Lecture 1: Introduction

Sergey Muravyov

25 January 2024

Outline

Artificial intelligence

How it works

Data representation

Basic tasks of machine learning

Other machine learning tasks

Outline

Artificial intelligence

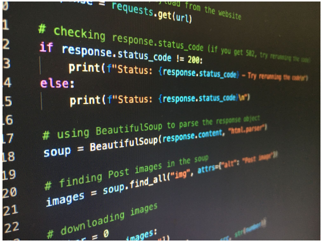
How it works

Data representation

Basic tasks of machine learning

Other machine learning tasks

Challenges in defining artificial intelligence (AI)



Empirical Definitions of Artificial Intelligence

Turing test [modern version]

The judge is corresponding with two respondents, one of which is a person, and the other is a computer program. Based on the answers to the questions, he must determine which of the respondents is who. The task of the computer program — is to mislead the judge, forcing him to make the wrong choice.

Turing test (Alan Turing)

The judge is corresponding with two respondents, one of whom is a human, **who is impersonating someone else**, and the other is a computer program, **who is also trying to impersonate the same person**. Based on the answers to the questions, he must determine which of the respondents is who. The task of the computer program — is to mislead the judge, forcing him to make the wrong choice.

Chinese Room (John Searle)

- There is a man in the room who communicates with the outside world in Chinese characters. He has a book,
  in which the rules are indicated, how to compose the output using the input combination of hieroglyphs. Is it possible to declare that a person knows Chinese?
- In this analogy, a book is an artificial intelligence,
  and a person is an interface.

Strong and weak AI

Artificial Narrow Intelligence, ANI

An artificial intelligence that implements a limited part of the mind or is focused on one narrow task.

Artificial General Intelligence, AGI

An artificial intelligence capable of reaching or surpassing human cognitive abilities.

The exclusive target problem

As AI tasks are solved, they are written out of the tasks of general AI and referred to as narrow AI.

Intellectual systems

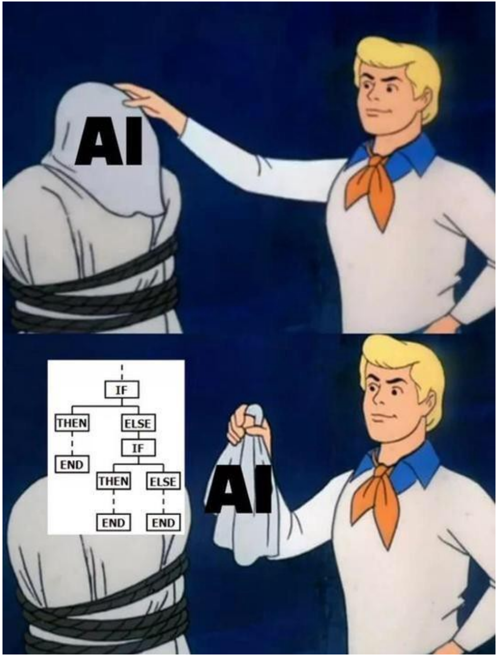
- An intelligent system — a system that solves one or more artificial intelligence tasks.
- Expert system — an intelligent system built on the basis of facts and rules extracted from experts.

Difference from machine learning

- Expert systems: from the general to the particular.
- ML System: from the general to the particular.

Example, translation from one language to another:

- An expert system requires the involvement of an expert (linguist) and the formalization of translation rules.
- Machine learning requires a dataset with many texts.


AI
IF
THEN
ELSE
END
IF
THEN
ELSE
END
END
AI

Outline

Artificial intelligence

How it works

Data representation

Basic tasks of machine learning

Other machine learning tasks

Definitions

Prerequisites:

- Machine learning has evolved from various domains (engineering, accounting).
- Machine learning is popular.

Consequence:

- A lot of «scientific pop» and other nonsense, including on Wikipedia.
- The same object can be called by different terms.
- The same term can mean different objects.
- Some terms contradict generally accepted analogues from other sciences.

The difference between machine learning and data analysis

- **Machine learning** is the development of machine learning algorithms.
- **Data analysis** is the application of various algorithms to data, including machine learning algorithms.

«Empty» term example

Big Data

- Data does not fit in RAM: algorithms in external memory.
- Data is stored or processed on different computers: distributed computing.
- There is enough data to extract a pattern from it: machine learning can be applied.
- There is too much data to analyze it «by hand»: forced to use machine learning.
- The data was not originally collected for analysis: raw data.
- ...

