Advanced Computer Networks

Stephan Sigg

Georg-August-University Goettingen, Computer Networks

26.06.2014

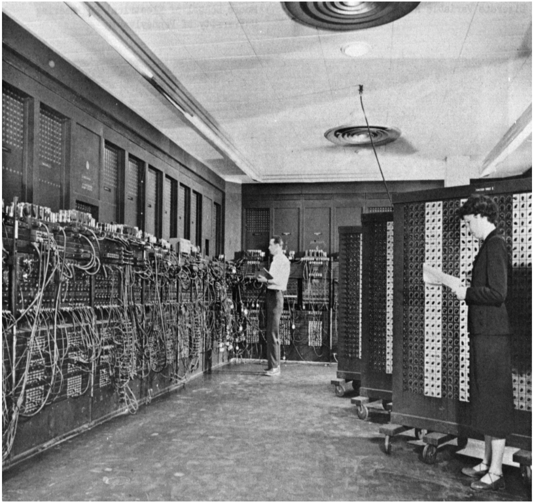
Outline

Introduction

Activity recognition

Data collection and training of the classifier

Conclusion



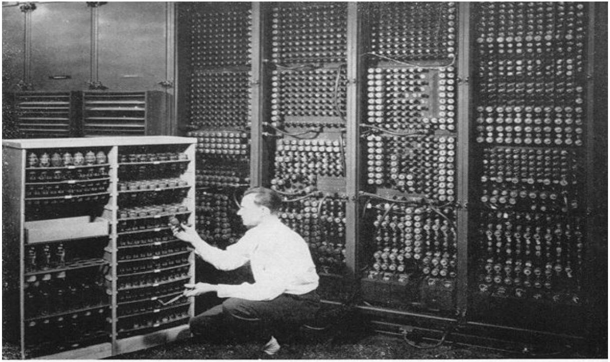
Monolithic machines

ENIAC, 1946
US army
Calculation of ballistic tables

Size:	10m x 17m x 2.7m
Weight:	27 tons
Performance:	174 kW (> 17000 tubes)
Price:	468.000 $
Comp.Power:	500 Additions per second

Monolithic machines

- Many people share a single machine
- No network required



Replacing a bad tube meant checking among ENIAC's 19,000 possibilities.

Interconnected, fixed machines

- Private devices for each single person
- Fixed networks
- Focus on service quality
- Error correction in transmission

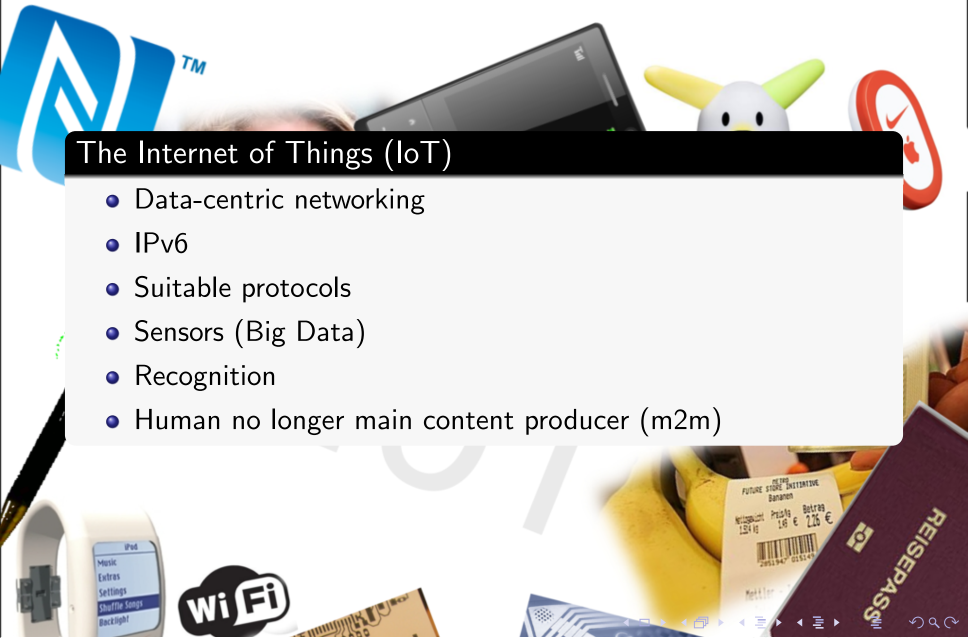
First Apple Macintosh (January 24, 1984)

