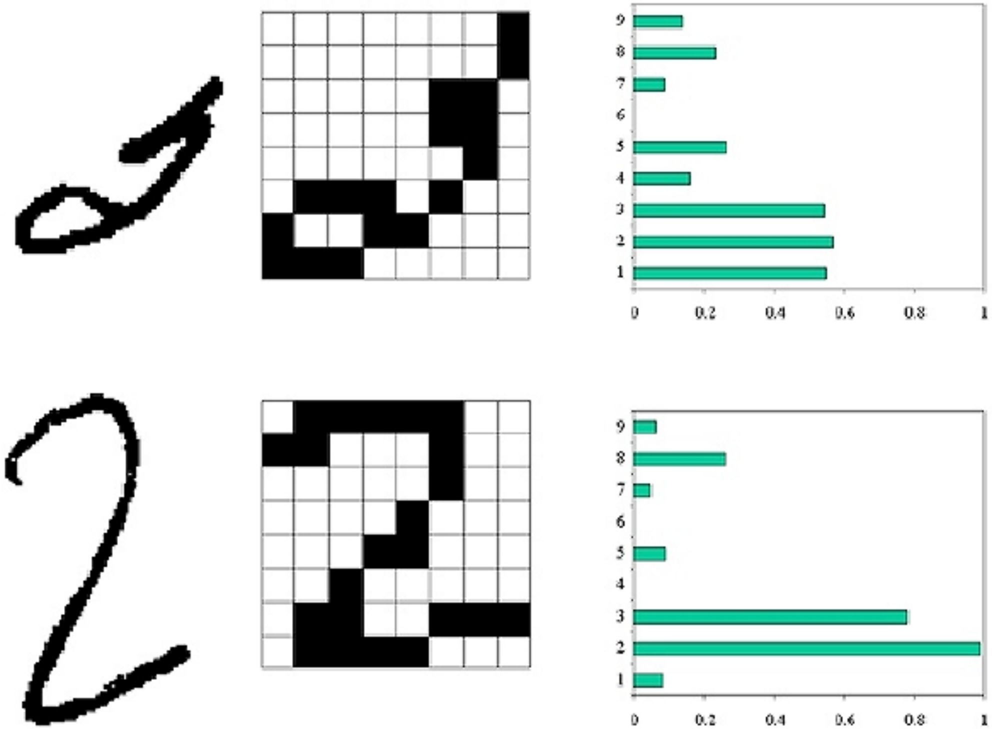
Discriminant: IF *income* > θ_1 AND *savings* > θ_2
THEN low-risk ELSE high-risk

Classification: Applications

- Aka Pattern recognition
- Character recognition: Different handwriting styles.
- Face recognition: Pose, lighting, occlusion (glasses, beard), make-up, hair style
- Speech recognition: Temporal dependency.
  - Use of a dictionary for the syntax of the language.
  - Sensor fusion: Combine multiple modalities; eg, visual (lip image) and acoustic for speech
- Medical diagnosis: From symptoms to illnesses
- Brainwave understanding: From signals to “states” of thought
- Reading text:
- ...

Character Recognition

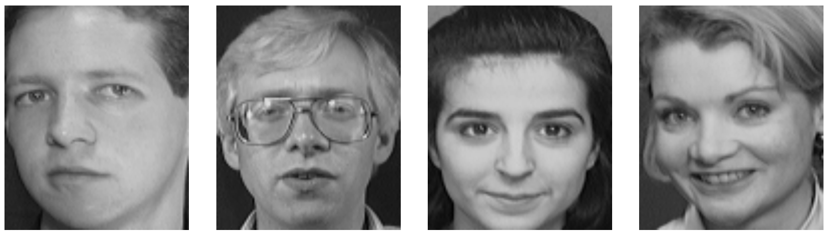
Want to learn how to recognize characters, even if written in different ways by different people



Face Recognition

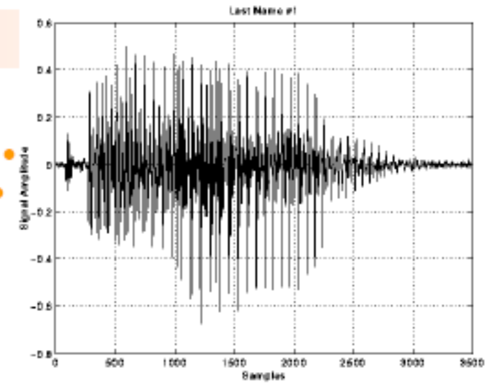
Training examples of a person

Test images



AT&T Laboratories, Cambridge UK

Example Pattern Recognition: Speech Recognition



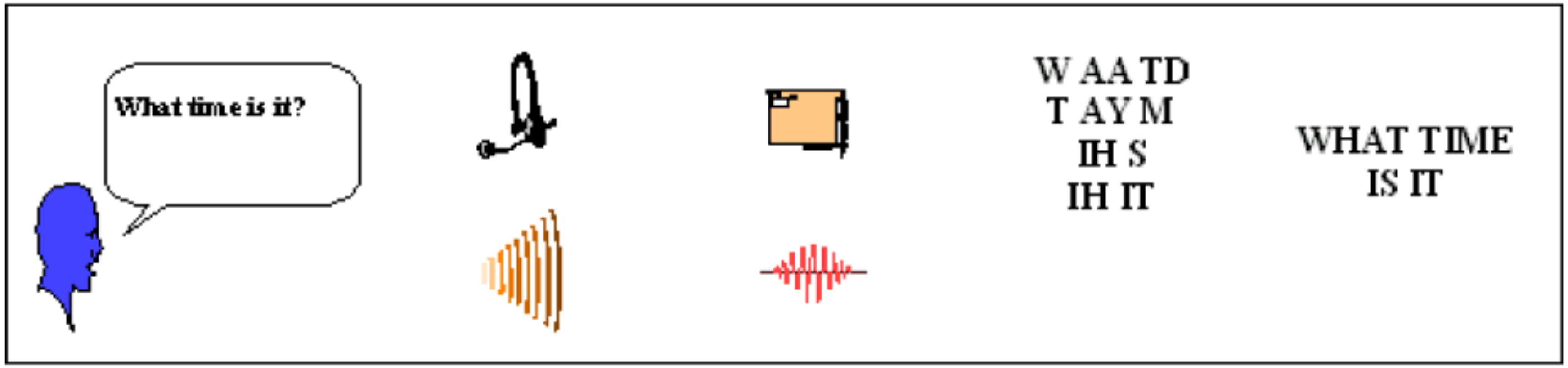
USER

MICROPHONE

SOUND CARD

SPEECH RECOGNITION ENGINE

SPEECH-AWARE APPLICATION



User speaks into the microphone.

Microphone captures sound waves and generates electrical impulses.

Sound card converts acoustical signal to digital signal.

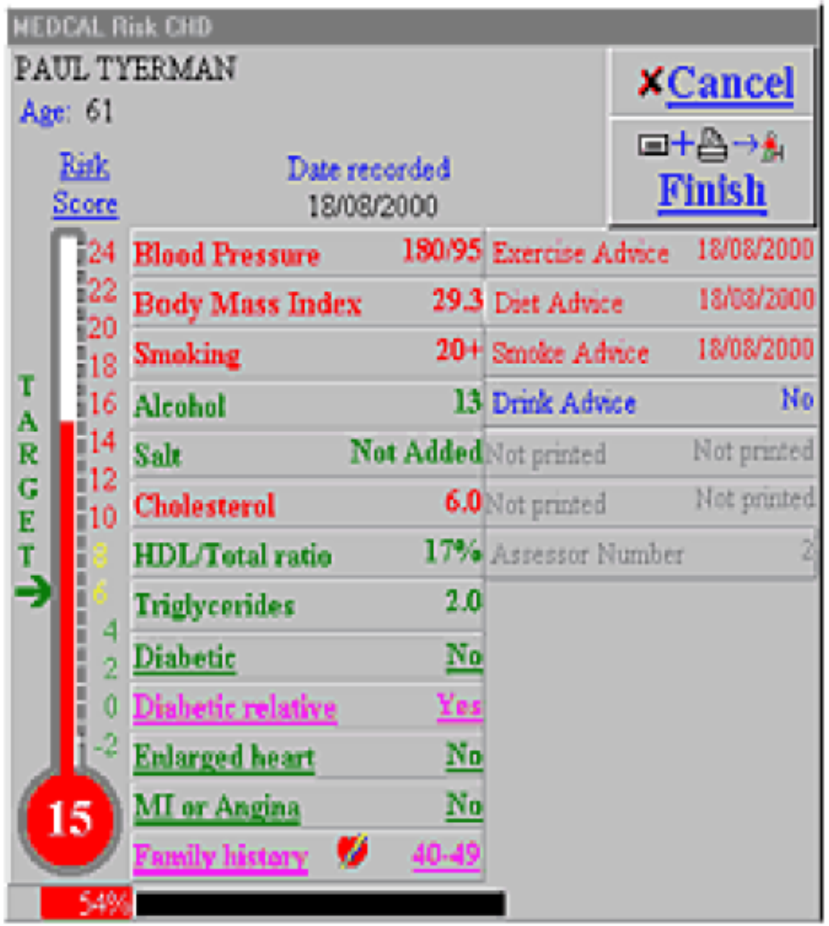
Speech recognition engine coverts digital signal to phonemes, then words.

