Introduction to Machine Learning

Alexander Sirotkin

HSE University, Saint-Petersburg
January 19, 2023

History

- Neural networks appeared even before AI and ML as a science.
- AI – Turing test (1950), Dartmouth seminar (1956). Then:

1950-60s big hopes and logical inference;
1970s knowledge-based systems based on rule combinations;
1980s second bubble of the neural networks;
1990-2000s machine learning, Bayesian methods, probabilistic learning;
2010s deep learning.

Main definitions and problems

- Supervised learning:
  - training set (training sample), where each example consists of *features* (attributes);
  - correct answers – response variable, which we are predicting;
  - categorical, continuous, or ordinal;

Main definitions and problems

- Supervised learning:
  - training set (training sample), where each example consists of *features* (attributes);
  - correct answers – response variable, which we are predicting;
  - categorical, continuous, or ordinal;
  - a model *trains* on this set (training phase, learning phase), then can be applied to new examples (test set);
  - the goal is to train a model that not only explains examples from the training set but also *generalizes* well to the test set;
  - one important problem – overfitting;

Main definitions and problems

- Supervised learning:
    - usually we simply have the training set; how do we know how well a model generalizes?
    - cross-validation: break the sample up into training and validation sets;
    - before feeding data into a model, it makes sense to do *preprocessing*:
        - feature extraction,
        - normalization/whitening,
        - encoding categorical features,
        - ...

Main definitions and problems

- Supervised learning:
  - *classification*: a certain discrete set of categories (classes), and we have to classify new examples into one of these classes;
    - text classification by topics (e.g., spam filter);
    - image/object/character recognition;
    - ...

Main definitions and problems

- Supervised learning:
  - *classification*: a certain discrete set of categories (classes), and we have to classify new examples into one of these classes;
    - text classification by topics (e.g., spam filter);
    - image/object/character recognition;
    - ...
  - *regression*: predicting the values of an unknown continuous function:
    - engineering applications (predict physical values, e.g., temperature, position etc.);
    - finances (predicting prices or effects);
    - ...
  - the same plus a time dimension: time series analysis, speech recognition etc.

Main definitions and problems

- Unsupervised learning – no correct answers, only data:
  - *clustering* – divide data into subsets so that the points are similar inside a cluster but dissimilar between them:
    - extract families of genes from a sequence of nucleotides;
    - cluster users and personalize an app for them;
    - cluster a mass-spectrometry image into subregions with similar composition;
  - *feature extraction* – when unsupervised learning is an auxiliary, instrumental goal for some subsequent supervised problems;
  - most generally, *density estimation*.

Main definitions and problems

- Unsupervised learning – no correct answers, only data:
  - *clustering* – divide data into subsets so that the points are similar inside a cluster but dissimilar between them:
    - extract families of genes from a sequence of nucleotides;
    - cluster users and personalize an app for them;
    - cluster a mass-spectrometry image into subregions with similar composition;
  - *feature extraction* – when unsupervised learning is an auxiliary, instrumental goal for some subsequent supervised problems;
  - most generally, *density estimation*.
- Other variations:
  - Dimensionality reduction: represent a high-dimensional sample in lower dimensions while preserving important properties;
  - Matrix completion: given a matrix with lots of unknown elements, predict them.
  - Often we know the correct answers for a small part of available data: *semi-supervised learning*.

Main definitions and problems

- *Reinforcement learning* – when an agent trains by trial and error:
  - *multiarmed bandits*: maximize expected revenue from an action;
  - *exploration vs. exploitation*: how and when to pass from exploring new possibilities to simply choosing the current best;
  - *credit assignment*: we get a response at the end but are now sure what exactly went right or wrong along the way.

Main definitions and problems

- *Active learning*: how do we choose the next (costly) test?
- *Learning to rank*: how do we generate an ordered list (e.g., Web search)?
- *Model combination*: how do we combine several models to get one better than any single component?
- *Model selection*: how do we choose between simpler and more complicated models?