Definition of machine learning

Machine learning is the process of giving computers the ability to learn new things without being directly programmed to do so.

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

The program **is trained** with experience E to solve some problem T according to the quality metric P if the quality of its solution T, measured according to P, grows along with the growth of experience E.

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

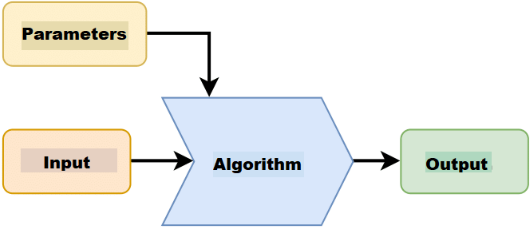
Machine learning (ML)

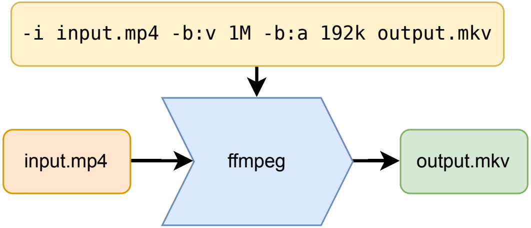
- Data Science: machine learning problems given a dataset $\mathcal{D}$ и
  - quality function (gain, likelihood) $\mathcal{Q}$, or
  - error function (risk, loss) $\mathcal{L}$.
- An algorithm «learns» to solve a problem if it maximizes $\mathcal{Q}_{\mathcal{D}}$ or minimizes $\mathcal{L}_{\mathcal{D}}$ — **empirical risk**.

Difference from an optimization problem

- Optimization: $\mathcal{L}(\theta) \underset{\theta}{\to} \min.$
- Machine Learning: $\mathcal{L}_{\mathcal{D}}(\theta) = \sum_{x \in \mathcal{D}} \mathcal{L}(x, \theta) \underset{\theta}{\to} \min.$

Algorithm parameters





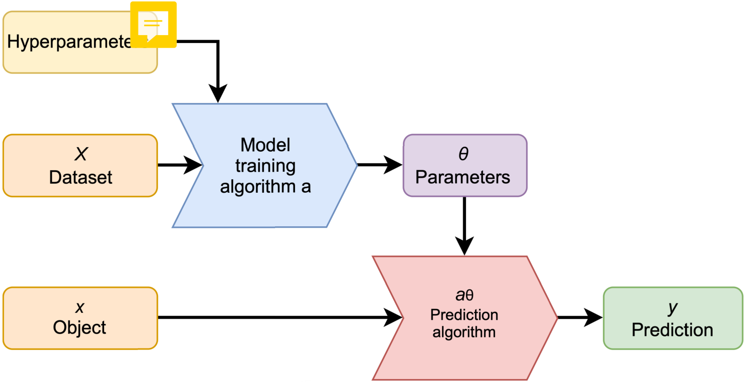
Separate the parameters of the algorithm and its input.

Example: console command arguments — parameters.

Another example: a class that implements the Function interface will have a constructor with «parameters» parameters.

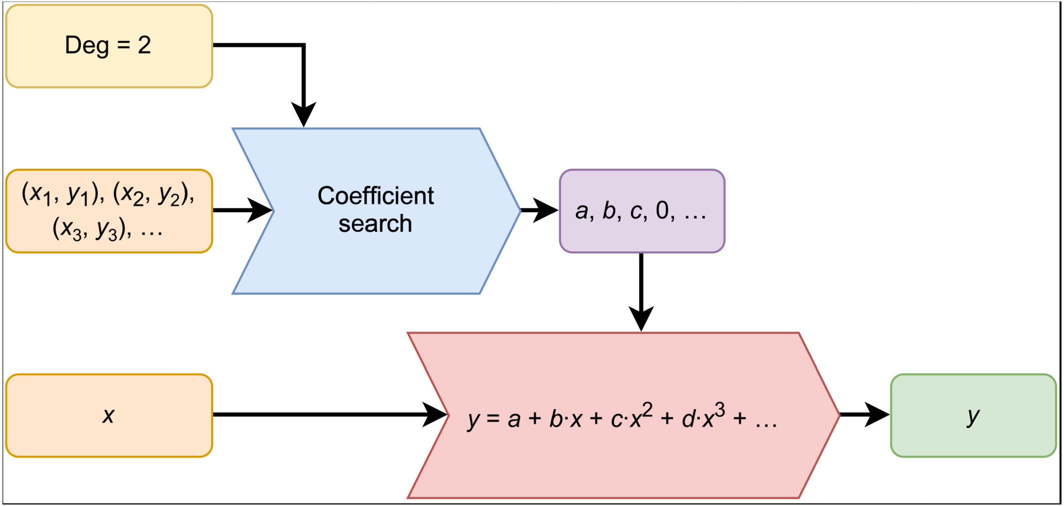
Building a model

Construction (construction, training, approximation, train, build, fit) model (algorithm, function).



