

Introduction to Machine Learning

CS195-5-2003
Thomas Hofmann

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-1

What is Machine Learning ?

- Machine learning deals with the **design of computer programs** and systems that are able to take advantage of **data, examples or experiences** to **improve** their accuracy or performance on a specific task or set of tasks.
- Tasks: decision problems, control problems or prediction problems

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-2

Abstraction Levels

Class of problems a learning machine can handle.

Training data
(indirect specification)

Specific problem from this class of problems.

Test data

Specific instance for which a solution needs to be computed

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-3

Automatic Document Categorization & Annotation

The image shows a screenshot of an XML document editor displaying an XML document with several annotations. The XML document is a Reuters BIP Coding Group document. The annotations are as follows:

- `<code code="UK">` is annotated with **M13 = MONEY MARKETS**.
- `<code code="M13">` is annotated with **M132 = FOREX MARKETS**.
- `<code code="MCAT">` is annotated with **MCAT = MARKETS**.

The XML document structure is as follows:

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<newstext itemid="2227" doc="root" date="1996-08-20" url="http://www.reuters.com" ?>
  <metaData>
    <codes class="bip:countries:1.0">
      <code code="UK">
        <editdetail attribution="Reuters BIP Coding Group"
          action="confirmed" date="1996-08-20" />
      </code>
    </codes>
    <codes class="bip:topics:1.0">
      <code code="M13">
        <editdetail attribution="Reuters BIP Coding Group"
          action="confirmed" date="1996-08-20" />
      </code>
      <code code="M132">
        <editdetail attribution="Reuters BIP Coding Group"
          action="confirmed" date="1996-08-20" />
      </code>
      <code code="MCAT">
        <editdetail attribution="Reuters BIP Coding Group"
          action="confirmed" date="1996-08-20" />
      </code>
    </codes>
  </metaData>
  <cd:element name="country" value="UK" />
  <cd:element name="source" value="Reuters" />
</newstext>
```

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-4

Categorization With Expert Rules



Expert



