INTRODUCTION: AI AND BAYESIAN INFERENCE

Sergey Nikolenko

Harbour Space University, Barcelona, Spain
March 13, 2017

WHAT IS MACHINE LEARNING

FIRST AI IDEAS

- Hephestus made giant android robots, e.g., Talos whom he presented to Minos to defent Cretes.
- Pygmalion made Galatea.
- Jehovah and Allah breathed life into pieces of clay.
- Especially wise rabbis could create golems.
- Albertus Magnus made an artificial talking head (which really discouraged his teacher, Thomas Aquinas).
- Starting from Dr. Frankenstein, AI is constantly appearing in literature...

TURING TEST

- AI as a science began with the *Turing test* (1950).
- A computer must successfully pose as a human in a (written) dialogue between a judge, a human, and a computer.
- The original formulation, by the way, was a bit different...

TURING TEST

- It is already obvious here how much we have to do to make an AI:
  - natural language processing;
  - knowledge representation;
  - inference from the knowledge;
  - learning from experience (machine learning per se).

DARTMOUTH SEMINAR

- The term AI and formulations of main problems appeared in 1956 on a seminar in Dartmouth.
- John McCarthy, Marvin Minsky, Claude Shennon, and Nathaniel Rochester.
- Probably the most ambitious grant proposal in the history of computer science.

DARTMOUTH SEMINAR

We propose that a 2 month, 10 man study of artificial intelligence be carried out during the summer of 1956 at Dartmouth College in Hanover, New Hampshire. The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. An attempt will be made to find how to make machines use language, form abstractions and concepts, solve kinds of problems now reserved for humans, and improve themselves. We think that a significant advance can be made in one or more of these problems if a carefully selected group of scientists work on it together for a summer.

1956-1960: GREAT EXPECTATIONS

- Very optimistic time. AI always seemed just around the corner...
- Allen Newell, Herbert Simon: *Logic Theorist*.
  - A program for logical inference.
  - Could re-prove a large part of *Principia Mathematica*, sometimes more elegant than Russell and Whitehead themselves.

1956-1960: GREAT EXPECTATIONS

- Very optimistic time. AI always seemed just around the corner...
- General Problem Solver – a program that tried to think like a human;
- Many programs that could do limited things (microworlds):
  - Analogy (IQ-tests);
  - Student (algebraic word problems);
  - Blocks World (moved 3D blocks).

1970s: knowledge-based systems

- Collect a large set of rules and knowledge about a problem domain, then perform probabilistic inference.
- First large success: MYCIN – diagnosing blood infections:
  - about 450 rules;
  - diagnosed like an experienced doctor, significantly better than a novice doctor.

1980s: commercial applications; the AI industry

- The first AI department was at DEC (Digital Equipment Corporation).
- Rumor has it that by 1986 it brought DEC $10 mln. per year.
- But the bubble burst by the end of the 1980s, when many companies could not meet high expectations.

1990-2010: Data mining, machine learning

- Lately, we moved on to data mining and machine learning.
- We have larger and larger datasets, especially after the Internet began in earnest.
- But how far are we from strong AI? Nobody really knows.

DEFINITION

- What does it mean for a program to “learn”? How do we define it?

DEFINITION

- What does it mean for a program to “learn”? How do we define it?

Definition

A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E.

- Very general definition. What specific examples can you think of?
- We will speak of the general theory of Bayesian inference where every machine learning algorithm can fit.
- But first — a brief survey of the main machine learning directions 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;

- 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;

- 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,
    - ...

- 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;
    - ...

- 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.

- 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*.

- 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*.

- *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.

- *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.

SOURCES

- Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2007.
- Kevin Murphy, *Machine Learning: A Probabilistic Perspective*, MIT Press, 2013.
- Trevor Hastie, Robert Tibshirani, and Jerome Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed., Springer, 2009.

BAYESIAN INFERENCE

- We won't need the whole formalism of probability measure defined on the sigma-algebra of Borel sets and so on.
- We only need the intuition that people usually get after an introductory probability course:
  - there are *discrete* random values, where nonnegative probabilities of outcomes sum up to one;
  - and *continuous* random values that integrate to one;
  - which probability distributions do you know?

MAIN DEFINITIONS

- *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

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

ON DISEASES AND PROBABILITIES

- We begin with a classical example.
- Suppose that a test for some disease has success probability 95% (i.e., 5% is the probability of both false positives and false negatives).
- In total, 1% of the population have the disease.
- Suppose that someone (taken uniformly from that population) got a positive test result. What is the probability that she is actually sick?