Outcome: model parameters.

Example for the polynomial fitting problem



Algorithm Comparison

Practice is the criterion of truth

- In machine learning, **cannot** be said/proved that algorithm A_1 is better than A_2. Exception: if A_2 is a special case of A_1 and the corresponding parameters are achievable during training.
- Instead, it can be only said that A_1 is better than A_2 on $\mathfrak{D}$ with respect to the quality or error function $\mathcal{L}(A, \mathfrak{D})$ and methods for its calculation.

Baseline

The base (existing) algorithm against which the current one is being compared. Sometimes a naive solution is used as a baseline.

Choise of an algorithm

No Free Lucnch

- If the algorithm works well on a certain set of data sets, then this will necessarily affect performance on the set of all remaining data sets.
- Formally, this is called No-Free-Lunch Theorem[1,2]
- For each data set, it is required to choose the best algorithm for it.

State-of-the-Art (SOTA)

The best algorithm for a specific task with a specific data set and validation technique.

[1] Wolpert D. H. The supervised learning no-free-lunch theorems // Soft computing and industry. 2002. P. 25–42.

[2] Wolpert D. H., Macready W. G. No free lunch theorems for optimization //IEEE transactions on evolutionary computation. 1997. Vol. 1. No. 1. P. 67–82.

Outline

Artificial intelligence

How it works

Data representation

Basic tasks of machine learning

Other machine learning tasks

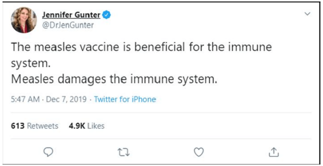
Types of data

	age (n)	job (c)	marital (c)	education (c)	balance (n)	housing (c)
0	30	unemployed	married	primary	1787	no
1	33	services	married	secondary	4789	yes
2	35	management	single	tertiary	1350	yes
3	30	management	married	tertiary	1476	yes
4	59	blue-collar	married	secondary	0	yes
5	35	management	single	tertiary	747	no

Structured (.csv, .excel, .parquet)


```
{
  "successCode":"1",
  "message":"Thank you for authenticating. Please wait...",
  "listOfPatients":{
    "Helen":{
      "Age":"19",
      "Phone":"777-777777",
      "smoker":"No"
    },
    "John":{
      "Age":"24",
      "Phone":"777-777778",
      "smoker":"Yes"
    },
    "Sarah":{
      "Age":"51",
      "Phone":"777-777779",
      "smoker":"Yes"
    }
  }
}
```


Semi-structured (.json, .yaml, .xml)



Non-structured (text, images, audios)

Tabular data representation

Dataset — table (matrix)
with n rows and m columns.

$$\begin{bmatrix} f_1 & f_2 & \dots & f_m \\ \hline x_{1,1} & x_{1,2} & \dots & x_{1,m} \\ x_{2,1} & x_{2,2} & \dots & x_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \dots & x_{n,m} \end{bmatrix}$$

Row
Object, instance, sample, example.

Column
Feature, attribute, characteristic,factor.

$$\begin{bmatrix} f_1(x_1) & f_2(x_1) & \dots & f_m(x_1) \\ f_1(x_2) & f_2(x_2) & \dots & f_m(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(x_n) & f_2(x_n) & \dots & f_m(x_n) \end{bmatrix}$$

Basic feature types

	Category	Number
Alternative naming	«Quality»	«Quantity»
Space	Discrete	Continuous
Number of elements	Finite	Infinite
Iteration over all values	Yes	No
Valid Operations	$=$	$<, +, -, \times, \sqrt{\cdot}, \ldots$
Examples	Gender, color, type, brand	Age, speed, price

Category

Something in between Enum and Object from Java:

- A finite number of elements, like Enum. All values are known in advance.
- There is no order relation above elements like Enum.
- Can be tested for equality.
- It is customary to associate each value with an integer or natural number (similar to *ordinal*), but these numbers are used by **only for convenience** of storage and operations on categories (similar to *hashCode*).
- Numbers are usually used: $[0; \ldots ; k - 1]$ or $[1; \ldots ; k]$.