Personal machines

- Wireless networking
- Mobility
- handover
- mobile IP

The Internet of Things (IoT)

- Data-centric networking
- IPv6
- Suitable protocols
- Sensors (Big Data)
- Recognition
- Human no longer main content producer (m2m)



This lecture

- ~~Protocols~~
- ~~Sensor hardware~~
- Sensing and activity recognition

Opportunity

WiSee

Outline

Introduction

Activity recognition

Data collection and training of the classifier

Conclusion

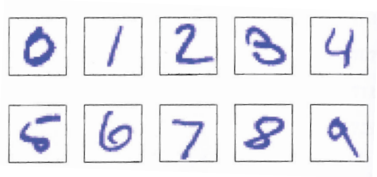
Recognition of patterns

Patterns can be described by a sufficient number of rules

Samples are inaccurate

Tremendous amount of rules to model all variations of one class

Therefore: Consider machine learning approaches

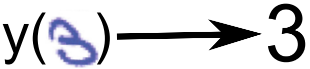


Recognition of patterns

Training set $x_1 \ldots x_N$ of a large number of N samples is utilised

Classes $t_1 \ldots t_N$ of all samples in this set known in advance

Machine learning algorithm computes a function $y(x)$ and generates a new target t‘



Polynomial curve fitting

Problem setting

A curve shall be approximated by a machine learning approach

- Vehicle speed from vibration
- Housing prices
- Season from temperature, humidity, pressure
- ...

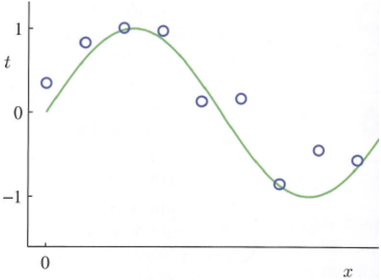
Polynomial curve fitting

Example

A curve shall be approximated by a machine learning approach

Artificial example here:

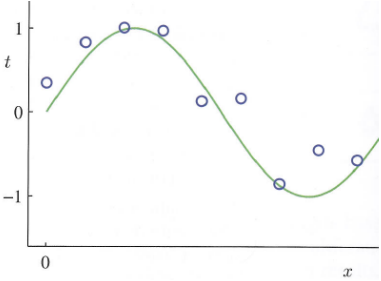
Sample points are created for the function $\sin(2\pi x) + \mathcal{N}$ where $\mathcal{N}$ is a random noise value



Polynomial curve fitting

We will try to fit the data points into a polynomial function:

$$y(x, \overrightarrow{w}) = w_0 + w_1 x + w_2 x^2 + \cdots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$



Polynomial curve fitting

We will try to fit the data points into a polynomial function:

$$y(x, \overrightarrow{w}) = w_0 + w_1 x + w_2 x^2 + \cdots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

This can be obtained by minimising an error function that measures the misfit between $y(x, \overrightarrow{w})$ and the training data set:

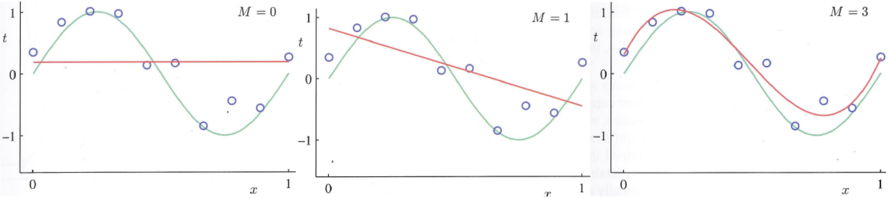
$$E(\overrightarrow{w}) = \frac{1}{2} \sum_{i=1}^{N} \left[y(x_i, \overrightarrow{w}) - t_i \right]^2$$

