Introduction

Machine Learning

Sergey Muravyov

20.02.2018

Lecture plan

- Course organization
- Concept of machine learning
- Examples of ML problems
- Books and materials
- Optimization problem
- Supervised learning

Lecture plan

- Course organization
- Concept of machine learning
- Examples of ML problems
- Books and materials
- Optimization problem
- Supervised learning

Parts of ML course

ML course consists of three parts:

1. Introduction track
2. Method-oriented track

Have I missed something?

Parts of ML course

ML course consists of three parts:

1. Introduction track
2. Method-oriented track

Have I missed something?

Yes, you will have an exam in the end!

Method-oriented track

Lectures are for learning new methods

Seminars are for applying them in practice

Lecture plan

- Course organization
- Concept of machine learning
- Examples of ML problems
- Books and materials
- Optimization problem
- Supervised learning

Machine learning definitions

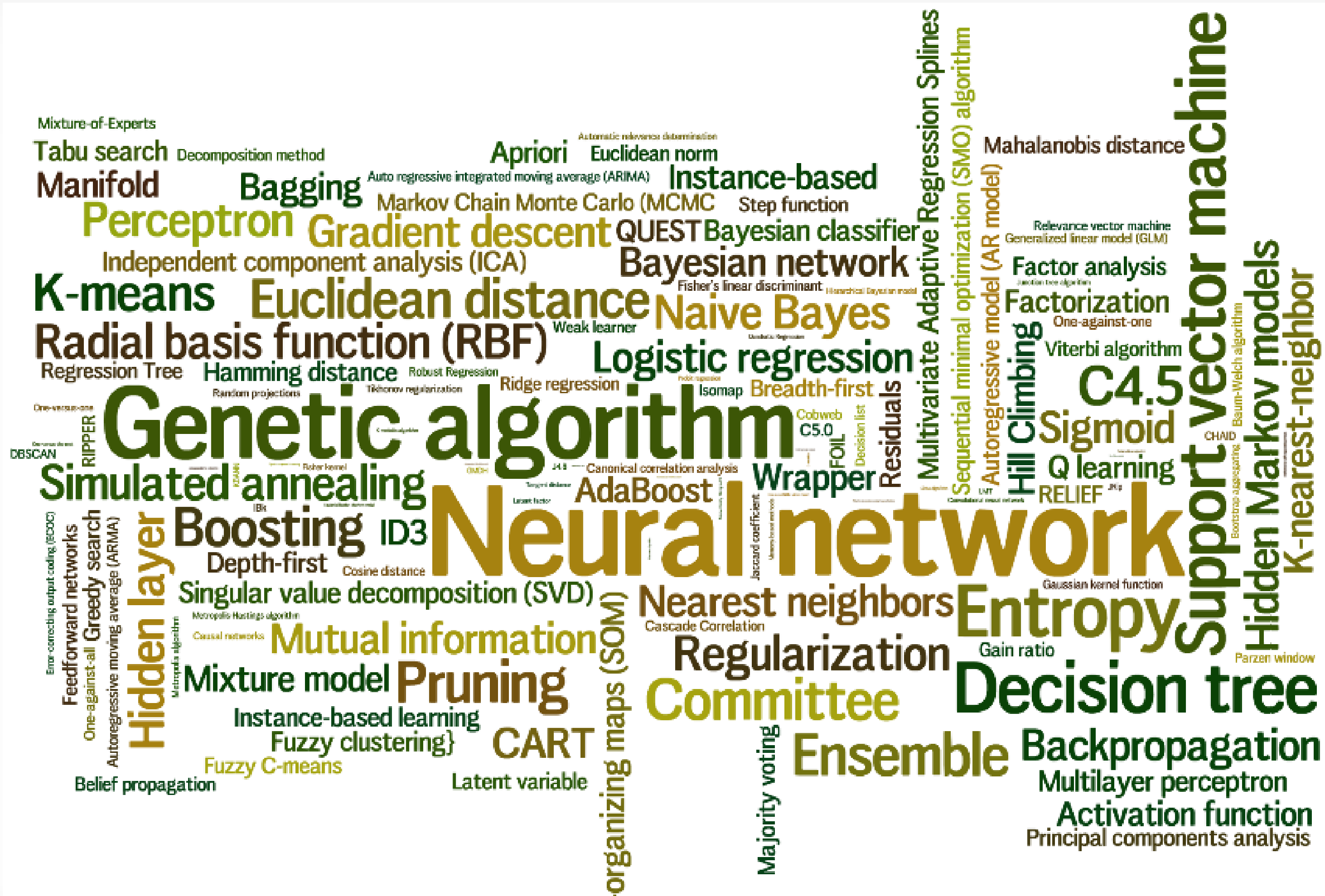
Machine learning is a process (field of study) that gives computers ability to learn without being explicitly programmed.