Application processes words as text input.

Medical diagnosis

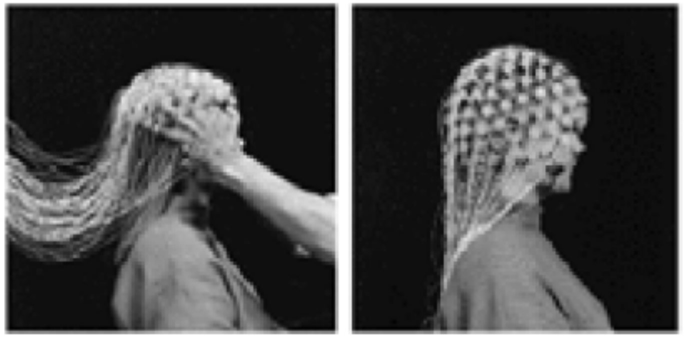
Inputs: relevant info about patient, symptoms, test results, etc.

Output: Expected illness or risk factors

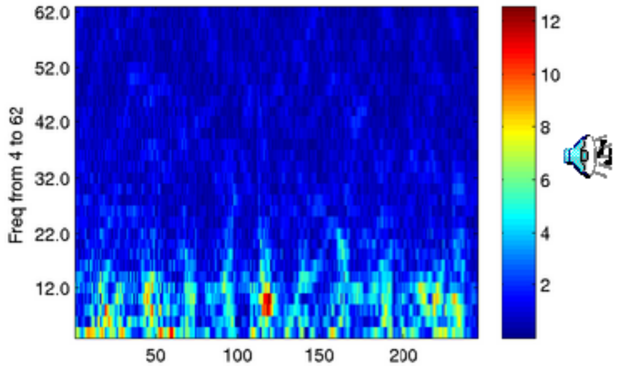


Example Pattern Recognition: Interpreting Brainwaves

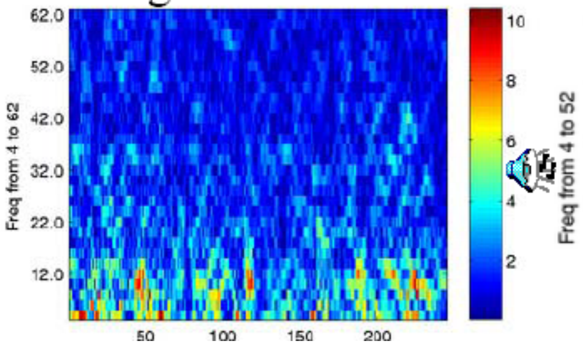
EEG electrodes reading brain waves:



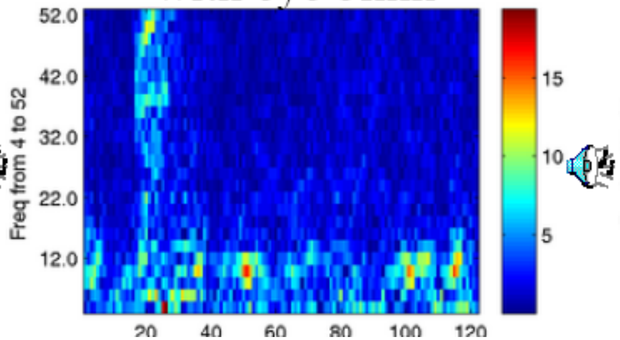
- Rotation task, left brain



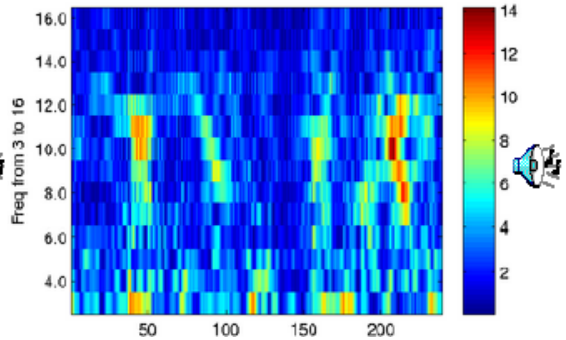
- Rotation task, right brain



- Resting task, with eye blink



- Counting task



Example Pattern Recognition: Reading text

- Can you read this?
  - Aircndcog to a rseerhcaer at Cbiardmge Urensvitiy, it dsoen't mtetar in waht oderr the letrtes in a wrod are, the olny ipnaotmrt tihng is taht the fsrit and lsat lteter be at the rgiht plcae. The rset can be a toatl mses and you can slitl raed it wutohit porlebm. Tehy spectluae taht tihs is bseuace the hmaun mnid deos not raed erevy leettr by iesltf but the wrod as a whloe. Wtehehr tihs is ture or not is a ponit of deabte.

- Clearly, the brain has learned syntax and semantics of language, including contextual dependencies, to make sense of of this ☺

- For fun: Here's a web page where you can create your own jumbled text: http://www.stevesachs.com/jumbler.cgi

Regression

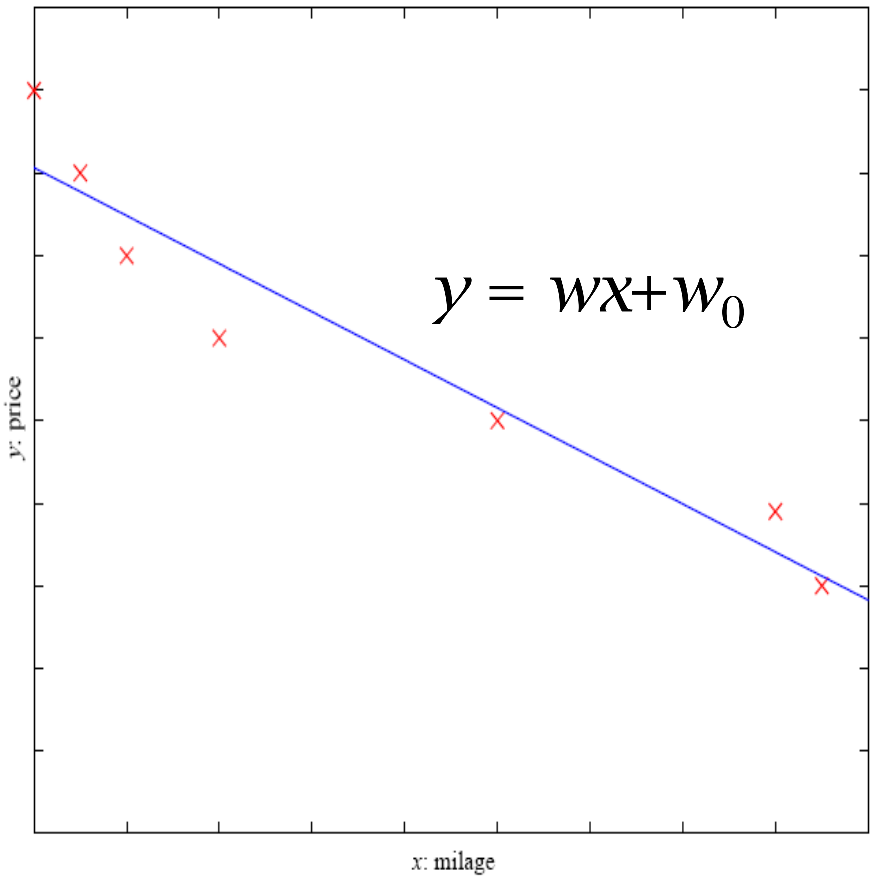
- Example: Price of a used plane
- x : plane attribute

  y : price

  $$y = g(x \mid \theta)$$

  $g(\,)$ model,

  θ parameters



Supervised Learning: Uses

- Prediction of future cases: Use the rule to predict the output for future inputs
- Knowledge extraction: The rule is easy to understand
- Compression: The rule is simpler than the data it explains
- Outlier detection: Exceptions that are not covered by the rule, e.g., fraud

Unsupervised Learning

- Learning "what normally happens"
- No output (we do not know the right answer)
- Clustering: Grouping similar instances
- Example applications
  - Customer segmentation in CRM
    - Company may have different marketing approaches for different groupings of customers
  - Image compression: Color quantization
    - Instead of using 24 bits to represent 16 million colors, reduce to 6 bits and 64 colors, if the image only uses those 64 colors
  - Bioinformatics: Learning motifs (sequences of amino acids in proteins)
  - Document Classification in unknown Domains.

Reinforcement Learning

- Learning a policy: A sequence of actions/outputs
- No supervised output but delayed reward
- Credit assignment problem
- Game playing
- Robot in a maze
- Multiple agents, partial observability, ...