```
if contains('yen') or  
contains('euro')  
then label=M132
```

M132 = FOREX MARKETS

Problem: Low coverage, moderate accuracy, difficulty of formalizing expert knowledge

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-5

Example-Based Categorization

Training Examples

M132 = FOREX MARKETS



Expert



Training



Learning Machine

Inductive Inference

/* some 'complicated' algorithm */

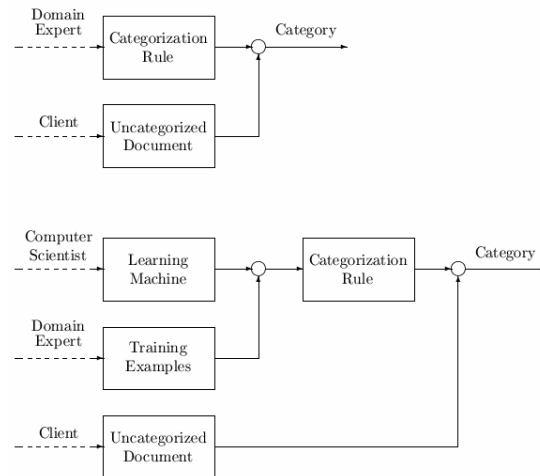
Recall

M132 = FOREX MARKETS

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-6

Direct & Indirect Approach to Categorization





© 2002,2003 Thomas Hofmann

CS195-5-2003-01-7



Living Longer With Machine Learning (1)

The Challenge

- Imagine that you visit an unknown country far away from here, where people use a large variety of mushrooms in their everyday diet.
-  Some of the mushrooms are poisonous while some are edible. There is obviously some incentive to knowing which one is which.
- Luckily, you have a digital mushroom field guide available that is uploaded to your handheld computer. 
- Huge table of mushroom descriptions along with the crucial bit of information...

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-8



Living Longer With Machine Learning (2)

The Data

- Here are some of the attributes:



1	cap-shape	bell=b, conical=c, convex=x, flat=f, knobbed=k, sunken=s
2	cap-surface	fibrous=f, grooves=g, scaly=y, smooth=s
3	cap-color	brown=n, buff=b, cinnamon=c, gray=g, green=r, pink=p, purple=u, red=e, white=w, yellow=y
4	bruises	bruises=t, no=f
5	odor	almond=a, anise=l, creosote=c, fishy=y, foul=f, musty=m, none=n, pungent=p, spicy=s
6	gill-attachment	attached=a, descending=d, free=f, notched=n
...		
22	habitat	grasses=g, leaves=l, meadows=m, paths=p, urban=u, waste=w, woods=d

© 2002,2003 Thomas Hofmann

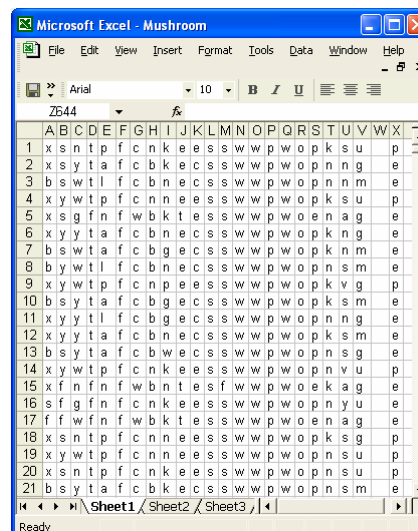
CS195-5-2003-01-9



Living Longer With Machine Learning (3)

The Data

- Here is what the table looks like
- Time for a little entrance examination ...



© 2002,2003 Thomas Hofmann

The training data

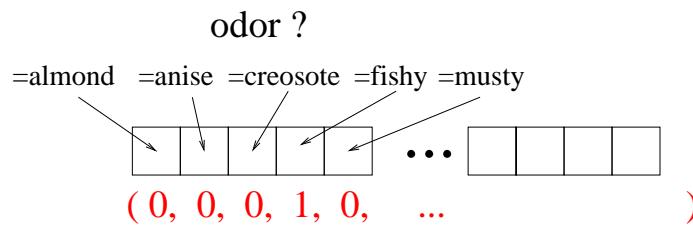
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
1	x	s	n	t	p	f	o	n	k	e	e	s	s	w	w	p	w	o	p	k	s	u		p	51	x	g	n	t	a	f	o	b	w	e	r	s	g	w	w	p	w	o	p	k	s	g		e
2	x	s	g	t	a	f	o	b	k	e	e	s	s	w	w	p	w	o	p	n	n	g		e	52	x	s	w	t	f	o	b	k	e	e	s	s	w	w	p	w	o	p	k	s	g		e	
