ADA511: Data science and data-driven engineering

Steffen Mæland PierGianLuca Porta Mana

2023-06-04

Table of contents

Preface **4**

1 Introduction **5**

2 Data: use and communication **6**
2.1 Sentences – or, what is "data"? 6
2.2 Well-posed and ill-posed sentences 7
Reading list 8

3 Inference **9**
3.1 What is inference? 9
3.2 Certain and uncertain inference 10

4 Truth inference **12**
4.1 Building blocks 12
4.1.1 Basic sentences 12
4.1.2 Connectives 13
4.1.3 Data or axioms 14
4.2 Truth-inference rules 15
4.3 Logical AI agents and their limitations 16

5 Probability inference **18**
5.1 When truth isn't known 18
5.2 Making room for uncertainty:Plausibility, credibility, degree of belief, probability 19
5.3 Inferences with uncertainty: the probability calculus 19
5.3.1 The Three Fundamental Laws of inference 20
5.3.2 Bayes's theorem 20
5.4 Common points of certain and uncertain inference 20

6 Data and information **21**
6.1 Kinds of data 21
6.1.1 Binary 21

6.1.2 Nominal 21
6.1.3 Ordinal 21
6.1.4 Continuous 21
6.1.5 Complex data 21
6.1.6 "Soft" data 21
6.2 Data transformations 22

7 Allocation of uncertainty among possible data values: probability distributions 23
7.1 The difference between Statistics and Probability Theory 23
7.2 What's "distributed"? 23
7.3 Distributions of probability 23
7.3.1 Representations 23
7.4 Summaries of distributions of probability 24
7.4.1 Location 24
7.4.2 Dispersion or range 24
7.4.3 Resolution 24
7.4.4 Behaviour of summaries under transformations of data and errors in data 24
7.5 Outliers and out-of-population data 24
7.6 Marginal and conditional distributions of probability 24
7.7 Collecting and sampling data 24
7.7.1 "Representative" samples 24
7.7.2 Unavoidable sampling biases 25
7.8 Quirks and warnings about high-dimensional data 25

8 Making decisions 26
8.1 Decisions, possible situations, and consequences . 26
8.2 Gains and losses: utilities 26
8.2.1 Factors that enter utility quantification . 26
8.3 Making decisions under uncertainty: maximization of expected utility 26

9 The most general inference problem 27

Preface

**WARNING: THIS IS A WORKING DRAFT. TEXT WILL CHANGE A LOT. MANY PASSAGES ARE JUST TEMPORARY, INCOHERENT, AND DISJOINTED.

To be written.

1 Introduction

To be written: motivation and structure of this course.

2 Data: use and communication

2.1 Sentences – or, what is "data"?

What is "data"?

"Data" (from Latin "given") is used more or less in the same sense as "information", and in these notes we'll use the two words as synonyms.

"Data" is often presented as numbers; but it's obviously more than that. I give you this number: "8". Is it "data"? what is it about? what should you do with it? We can hardly call this number a piece of information, since we have no clue what we could do with it. Instead, if I tell you: "The number of official planets in the solar system is 8", then we can say that I've given you data. So "data" is not just numbers. A number is not "data" unless there's some verbal, non-numeric context associated with it – even if this context is only implicitly understood.

Data can also be completely non-numeric. A clinician saying "The patient has fully recovered from the disease" (we imagine to know who's the patient and what was the disease) is giving us a piece of information that we could further use, for instance, to make prognoses about other, similar patients. The clinician's statement surely is "data". It is essentially non-numeric data, even if in some situations we can represent it as "1", say, while "0" would represent "not recovered".

From these two examples, and with some further thought, we realize that "data" – and in general any piece of information or hypothesis – can universally be represented and communicated by sentences , also called statements or propositions[1]. In some

[1] These terms are not equivalent in Logic, but sometimes we'll use them as synonyms.

cases we can summarize or represent such sentences as numbers. But the numbers alone, by themselves, are not data.

So our conclusion is that information or data is represented by sentences.

