ITMO

Introduction to ML

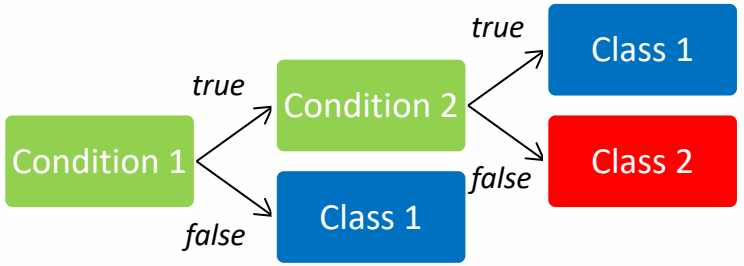
Computer Vision

Introduction to Machine Learning

Recognition Problems

- **The main problem:** the rules must be selected manually.

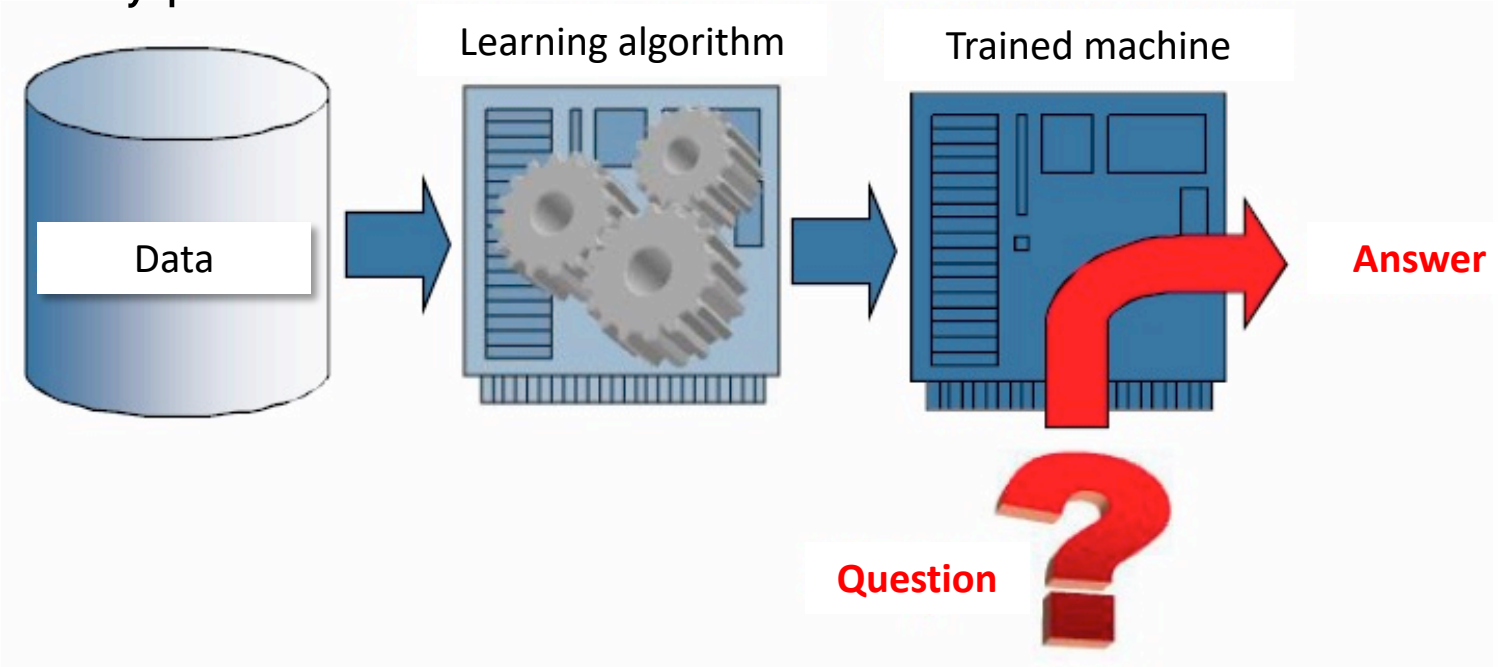
Geometric features for selected segments



- **Consequences:**
  1. Meaningful and informative features should be used.
  2. There are very few such features, and their complex combinations cannot be processed.