$E(\overrightarrow{w})$ is non-negative and zero if and only if all points are covered by the function

Polynomial curve fitting

One problem is the right choice of the dimension M

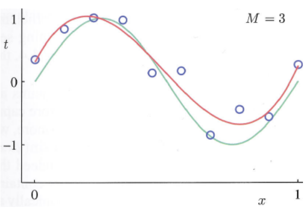
When M is too small, the approximation accuracy might be low

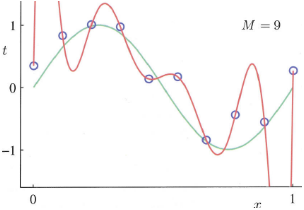


Polynomial curve fitting

However, when M becomes too big, the resulting polynomial will cross all points exactly

When M reaches the count of samples in the training data set, it is always possible to create a polynomial of order M that contains all values in the data set exactly.





Polynomial curve fitting

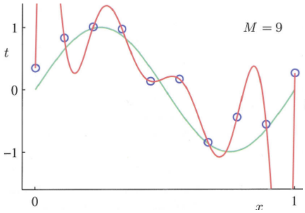
This event is called overfitting

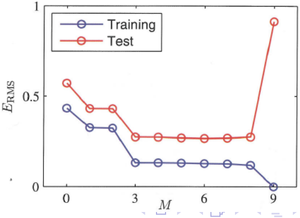
The polynomial now trained too well to the training data

It will therefore perform badly on another sample of test data for the same phenomenon

We visualise it by the Root of the Mean Square (RMS) of $E(\overrightarrow{w})$

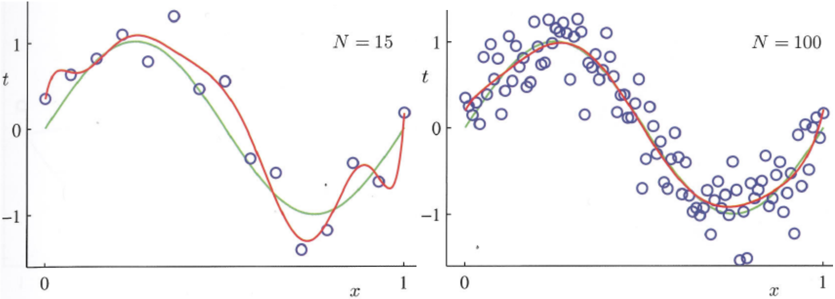
$$E_{RMS} = \sqrt{\frac{2E(\overrightarrow{w})}{N}}$$





Polynomial curve fitting

With increasing number of data points, the problem of overfitting becomes less severe for a given value of M



Polynomial curve fitting

One solution to cope with overfitting is regularisation

A penalty term is added to the error function

This term discourages the coefficients of $\overrightarrow{w}$ from reaching large values

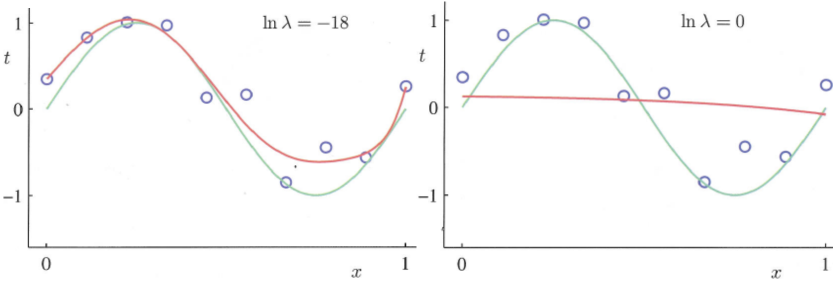
$$\overline{E}(\overrightarrow{w}) = \frac{1}{2}\sum_{i=1}^{N}\left[y(x_i, \overrightarrow{w}) - t_i\right]^2 + \frac{\lambda}{2}||\overrightarrow{w}||^2$$

with

$$||\overrightarrow{w}||^2 = \overrightarrow{w}^T\overrightarrow{w} = w_0^2 + w_1^2 + \cdots + w_M^2$$