3	b	s	w	t	f	o	b	n	e	e	s	s	w	w	p	w	o	p	n	n	m		e	53	b	s	w	t	f	o	b	k	e	e	s	s	w	w	p	w	o	p	k	s	g		e		
4	x	g	w	t	p	f	o	n	n	e	e	s	s	w	w	p	w	o	p	k	s	u		p	54	x	g	n	t	p	f	o	n	k	e	e	s	s	w	w	p	w	o	p	n	u		p	
5	x	s	g	f	n	f	w	b	k	t	e	s	s	w	w	p	w	o	e	n	a	g		e	55	x	s	w	t	p	f	o	n	k	e	e	s	s	w	w	p	w	o	p	k	u		p	
6	x	g	g	t	a	f	o	b	n	e	e	s	s	w	w	p	w	o	p	k	n	g		e	56	b	g	g	t	a	f	o	b	w	e	e	s	s	w	w	p	w	o	p	k	s	m		e
7	b	s	w	t	a	f	o	b	g	e	e	s	s	w	w	p	w	o	p	k	n	m		e	57	f	g	f	n	f	w	b	n	t	e	s	s	w	w	p	w	o	e	n	a	g		e	
8	b	g	w	t	f	o	b	n	e	e	s	s	w	w	p	w	o	p	n	s	m		e	58	b	s	w	t	a	f	o	b	w	e	e	s	s	w	w	p	w	o	p	n	n	g		e	
9	x	g	w	t	p	f	o	n	p	e	e	s	s	w	w	p	w	o	p	k	v	g		p	59	x	s	g	t	f	o	b	k	e	e	s	s	w	w	p	w	o	p	n	n	g		e	
10	b	s	g	t	a	f	o	b	g	e	e	s	s	w	w	p	w	o	p	k	s	m		e	60	x	g	n	t	a	f	o	b	p	e	r	s	g	w	w	p	w	o	p	k	j	p		e
11	x	g	g	t	f	o	b	g	e	e	s	s	w	w	p	w	o	p	n	n	g		e	61	s	f	g	f	n	f	o	n	k	e	e	s	s	w	w	p	w	o	p	n	u		e		
12	x	g	g	t	a	f	o	b	n	e	e	s	s	w	w	p	w	o	p	k	s	m		e	62	b	g	g	t	a	f	o	b	k	e	e	s	s	w	w	p	w	o	p	n	s	m		e
13	b	s	g	t	a	f	o	b	w	e	e	s	s	w	w	p	w	o	p	n	s	g		e	63	b	s	g	t	f	o	b	g	e	e	s	s	w	w	p	w	o	p	n	s	m		e	
14	x	g	w	t	p	f	o	n	k	e	e	s	s	w	w	p	w	o	p	n	u		p	64	b	g	g	t	f	o	b	g	e	e	s	s	w	w	p	w	o	p	n	n	m		e		
15	x	f	n	f	n	f	w	b	n	t	e	s	s	w	w	p	w	o	e	k	a	g		e	65	b	g	w	t	f	o	b	n	e	e	s	s	w	w	p	w	o	p	n	s	g		e	
16	s	f	g	f	n	f	o	n	k	e	e	s	s	w	w	p	w	o	p	n	u		e	66	f	s	n	f	w	b	k	t	e	s	s	w	w	p	w	o	e	k	a	g		e			
17	f	f	w	f	n	f	w	b	k	t	e	s	s	w	w	p	w	o	e	n	a	g		e	67	x	s	w	t	f	o	b	n	e	e	s	s	w	w	p	w	o	p	k	s	g		e	
18	x	s	n	t	p	f	o	n	n	e	e	s	s	w	w	p	w	o	p	k	s	g		p	68	f	g	g	t	a	f	o	b	w	e	r	s	g	w	w	p	w	o	p	n	s	g		e
19	x	g	w	t	p	f	o	n	n	e	e	s	s	w	w	p	w	o	p	n	s	u		p	69	x	g	g	t	a	f	o	b	w	e	s	s	w	w	p	w	o	p	k	n	g		e	
20	x	s	n	t	p	f	o	n	k	e	e	s	s	w	w	p	w	o	p	n	s	u		p	70	f	g	f	n	f	o	n	p	e	e	s	s	w	w	p	w	o	p	n	u		e		
21	b	s	g	t	a	f	o	b	k	e	e	s	s	w	w	p	w	o	p	n	s	m		e	71	f	g	t	f	w	n	p	t	b	s	w	w	p	w	o	p	n	v	d		e			
22	x	g	n	t	p	f	o	n	n	e	e	s	s	w	w	p	w	o	p	n	v	g		p	72	b	g	w	t	f	o	b	g	e	e	s	s	w	w	p	w	o	p	n	s	m		e	
23	b	g	g	t	f	o	b	k	e	e	s	s	w	w	p	w	o	p	n	s	m		e	73	f	g	t	f	w	n	w	t	b	s	w	w	p	w	o	p	n	v	d		e				
24	b	g	w	t	a	f	o	b	w	e	e	s	s	w	w	p	w	o	p	n	n	m		e	74	x	g	n	t	a	f	o	b	p	e	r	s	g	w	w	p	w	o	p	k	s	p		e
25	b	s	w	t	f	o	b	g	e	e	s	s	w	w	p	w	o	p	k	s	m		e	75	b	s	g	t	a	f	o	b	k	e	e	s	s	w	w	p	w	o	p	k	s	g		e	
26	f	s	w	t	p	f	o	n	n	e	e	s	s	w	w	p	w	o	p	n	v	g		p	76	f	g	t	f	w	n	p	t	b	s	w	w	p	w	o	p	n	v	d		e			