Recognizing that data and information are ultimately sentences has important practical consequences:

Clarity and goal-orientation. As a data engineer you'll have to acquire information and convey information. Acquiring information is not simply making some measurement or counting something: you must understand what you are measuring and why. If you gather data from third parties, you have to ask what exactly the data mean and how they were acquired. In designing and engineering a solution, you'll have to understand what information or outcomes the end user exactly wants. It will often happen that you ask "wait, what do you mean by that?"; this question is not just an unofficial parenthesis in the official data-transfer workflow between you and someone else: it is an integral part of that workflow, it means that the data has not been completely transferred yet.

Artificial Intelligence Sentences are the central components of knowledge representation and inference in artificial-intelligence agents.

Reading material

§ 7.1 in Artificial Intelligence

2.2 Well-posed and ill-posed sentences

We face problems when the sentences that should convey information and data are not clear. Suppose that an electric-car model consumes 150 Wh/km and has a range of 200 km; a second car model consumes 250 Wh/km and has a range of 600 km. Someone says "I think the second model is better; what do you think?". It isn't clear how we should answer; what does "better" mean? If it refers to consumption, then the first car model is "better". If it refers to range, then the second model is "better". If it refers to a combination of these two characteristics, or to something else, then we simply can't answer. Here we have a

problem with querying and giving data, because the sentence underlying such query is not clear.

We say that such sentences are **not well-posed**, or that they are **ill-posed**.

This may seem an obvious discussion to you. Yet you'd be surprised by how often unclear sentences appear in scientific papers about data engineering! Not seldom we find discussions and disagreements that actually come from unclear underlying sentences, that two parties interpret in different ways.

As a data engineer, you'll often have the upper hand if you are on the lookout for ill-posed sentences. Whenever you face an important question, or you're given an important piece of information, or you must provide an important piece of information, always take a little time to examine whether the question or information is actually well-posed.

- [TODO] Exercise: give actual paper to analyse

Reading list

3 Inference

3.1 What is inference?

The first core problem in all data-driven engineering applications – and in daily life too – is to draw inferences, that is, acquire information. We may wish to acquire information out of simple curiosity, or for some specific engineering reason or goal, as we'll discuss later. Examples:

1. We'd like to know whether it'll rain today, so we can decide whether to get an umbrella or rain clothes.
2. A clinician would like to know which disease affects a patient, so as to decide for the optimal treatment.
3. The X-player of this game of Xs & Os: needs to know where put the next **X** in order to win.
4. The computer of a self-driving car needs to know whether a particular patch of colours in the visual field is a person, so as to slow down the car and stop.
5. In order to launch a rocket to the Moon, a rocket engineer needs to know, within two significant digits, how much is the velocity $\sqrt{2\,G\,M/r}$, where $G = 6.67 \cdot 10^{-11}\,\mathrm{m^3\,s^{-2}\,kg^{-1}}$, and $M = 5.97 \cdot 10^{24}\,\mathrm{kg}$ and $r = 6.37 \cdot 10^{6}\,\mathrm{m}$ are the mass and radius of the Earth.
6. We'd like to know whether the rolled die will show , so we can win a bet.
7. An aircraft's autopilot system needs to predict how much the aircraft's roll will change by increasing the right wing's angle of attack by 0.1 rad.
8. An archaeologist would like to know whether the fossil bone just dug out belonged to a Tyrannosaurus rex.

9. An automated system in an assembly line needs to predict whether an electric component of a widget will fail within the next two years.

Note how each of these inferences boils down to determining whether some sentences are true or false. In example 1. we want to know whether the sentence ‘It rains today‘ is true or not. In example 2. the clinician wants to know which of the sentences ‘The patient has pneumonia‘, ‘The patient has asthma‘, ‘The patient has bronchitis‘, and so on, are true (several can be true at the same time). In example 5. the rocket engineer wants to know which among the sentences ‘The velocity is 0.010 m/s‘, ‘The velocity is 0.011 m/s‘, ..., ‘The velocity is 130 m/s‘, and so on, is true. The sentences that underlie an inference can be extremely many and complex, and yet we must have an idea of what they are (otherwise, do we really know what our inference is about?).

> **Exercise**
>
> Try to identify which sentences underlie the other example inferences above.

3.2 Certain and uncertain inference

The example inferences above present very different levels of difficulty.