Polynomial curve fitting

Depending on the value of λ, overfitting is controlled



$$\overline{E}(\overrightarrow{w}) = \frac{1}{2}\sum_{i=1}^{N}\left[y(x_i, \overrightarrow{w}) - t_i\right]^2 + \frac{\lambda}{2}||\overrightarrow{w}||^2$$

Parameter optimisation with gradient descent

Repeatedly modify the weights w_i with the weighted derivation of their previous value

$$w_i = w_i - \alpha \cdot \overline{E}'$$

Outline

Introduction

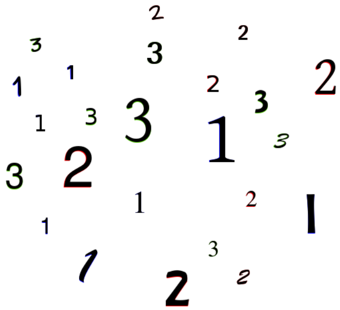
Activity recognition

Data collection and training of the classifier

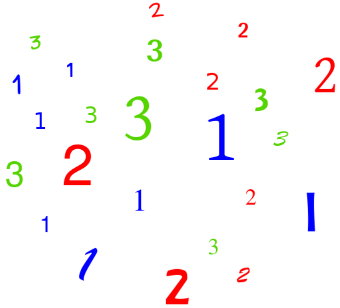
Conclusion

Training of a machine learning system

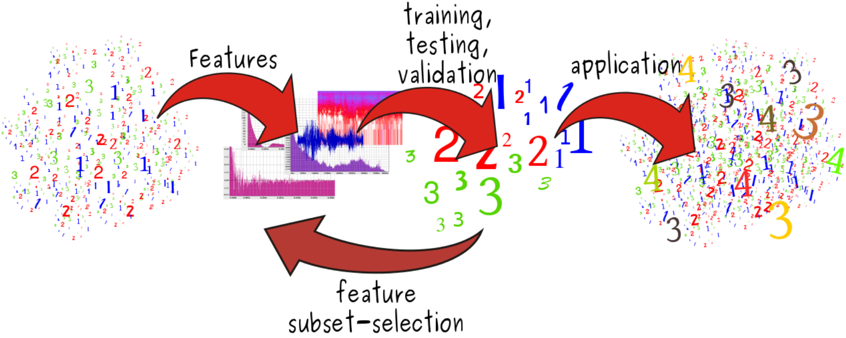
Training of a machine learning system

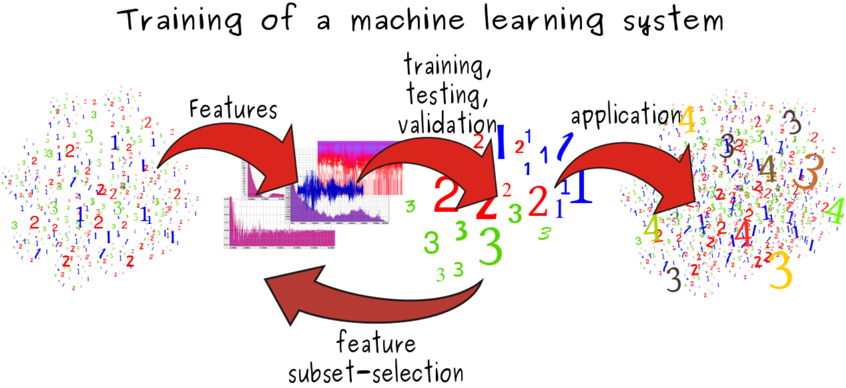


Training of a machine learning system



Training of a machine learning system

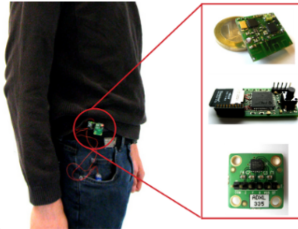




- Mapping of features onto classes by using prior knowledge
- What are characteristic features?
- Which approaches are suitable to obtain these features?