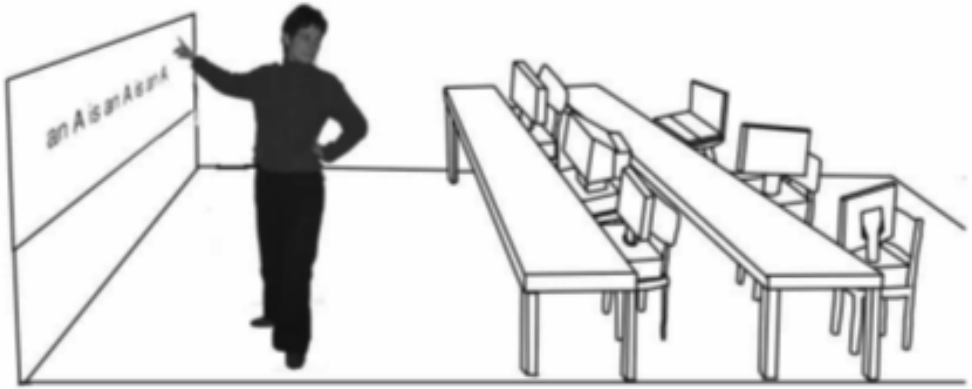
Ideal Solution

- The machine gives answers to questions based on the data already processed.



Learning Process

- Learning is not the same as memorization, memorization is not a problem for a machine.
- The machine must learn to draw inferences from a set of training data.
- The machine must work correctly based on new data that was not given to it before.

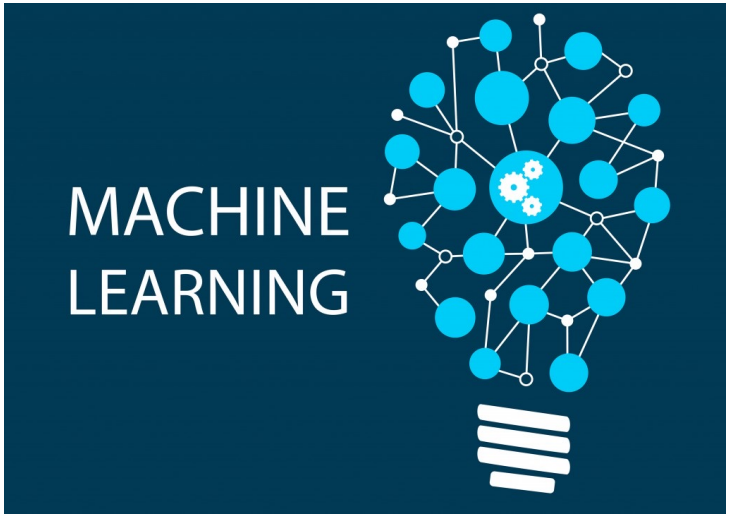


Definition

- «A computer program is said to 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, 1997.

Applications

- computer vision,
- speech recognition,
- computational linguistics and natural language processing,
- medical diagnostics,
- bioinformatics,
- technical diagnostics,
- financial applications,
- search and rubricating of texts,
- expert systems,
- etc.



Machine Learning Classes

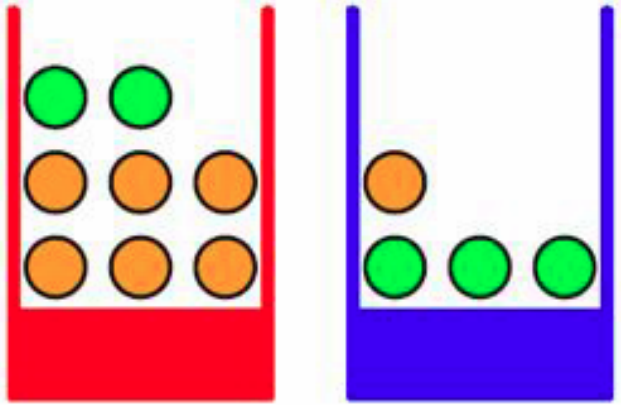
ITMO

1. **Deductive learning** (from general to particular).
   - There are formalized data.
   - It is required to derive a rule applicable to a particular case based on formalized data.
   - Typical example: expert systems.
2. **Inductive learning** (from particular to general).
   - There are empirical data. Need to restore some dependency.
   - Subdivided into:
     a. Supervised learning;
     b. Unsupervised learning;
     c. Reinforcement learning;
     d. Active learning etc.

- What is a probability?
  1. In frequency interpretation: the probability is the frequency of a repeating event.
  2. In the Bayesian interpretation: the probability is a measure of the experiment’s outcome uncertainty.