Inferences 3. and 5. are special because they can actually be drawn exactly, that is, we really find out which of their underlying sentences are true and false. In example 3. it is trivial that putting the next **X** in the mid-right slot makes the X-player win. In example 5. a couple of mathematical operations show that the sentence ‘The velocity is 11 km/s‘ is true. When we can obtain the data we want from the data we have by using “only”[2] logic and mathematical operations, our inference is certain, also called a “deduction”; in these notes we shall call it a truth inference. But every deduction can be basically drawn by repeatedly applying the rules of logic.

[2] “Only” in quotation marks because the logical analysis and operations leading to the answer can still be computationally very expensive.

The other example inferences cannot be drawn exactly, in the sense that we cannot know for sure whether all their underlying sentences are true or false. But this doesn't mean that we cannot say anything whatsoever. In example 6. we consider the sentence 'The die shows ' to be more likely false than true. In example 2. the clinician might be quite sure about the disease, after observing the symptoms. On the other hand, in example 1. we might really have no clue whether 'It rains today' will turn out to be true or false. These inferences are uncertain. Certain inferences can be considered as a limit case of uncertain ones, in which the uncertainty vanishes or is extremely small.

To draw certain inferences, we follow the rules of Logic. What rules do we follow to draw uncertain inferences?

4 Truth inference

4.1 Building blocks

Consider the following trivial but certain inference:

$$\frac{\text{‘My umbrella is either blue or red‘} \quad \text{‘My umbrella is not red‘}}{\text{‘My umbrella is blue‘‘My umbrella is not red‘}}$$

Above the line we write the sentences representing the data we have. Below the line we infer the information that supposedly interests us.

How could we draw this obvious inference? Which rules did we follow?

Logic is a huge field that formalizes and makes rigorous the rules that a rational person or an artificial intelligence should use in drawing certain inferences. We'll get a glimpse of it here, as a trampoline for jumping towards our data-driven engineering problems.

4.1.1 Basic sentences

We start by writing down the basic[3] sentences that constitute our data and that underlie the inferences we want to draw. "Basic" in the sense that we will not analyse these sentences into further sub-sentences. In the trivial example above we identify two such sentences: ‘My umbrella is blue‘, and ‘My umbrella is pink‘. Let's represent them by symbols:

$b :=$ ‘My umbrella is blue‘

$r :=$ ‘My umbrella is red‘

[3] A more technical term is "atomic"

Note a subtlety in our data – and again why we need to make their underlying sentences as clear as possible: it is understood here that my umbrella is all of one colour.

4.1.2 Connectives

You notice that we didn't consider 'My umbrella is either blue or red' and 'My umbrella is not red' as basic sentences. These sentences can indeed be expressed in terms of the basic sentences b and r. We consider one way or operation to change a sentence into another related to it, and two ways or operations to combine two or more sentences together. These operations are called "connectives". Our natural language offer many more operations to combine sentences, but these three turn out to be all we need in virtually all engineering problems. The three connectives are:

Not: $\neg$ for example,

$$\neg r = \text{'My umbrella is not red'}$$

And: $\wedge$ for example,

$$b \wedge r = \text{'My umbrella is blue, and it is red'}$$

Or: $\vee$ for example,

$$b \vee r = \text{'My umbrella is blue, or red, or both'}$$

Note some subtleties of the connectives:

- "Not" doesn't mean some kind of complementary quality, but only the negation. For instance, ¬'The chair is black' does not mean 'The chair is white'.
- $b \vee r$ does not exclude, a priori, that my umbrella cannot be both blue and black (there is a connective for that: "exclusive-or", but it can be constructed out of the three we already have.)

From this last remark we see that the sentence 'My umbrella is either blue or red' does not correspond to $b \vee r$. The sentence also means implicitly that my umbrella cannot be both blue or red. We could rewrite it as 'My umbrella is either blue or red, and it is not both blue and red'. Convince yourself that in symbols we can write it like this:

$$(b \vee r) \wedge \neg(b \wedge r) = \text{'My umbrella is either blue or red'}$$

4.1.3 Data or axioms

Now we have the sentences to represent our data, and even symbols to represent it in a compact way. But what do our data actually say? They say that the sentences 'My umbrella is either blue or red' and 'My umbrella is not red' are true. Here's how we express this in symbols.