An Extended Example

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

Species

- Sea bass (cheap)
- Salmon (expensive)

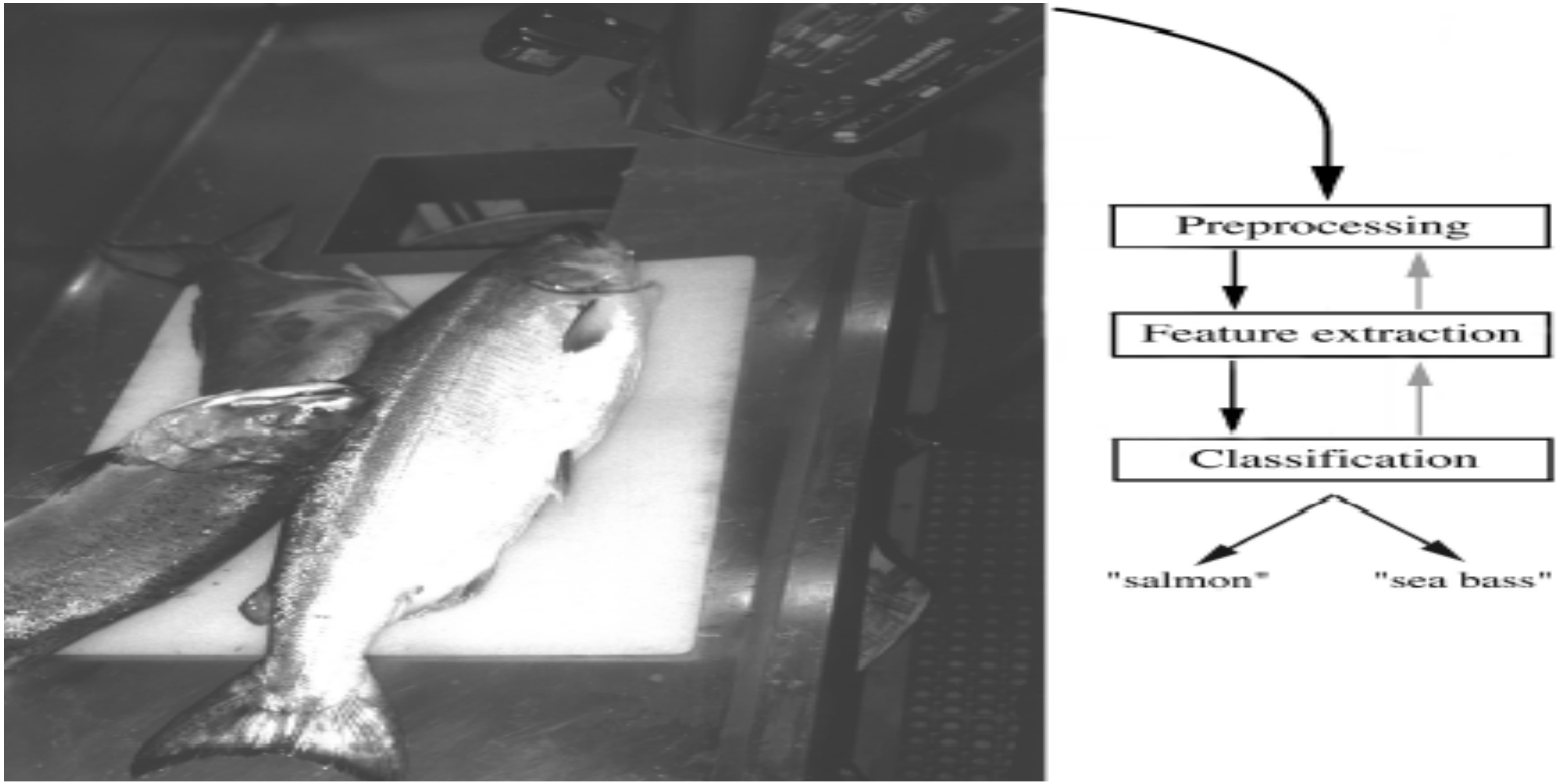
- Problem Analysis

  - Set up a camera and take some sample images to extract features

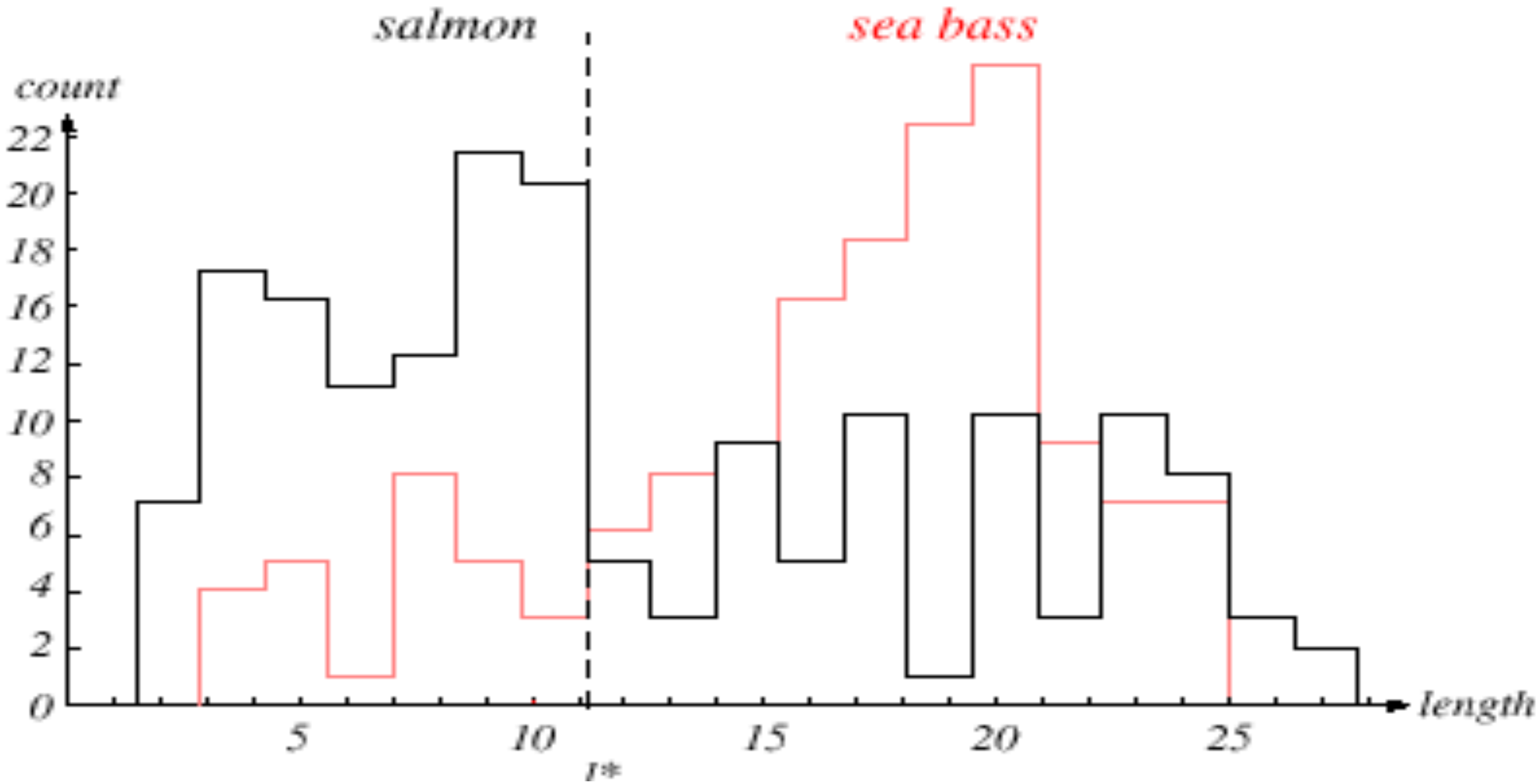
    - Length
    - Lightness
    - Width
    - Number and shape of fins
    - Position of the mouth, etc…

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

- Preprocessing
  - 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"