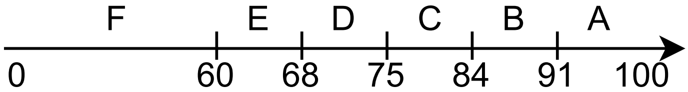
The algorithm must be statistically invariant under different comparisons of numbers and categories.

Ordinal type

- Somewhere between a category and a number: discrete, but there is an order above the elements, and the number of elements can change.
- Called «category» in the outside world.
- Not popular.

Sampling

- Transformation to an ordinal attribute. Example:



- Transformation into a categorical feature.
  Order information is lost.
- Rarely used, as it is more convenient to work with numbers.

Ordinal Type Conversion

- Can be converted to a number via ordinal.
- If the number of values is finite and equal to k, then it can be converted to k binary categories: $c_i(ord) := (ord < \text{ord}_i)$, where $\{\text{ord}_1, \dots, \text{ord}_k\}$ – set of ordinal feature values. Example, let $A < B < C$:

$$\begin{bmatrix} \text{ord} \\ \hline A \\ B \\ C \end{bmatrix} \Rightarrow \begin{bmatrix} < A & < B & < C \\ \hline false & true & true \\ false & false & true \\ false & false & false \end{bmatrix}$$

- If the category **binary** (it has only two values c_1, c_2), can be converted to a number: $c_1 \Rightarrow 0$, $c_2 \Rightarrow 1$ or $c_1 \Rightarrow -1$, $c_2 \Rightarrow +1$.
- A category of k values $\{\mathsf{c}_1, \ldots, \mathsf{c}_k\}$ can be **binarized** by getting k binary categories: $b_i(c) := (c = \mathsf{c}_i)$. Example:

$$\begin{bmatrix} \mathsf{c} \\ \hline A \\ B \\ C \end{bmatrix} \Rightarrow \begin{bmatrix} =A & =B & =C \\ \hline true & false & false \\ false & true & false \\ false & false & true \end{bmatrix}$$

- **One-hot encoding** (~~unitary code~~) — another name for binarization, or a conversion option when going straight to numbers (0 and 1):

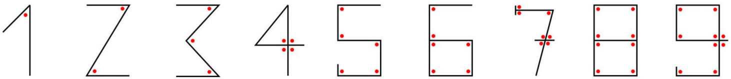
$$\text{one-hot}_i(c) = [c = \mathsf{c}_i]$$

Why is a number not a number?

Consider the problem of determining a digit from an image:

- Naive idea: match numbers «0», «1», . . . «9» with numbers 0, 1 . . . 9, and work with one numeric feature.
- Image «3» — is not something between images «2» and «4» or «1» and «5».
- Image «5» looks more like «6» than image «7», but in terms of this mapping they are equally similar.
- . . .

Or is it a number?



Example of converting other types

Time

1. Find out with what periods T_e the events $e_1, \ldots e_m$ occur, which can affect the studied dependence. For example, to analyze traffic congestion or energy consumption, periods
   of 1 and 7 days can be useful.
2. Add $2m$ new numeric features: $f_{2e-1} = \sin\left(\frac{2\pi t}{T_e}\right)$ and $f_{2e} = \cos\left(\frac{2\pi t}{T_e}\right)$.

Color

- Use RGB model.
- Why is it bad to use HSB (HSV)?

Selecting and converting to the correct type is the most important part of data analysis.

Formats

Comma Separated Values (CSV)

- Most popular format.
- Poorly standardized (even the column separator).
- Designed to store tables, not datasets.

Tab-separated values (TSV)

- Like CSV, but «tabs» are used as delimiter (\t).

Attribute-Relation File Format (ARFF)

- The heading is formalized, which stores the name, the text description and **formal** description of feature types. Possible types: number (Numeric), category (Nominal), string (String), date (Date).
- The body of the file is similar to CSV, but more standardized.