Probability in ML

- In all methods and approaches of machine learning, the central notion is *uncertainty*.
- We don't know the answers, and the answers in the training set do not perfectly match our models.
- Moreover, it would be great to know how certain we are.
- Therefore, *probability theory* is crucial for ML.
- To be honest, this is mostly a course in applied probability theory.

Bayes theorem

- *Discrete* and *continuous* random values.
- *Joint probability* – $p(x, y)$ is the probability of both x and y at the same time; marginalization:

$$p(x) = \sum_y p(x, y).$$

- *Conditional probability* – probability of one event if we know that another occurred, $p(x \mid y)$:

$$p(x, y) = p(x \mid y)p(y) = p(y \mid x)p(x).$$

- From this definition, we can immediately see *Bayes theorem*:

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)} = \frac{p(x|y)p(y)}{\sum_{y'} p(x|y')p(y')}.$$

- *Independence*: x and y are independent if

$$p(x, y) = p(x)p(y).$$

Bayes theorem

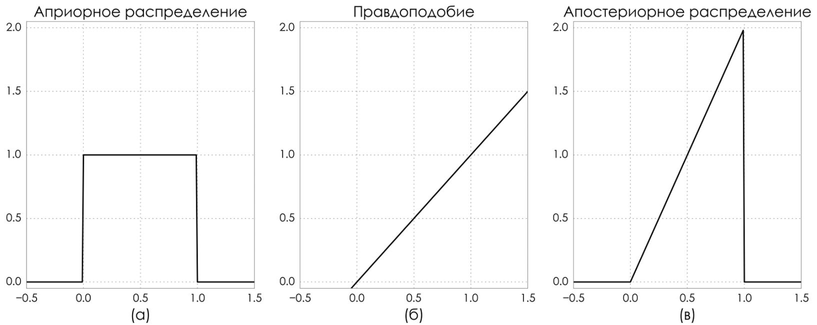
- Bayes theorem – the main formula in machine learning:

$$p(\theta|D) = \frac{p(\theta)p(D|\theta)}{p(D)}.$$

- Here
  - $p(\theta)$ is the *prior probability*,
  - $p(D|\theta)$ is the *likelihood*,
  - $p(\theta|D)$ is the *posterior probability*,
  - $p(D) = \int p(D \mid \theta)p(\theta)\mathrm{d}\theta$ is the *evidence* (probability of the data).

Bayesian inference for a coin

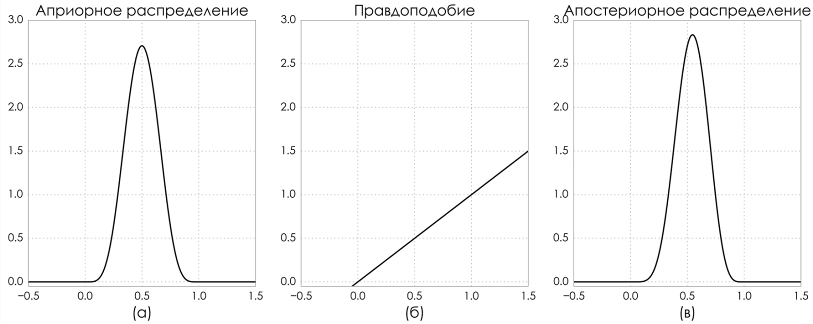
- Example – a completely unknown coin:



- Multiplying $p(\theta) = 1$ on $[0, 1]$ by $p(s \mid \theta) = \theta$, we get $p(\theta \mid s) = 2\theta$ на $[0, 1]$.

Bayesian inference for a coin

- Example – a coin taken from my pocket:



- Multiplying $p(\theta) = \mathrm{Beta}(\theta; 10, 10)$ on $[0, 1]$ by $p(s \mid \theta) = \theta$, we get $p(\theta \mid s) = \mathrm{Beta}(\theta; 10, 11)$ on $[0, 1]$.

Contacts

- email: avsirotkin@hse.ru
- email: alexander.sirotkin@gmail.com
- Telegram: @avsirotkin