27	x	g	g	t	a	f	o	b	n	e	e	s	s	w	w	p	w	o	p	n	n	m		e	77	x	s	w	t	f	w	n	n	t	b	s	w	w	p	w	o	p	u	v	d		e		
28	x	g	w	t	f	o	b	w	e	e	s	s	w	w	p	w	o	p	n	n	m		e	78	f	g	n	t	f	o	b	p	e	r	s	g	w	w	p	w	o	p	n	g	p		e		
29	x	f	n	f	n	f	o	n	k	e	e	s	s	w	w	p	w	o	p	k	g	u		e	79	x	g	n	t	p	f	o	n	w	e	e	s	s	w	w	p	w	o	p	n	u		p	
30	x	s	g	t	a	f	w	n	n	t	b	s	w	w	p	w	o	p	n	v	d		e	80	f	g	n	t	a	f	o	b	n	e	r	s	g	w	w	p	w	o	p	n	g		e		
31	b	s	g	t	f	o	b	g	e	e	s	s	w	w	p	w	o	p	n	n	m		e	81	x	s	n	f	w	b	k	t	e	f	s	w	w	p	w	o	e	n	s	g		e			
32	x	g	w	t	p	f	o	n	k	e	e	s	s	w	w	p	w	o	p	n	s	u		p	82	x	g	w	t	p	f	o	n	w	e	e	s	s	w	w	p	w	o	p	k	s	g		p
33	x	g	g	t	f	o	b	n	e	e	s	s	w	w	p	w	o	p	n	n	m		e	83	f	g	f	n	f	o	n	n	e	e	s	s	w	w	p	w	o	p	n	u		e			
34	x	g	n	t	f	o	b	p	e	r	s	g	w	w	p	w	o	p	n	g	p		e	84	x	f	g	f	n	f	w	b	n	t	e	s	s	w	w	p	w	o	e	n	s	g		e	
35	b	g	g	t	f	o	b	n	e	e	s	s	w	w	p	w	o	p	k	s	m		e	85	x	g	g	t	f	o	b	w	e	r	s	g	w	w	p	w	o	p	k	s	g		e		
36	x	f	g	t	f	w	n	w	t	b	s	w	w	p	w	o	p	n	v	d		e	86	x	n	f	w	b	k	t	e	s	s	w	w	p	w	o	e	k	s	g		e					
37	s	f	g	f	n	f	o	n	k	e	e	s	s	w	w	p	w	o	p	k	v	u		e	87	b	s	w	t	a	f	o	b	w	e	e	s	s	w	w	p	w	o	p	k	s	g		e
38	x	g	n	t	p	f	o	n	w	e	e	s	s	w	w	p	w	o	p	n	s	u		p	88	x	s	w	t	f	o	b	n	e	e	s	s	w	w	p	w	o	p	n	s	g		e	
39	f	g	t	a	f	w	n	p	t	b	s	w	w	p	w	o	p	n	v	d		e	89	f	g	n	t	f	o	b	w	e	r	s	g	w	w	p	w	o	p	k	g		e				
40	b	s	g	t	f	o	b	k	e	e	s	s	w	w	p	w	o	p	k	s	m		e	90	s	f	n	f	o	n	n	e	e	s	s	w	w	p	w	o	p	n	u		e				
41	b	g	g	t	a	f	o	b	n	e	e	s	s	w	w	p	w	o	p	n	s	g		e	91	x	f	n	f	o	n	n	e	e	s	s	w	w	p	w	o	p	n	u		e			
42	x	g	g	t	f	o	b	n	e	r	s	g	w	w	p	w	o	p	k	j	p		e	92	b	s	w	t	f	o	b	k	e	e	s	s	w	w	p	w	o	p	k	s	g		e		
43	x	f	n	f	n	f	o	n	g	e	e	s	s	w	w	p	w	o	p	k	g	u		e	93	x	g	g	t	a	f	o	b	g	e	e	s	s	w	w	p	w	o	p	k	s	g		e
44	x	g	w	t	p	f	o	n	p	e	e	s	s	w	w	p	w	o	p	n	v	g		p	94	x	g	g	t	f	o	b	g	e	e	s	s	w	w	p	w	o	p	k	n	m		e	
45	x	s	g	t	a	f	o	b	w	e	e	s	s	w	w	p	w	o	p	k	n	m		e	95	x	s	n	f	w	b	n	t	e	s	s	w	w	p	w	o	e	n	a	g		e		
46	x	g	w	t	a	f	o	b	n	e	e	s	s	w	w	p	w	o	p	n	n	g		e	96	b	s	w	t	a	f	o	b	g	e	e	s	s	w	w	p	w	o	p	n	s	g		e
47	x	g	g	t	f	o	b	k	e	e	s	s	w	w	p	w	o	p	k	s	m		e	97	f	g	n	t	f	o	b	p	e	r	s														



Living Longer With Machine Learning (4)

The Representation

- Multi-valued attributes are mapped to binary representation (also known as *orthogonal encoding*)



© 2002,2003 Thomas Hofmann

CS195-5-2003-01-13



Living Longer With Machine Learning (5)

The Representation

x s n t p f c n k e e s w w p w o p k s u