Example of ARFF file

```
% 1. Title: Iris Plants Database
%
% 2. Sources:
%       (a) Creator: R.A. Fisher
%       (b) Donor: Michael Marshall (MARSHALL%PLU@io.arc.nasa.gov)
%       (c) Date: July, 1988
%
@RELATION iris

@ATTRIBUTE sepallength  NUMERIC
@ATTRIBUTE sepalwidth   NUMERIC
@ATTRIBUTE petallength  NUMERIC
@ATTRIBUTE petalwidth   NUMERIC
@ATTRIBUTE class        {Iris-setosa,Iris-versicolor,Iris-virginica}

@DATA
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
4.6,3.1,1.5,0.2,Iris-setosa
5.0,3.6,1.4,0.2,Iris-setosa
5.4,3.9,1.7,0.4,Iris-setosa
4.6,3.4,1.4,0.3,Iris-setosa
5.0,3.4,1.5,0.2,Iris-setosa
4.4,2.9,1.4,0.2,Iris-setosa
4.9,3.1,1.5,0.1,Iris-setosa
```

Other object types

Images

- 2D or 3D matrix.
- Basic conversion to vector: reversal by rows.

Text

- Sequence of words of variable length.
- Basic transformation to a vector: for each word, create a feature: TF-IDF, whether the word was encountered or not, how many times the word was encountered.
- After transformations, you need to use a sparse dataset (eg Sparse ARFF).

Multimodal objects (data)

- Objects consisting of different types («modalities»).

Dataset normalization

Motivation

- Features with more variance may have more effect on the result.
- Numerical attributes have units of measurement: [kg], [m], [s], etc.
  For convenience of storage, units of measurement are discarded.
- Eliminating units is a necessary but not sufficient step.
- Since the change in the units of change does not change the hidden dependence in the data, the algorithm must be statistically invariant to linear transformations over features.

- Applies independently to column X.
- Do not use the sklearn.preprocessing.normalize method
- Normalization is the part of the tutorial!

Minimax, [0; 1] scaling

$$x_{\text{new}} = \frac{x_{\text{old}} - \min\left[X\right]}{\max\left[X\right] - \min\left[X\right]}$$

After normalization: $\min\left[X_{\text{new}}\right] = 0$ и $\max\left[X_{\text{new}}\right] = 1$.

Standartization, Z-scaling

$$x_{\text{new}} = \frac{x_{\text{old}} - \mathbb{E}\left[X\right]}{\sqrt{\mathbb{D}\left[X\right]}}$$

After normalization: $\mathbb{E}\left[X_{\text{new}}\right] = 0$ и $\mathbb{D}\left[X_{\text{new}}\right] = 1$.

Weights of objects and features

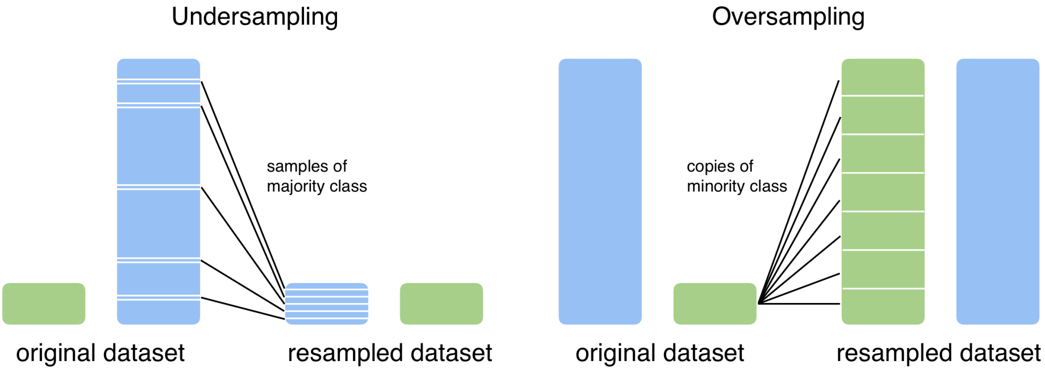
Strict definition

- The algorithm takes into account an object / feature with a weight of w if it affects the result w times more.
- If the weight of an object / feature is n, then this is equivalent to the fact that it occurs (repeated) n times in the data set.
- It is difficult to formally follow this definition.

Informal definition

- The greater the weight of the object / feature, the more it affects the result.

Sampling balancing example



Outline

Systematization of the main tasks of machine learning

$a: X \to Y$	$Y = \{y_1, \ldots, y_k\}$	$Y = \mathrm{Pr}^k$	$Y = \mathbb{R}^k$
Supervised learning	Classification	Soft Classification	Regression
One-class classification	Anomaly detection	Density recovery	*Object generation*
Unsupervised learning	Clustering	Fuzzy Clustering	Feature extraction

Supervised learning

Supervised learning, Learning from labeled data (examples, use cases), approximation, Supervised learning

- A machine learning problem in which the training set contains the correct answers that the algorithm must learn to predict for new data.
- Labeled object — object that has the value of the target feature.