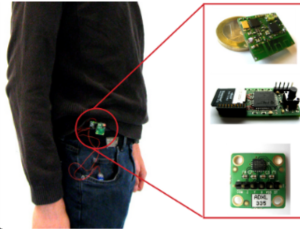
Data sampling

- Record sufficient training data
  - Annotated! (Ground-truth)
  - Multiple subjects
  - Various environmental conditions (time of day, weather, ...)



Data sampling

- Record sufficient training data
  - Annotated! (Ground-truth)
  - Multiple subjects
  - Various environmental conditions (time of day, weather, ...)

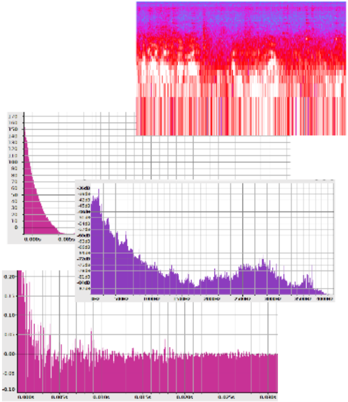


Example

- Electric supply data over 15 years covers 5000 days but only 15 christmas days
- Especially critical events like accidents (e.g. plane, car, earthquake) are scarce

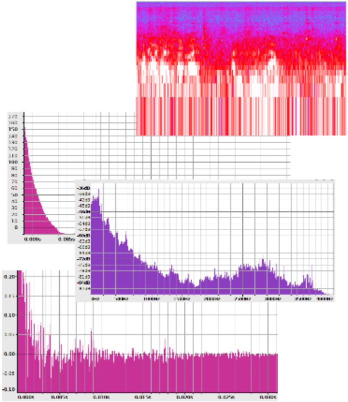
Feature extraction

- Identify meaningful features
  - remove irrelevant/redundant features



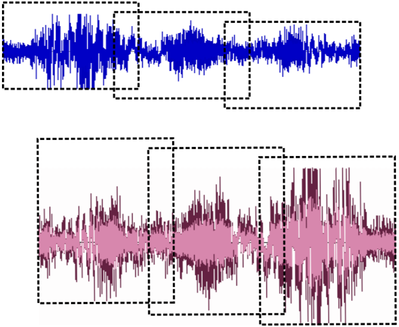
Feature extraction

- Identify meaningful features
    - remove irrelevant/redundant features
- Features can be contradictory!



Feature subset-selection

- Pre-process data
    - Framing
    - Normalisation



Feature subset-selection

- Pre-process data
  - Framing
  - Normalisation

Domain knowledge?
-> better set of ad-hoc features

Features commensurate?
-> normalise

Pruning of input required?
-> if no, create disjunctive features or weithted sums of features

Independent features?
-> construct conjunctive features or products of features

Is the data noisy?
-> detect outlier examples

Do you know what to do first?
-> If not, use a linear predictor

Feature subset-selection

Simple ranking of features with correlation coefficients

Example: Pearson Correlation Coefficient

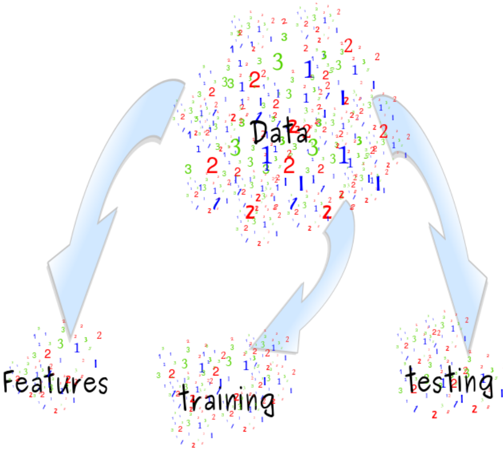
$$\varrho(X, Y) = \frac{\mathrm{Cov}(X, Y)}{\sqrt{\mathrm{Var}(X)\mathrm{Var}(Y)}} \tag{1}$$

- Identifies linear relation between input variables x_i and an output y

Feature subset-selection

How to do reasonable feature selection

- Utilise dedicated test- and training- data-sets
- Pay attention that a single raw-data sample could not impact features in both these sets
- Don't train the features on the training- or test- data-set