```

00100000011000000000100000000100010100011000000000
00100001000000100010000000100000000101000100100000
0010010000000000001000000100

```

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-14

Discriminant Function

- Formally (for binary inputs)

$$f : \{0, 1\}^d \rightarrow \{-1, 1\}$$

- More generally:

$$f : \mathcal{X} \subseteq \mathbb{R}^d \rightarrow \{-1, 1\}$$

Edible or not

- Maps a pattern to its class or label

x **y**

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-15

Linear Discriminant

- Linear discriminant

$$f(\mathbf{x}) = \text{sign} \left(\sum_{i=1}^d x_i w_i + b \right), \quad w_i, b \in \mathbb{R}.$$

- Intuitively, each attribute which is present contributes with a **weight** that may be positive or negative.
- A **positive weight** indicates that the attribute provides evidence for the mushroom being edible, a **negative weight** provides evidence for the opposite hypothesis.
- The **magnitude of the weight** determines the influence of this attribute relative to other attributes.
- The **bias** specifies the (negative of the) threshold for making the final decision based on the accumulative evidence.

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-16

Learning Linear Discriminant

- Goal: Adjust weights and bias in way that maximizes the classification accuracy.
- Problem has been restricted to a pre-specified hypothesis class (linear discriminants)
- Learning procedure = parameter fitting

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-17

Perceptron Algorithm (1)

- Initialize the weights & bias
- Cycle through the training data
- Test whether current example is correctly classified
- If not, perform an update step by adjusting w, b . **Learning from mistakes.**
- Until all training data are correctly classified

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-18

Perceptron Algorithm (2)

Algorithm 1 Perceptron algorithm

```
1:  $\mathbf{w} \leftarrow 0, b \leftarrow 0$ 
2: repeat
3:    $errors \leftarrow 0$ 
4:   /* cycle through all training examples */
5:   for  $i = 1, \dots, n$  do
6:     compute  $f(\mathbf{x}_i) = \text{sign}(\langle \mathbf{w}, \mathbf{x}_i \rangle + b)$ 
7:     if  $f(\mathbf{x}_i) \neq y_i$  then
8:        $\mathbf{w} \leftarrow \mathbf{w} + y_i \mathbf{x}_i$ 
9:        $b \leftarrow b + y_i$ 
10:     $errors \leftarrow errors + 1$ 
11:   end if
12: end for
13: until  $error = 0$ 
14: output  $\mathbf{w}, b$ 
```

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-19

Matlab (1)

Matlab 1 Reading in a matlab data file.

```
% load data
load 'mushroom';

% get dimensions
[dim num] = size(features);
pos      = length(find(labels==1));
neg      = num - pos ;

% print some info
fprintf('Number of training data = %d\n', num);
fprintf('Pos/neg examples      = %d / %d \n', pos, neg);
fprintf('Number of input features = %d\n', dim);
```

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-20

Matlab (2)

Creating zero
vectors, no variable
declaration

for loop

Matrix indexing,
transposing,
multiplications

fprintf

```
Matlab 2 Perceptron algorithm, primal form.
% Initialize parameters
w = zeros(dim,1);
b = zeros(1,1);