Variants of supervised learning problem depending on Y:

- Classification problem: $Y = \{c_1, c_2, \ldots, c_k\}$.
- Probabilistic classification problem: $Y = \text{Pr}^k$.
- Regression recovery problem: $Y = \mathbb{R}^k$.

Classification

Classification problem

- A supervised learning problem where the type of target feature Y is **category**. This attribute is called: class, class label, label, type.
- Naive solution $a(x) = \text{Mode}[Y]$ (most common value).

Language nuance

- The number of classes — the number of elements in the set of values of the categorical target feature, not the number of target features. Multi-class not multi-purpose (Multi-label, Multi-task).

Examples

- Text classification: determine if an email is spam or not.
- Image classification: determine which number is shown in the photo.

Soft and probabilistic classification

Soft classification problem

- For object x, the algorithm predicts an array of numbers $(p_1, \dots p_k)$, where p_c — confidence that x belongs to class c and k — number of classes.
- The algorithm is trained to solve a common classification problem.

Probabilistic classification problem

- The $(p_1, \dots p_k)$ array is required to be a valid probability vector: $(\forall c : 0 \leq p_c \leq 1)$ and $\sum_c p_c = 1$.
- You can use a probability vector as a target feature, for example onehot$(y(x))$. Then the error function — comparison of two probability vectors, for example Cross entropy.

Reduction to binary classification

Let a^b be a binary **probabilistic** classification algorithm. Let's make it an algorithm for a multiclass classification into k classes.

«One vs all» approach: train k classifiers $a^b c$.

1. For each class c the classifier $a^b_c(x)$ predicts $\Pr(y(x) = c)$, where $y(x)$ is the real class of object x.
2. Get the algorithm: $a(x) = \underset{c}{\operatorname{argmax}}\, a^b_c(x)$.

Reduction to binary classification

«One vs one» approach: train $k \cdot (k-1)/2$ classifiers $a^b_{u,v}$.

1. For each pair of classes u, v we choose a subset of objects:

$$X_{u,v} = \{x_i \mid (y(x_i) = u) \vee (y(x_i) = v)\}$$

2. Let's train the algorithm $a^b_{u,v}(x)$ on $X_{u,v}$ to predict $Pr(y(x) = u)$.
3. We get the algorithm:

$$a(x) = \operatorname*{argmax}_c \prod_v a^b_{c,v}(x) \cdot \prod_u \left(1 - a^b_{u,c}\right)(x)$$

These approaches also work for soft classification.

Regression problem

- Regression recovery problem, Regression, Regression analysis — за supervised learning problem where the type of target feature Y is **number**.
- Naive solution $a(x) = \mathbb{E}[Y]$.

Language nuance

- The word regression — is a synonym for the word return.
- Sometimes ordinary regression is understood as one-dimensional regression, when the dependence is built on one attribute.
- Multivariate regression: dependence is built on several features.

Example

- Predict the performance (grade) of a student.

Time series forecasting

- Given a set of values $y_1, y_2, \ldots, y_t$.
- It is required to predict y_{t+1}

Where should we take X?

- $x_t = (y_{t-m}, \ldots, y_{t-2}, y_{t-1},)$ — **autoregression**.
- $x = f(t)$ — feature construction.

Naive solution

- Moving average:

$$\hat{y}_t = (y_{t-m+1} + \cdots + y_{t-1} + y_t)\,/m$$

- Exponentially weighted moving average:

$$\hat{y}_t = \alpha \cdot y_t + (1 - \alpha) \cdot \hat{y}_{t-1}$$

One class classification problem

One-class classification,
Positive labeled data classification

- Almost all objects in the training set belong to the same class.
- Even if there are objects of another class, it is not known which objects.

Problems:

- Anomaly detection: find objects of another class among **existing ones**.
- Search for novelty: find objects of a different class among **new ones**.

Example: determining the authenticity of a signature from a photograph.
Error function: expert judgment or any for the classification problem, but the test set must be labeled.

Noise elimination

Anomalies, noise, errors, outliers

- Anomalies — bad objects for building a model.
- Mistakes — bad objects in terms of reality.