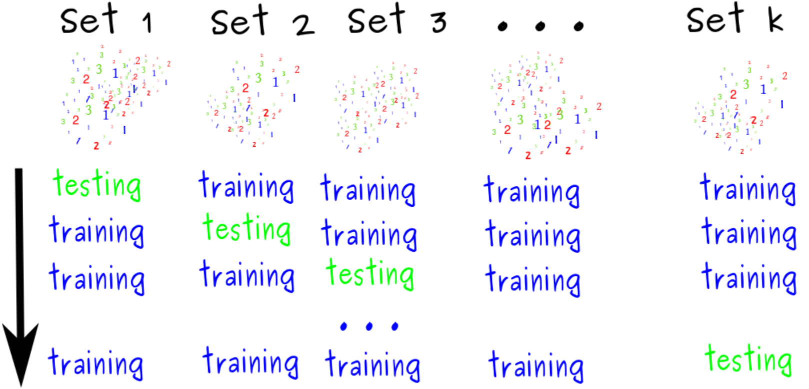


Training of the classifier

Evaluation of classification performance

k-fold cross-validation

- Standard: k=10

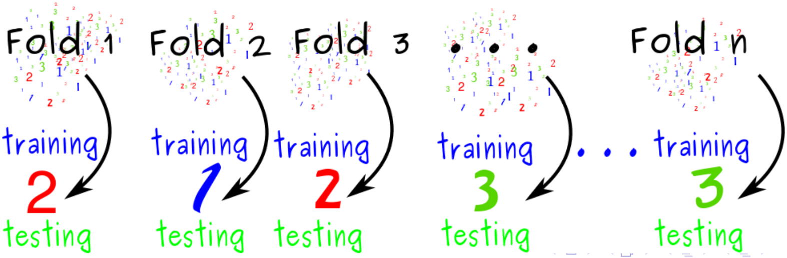

Set 1
Set 2
Set 3
. . .
Set k
testing training training training training
training testing training training training
training training testing training training
. . .
training training training training testing

Training of the classifier

Evaluation of classification performance

Leave-one-out cross-validation

- n-fold cross validation where n is the number of instances in the data-set
- Each instance is left out once and the algorithm is trained on the remaining instances
- Performance of left-out instance (success/failure)

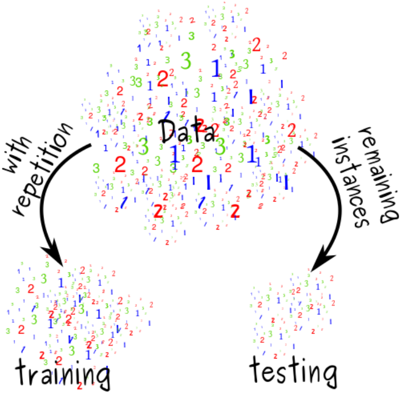


Training of the classifier

Evaluation of classification performance

0.632 Bootstrap

- Form training set by choosing n instances from the data-set with replacement
- All not picked instances are used for testing
- Probability to pick a specific instance: $1 - \left(1 - \frac{1}{n}\right)^n \approx 1 - e^{-1} \approx 0.632$

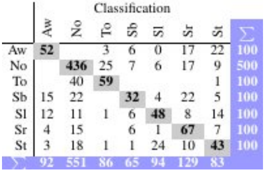


Training of the classifier

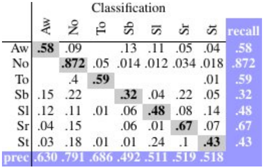
Evaluation of classification performance

Classification accuracy

- Confusion matrices
- Precision
- Recall



	Classification							
	Aw	No	To	Sb	Sl	Sr	St	∑
Aw	**52**		3	6	0	17	22	100
No		**436**	25	7	6	17	9	500
To		40	**59**				1	100
Sb	15	22		**32**	4	22	5	100
Sl	12	11	1	6	**48**	8	14	100
Sr	4	15		6	1	**67**	7	100
St	3	18	1	1	24	10	**43**	100
∑	92	551	86	65	94	129	83	