% Perceptron learning, main loop
max_iter = 100;
for iter=1:max_iter
    errors = 0;
    for i=1:num
        if ( labels(i) * ( w' * features(:,i) + b ) <= 0 )
            w = w + labels(i) * features(:,i);
            b = b + labels(i);
            errors = errors + 1;
        end
    end
    fprintf('Iteration %d, errors = %d\n', iter, errors);
    if (errors==0)
        break;
    end
end
end
```

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-21

Generalization

- Assuming the perceptron converges, what do we know about the solution?
- It classifies all training examples correctly!
- Yet, how will the discriminant function perform on unseen new data? (Generalization)
- **Inductive inference**

How can we derive general rules
that apply to an infinite number of
cases from a finite set of examples?

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-22

Empirical vs. Expected Risk

- **Empirical risk** (based on zero-one loss)

$$J_{emp}(f) = \frac{1}{2n} \sum_{i=1}^n |f(\mathbf{x}_i) - y_i|$$

- **Expected risk**

$$J(f) = \frac{1}{2} \int |f(\mathbf{x}) - y| dP(\mathbf{x}, y)$$

Underlying probability law that generates the data.

© 2002,2003

CS195-5-2003-01-23

Estimating Generalization Performance

- Statistical learning theory: derive bounds of the type

$$J(f^*) \leq J_{emp}(f^*) + \text{some function of } n \text{ and } \mathcal{H}$$

- Empirical estimate via **hold out data**
 - Leave out some random subset of the examples during training (e.g. 20%)
 - Test of these hold-out examples after training

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-24

Applications of Pattern Recognition

- Optical character recognition

- Recognize hand written characters
- *Zip code recognition in mail sorting facilities*
- *Reading numbers on checks*



- Object recognition

- *Recognize object types in images*
- *Face detection, recognizing animals, cars, etc.*
- *Image annotation for content-based retrieval*



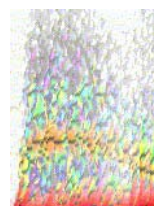
© 2002,2003 Thomas Hofmann

CS195-5-2003-01-25

Applications of Pattern Recognition

- Automatic speech recognition

- Transcribing speech signals into written text
- Usually: mapping windows (10ms) of speech to phonemes (approx. 20 different sound classes in English)
- *Use in voice computer interfaces, calling centers, etc.*



- Text Categorization

- Annotating documents with content descriptors (topics, categories, taxonomies)
- *Content management, filtering, routing*



© 2002,2003 Thomas Hofmann

CS195-5-2003-01-26

Applications of Pattern Recognition

- Medical diagnosis
 - Prediction based on clinical measurement (lab tests)
 - *Disease prediction or decision about treatment*

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-27

Classification & Regression

- Concept learning

$$f : \mathbb{R}^d \rightarrow \{-1, 1\} \quad \text{dichotomy}$$

- Multi-class classification

$$f : \mathbb{R}^d \rightarrow \{1, \dots, K\} \quad \begin{array}{l} \text{finite number} \\ \text{of responses} \\ = \text{qualitative} \end{array}$$

- Regression

$$f : \mathbb{R}^d \rightarrow \mathbb{R} \quad \text{quantitative}$$

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-28

Applications of Regression

- Financial forecasting

- Given the past values, observe the next value in a series
- *Stock market prediction*
- *Time series prediction*



- Optimal Control

- Given direct or indirect observations of the state of a system and a desired state or trajectory: compute optimal control output
- Robot arms, Space crafts, etc.



© 2002,2003 Thomas Hofmann

CS195-5-2003-01-29

Applications of Regression

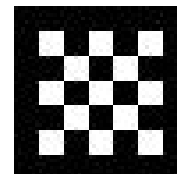
- Drug Design

- *Predict chemical and biological properties of new compounds*



- Game Heuristics

- *Evaluate game positions (approximately) for use in a heuristic search procedure*
- *Map positions to a score*



© 2002,2003 Thomas Hofmann

CS195-5-2003-01-30

Unsupervised Learning

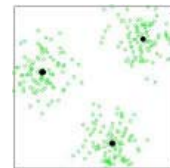
- Supervised learning (classification, regression)
 - Input patterns along with target values are given as training data
 - Targets can be class labels or numerical values
 - Find (stochastic or deterministic) mapping from inputs to outputs
- Unsupervised learning
 - Only a set of input patterns is given

© 2002,2003 Thomas Hofmann

CS195-5-2003-01-31

Unsupervised Learning

- Structure detection
 - Data clustering
 - finding groups of similar patterns
 - Document clustering, protein clustering, texture-based image segmentation
 - Dimension reduction
 - Finding low-dimensional subspaces or data manifolds
 - Data mining
 - Discovering unexpected regularities



© 2002,2003 Thomas Hofmann

CS195-5-2003-01-32

Unsupervised Learning

- Probabilistic modeling (density estimation)

- Medical diagnosis:

- learn about dependencies between symptoms and diseases
- *E.g. Bayesian networks*



- Data compression

- Shannon: optimal expected codeword length $E[-\log p(x)] = \text{entropy}$
- For example: image compression



© 2002,2003 Thomas Hofmann

CS195-5-2003-01-33

The End



© 2002,2003 Thomas Hofmann

CS195-5-2003-01-34