Example

Consider a dataset with information about cars. One of them has a suspiciously high fuel consumption: 30l/100km.

- If it's a truck and the rest are regular vehicles (sedans, SUVs), then it's a **anomaly**.
- If you meant miles per gallon, then this is a **error**.

Solution Approaches

- Reduction to one-class classification (density recovery).
- Anomaly — the object on which the error of the prediction algorithm is higher.

Object generation

The task of generating (synthesis) new objects

Based on the given set of objects, generate new ones.

- Do not confuse with **sampling**, when objects are selected from existing ones.
- Do not confuse with **augmentation**. Most often, augmentation is understood as an analytical solution to the problem of generating new objects.

These tasks can be used as a naive solution / baseline.

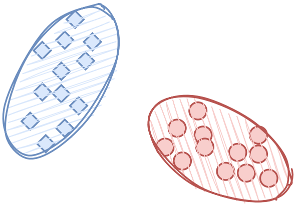
Quality evaluation

- Reduction to classification into two classes: real or generated.
- Expert evaluation by people (assessors).

Generative and discriminative models

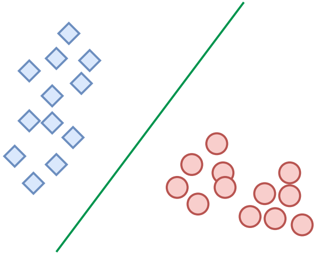
Generative models

This is a class of models that train the joint distribution of $p(x,y)$ data. They reduce the classification problem to the density recovery problem.



Discriminative models

This is a class of models that only train the conditional distribution $p(y|x)$. Trying to find a separating rule.



Unsupervised learning

Unsupervised learning

A machine learning problem in which the training set does not contain target features. The algorithm itself needs to come up with new features $\hat{Y}$ based on the existing X.

Types of the unsupervised learning problem depending on $\hat{Y}$:

- (Hard-)Clustering: $\hat{Y} = \{c_1, c_2, \ldots, c_k\}$.
- Fulzzy-clustering: $\hat{Y} = \mathrm{Pr}^k$.
- Feature extraction task: $\hat{Y} = \mathbb{R}^k$.

Clustering

Cluster analysis, Clustering, Clusterization

An unsupervised learning problem in which the algorithm needs to extract (invent) a new categorical feature.

Language nuance

In the outside world, clustering is called «classification».

Examples:

- You want to split your music collection by genre, but you're too lazy to come up with a genre for each track.
- You want to split the linked graph into possible communities (social networks).

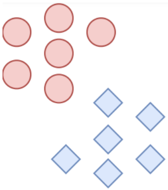
Evaluation of the clustering problem

Internal measures

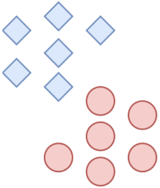
- Uses X and $\hat{Y}$.
- Examples: intra-cluster or inter-cluster distance.

External measures

- Used to evaluate $\hat{Y}$ and Y, which is taken from the classification data set.
- You cannot use measures for a classification task.
  - $\hat{y}_i$ and y_i are taken from different spaces, they cannot be checked for equality.
  - The number of clusters may not match the number of classes.
- An analogue of validation, but instead of rows in the training set, a column is removed.



Real class labels



Clustering

The problem of feature extraction

Feature extraction, Embedding, Dimensionality reduction

- The algorithm must learn to map an object from X to the space of numerical features $\hat{Y}$, which it will come up with.
- **Naive solution**: multiplication by a random matrix.
- **Example**: Dataset visualization.

Feature engineering

Sometimes a this task is separated from the feature extraction task.

Two different definitions:

- More general problem, when $x \in X \neq \mathbb{R}^m$ is not a feature vector, but an abstract object: picture, text, etc.
- The problem is solved explicitly, not by Machine Learning methods.

Example 1. Vectorizing an Image with Convolutions

- Let the j-th feature x_j – be the sum of similarity of θ_j template to the image part, which is taken over all possible overlays of the template on the image.
- θ_j patterns are searched using Machine Learning.

Example 2. Generation of features by a polynomial of the second degree

Addition to existing features $x_1, x_2, x_3, \dots$ of all possible pairwise products:
$x_1 \cdot x_1, x_1 \cdot x_2, x_1 \cdot x_3, \dots, x_2 \cdot x_2, x_2 \cdot x_3, \dots$.