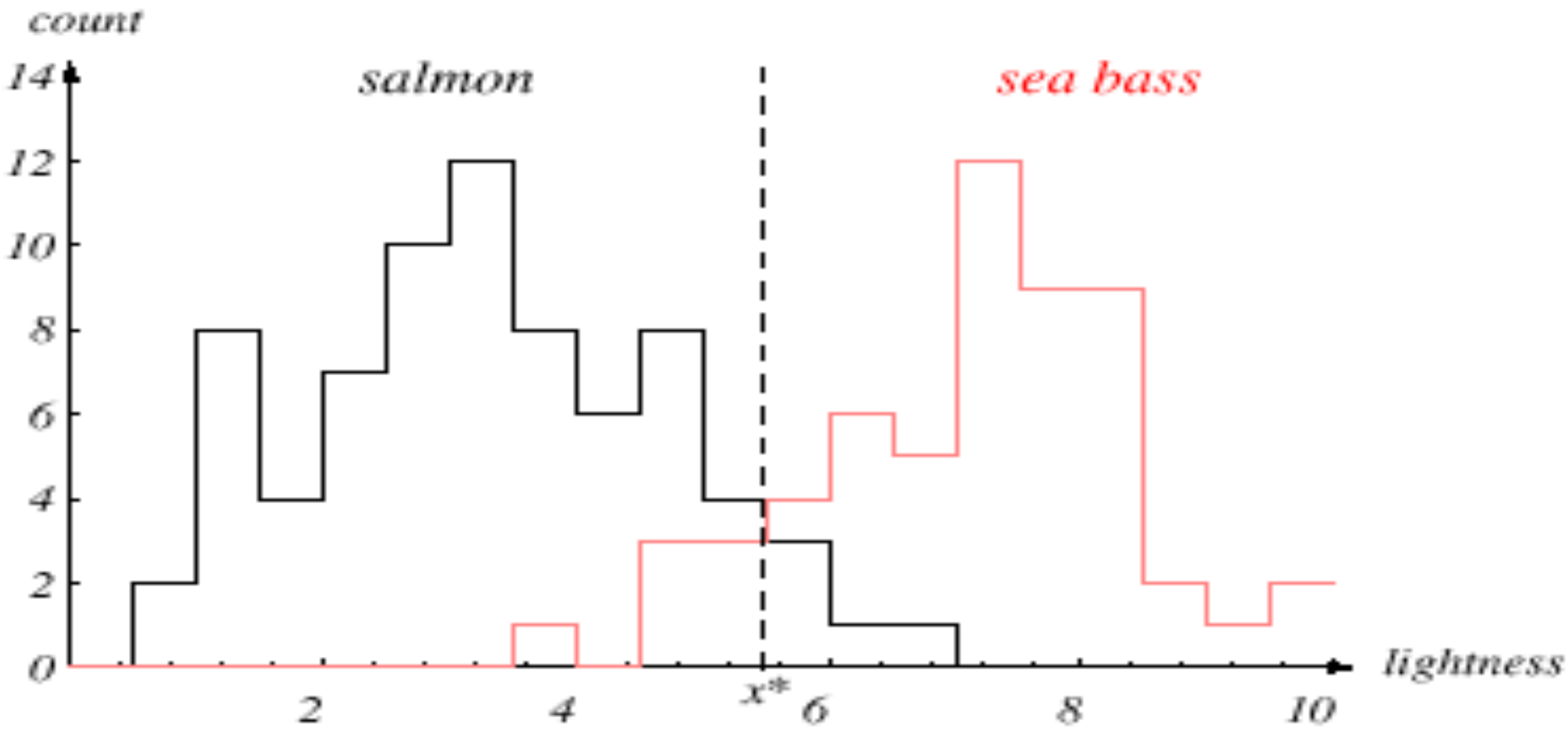
- Classification
  - Now we need (expert) information to find features that enables us to distinguish the species.
  - “Select the length of the fish as a possible feature for discrimination”



salmon
sea bass
count
22
20
18
16
12
10
8
6
4
2
0
5
10
15
20
25
length
l*

The length is a poor feature alone!
→ Cost of decision

Select the lightness as a possible feature.


count
14
12
10
8
6
4
2
0
salmon
sea bass
lightness
2
4
x*
6
8
10

- 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!)