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 **learnt** 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
- Informational Retrieval
- Natural Language Processing
- Neural Computation
- …

Related concepts

- Artificial intelligence
  Strong AI vs Weak AI
- Intellectual systems
  Expert system vs ML systems
- Mathematical modelling
- Way of knowledge representation and using

Knowledge vs data

Knowledge ≠ data

Knowledge consists of 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. Pre-process data
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
- Reinforcement learning
- Semi-supervised learning
- Active learning
- Online learning
- Structured prediction
- Model selection and tuning

Supervised learning

Set of examples with answers is given.
A 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. A rule for finding answers or some regularity is required:

- clustering;
- dimension reduction;
- association rules learning;
- model selection (very general problem).

Model selection and tuning

How to choose an algorithm?

There are many parametrized families of algorithms. You should choose both a family (model selection) and its parameters (tuning).

Machine learning problems

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

Lecture plan

- Course organization
- Concept of machine learning
- Examples of ML problems
- Books and materials
- Optimization problem
- Supervised learning

Examples (1/3)

1. Medical diagnosis problem
For a patient, decide, what is his/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 that are represented in it.

Examples (2/3)

5. Immobile property cost forecasting
For a house or land, predict its 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 the 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 someone's signature, define if it is 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.

Lecture plan

- Course organization
- Concept of machine learning
- Examples of ML problems
- **Books and materials**
- Optimization problem
- Supervised learning

Books

1. Hastie T., Tibshirani R., Friedman J. The elements of statistical learning: Data Mining, Inference, and Prediction. 2nd Edition. Springer, 2009

2. Bishop C.M. Pattern recognition and machine learning. Springer, 2006.

3. Mitchell T. Machine learning. McGraw Hill, 1997.

4. Vapnik V.N. The nature of statistical learning theory. NY: Springer, 1995.

5. Rassel S., Norvig P. Artificial Intelligence: Modern Approach. Prentice Hall Inc., 1995.

6. Givens G.H., Hoeting J.A. Computational Statistics, 2nd Edition. Wiley, 2012

Web sources

MOOC courses (coursera.org):

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

YouTube courses:

- N. de Freitas "Deep Leaning" (at Oxford, 2015)
https://www.youtube.com/playlist?list=PLE6Wd9FR--EfW8dtjAuPoTuPcqmOV53Fu
- N. de Freitas "Machine Learning" (at UBC, 2013)
https://www.youtube.com/playlist?list=PLE6Wd9FR--EdyJ5lbFl8UuGjecvVw66F6
- A. Ng "Machine Learning (at Stanford, 2014)
https://www.youtube.com/playlist?list=PLA89DCFA6ADACE599

Lecture plan

- Course organization
- Concept of machine learning
- Examples of ML problems
- Books and materials
- **Optimization problem**
- Supervised learning

Optimization problem (simple)

Optimization is the process of finding the minima or the maxima of a function.

A point x^* is a **global maximum** if $f(x) \leq f(x^*)\ \forall x$ and is a **global minimum** if $f(x) \geq f(x^*)\ \forall x$.

Consider $f\colon \mathbb{R} \to \mathbb{R}$, such that f' and f'' are continuous. Then necessary conditions for maximum are:

1. $f'(x) = 0$.
2. $f''(x) \leq 0$.

Optimization problem (general)

Optimization is the process of finding the minima or the maxima of a function.

A point x^* is a **global maximum** if $f(x) \leq f(x^*)\ \forall x$ and is a **global minimum** if $f(x) \geq f(x^*)\ \forall x$.

Consider $f: \mathbb{R} \to \mathbb{R}$, such that f' and f'' are continuous. Then necessary conditions for maximum are:

1. $f'(x) = 0$.

2. $f''(x) \leq 0$.

Optimization problem (general)

1. It is not necessary defined on $\mathbb{R}$.
2. It is not univariate.
3. Solutions for $f'(x) = 0$ may be hard to compute.
4. It may be not smooth.
5. We may lack time.

Optimization problem (general)

Assume f, g and h_j are defined on a variable space X, $x \in X$. Then, the problem is stated as follows:

$$\begin{cases} f(x) \to \min_x \\ \quad g_i(x) \le 0, & \qquad i = 1, \dots, m; j = 1, \dots, k. \\ \quad h_j(x) = 0. \end{cases}$$

Optimization methods

- Bisection method
- Fixed point integration
- Gradient descent
- Newton's method
- Coordinate descent
- Maximum likelihood
- …

Lecture plan

- Course organization
- Concept of machine learning
- Examples of ML problems
- Books and materials
- Optimization problem
- Supervised learning

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**).
$\{x_1, \ldots, x_\ell\} \subset X$ is **training sample**;
$y_i = y(x_i),\ i = 1, \ldots, \ell$ are **known values** of the function.

Problem: find $a : X \to Y$, **solving function** (decision function), which approximates 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?

$f_j : X \to D_j$, $j = 1, \dots, n$ 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 is usually represented with matrix objects-features:

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

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

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

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 a 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 $L(a, x)$ is the error size of 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 is a 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.