Lecture 1

Introduction

Intellectual systems
(Machine Learning)

Andrey Filchenkov

05.09.2016

Lecture plan

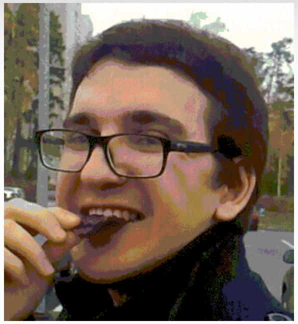
- Organizational questions
- Concept of machine learning
- Supervised learning
- Overfitting and model validation
- Examples

- The presentation is prepared with materials of the K.V. Vorontsov's course "Machine Leaning".

Lecture plan

- Organizational questions
- Concept of machine learning
- Supervised learning
- Overfitting and model validation
- Examples

Teaching Staff



Andrey Filchenkov (lectures)
PhD in Math, Ass. Prof., researcher

Ivan Smetannikov (seminars)
PhD student, senior researcher

Sergey Muravyov (seminars)
PhD student, junior researcher

Valeria Efimova (seminars)
Master student, junior researcher

Andrey Danilchenko (seminar + lecture)
developer in Yandex

ML research group

- Part of “Computer technologies lab”
- The four full-time researchers + students

- Research interests:
  - meta-learning
  - feature selection
  - deep networks
  - practical applications
  - …

Preliminary lectures plan

- Introduction (1 lecture)
- Simple classification (5 lectures)
- Regression and ranks (2 lectures)
- Complex classification (2 lectures)
- Recommender systems (1 lecture)
- Unsupervised learning (3 lectures)
- Data mining workflow (1 lecture)

Note: it can be dramatically changed!

Course schedule

Starts at 11:40

Usually one class is lecture class, and another class is practical seminar.

Today you have two lectures.

Many shrifts and changes are expected.

“How can I get grade?”

The course ends with examination.

You can have automatically given grade before.

You will have to

- submit solutions on seminars
- pass theoretical test on lectures
- solve individual/group assignment.

Ways you can improve your grade

You can improve your grade by:

- writing a coursework / thesis on machine learning / data mining
- improving this course (limited)
- writing papers on machine learning / data mining to conferences and journals
- attending ML hackathon course

Ways you cannot improve you grade

You cannot improve your grade by:

- attending lectures
- giving money to teaching staff (we consider only blue chips and immobile property)
- being a relative or a friend of teachers

Lecture plan

- Organizational questions
- Concept of machine learning
- Supervised learning
- Overfitting and model validation
- Examples