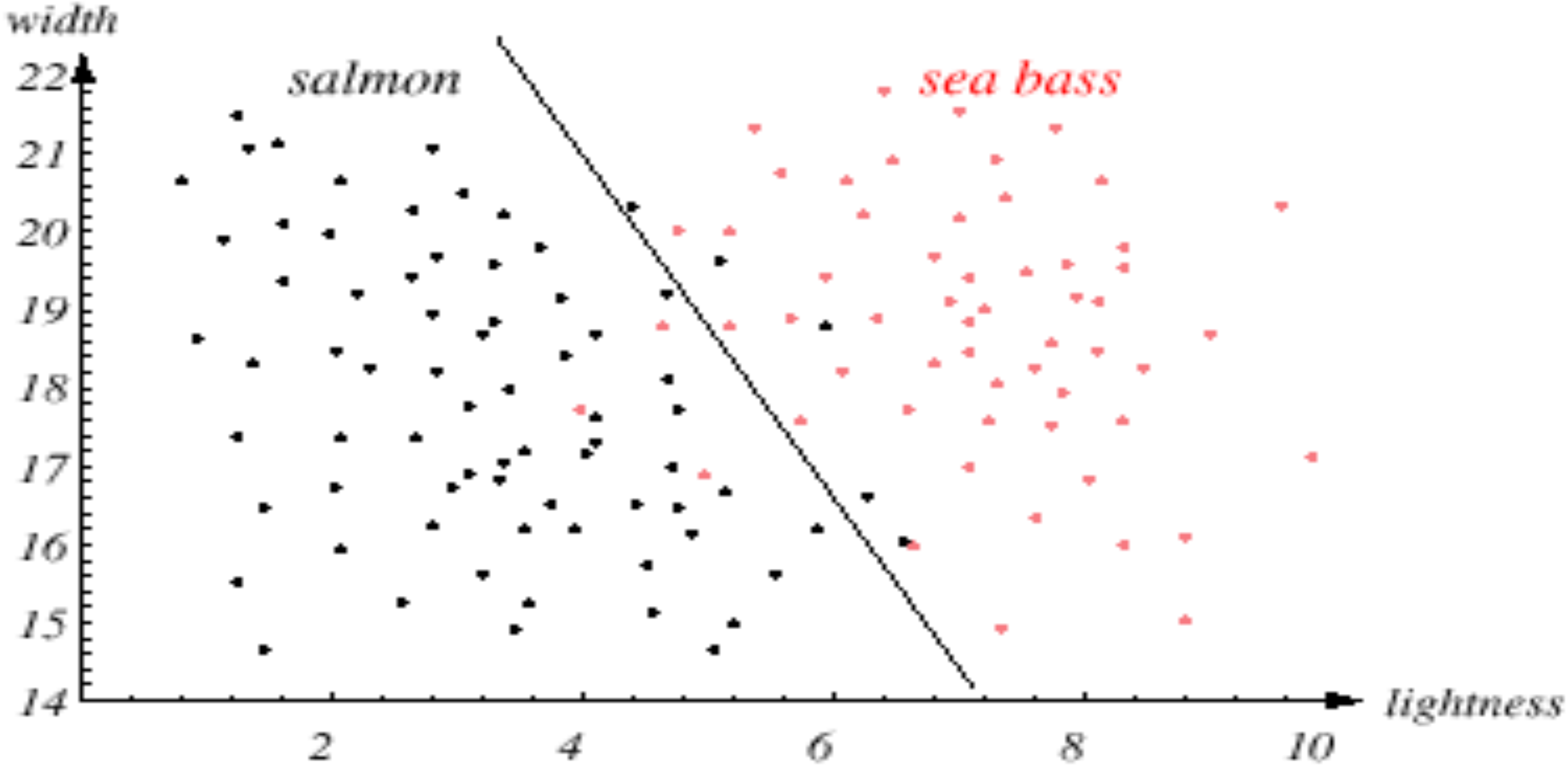
Task of decision theory

- Adopt the lightness and add the width of the fish

Fish $\Rightarrow$ $x^T = [x_1, x_2]$

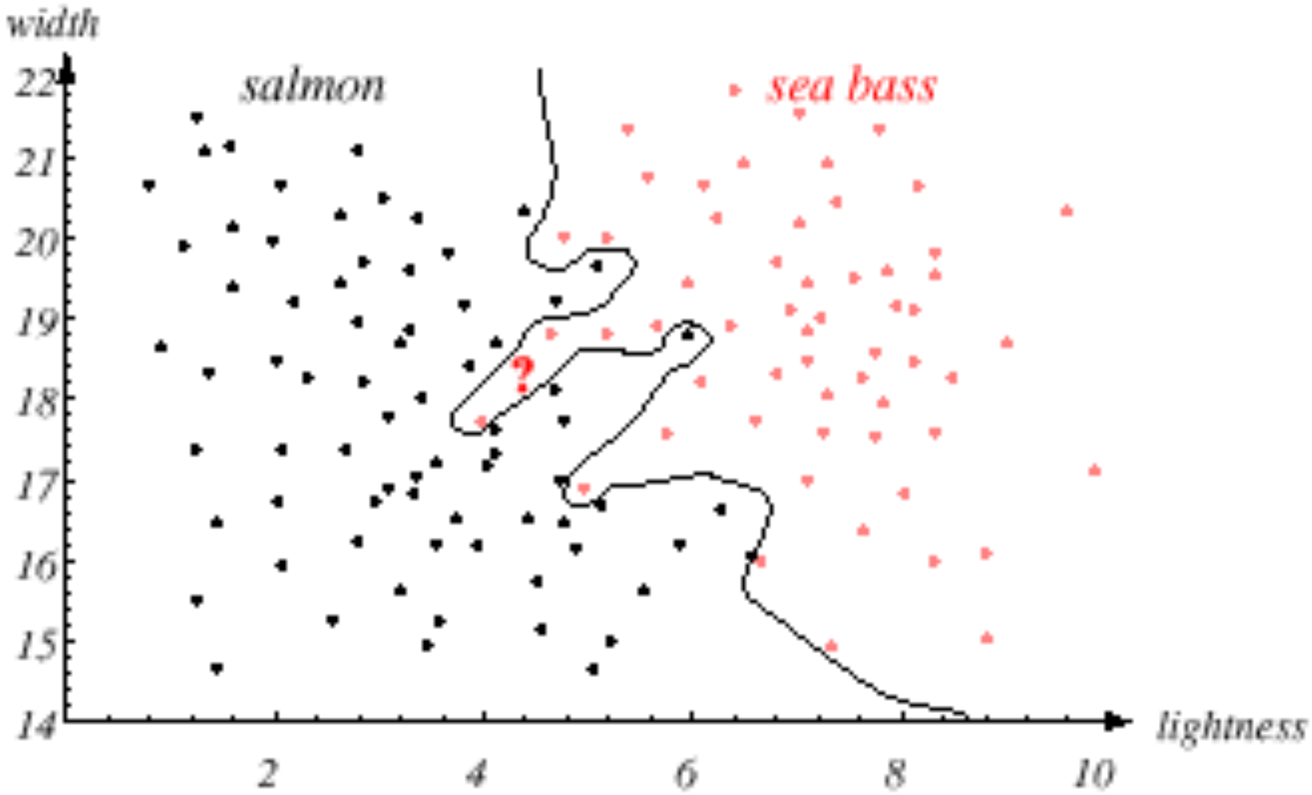
Lightness

Width


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

- We might add other features that are not correlated with the ones we already have. Precaution should be taken not to reduce the performance by adding such “noisy features”

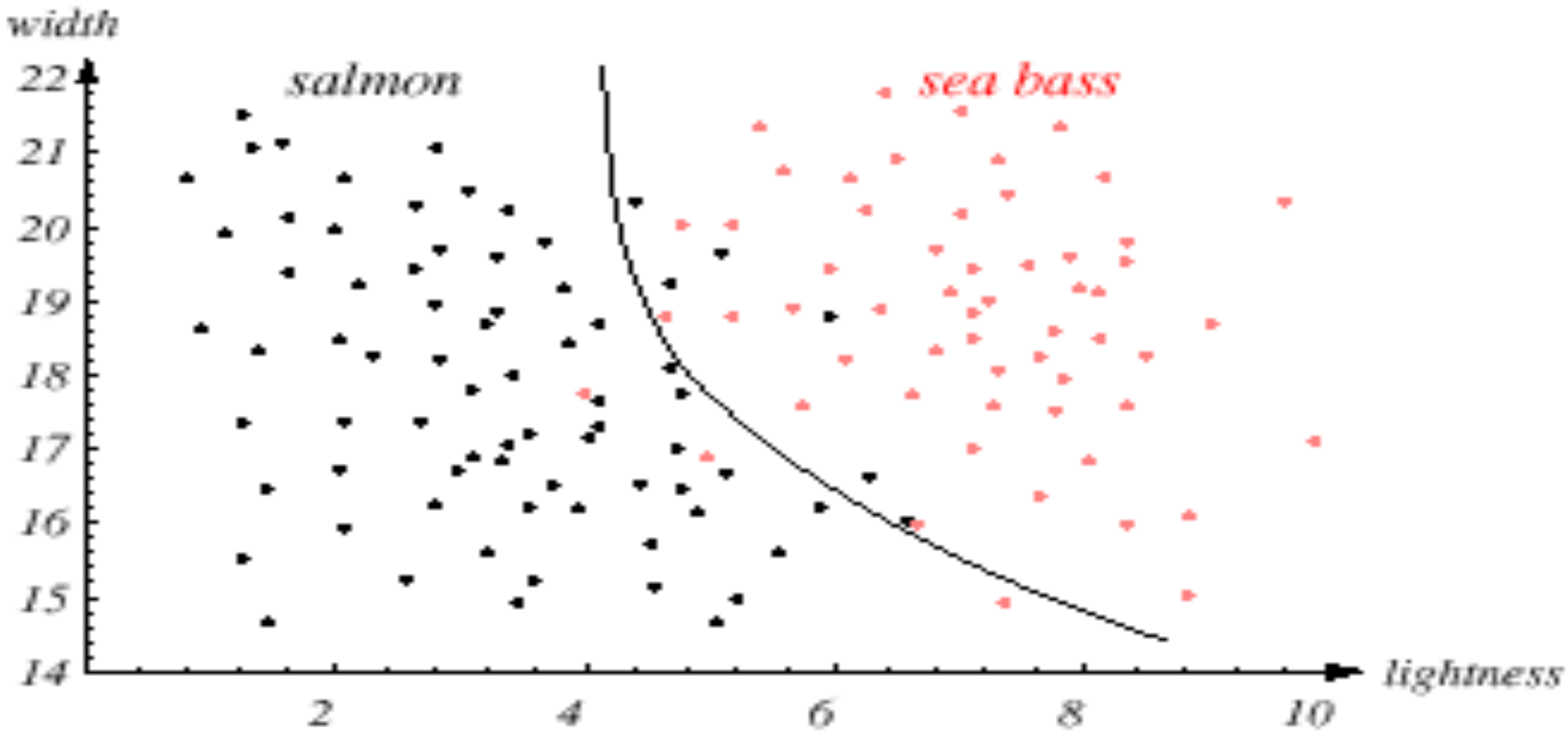
- Ideally, the best decision boundary should be the one which provides an optimal performance such as in the following figure:



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

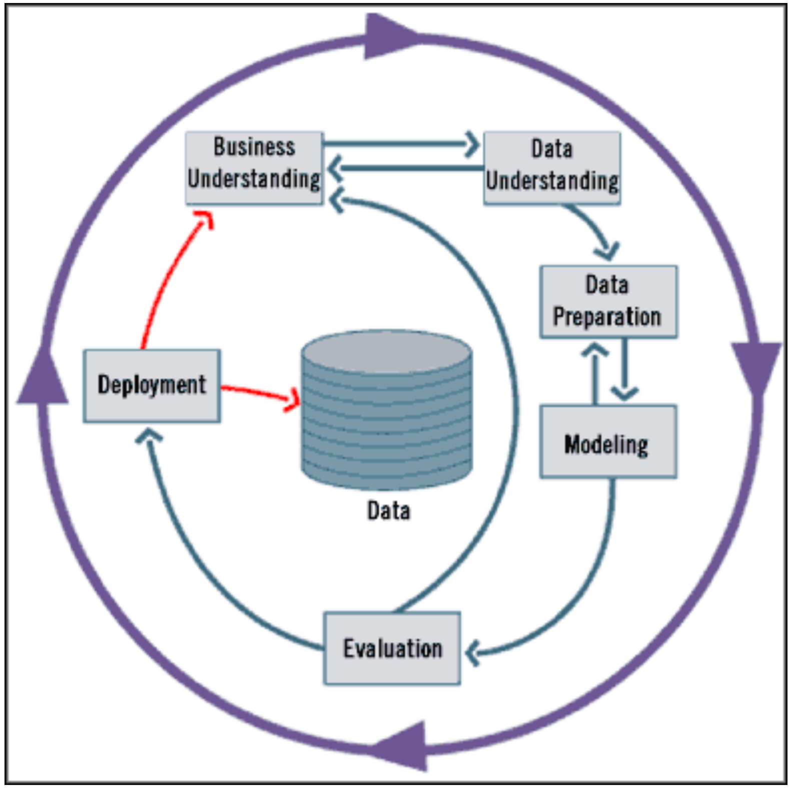
- However, our satisfaction is premature because the central aim of designing a classifier is to correctly classify novel input

Issue of generalization!


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

Standard data mining life cycle

- It is an iterative process with phase dependencies
- Consists of six (6) phases:



Phases (1)

- Business Understanding
  - Understand project objectives and requirements
  - Formulation of a data mining problem definition
- Data Understanding
  - Data collection
  - Evaluate the quality of the data
  - Perform exploratory data analysis
- Data Preparation
  - Clean, prepare, integrate, and transform the data
  - **Select** appropriate attributes and variables

Phases (2)

- Modeling
  - Select and apply appropriate modeling techniques
  - Calibrate/learn model parameters to optimize results
  - If necessary, return to data preparation phase to satisfy model's data format
- Evaluation
  - Determine if model satisfies objectives set in phase 1
  - Identify business issues that have not been addressed
- Deployment
  - Organize and present the model to the “user”
  - Put model into practice
  - Set up for continuous mining of the data

Fallacies of Data Mining (1)

- Fallacy 1: There are data mining tools that automatically find the answers to our problem
  - Reality: There are no automatic tools that will solve your problems "while you wait"
- Fallacy 2: The DM process require little human intervention
  - Reality: The DM process require human intervention in all its phases, including updating and evaluating the model by human experts
- Fallacy 3: Data mining have a quick ROI
  - Reality: It depends on the startup costs, personnel costs, data source costs, and so on

Fallacies of Data Mining (2)

- Fallacy 4: DM tools are easy to use
  - Reality: Analysts must be familiar with the model
- Fallacy 5: DM will identify the causes to the business problem
  - Reality: DM tools only identify patterns in your data, analysts must identify the cause
- Fallacy 6: Data mining will clean up a data repository automatically
  - Reality: Sequence of transformation tasks must be defined by an analysts during early DM phases

* Fallacies described by Jen Que Louie, President of Nautilus Systems, Inc.