On diseases and probabilities

- We begin with a classical example.
- Suppose that a test for some disease has success probability 95% (i.e., 5% is the probability of both false positives and false negatives).
- In total, 1% of the population have the disease.
- Suppose that someone (taken uniformly from that population) got a positive test result. What is the probability that she is actually sick?
- Answer: 16%.

PROOF

- We denote by t the test result; by d, the presence of a disease.
- $p(t=1) = p(t=1|d=1)p(d=1) + p(t=1|d=0)p(d=0)$.
- Using Bayes theorem:

$$p(d=1|t=1) =$$
$$= \frac{p(t=1|d=1)p(d=1)}{p(t=1|d=1)p(d=1) + p(t=1|d=0)p(d=0)} =$$
$$= \frac{0.95 \times 0.01}{0.95 \times 0.01 + 0.05 \times 0.99} = 0.16.$$

BAYESIAN INFERENCE

- This is the kind of problems usually solved by probabilistic inference.
- Since it is based on Bayes theorem, it is often called *Bayesian inference*.
- But there is another reason.

PROBABILITY AS FREQUENCY

- In classical probability theory, probability is usually understood as the limit ratio of a certain experiment result to the total number of experiments.
- E.g., tossing a coin.

PROBABILITY AS DEGREE OF BELIEF

- We want and often need to talk about how “probable” it is that
  - FC Barcelona will win the current Champions League,
  - “Odyssey” was written by a woman,
  - average temperature on Earth will rise by 5 degrees in 50 years,
  - and so on.
- But there is only one experiment!

PROBABILITY AS DEGREE OF BELIEF

- Here probabilities are understood as *degrees of belief*.
- The Bayesian approach to probabilities.
- Fortunately, both kinds of probabilities obey exactly the same laws, and very natural axioms of probabilistic logic lead to a very narrow class of possible functions (Cox, 19).

DIRECT AND INVERSE PROBLEMS

- In probability theory, we have direct and inverse problems.
- Direct problem: there are 10 balls in the urn, 3 of them black. What is the probability of choosing a black ball?
- Or: there are 10 balls in the urn numbered 1 through 10. What is the probability that three balls drawn sequentially from the urn will sum up to 12?
- Inverse problem: we have two urns, 10 balls each, but one has 3 black balls and another 6. Someone took a ball from an urn (chosen at random), and it is black. How probable it is that he took it from the first urn?

DIRECT AND INVERSE PROBLEMS

- Direct problems define a random process and ask to compute the probability of some event (given a model, predict behaviour).
- Inverse problems usually contain *latent variables* and ask to derive their values from the data (given behaviour, construct a model).
- ML problems usually fall into the latter category.
- Note that the probabilities in an inverse problem are usually Bayesian.

DEFINITIONS

- We begin with Bayes theorem:

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

- 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).
- Generally speaking, *likelihood* is a function of the form

$$a \mapsto p(y|x = a)$$

  for some random value y.

ML VS. MAP

- In classical statistics, one often looks for the *maximum likelihood hypothesis*:

$$\theta_{ML} = \arg\max_\theta p(D \mid \theta).$$

- In the Bayesian approach, we are looking for the *posterior distribution*:

$$p(\theta|D) \propto p(D|\theta)p(\theta)$$

  and, possibly, the *maximum a posteriori hypothesis* (MAP):

$$\theta_{MAP} = \arg\max_\theta p(\theta \mid D) = \arg\max_\theta p(D \mid \theta)p(\theta).$$

EXAMPLE

- For example, suppose that we are given a (possibly unfair) coin, it has been tossed N times, and we know the results. The problem is to find out "how unfair" it is and predict the next result.
- The maximal likelihood hypothesis will say that the probability of heads equals the number of heads divided by the number of experiments.

EXAMPLE

- For example, suppose that we are given a (possibly unfair) coin, it has been tossed N times, and we know the results. The problem is to find out "how unfair" it is and predict the next result.
- The maximal likelihood hypothesis will say that the probability of heads equals the number of heads divided by the number of experiments.
- That is, if you took a coin, tossed it once, and heads came up, you'd expect it to always come out heads?
- Kinda strange... we will go over this in detail later.

EXERCISES FOR DISCUSSION

1. A friend of mine has two children; we assume that boys and girls appear with probability $\frac{1}{2}$. Two questions:
   (1) I asked if she has a boy, and she said “yes”; what is the probability that one of the children is a girl?
   (2) I met one of her children, and it’s a boy; what is the probability that the other one is a girl?