Example: Extraction of Fruit From Two Boxes

- **Experiment:**
  - Box selection;
  - Fruit extraction;
  - Putting the fruit back.
- **Two random variables:**
  - X – the color of the box (red or blue);
  - Y – fruit (orange or apple).

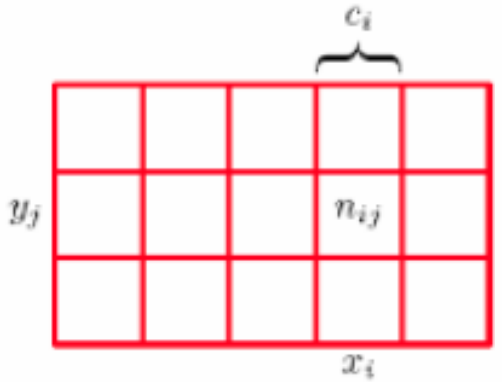


$$P(X = red) = \frac{How\ many\ times\ the\ red\ box\ was\ chosen}{How\ many\ experiments\ were\ done}$$

- P is the probability of choosing the red box.

Example: Extraction of Fruit From Two Boxes

- We will perform an experiment and enter the number of outcomes in the table (horizontally – the colors of the box, vertically – fruits).



- ***Event intersection probability***:

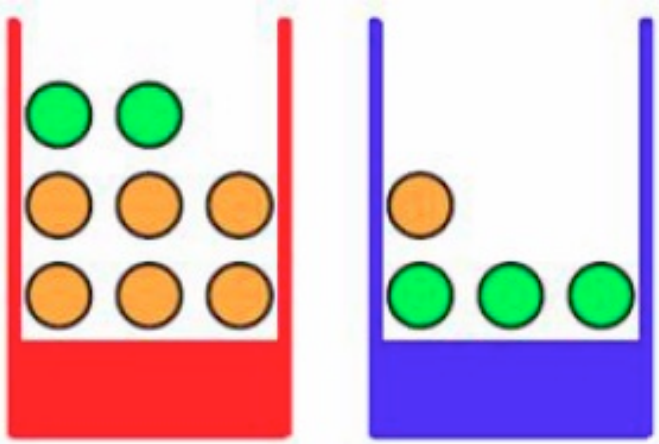
$$P(X = x_i, P = y_j) = \frac{n_{ij}}{N},$$

where N – the number of experiments, the number of outcomes n_{ij}.

- ***Conditional Probability:***

$$P(P = y_j | X = x_i,) = \frac{n_{ij}}{c_i}.$$

Example: Extraction of Fruit From Two Boxes



P(Y|X)=red

P(Y|X)=blue

Conditional Probability:
75% that an orange in a red box
25% that it in blue.

Bayes Formula

- What is the probability that we have a dinosaur in front of us (x – observation)?

$$P(y|x) = \frac{P(x|y) \cdot P(y)}{P(x)} - \text{Bayes formula,}$$

where $P(x|y)$ – is the probability that the dinosaur looks like this;
$P(y)$ – the probability of meeting a dinosaur;
$P(x)$ – the probability of seeing such a scene.

Probability Theory

- Sum rule:

$$P(x) = \int_y P(x, y)dy \leftrightarrow P(y) = \int_x P(x, y)dx$$

- Product rule:

$$P(x, y) = P(y|x)P(x) = P(x|y)P(y)$$

- If two random variables are independent:

$$P(x, y) = P(x)P(y)$$

Machine Learning Tasks

- Setting the task of *supervised learning*:
  - $\mathbb{X}$ – a set of objects or examples, situations, inputs (samples);
  - $\mathbb{Y}$ – a set of answers or labels, outputs (responses).
- There is some dependence that allows predicting $y \in \mathbb{Y}$ from $x \in \mathbb{X}$.
- If the dependence is deterministic, then there is a function $f^*: \mathbb{X} \to \mathbb{Y}$.
- The dependence is known only on the objects of the training sample, it means that we know some finite number of data:

$$\left\{\left(x^{(i)}, y^{(i)}\right): x^{(i)} \in \mathbb{X}, y^{(i)} \in \mathbb{Y}(i = 1, \dots, N)\right\}.$$

Machine Learning Tasks

- An ordered object-response pair $\left(x^{(i)}, y^{(i)}\right) \in (\mathbb{X} \times \mathbb{Y})$ is called a *precedent*.
- **The task of supervised learning** is to restore the relationship between input and output based on the existing training sample,
  - i.e., it is necessary to design a function (decision rule) $f: \mathbb{X} \to \mathbb{Y}$, for new objects $x \in \mathbb{X}$ predicting the answer $f(x) \in \mathbb{Y}$:

$$y = f(x) \approx f^*(x).$$

Basic Definitions

- The functions f are chosen from the parametric family F, i.e. from a set of possible models.
- The process of finding the function f is called *learning*, as well as tuning or fitting the model.
- An algorithm for designing a function f from a given training set is called a *learning algorithm*.
- Some class of algorithms is called a *learning method*.