Remember

- Problems suitable for Data Mining:
  - Require to discover knowledge to make right decisions
  - Current solutions are not adequate
  - Expected high-payoff for the right decisions
  - Have accessible, sufficient, and relevant data
  - Have a changing environment

- IMPORTANT:
  - **ENSURE privacy if personal data is used!**
  - **Not every data mining application is successful!**

Overview

Supervised Learning

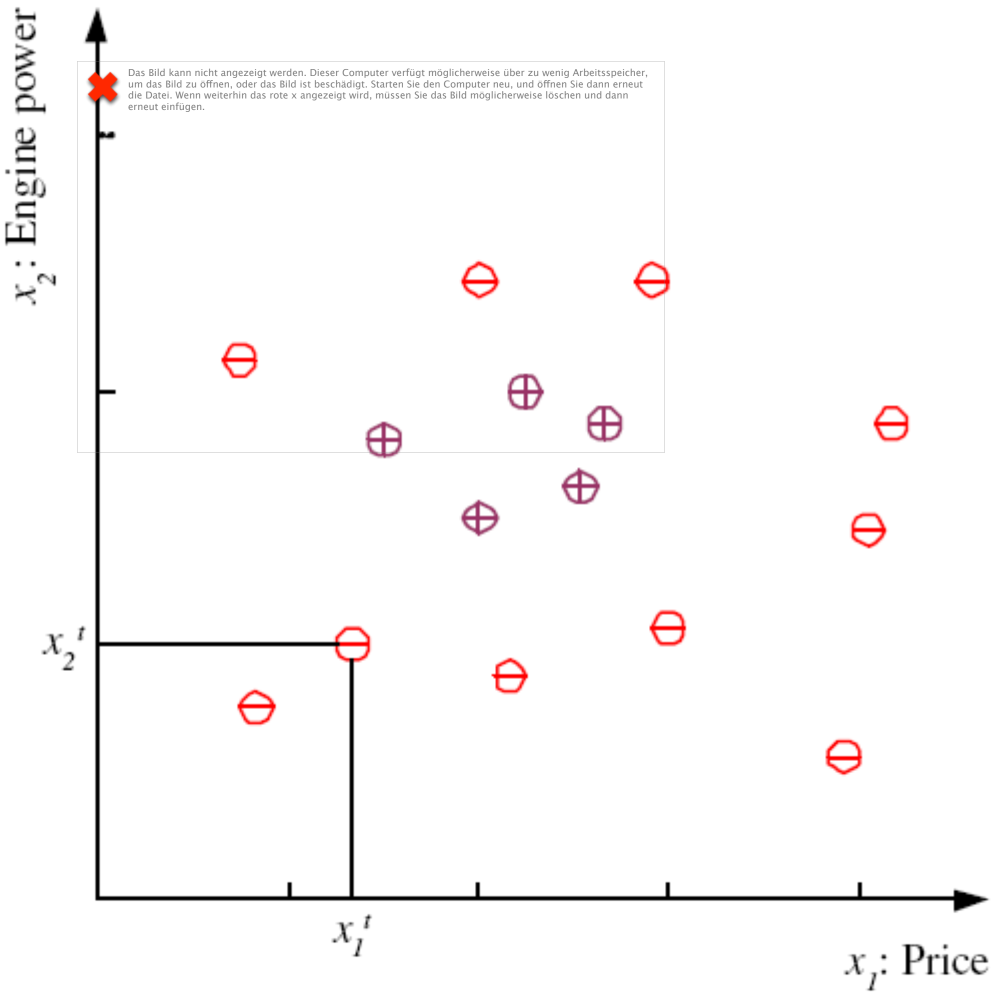
Learning a Class from Examples

- Class C of a "family car"
  - Prediction: Is car x a family car?
  - Knowledge extraction: What do people expect from a family car?
- Output:

  Positive (+) and negative (–) examples
- Input representation:

  x_1: price, x_2 : engine power

Training set $\mathcal{X}$



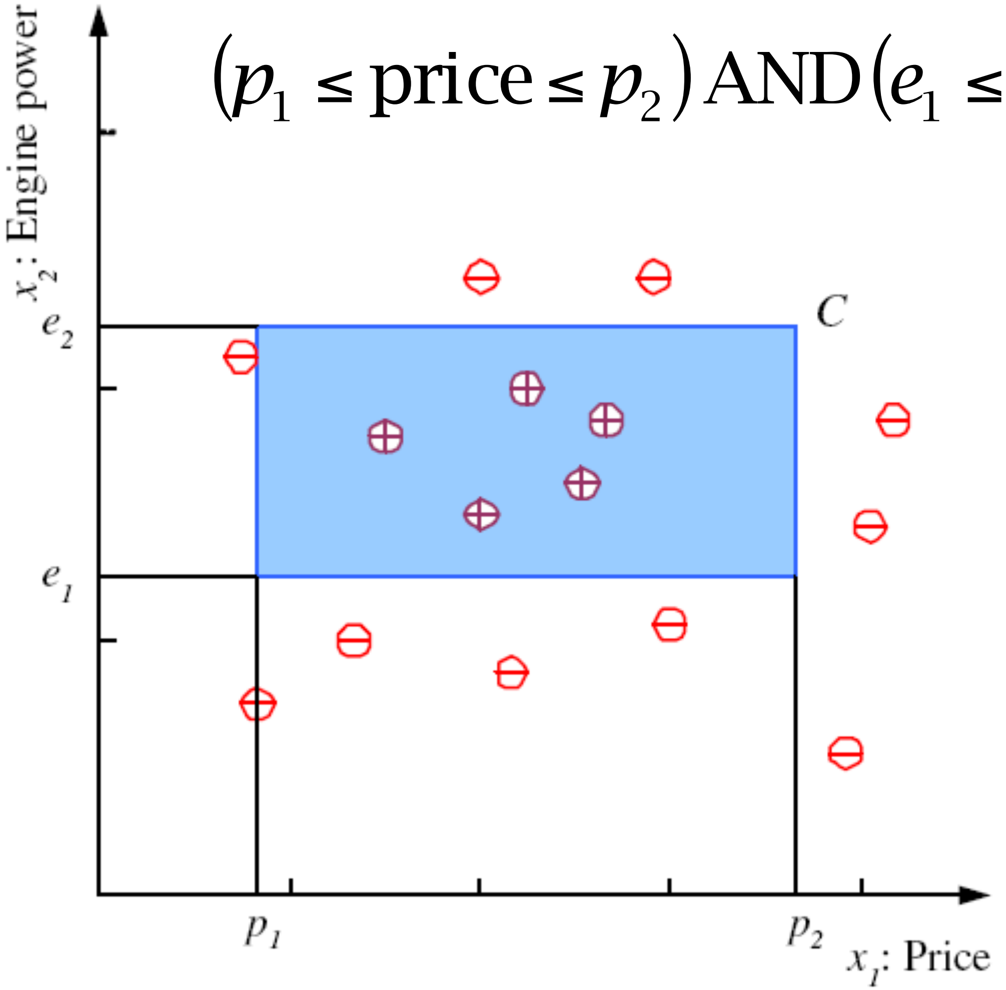
$$\mathcal{X} = \{\boldsymbol{x}^t, r^t\}_{t=1}^N$$

$$r = \begin{cases} 1 \text{ if } \boldsymbol{x} \text{ is positive} \\ 0 \text{ if } \boldsymbol{x} \text{ is negative} \end{cases}$$