EXERCISES FOR DISCUSSION

2. A murder has occurred. Blood was found at the murder scene, which obviously belongs to the killer. It is a rare blood type, only 1% of the population have it, including the accused.
   (1) The prosecutor says: "The chance that the accused would have this blood type if he was innocent is only 1%; hence, with probability 99% he is guilty". What is wrong with this reasoning?
   (2) The defender says: "A million people live in the city, so 10000 of them have this blood type. Therefore, all that it says is that the accused is guilty with probability 0.01%; pretty weak evidence". What is wrong with this reasoning?

PRIOR DISTRIBUTIONS

ML VS. MAP

- Recall that in classical statistics, one often looks for the *maximum likelihood hypothesis*:

$$\theta_{ML} = \arg\max_\theta p(D \mid \theta).$$

- In the Bayesian approach, we are looking for the *posterior distribution*:

$$p(\theta|D) \propto p(D|\theta)p(\theta)$$

  and, possibly, the *maximum a posteriori hypothesis* (MAP):

$$\theta_{MAP} = \arg\max_\theta p(\theta \mid D) = \arg\max_\theta p(D \mid \theta)p(\theta).$$

PROBLEM SETTING

- We are given a (possibly unfair) coin, it has been tossed N times, and we know the results. We need to find out "how unfair" it is and predict the next result.
- Denote by θ the probability of the coin showing heads (the probability of tails is then $1 - \theta$).
- What is the probability of a sequence s with n_h heads and n_t tails?

PROBLEM SETTING

- We are given a (possibly unfair) coin, it has been tossed N times, and we know the results. We need to find out "how unfair" it is and predict the next result.
- Denote by θ the probability of the coin showing heads (the probability of tails is then $1 - \theta$).
- What is the probability of a sequence s with n_h heads and n_t tails?

$$p(s|\theta) = \theta^{n_h}(1-\theta)^{n_t}.$$

- We will assume that θ has uniform prior distribution, i.e., we do not know anything at all about θ (note that this is not true about real coins).
- We now simply take the Bayes theorem and compute.

SAMPLE APPLICATION OF THE BAYES THEOREM

- Likelihood: $p(\theta|s) = \frac{p(s|\theta)p(\theta)}{p(s)}$.
- $p(\theta)$ is a continuous random variable on $[0, 1]$. Our uniformity assumption means that $p(\theta) = 1$, $\theta \in [0, 1]$. And we already know $p(s|\theta)$.
- We get that

$$p(\theta|s) = \frac{\theta^{n_h}(1-\theta)^{n_t}}{p(s)}.$$

SAMPLE APPLICATION OF THE BAYES THEOREM

- $p(s)$ can be computed as

$$p(s) = \int_0^1 \theta^{n_h}(1-\theta)^{n_t} d\theta = \\ = \frac{\Gamma(n_h+1)\Gamma(n_t+1)}{\Gamma(n_h+n_t+2)} = \frac{n_h!\, n_t!}{(n_h+n_t+1)!},$$

  but we could find $\arg\max_\theta p(\theta \mid s) = \frac{n_h}{n_h+n_t}$ without it.

SAMPLE APPLICATION OF THE BAYES THEOREM

- But that it not all. To predict the next outcome, we have to find $p(\text{heads}|s)$:

$$p(\text{heads}|s) = \int_0^1 p(\text{heads}|\theta)p(\theta|s)d\theta =$$
$$= \int_0^1 \frac{\theta^{n_h+1}(1-\theta)^{n_t}}{p(s)}d\theta =$$
$$= \frac{(n_h+1)!\,n_t!}{(n_h+n_t+2)!} \cdot \frac{(n_h+n_t+1)!}{n_h!\,n_t!} = \frac{n_h+1}{n_h+n_t+2}.$$

- This is called *Laplace's rule*.

SAMPLE APPLICATION OF THE BAYES THEOREM

- This is an illustration of the two main problems of Bayesian inference:
  (1) find a posterior distribution on hypotheses or parameters:

$$p(\theta \mid D) \propto p(D|\theta)p(\theta)$$

  (and/or find the maximal a posteriori hypothesis $\arg\max_\theta p(\theta \mid D)$);
  (2) find the posterior distribution of outcomes for further experiments:

$$p(x \mid D) \propto \int_{\theta\in\Theta} p(x \mid \theta)p(D|\theta)p(\theta)\mathrm{d}\theta.$$

THANK YOU!

Thank you for your attention!