	Classification							
	Aw	No	To	Sb	Sl	Sr	St	recall
Aw	**.58**	.09		.13	.11	.05	.04	.58
No		**.872**	.05	.014	.012	.034	.018	.872
To		.4	**.59**				.01	.59
Sb	.15	.22		**.32**	.04	.22	.05	.32
Sl	.12	.11	.01	.06	**.48**	.08	.14	.48
Sr	.04	.15		.06	.01	**.67**	.07	.67
St	.03	.18	.01	.01	.24	.1	**.43**	.43
prec	.630	.791	.686	.492	.511	.519	.518	

Training of the classifier

Evaluation of classification performance

Information score

Let C be the correct class of an instance and $\mathcal{P}(C)$, $\mathcal{P}'(C)$ be the prior and posterior probability of a classifier

We define:[1]

$$I_i = \begin{cases} -\log(\mathcal{P}(C)) + \log(\mathcal{P}'(C)) & \text{if } \mathcal{P}'(C) \geq \mathcal{P}(C) \\ -\log(1-\mathcal{P}(C)) + \log(1-\mathcal{P}'(C)) & \text{else} \end{cases} \tag{2}$$

The information score is then

$$\mathrm{IS} = \frac{1}{n}\sum_{i=1}^{n} I_i \tag{3}$$

[1] I. Kononenko and I. Bratko: Information-Based Evaluation Criterion for Classifier's Performance, Machine Learning, 6, 67-80, 1991.

Training of the classifier

Evaluation of classification performance

Brier score

The Brier score is defined as

$$\text{Brier} = \sum_{i=1}^{n} (t(x_i) - p(x_i))^2 \tag{4}$$

where

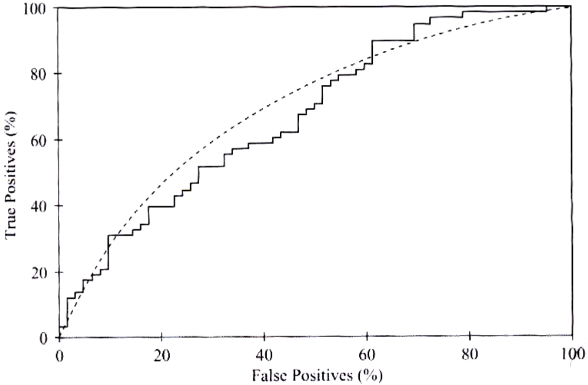
$$t(x_i) = \begin{cases} 1 & \text{if } x_i \text{ is the correct class} \\ 0 & \text{else} \end{cases} \tag{5}$$

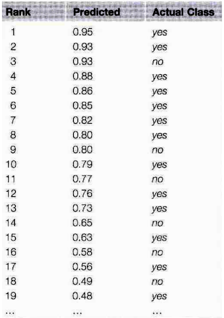
and $p(x_i)$ is the probability the classifier assigned to the class x_i.

Training of the classifier

Evaluation of classification performance

Area under the receiver operated characteristic (ROC) curve (AUC)





Rank	Predicted	Actual Class
1	0.95	*yes*
2	0.93	*yes*
3	0.93	*no*
4	0.88	*yes*
5	0.86	*yes*
6	0.85	*yes*
7	0.82	*yes*
8	0.80	*yes*
9	0.80	*no*
10	0.79	*yes*
11	0.77	*no*
12	0.76	*yes*
13	0.73	*yes*
14	0.65	*no*
15	0.63	*yes*
16	0.58	*no*
17	0.56	*yes*
18	0.49	*no*
19	0.48	*yes*
...	...	...

Pattern recognition and classification

Data mining frameworks

- Orange Data Mining (http://orange.biolab.si/)
- Weka Data Mining (http://www.cs.waikato.ac.nz/ml/weka/)

orange

WEKA
The University
of Waikato

Questions?

Stephan Sigg
stephan.sigg@cs.uni-goettingen.de

Literature

- C.M. Bishop: Pattern recognition and machine learning, Springer, 2007.
- P. Tulys, B. Skoric, T. Kevenaar: Security with Noisy Data – On private biometrics, secure key storage and anti-counterfeiting, Springer, 2007.
- R.O. Duda, P.E. Hart, D.G. Stork: Pattern Classification, Wiley, 2001.