Basic Definitions

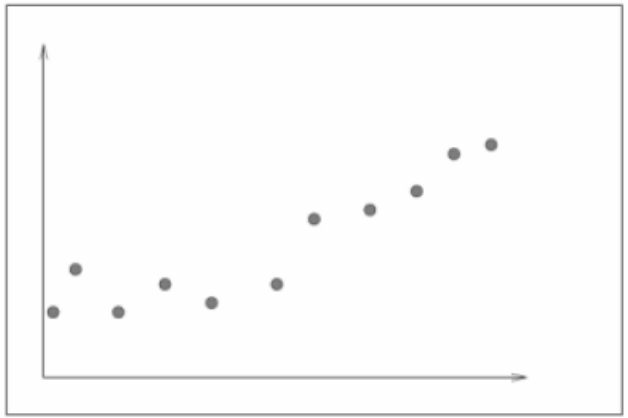
- Learning algorithms operate with object descriptions: each sample element is described by a set of features $x = (x_1, x_2, \dots, x_d)$ (**feature vector**), where $x_j \in Q_j, j = \overline{1,d}, \mathbb{X} = Q_1 \times Q_2 \times \cdots Q_d$.
- The set $\mathbb{X}$ is called the *feature space*.
- It is necessary to design such a function $y = f(x)$ from the feature vector $x = (x_1, x_2, \dots, x_d)$, that would give the answer y for any possible observation x.
- The component x_j is called the j-th *feature*, or property, or *attribute* of the object x.

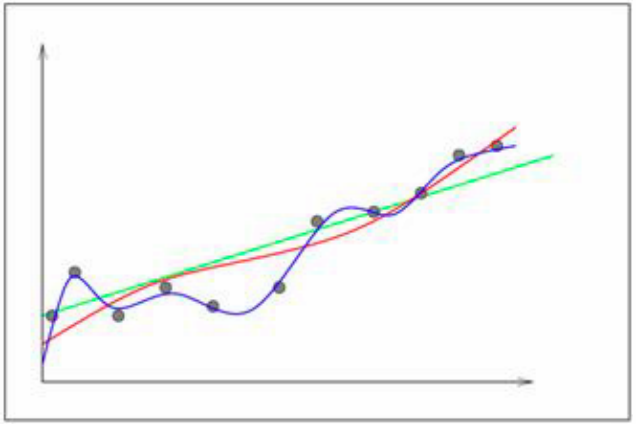
Basic Definitions

- If $Q_j = \mathbb{R}$, then the j-th attribute is called *quantitative* or *real*.
- If Q_j is finite, then the j-th feature is called *nominal, or categorical, or factor*.
  - If $\dim Q_j = 2$, then the feature is called *binary*.
  - If Q_j is ordered, then the feature is called *ordinal*.

Regression Retrieval Task

- If $\mathbb{Y} = \mathbb{R}$, then it is a *regression retrieval* task.
  - decision rule f called *regression*.
- If $\mathbb{Y}$ is finite $\mathbb{Y}$={1,2,...,K}, then this is a *classification* task.
  - The decision rule f is called the *classifier*.





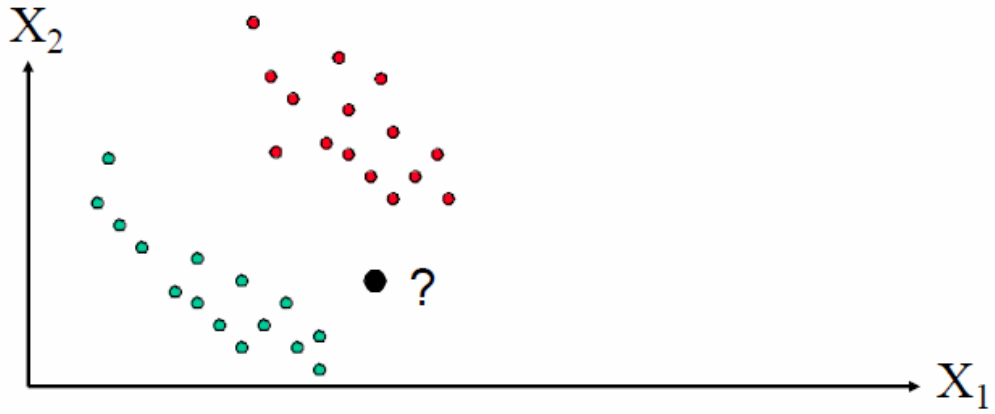
An example of regression retrieval, y is a continuous value.