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

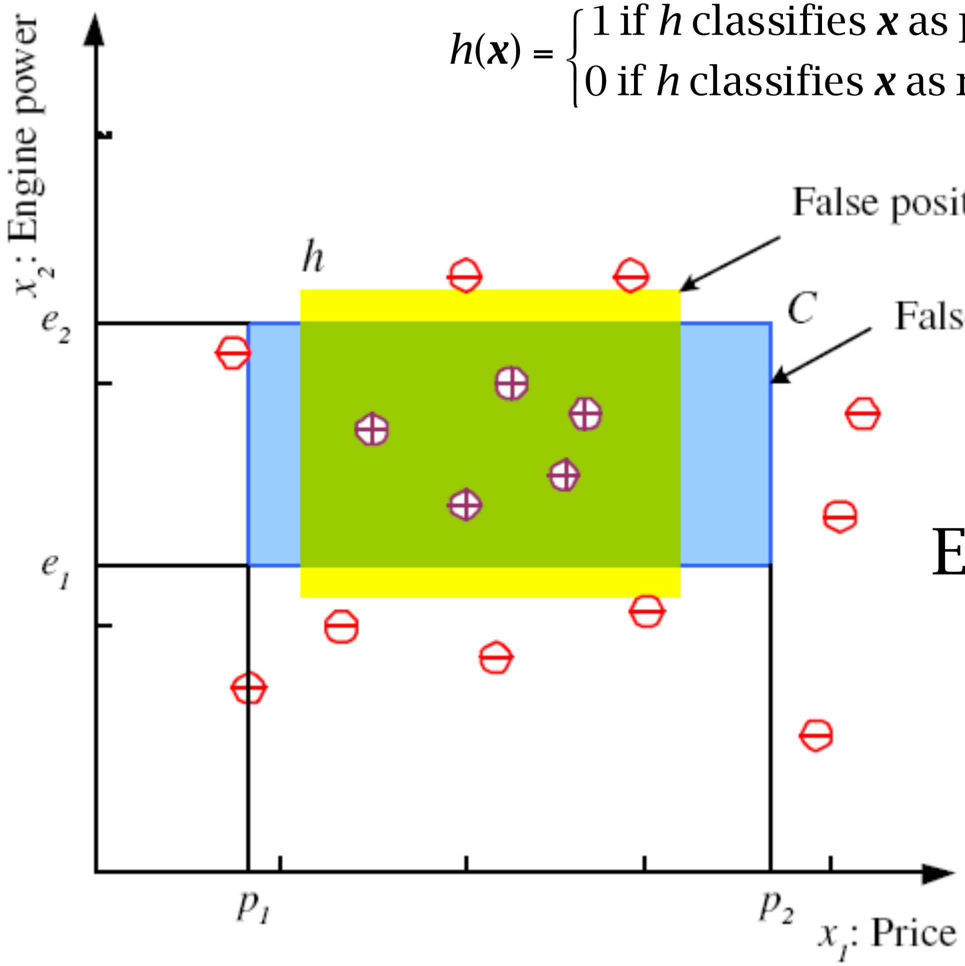
Class C

$$(p_1 \le \text{price} \le p_2) \text{ AND } (e_1 \le \text{engine power} \le e_2)$$



Hypothesis class $\mathcal{H}$

$$h(\boldsymbol{x}) = \begin{cases} 1 \text{ if } h \text{ classifies } \boldsymbol{x} \text{ as positive} \\ 0 \text{ if } h \text{ classifies } \boldsymbol{x} \text{ as negative} \end{cases}$$

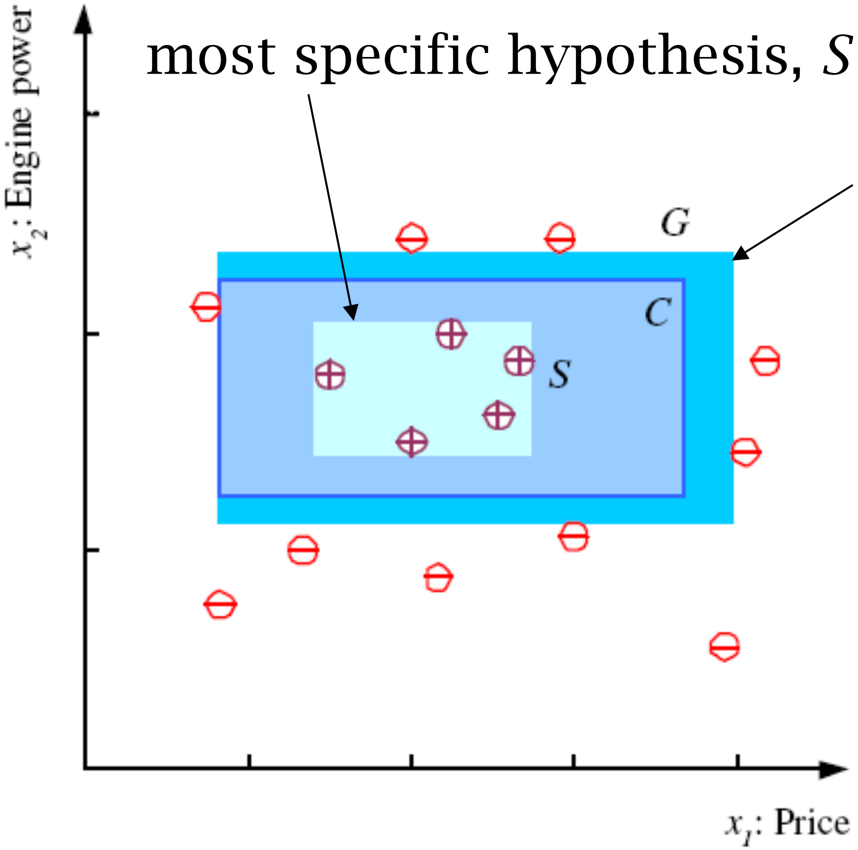


Error of h on $\mathcal{H}$

$$E(h|\mathcal{X}) = (1/N)\sum_{t=1}^{N}\left(h\left(\mathbf{x}^t\right) \neq r^t\right)$$

$(a \neq b) = 1$ if $\neq$, 0 otherwise

S, G, and the Version Space



most specific hypothesis, S

most general hypothesis, G

$h \in \mathcal{H}$, between S and G is consistent

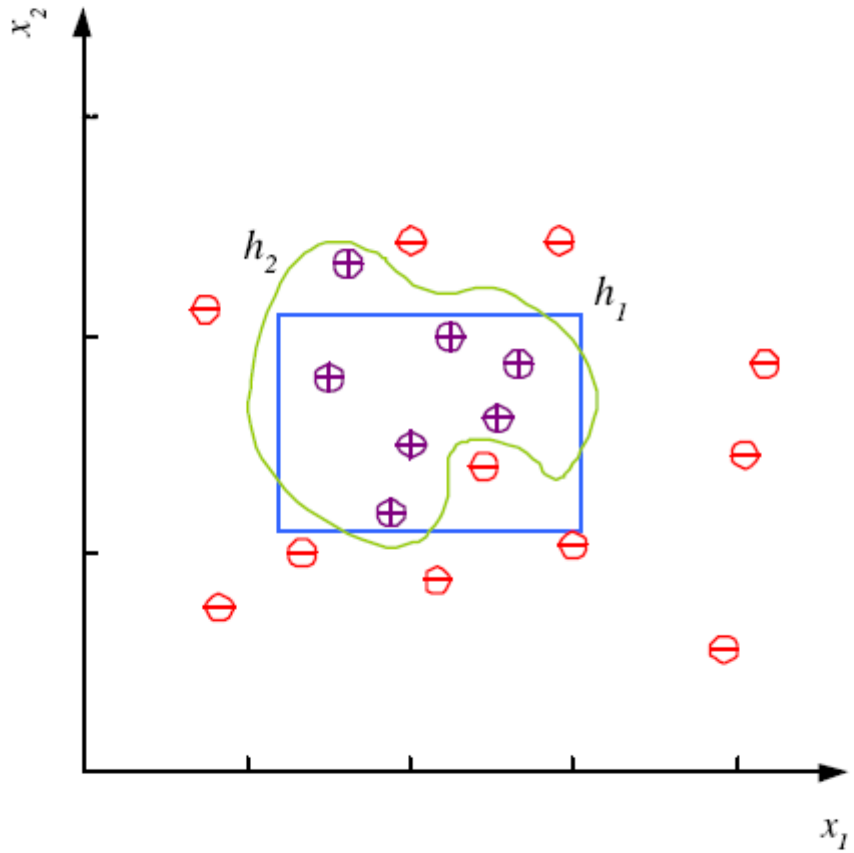
and make up the version space

(Mitchell, 1997)