Machine learning definitions

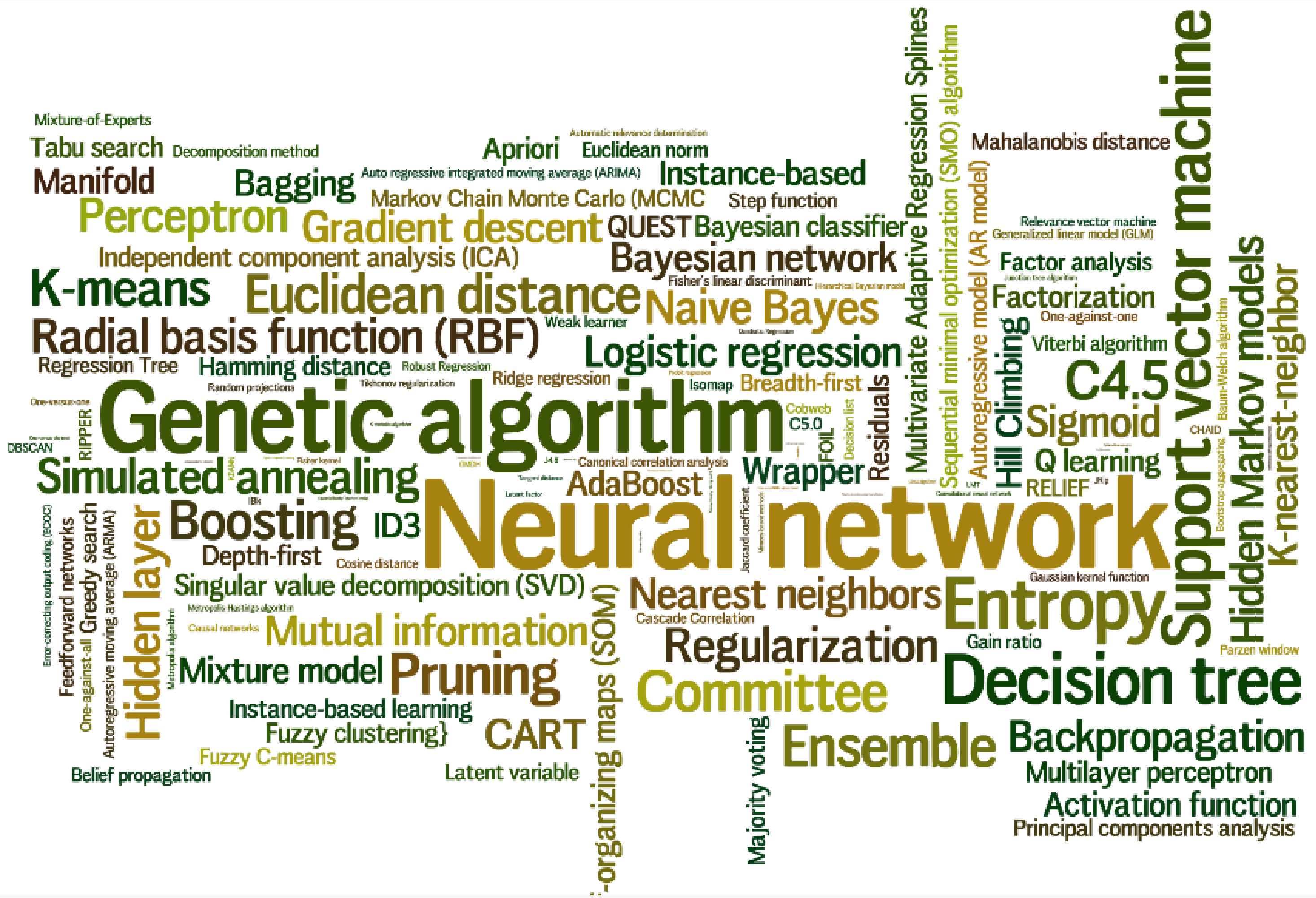
Machine learning is process (field of study) that gives computers ability to learn without being explicitly programmed.

A.L. Samuel Some Studies in Machine Learning Using the Game of Checkers // IBM Journal. July 1959. P. 210–229.

A computer program is said to be **learn** from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.

T.M. Mitchell Machine Learning. McGraw-Hill, 1997.

Machine Learning Approaches



Machine Learning Applications

Related fields

- Pattern recognition
- Computer vision
- Data mining (DM)
- Information Retrieval (IR)
- Natural Language Processing (NLP)
- Neural Computation
- …

Related concepts

- Artificial intelligence

  Strong AI vs Weak AI
- Intellectual systems

  Expert system vs ML systems
- Mathematical modeling

- Way of knowledge representation and using

Knowledge vs data

Knowledge ≠ data

Knowledge is patterns in a certain domain (principals, regularities, relations, rules, laws), gained with practice and professional experience, which helps to formulate and solve problems in a certain field.

Machine Learning vs Data Mining

Formally, DM is a step in **knowledge discovery in databases** (KDD). Usually, these two terms are synonyms.

1. Collect data
2. Engineer features
3. Apply machine learning algorithms

Required background

- Probability theory and mathematical statistics
- Optimization
- Computational science
- Linear algebra
- Discrete math
- Computational complexity theory
- …

Machine learning problems

- Supervised learning
- Unsupervised learning
- Semi-supervised learning
- Reinforcement learning
- Active learning
- Online learning
- Structured prediction
- Model selection and validation

Supervised learning

Set of examples with answers is given. Rule for giving answers for all possible examples is required:

- classification;
- regression;
- learning to rank;
- forecasting.

Unsupervised learning

Set of examples is given, but no answers. Rule for finding answers or some regularity is required:

- clustering;
- association rules learning;
- recommender systems*;
- dimension reduction**.

Machine learning problems classification

- Supervised learning
- Unsupervised learning
- Semi-supervised learning
- Reinforcement learning
- Active learning
- Online learning
- Structured prediction
- Model selection and validation

Machine learning problems we are to discuss in this course

- **Supervised learning**
- **Unsupervised learning**
- Semi-supervised learning
- Reinforcement learning
- Active learning
- Online learning
- Structured prediction
- Model selection and validation

Lecture plan

- Organizational questions
- Concept of machine learning
- Supervised learning
- Overfitting and model validation
- Examples

Supervised learning

- We are going to talk about supervised learning most of the time

The problem

X is **object set**, or input set;
Y is **label set**, or **answer set,** or output set;
$y : X \to Y$ is unknown **target function (dependency**).
$\{x_1, \dots, x_\ell\} \subset X$ is **training sample**;
$y_i = y(x_i),\ i = 1, \dots, \ell$ are **known values** of the function.