Binary Classification Task

- Given a training sample:

$$X_m = \{(x_1, y_1), \ldots, (x_m, y_m)\}, (x_i, y_i) \in R^m \times Y, Y = \{-1, +1\}.$$

- Objects belong to one of two classes.
  - We mark the main class as “+1”, the secondary “background” as “−1”.
- It is required for all new values x to define the class "+1" or "−1".

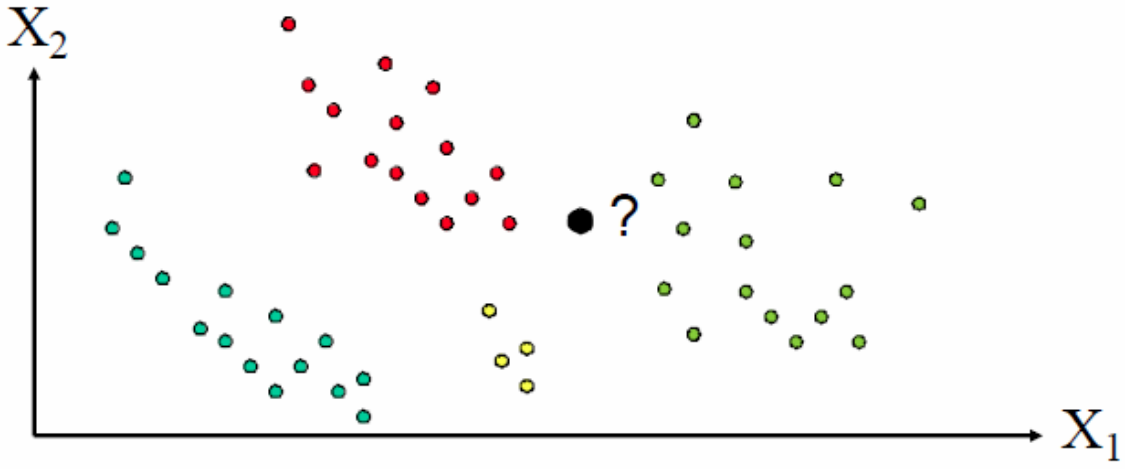


Multiclass Classification Task

- Given a training sample:

$$X_m = \{(x_1, y_1), \ldots, (x_m, y_m)\}, (x_i, y_i) \in R^m \times Y, Y = \{1, \ldots, K\}.$$

- Objects belong to one of the K classes.
- It is required for all new values x to define a class and put a label from 1 to K.

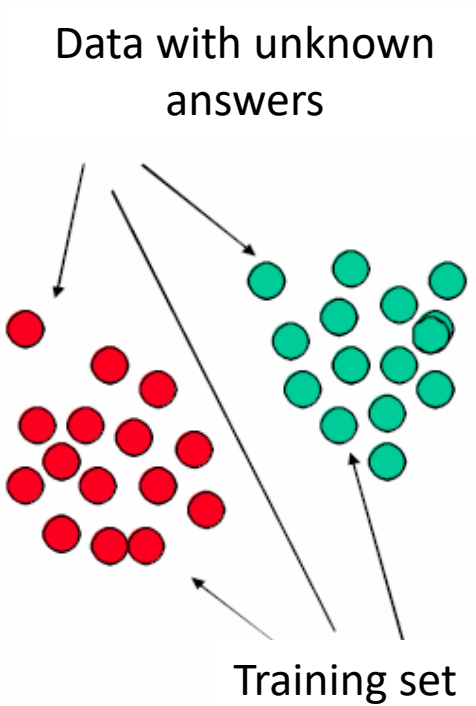


Remarks

1. The found decision rule should have a generalizing ability (the constructed classifier or regression function should reflect the overall dependence of the output on the input, based only on known data about the precedents of the training sample).
2. Attention should be paid to the problem of effective computability of the function f and to the learning algorithm: the model tuning should take place in an acceptable time.

Machine Learning Tasks

- The quality of the algorithm on new data is interesting: it is necessary to connect the existing data with those that we will process in the future.
- For this, the values of the features will be considered random variables.
- We will assume that the data that will have to be processed in the future and the available data are distributed equally.