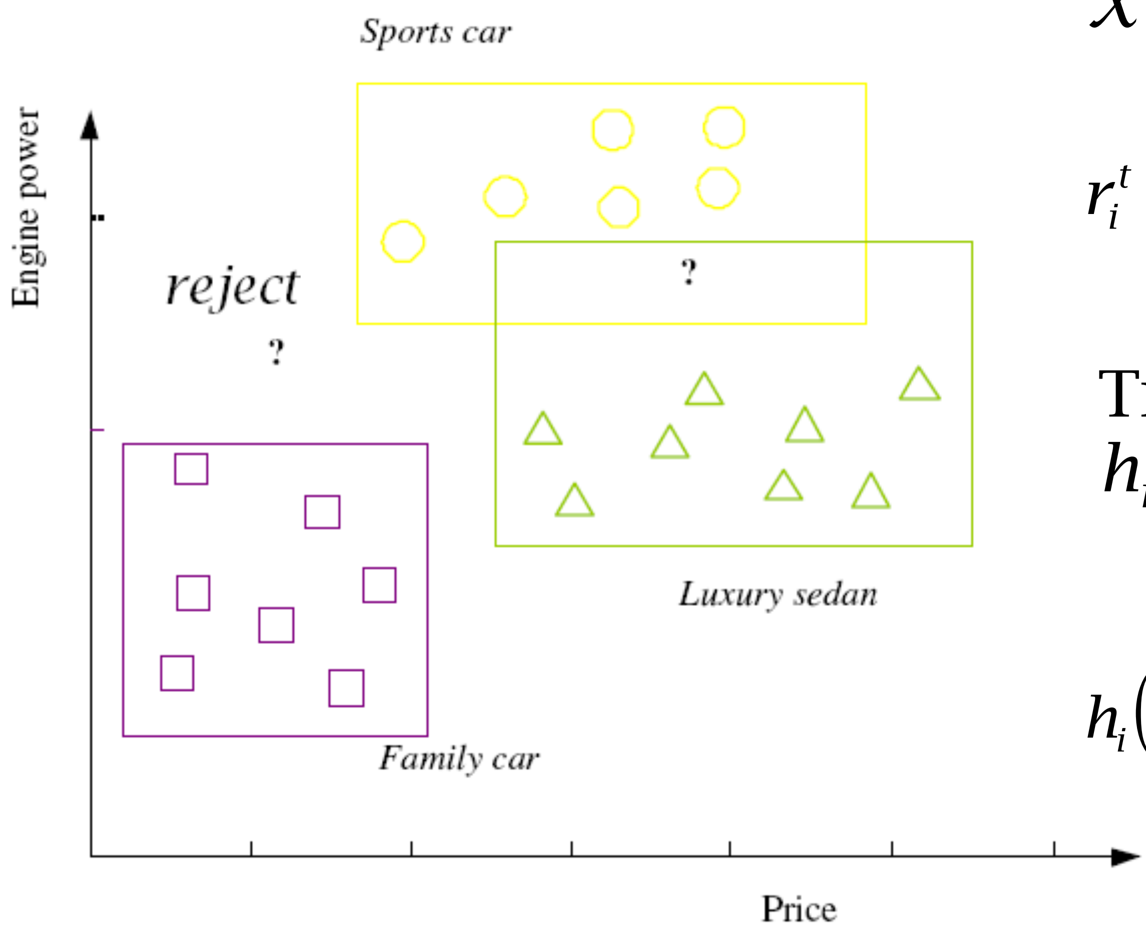
Noise and Model Complexity

Use the simpler one because

- Simpler to use (lower computational complexity)
- Easier to train (lower space complexity)
- Easier to explain (more interpretable)
- Generalizes better (lower variance - Occam's razor)



Multiple Classes, C_i i=1,...,K



$$\mathcal{X} = \{\mathbf{x}^t, r^t\}_{t=1}^{N}$$

$$r_i^t = \begin{cases} 1 \text{ if } \mathbf{x}^t \in C_i \\ 0 \text{ if } \mathbf{x}^t \in C_j, j \neq i \end{cases}$$

Train hypotheses
$h_i(\mathbf{x})$, i =1,...,K:

$$h_i\left(\mathbf{x}^t\right) = \begin{cases} 1 \text{ if } \mathbf{x}^t \in C_i \\ 0 \text{ if } \mathbf{x}^t \in C_j, j \neq i \end{cases}$$

Regression

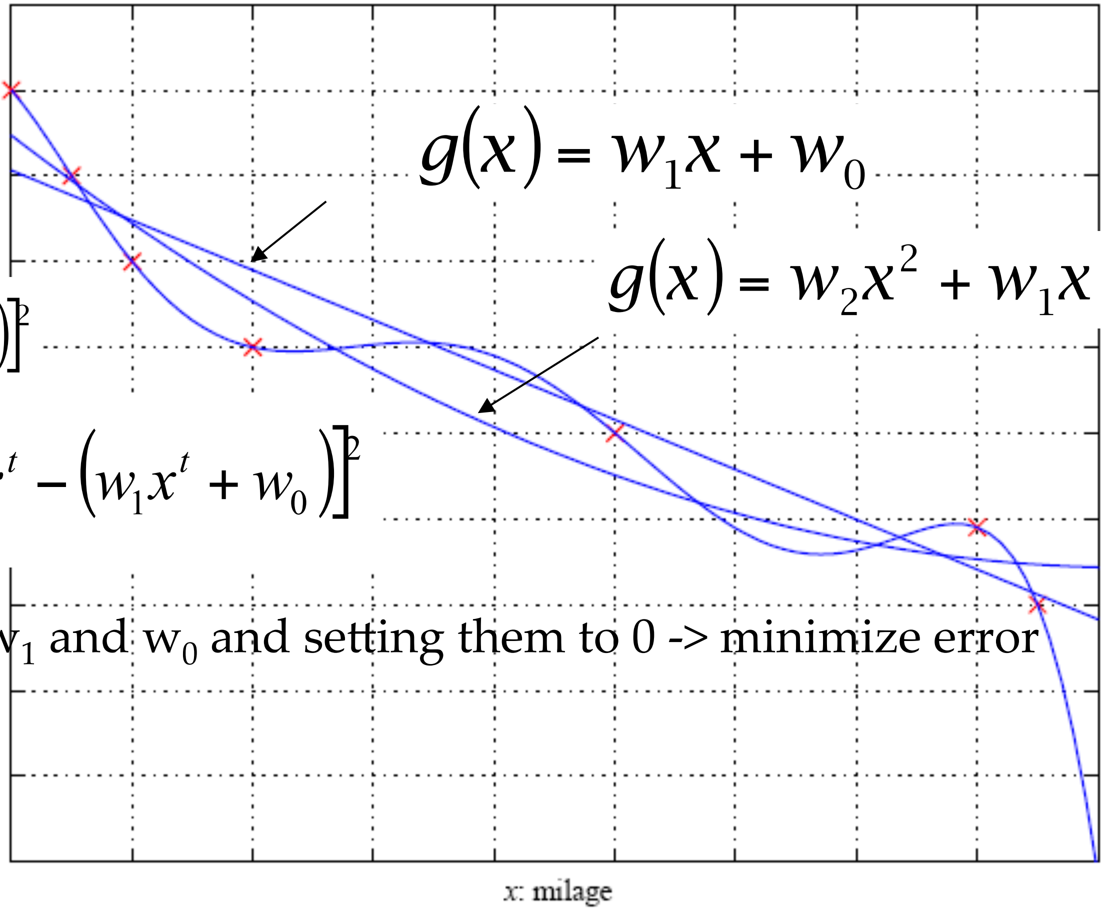
$$\mathcal{X} = \{x^t, r^t\}_{t=1}^{N}$$

$$r^t \in \Re$$

$$r^t = f(x^t)$$

$$E(g \mid \mathcal{X}) = \frac{1}{N}\sum_{t=1}^{N}\left[r^t - g(x^t)\right]^2$$

$$E(w_1, w_0 \mid \mathcal{X}) = \frac{1}{N}\sum_{t=1}^{N}\left[r^t - (w_1 x^t + w_0)\right]^2$$



$$g(x) = w_1 x + w_0$$

$$g(x) = w_2 x^2 + w_1 x + w_0$$

Partial derivatives of E w.r.t w_1 and w_0 and setting them to 0 -> minimize error

$$w_1 = \frac{\sum_t x^t r^t - \overline{x}\overline{r}N}{\sum_t (x^t)^2 - N\overline{x}^2}$$

$$w_0 = \overline{r} - w_1 \overline{x}$$

Model Selection & Generalization

- Learning is an ill-posed problem; data is not sufficient to find a unique solution
- The need for inductive bias, assumptions about $\mathcal{H}$
- Generalization: How well a model performs on new data
- Overfitting: $\mathcal{H}$ more complex than C or f
- Underfitting: $\mathcal{H}$ less complex than C or f

Triple Trade-Off

- There is a trade-off between three factors (Dietterich, 2003):
  1. Complexity of $\mathcal{H}$, $c(\mathcal{H})$,
  2. Training set size, N,
  3. Generalization error, E, on new data
- As $N\uparrow$, $E\downarrow$
- As $c(\mathcal{H})\uparrow$, first $E\downarrow$ and then $E\uparrow$

Cross-Validation

- To estimate generalization error, we need data unseen during training. We split the data as
  - Training set (50%)
  - Validation set (25%)
  - Test (publication) set (25%)
- Resampling when there is few data

Dimensions of a Supervised Learner

1. Model : $g(\boldsymbol{x} \mid \theta)$
2. Loss function: $E(\theta \mid \mathcal{X}) = \sum_t L\left(r^t, g\left(\boldsymbol{x}^t \mid \theta\right)\right)$
3. Optimization procedure:

$$\theta^* = \arg\min_\theta E(\theta \mid \mathcal{X})$$