Problem: find $a : X \to Y$ is **solving function** (decision function), which approximate y on X.

We are going to speak only about **algorithms**. What is the difference between algorithms and functions?

Main questions

1. How are the objects described?
2. How do the answers look like?
3. What is algorithm set from which a is being chosen?
4. How to measure quality of how a approximates y?

How are the objects described?

$f_j : X \to D_j$, $j = 1, \ldots, n$ are **features** or **attributes**.

Feature types:

- **binary**: $D_j = \{0, 1\}$;
- **categorical**: D_j is finite;
- **ordinal**: D_j is finite and ordered;
- **numerical**: $D_j = \mathbb{R}$.

Features data

$(f_1(x), \dots, f_n(x))$ is feature description of an object x.
Object is its feature description.

Data are usually represented with the matrix objects-features (feature data):

$$F = \left\| f_j(x_i) \right\|_{\ell \times n} = \begin{pmatrix} f_1(x_1) & \dots & f_n(x_1) \\ \dots & \dots & \dots \\ f_1(x_\ell) & \dots & f_n(x_\ell) \end{pmatrix}.$$

How do the answers look like?

Classification:

- $Y = \{-1, +1\}$ – binary;
- $Y = \{1, \dots, M\}$ – M non-overlapping classes;
- $Y = \{0, 1\}^M$ – M classes that can overlap.

Ranking:

- Y – finite (partially) ordered set.

Regression:

- $Y = \mathbb{R}$ or $Y = \mathbb{R}^m$.

What is algorithm set from which a is being chosen?

Algorithms model is a parametric family of mappings

$$A = \{g(x, \theta) | \theta \in \Theta\},$$

where $g : X \times \Theta \rightarrow Y$ is fixed function, Θ is set of possible values of parameter θ.

Example: **linear model** with parameter vector $\theta = (\theta_1, \dots, \theta_n)$, $\Theta = \mathbb{R}^n$.

Which type of problem is the one where the function is

$$g(x, \theta) = \sum_{j=1}^{n} \theta_j f_j(x)\,?$$

Learning method

Learning method is mapping

$$\mu\colon (X \times Y)^\ell \to A,$$

which for a certain training set $T^\ell = \{(x_i, y_i)\}_{i=1}^{\ell}$ returns an algorithm $a \in A$.

Two steps:

1. **Training**:
with method μ on training set T^ℓ build $a = \mu(T^\ell)$.
2. **Testing**:
apply a for new object x to find answer $a(x)$.

How to measure quality of how a approximates y?

Loss function $L(a, x)$ — error size of an algorithm a on object x

- for classification problem:

$$L(a, x) = [a(x) \neq y(x)]$$

- for regression problem:

$$L(a, x) = d\big(a(x) - y(x)\big),$$

usually quadratic loss function:

$$d(x) = x^2, L(a, x) = \big(a(x) - y(x)\big)^2.$$

Empirical risk — quality measure of algorithm a on T^ℓ:

$$Q(a, T^\ell) = \frac{1}{\ell}\sum_{i=1}^{\ell} L(a, x_i).$$

Empirical risk minimization

Empirical risk minimization method

$$\mu\left(T^{\ell}\right) = \operatorname{argmin}_{a \in A} Q(a, T^{\ell}).$$

Decreasing error on train set can lead to a certain problem of lack of generalization.