Machine Learning Paradigms

- **Discriminative Approach**
- We will choose functions f from the parametric family F, i.e. from some set of possible models.
- Let us introduce some loss function $L(y, f(x))$ of the true value of the output y and the predicted value $f(x)$:

Discriminative Approach

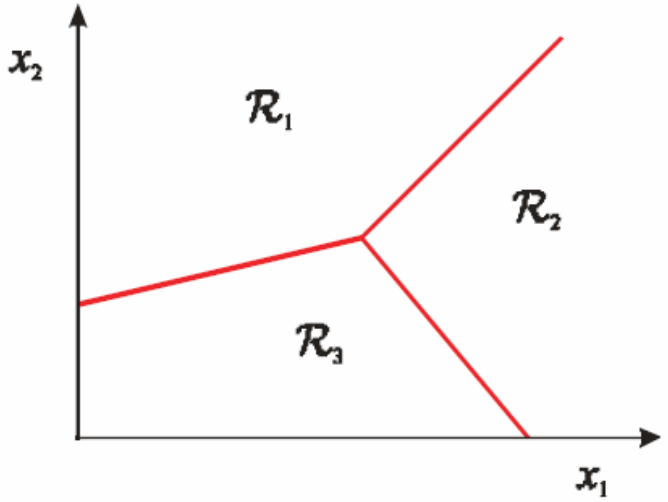
- Let us introduce some loss function $L(y, f(x))$ of the true value of the output y and the predicted value $f(x)$:
  - In a regression retrieval task,
    - a quadratic error:
$$L\big(y, f(x)\big) = \frac{1}{2}\big(y - f(x)\big)^2,$$
    - an absolute error:
$$L\big(y, f(x)\big) = |y - f(x)|.$$
  - In a classification task,
    - a prediction error:
$$L\big(y, f(x)\big) = I\big(y \neq f(x)\big),$$

where $f(x)$ is the predicted class,

$I = \begin{cases} 1, \text{condition met} \\ 0, \text{condition not met} \end{cases}$ – indicator function.

Discriminative Approach

- It is necessary to design a function $y=f(x)$ – a decision rule or a *classifier*.
- Any decision rule divides space into decision regions separated by decision boundaries.



Machine Learning Paradigms

- **Generative approach**
  - Bayes formula $P(y|x) = \frac{P(x|y) \cdot P(y)}{P(x)}$ is used;
  - Each class is modeled separately, evaluate $P(x|y), P(y)$;
  - The problem statement is like the *classification*.
- **Discriminative approach**
  - Since we are interested in $P(y \mid x)$, we will evaluate it;
  - Problem statement is like *regression*.

Risks

- The task of learning is to find a set of classifier parameters f, in which the losses for new data will be minimal.
- Let's introduce the concept of *general (average) risk* – this is the mathematical expectation of losses:

$$R(f) = E\big(L(f(x), y)\big) = \int_{x,y} L(f(x), y) dP.$$

- Unfortunately, due to the unknown probability distribution P of the joint random variable (x,y), the *total risk* cannot be calculated.

Risks

- Let us introduce the concept of *empirical risk*. Let $X=\{x_1,\ldots, x_m\}$, $Y=\{y_1,\ldots, y_m\}$ be the training sample. Empirical risk or *training error*:

$$R_{emp}(f,X)=\frac{1}{m}\sum_{i=1}^{m}L\big(y_i,f(x_i)\big).$$

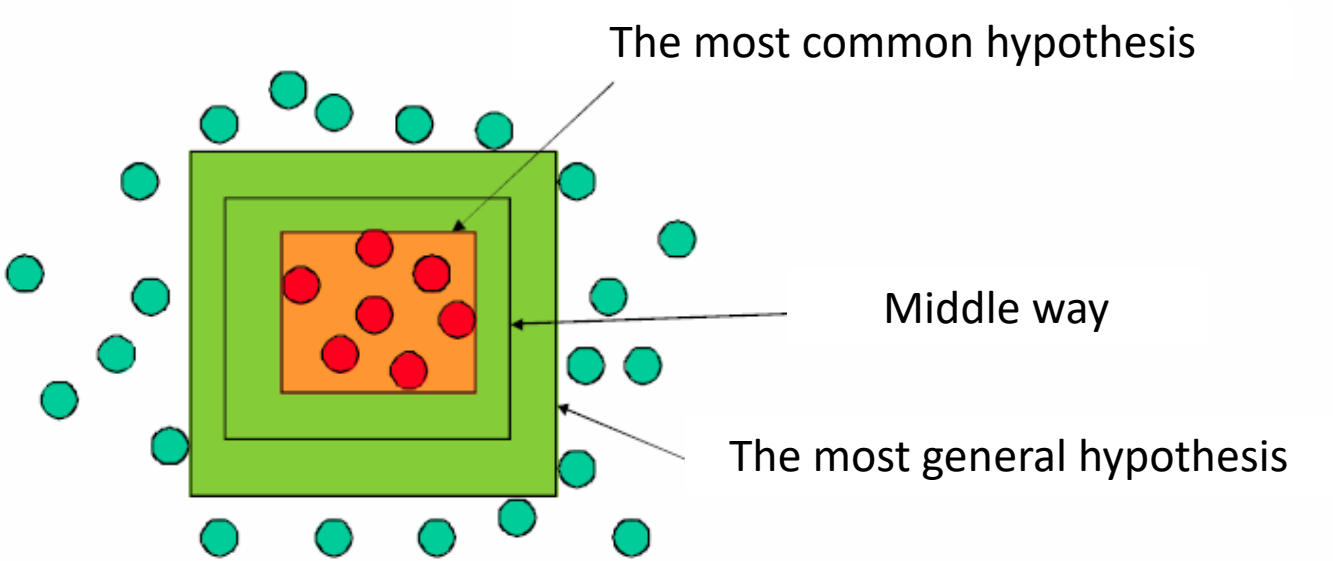
- To minimize the empirical risk, it is necessary to find the function f in accordance with the condition:

$$f=\arg\min_{f\in F}R_{emp}(f,X).$$

- The condition is called: the *empirical risk minimization principle.*

Comment

- There can be an unlimited number of hypotheses that have zero empirical risk:



Supervised Learning Challenge

- The problem was reduced to finding a function f from an admissible set F that satisfies the condition:

$$f = \arg\min_{f \in F} R_{emp}(f, X),$$

F and L are fixed and known.

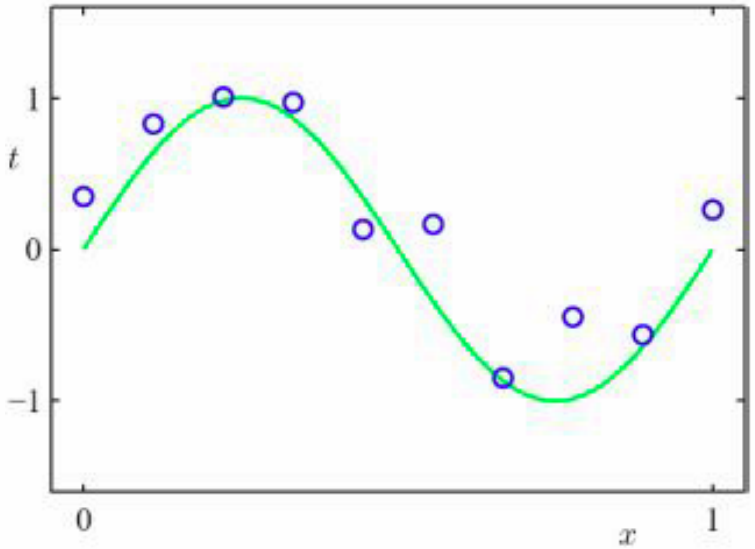
- The class of models F is parametrized, i.e. there is a description of the species $F = \{f(x) = f(x, \theta) : \theta \in \Theta\}$, where Θ is some known set.
- Model tuning process:
  - the learning algorithm selects the values of the set of parameters Θ that ensure the fulfillment of the condition f, i.e. minimizing the error on the precedents of the training sample.

Overfitting

- The considered condition is not suitable for evaluation the generalizing ability of the algorithm.
- All available data is divided into training and test sets:
  - Training is performed using a training set,
  - Evaluation of the prediction quality based on test sample data.
- Values $R(f)$ and $R_{emp}(f, X)$ can differ significantly.
- The phenomenon when $R_{emp}(f, X)$ is small and $R(f)$ is too large is called *overfitting*.