Systematization of the main tasks of machine learning

$a : X \to Y$	$Y = \{y_1, \ldots, y_k\}$	$Y = \text{Pr}^k$	$Y = \mathbb{R}^k$
Supervised learning	Classification	Soft Classification	Regression
One-class classification	Anomaly detection	Density recovery	*Object generation*
Unsupervised learning	Clustering	Fuzzy Clustering	Feature extraction

Outline

Artificial intelligence

How it works

Data representation

Basic tasks of machine learning

Other machine learning tasks

Missing values in the dataset

Where they come from:

From a sparse dataset or when you combine data from different sources.

How they are encoded:

- CSV : «?», « », «_», empty string
- ARFF : «?»
- String / object: Null, None, empty string
- Category (от 0 to $k-1$): -1 or k
- Number: NaN

Basic soultion:

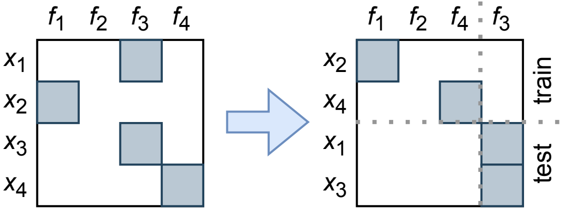
Delete, replace, add something new.

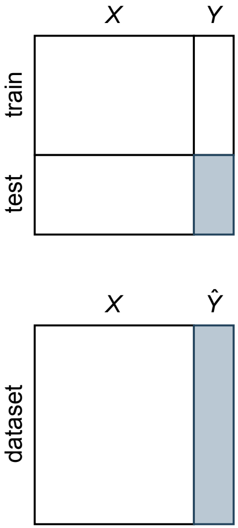
Algorithm refusals

Some algorithms can not only accept missing values as input, but also return them. This can be interpreted as a refusal to work with the object in question.

- Reject classification: used in ensembles.
- Reject clustering: used to find anomalies.

The task of predicting and filling in the gaps



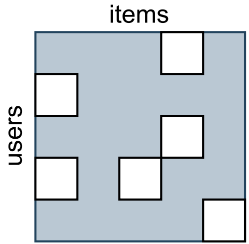


- The gap filling problem can be reduced to a prediction problem.
- Other machine learning problems can be thought of as gap filling problems. Example is on the right side.

Recommending systems

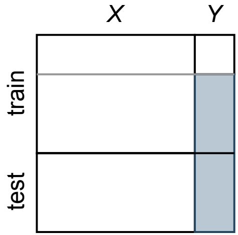
Collaborative filtering

- Given a set of evaluations of things (items) by users (users).
- The number of ratings is much less than the product of the number of users and things.
- It is required to predict the value of an arbitrary item by an arbitrary user.



Решения:

- The solution of the gap filling problem is difficult to apply to collaborative filtering.
- You can apply it in the opposite direction.



Semi-supervised learning

A supervised learning problem in which only a small part of the training data contains the target feature.

Basic solution:

- Do not use objects that have a missing target feature.
- Do not use the target feature for learning (unsupervised learning problem). Labeled objects can be used for testing.

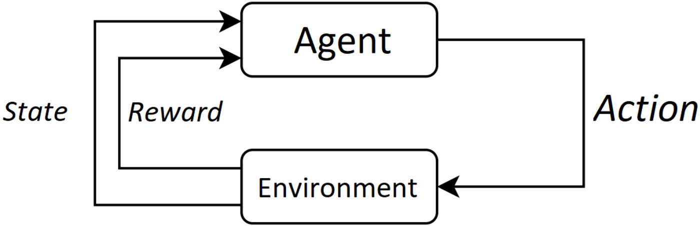
Labeled objects can be statistically different from unlabeled ones.

Active learning

- There is access to a large number of objects, but not all of them have labels.
- Data is collected quickly, but labeled up slowly and in portions, the speed of model learning is faster than labeling.
- In active learning, the conditions are the same as in partial learning, but you can ask the Oracle questions about the meaning of labels.
- It is required to restore $f : X \to Y$ in the least number of Oracle calls (find an Oracle call strategy that optimizes the quality of f approximation).

Reinforcement Learning, RL

- The agent interacts with the environment by telling it some action for the current state.
- The environment tells the agent the reward for the action and the new state.
- The task of the agent is to maximize the total reward.
- This task is more like learning in the real world.



Thank you for your attention!

ITsMOre than a UNIVERSITY