Lecture plan

- Organizational questions
- Concept of machine learning
- Supervised learning
- Overfitting and model validation
- Examples

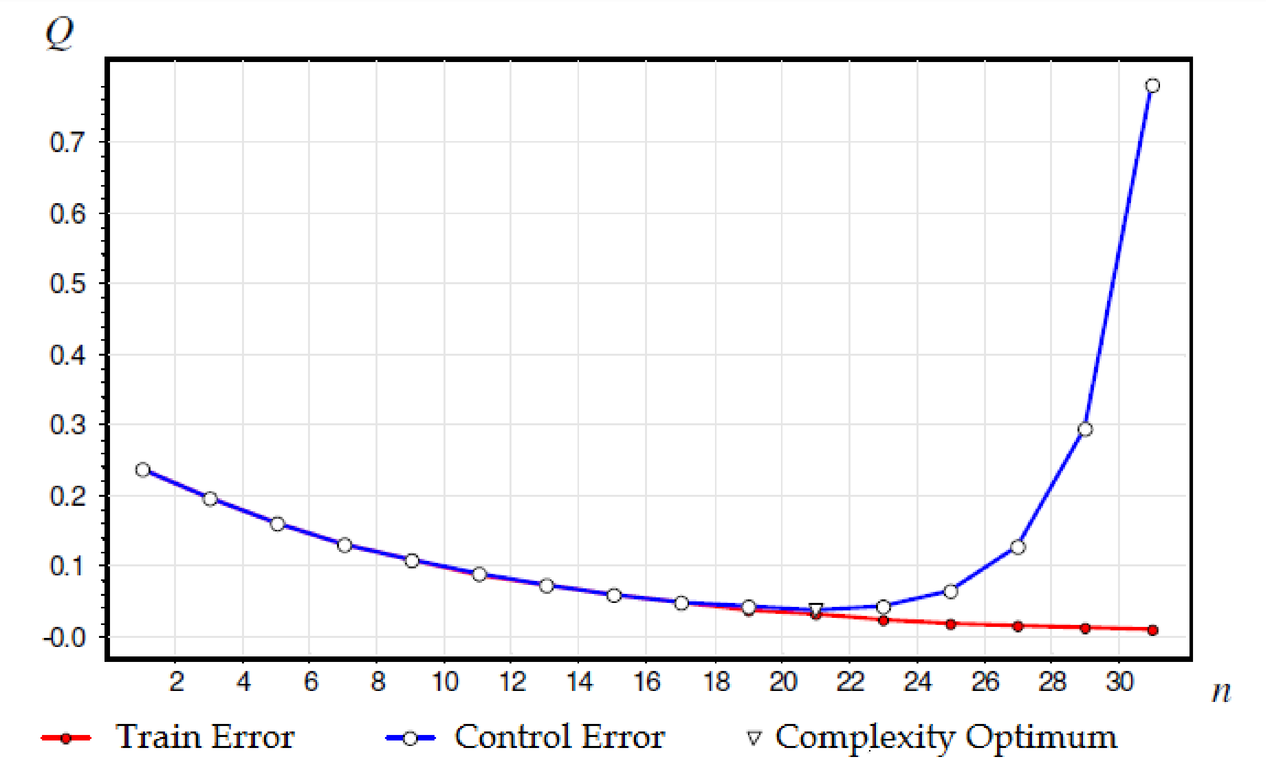
Overfitting problem

Overfitting problem: from a certain model complexity level, the better an algorithm performs on train set X^ℓ, the worse it performs on real world objects.