Overfitting

- Let there be a regression problem.
- $t = \sin(2\pi x) + \epsilon$, where ϵ – normally distributed noise, but we don't know that.
- Let there be a training sample and it is required to restore the dependence:



Overfitting

- We will choose the target dependence among polynomials of order M (parametrized set):

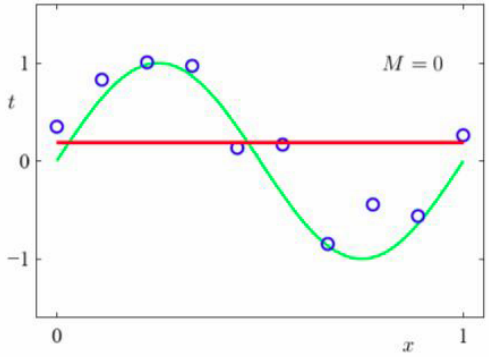
$$y(x, w) = w_0 + w_1 x + w_2 x^2 + \cdots + w_M x^M = w^{\mathrm{T}} \phi_M(x).$$

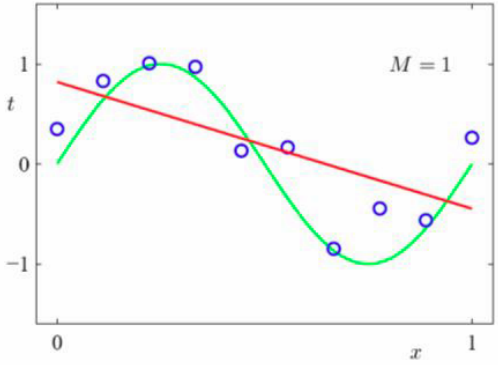
- We introduce the loss function:

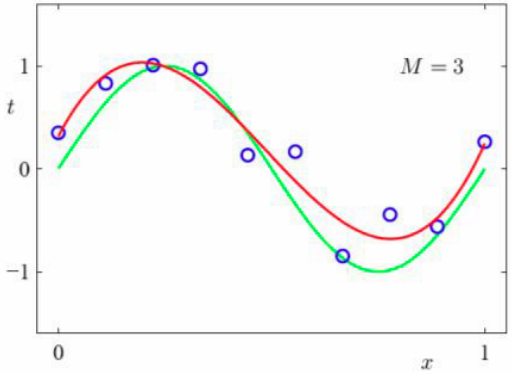
$$L\big((x, t), y\big) = \frac{1}{2}(y(x, w) - t)^2.$$

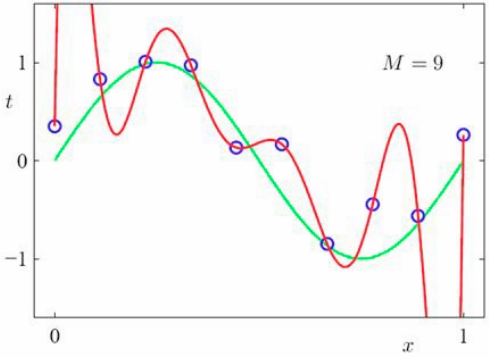
- Among the set of polynomials, we will choose the one that brings the least total loss on the training set.

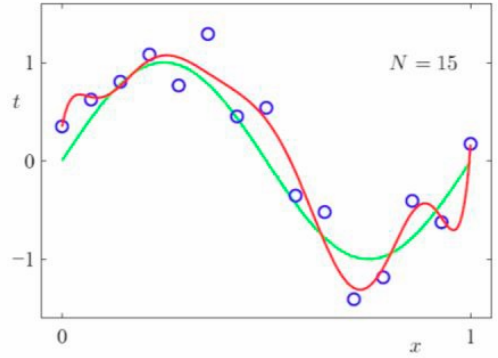
Overfitting

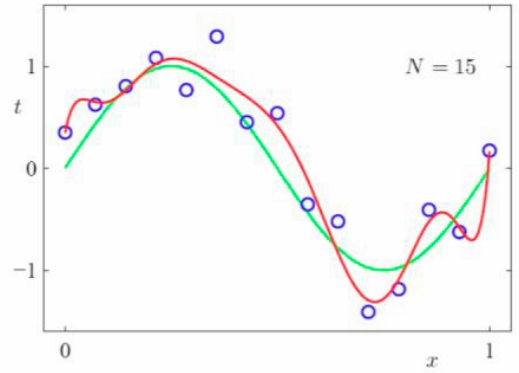












Overfitting

- Reason: the hypothesis well describes the properties of not objects in general, but only objects from the training sample:
  - Too many degrees of freedom of the algorithm model parameters (a complex model);
  - Noisy data;
  - Bad training set.

THANK YOU FOR YOUR TIME!

ITMO more than a UNIVERSITY

s.shavetov@itmo.ru