Example of overfitting

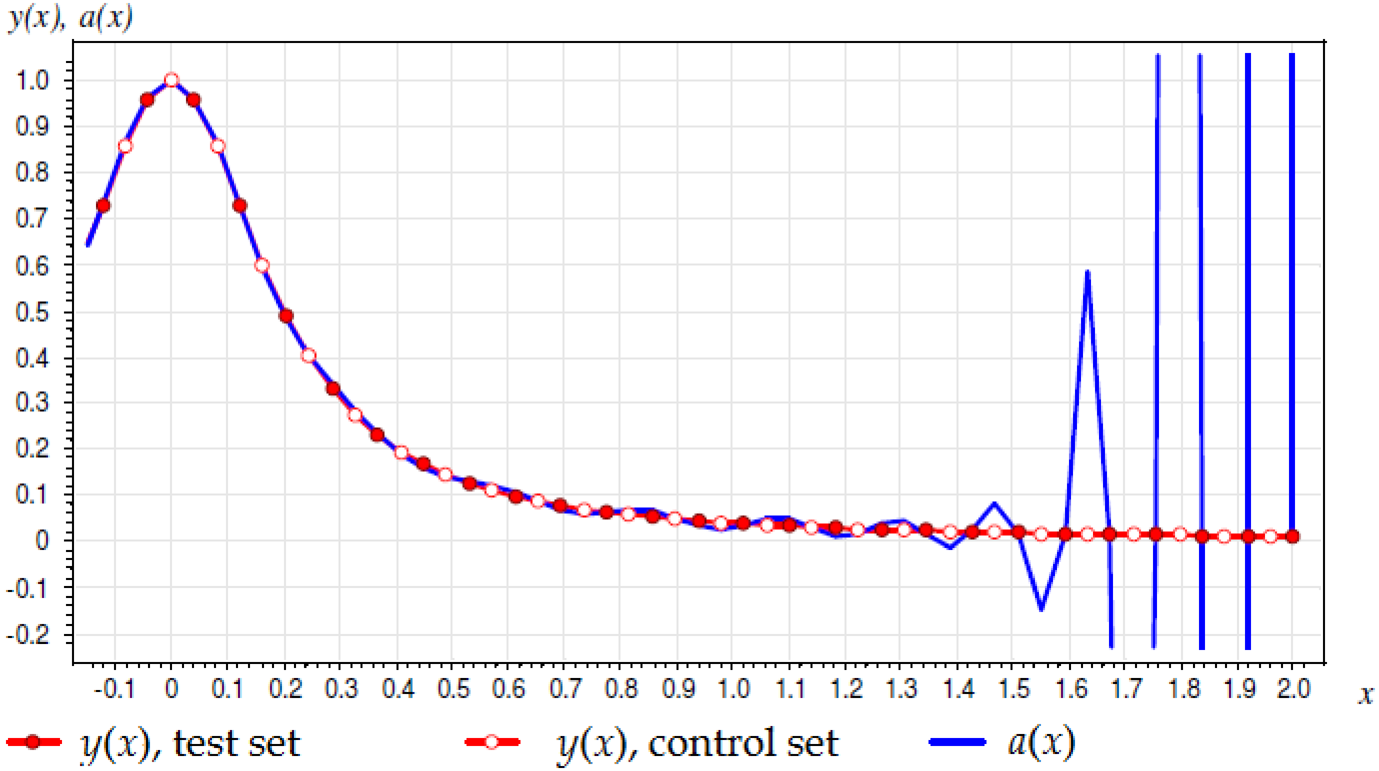
Dependency $y(x) = \frac{1}{1+25x^2}$ defined on $x \in [-2, 2]$.

Let search a function among polynomials with degree n.



Overfitted algorithm

$$y(x) = \frac{1}{1+25x^2}; \quad a(x) \text{ — polynomial of degree } n = 38$$



Lecture plan

- Organizational questions
- Concept of machine learning
- Supervised learning
- Overfitting and model validation
- Examples

Examples (1/3)

1. Medical diagnosis problem
For a patient decide, what is his or her illness, risks and treatment.

2. Credit scoring
For an applicant decide, if he or she return a credit.

3. Spam filtering
For a letter decide if it is spam or not.

4. Documents categorization
For a document pick categories, to which it belongs, or topics, which are represented in it.

Examples (2/3)

5. Immobile property cost forecasting
For a house or land predict it cost or factors that have impact on its cost.

6. Sales rate forecasting
For history of sales predict, how much a certain shop will sell goods or how many certain goods will be sold.

7. Search engine results ranking
For a search query return most relevant links.

8. Collaborative filtering
For a user determine his preferences (movies, books, music, goods).

Examples (3/3)

9. Detecting consumers categories

For a set of consumers find groups with similar degree of interest in a certain product.

10. Signature authentication

For smb's signature define if it's real or fake.

11. Forecasting stock indices

Predict values and dynamics of stock indices.

12. Computational synthesis of drugs

Predict if a molecule can be used in a certain drug.

References (1/2)

1. Bishop C.M. Pattern recognition and machine learning. Springer, 2006.
2. Duda R. O., Hart P. E., Stork D. G. Pattern classification. New York: JohnWiley and Sons, 2001.
3. Hastie T., Tibshirani R., Friedman J. The elements of statistical learning: Data Mining, Inference, and Prediction. 2nd Edition. Springer, 2009
4. Mitchell T. Machine learning. McGraw Hill,1997.
5. Vapnik V.N. The nature of statistical learning theory. NY: Springer, 1995.
6. Rassel S., Norvig P. Artificial Intelligence: Modern Approach. Prentice Hall Inc., 1995.

References (2/2)

MOOC courses (coursera.org):

- A. Ng "Machine Learning"
- D. Koller "Probabilistic Graphical Model"
- G. Hinton "Neural Networks for Machine Learning"

MOOC in Russian

- K.V. Vorontsov "Machine Leaning".