We represent our data by the symbol D and use the notation[4]

$$\begin{aligned} \neg r \mid D \\ (b \vee r) \wedge \neg(b \wedge r) \mid D \end{aligned}$$

[4] Current notation in logic writes $D \models \neg r$. We use a different notation for an easier transition to probability logic.

to mean that 'My umbrella is not red' and 'My umbrella is either blue or red' are `true` according to our data.

With this notation we can also augment our data with additional assumptions or hypotheses, even if just temporarily. For example,

$$\neg r \mid b \wedge D$$

means that 'My umbrella is not red' is `true` according to data D together with the additional assumption that 'My umbrella is blue' is `true`.

4.2 Truth-inference rules

Deduction systems in formal logic give us a set of rules for making correct inferences. These rules can be represented in a wide variety of ways. For instance as lines: above, we write what the data say; below, what inference we can draw. An example is this:

$$\frac{a \wedge b \mid D}{a \mid D} \tag{4.1}$$

In total there are a dozen or so rules of this kind.

But we can compactly encode all these rules in the following way. First, represent `true` with the number 1, and `false` with 0. Second, express the fact that a sentence a – which can be made of subsentences combined by connectives – is `true` according to data D by writing

$$\mathrm{T}(a \mid D) = 1$$

and that it is `false` according to D by writing

$$\mathrm{T}(a \mid D) = 0$$

Then the rules of truth inference are summarized by the following equations, which must always hold:

Let's see how the inference rule (4.1), for example, is encoded in these equations. The rule starts with saying that $a \wedge b$ is

Rule for "not":

$$\mathrm{T}(\neg a \mid D) + \mathrm{T}(a \mid D) = 1 \tag{4.2}$$

Rule for "and":

$$\mathrm{T}(a \wedge b \mid D) = \mathrm{T}(a \mid b \wedge D) \cdot \mathrm{T}(b \mid D) = \mathrm{T}(b \mid a \wedge D) \cdot \mathrm{T}(a \mid D) \tag{4.3}$$

Rule for "or":

$$\mathrm{T}(a \vee b \mid D) = \mathrm{T}(a \mid D) + \mathrm{T}(b \mid D) - \mathrm{T}(a \wedge b \mid D) \tag{4.4}$$

Rule of self-consistency:

$$\mathrm{T}(a \mid a \wedge D) = 1 \tag{4.5}$$

`true` according to D. This means that $\mathrm{T}(a \wedge b \mid D) = 1$. But, by rule (4.3), we must then have $\mathrm{T}(b \mid a \wedge D) \cdot \mathrm{T}(a \mid D) = 1$. This can only happen if both $\mathrm{T}(b \mid a \wedge D)$ and $\mathrm{T}(a \mid D)$ are equal to 1. So we can conclude that $\mathrm{T}(a \mid D) = 1$, which is exactly the conclusion under the line in rule (4.1).

Exercise

Try to prove our initial inference

$$\frac{(b \vee r) \wedge \neg(b \wedge r) \mid D \qquad \neg r \mid D}{b \mid D}$$

using the basic rules (4.2, 4.3, 4.4, 4.5). Remember that you can use each rule as many times as you like, and that there is not only one way of constructing a proof.

4.3 Logical AI agents and their limitations

The basic rules above are also the rules that a logical artificial-intelligent agent should follow.

Many – if not most – inference problems that a data engineer must face are, however, of the uncertain kind: it is not possible to surely infer the truth of some data, and the truth of some initial data may not be known either. In the next chapter we shall see how to generalize the logic rules to uncertain situations.

Reading material

Ch. 7 in Artificial Intelligence

Extra reading material

Our cursory visit of formal logic only showed a microscopic part of this vast field. The study of logic rules continues still today, with many exciting developments and applications. Feel free take a look at Logic in Computer Science, Mathematical Logic for Computer Science, Natural Deduction Systems in Logic

5 Probability inference

5.1 When truth isn't known

In most real-life and engineering situations we don't know the truth or falsity of sentences that interest us. But this doesn't mean that nothing can be said or done in such situations.

When we cross a busy city street we look left and right to check whether any cars are approaching. We typically don't look up to check whether something is falling from the sky. Yet, couldn't it be `false` that cars are approaching? and couldn't it be `true` that some object is falling from the sky? Of course both events are possible. Then why do we look left and right, but not up?

The main reason[5] is that we believe strongly that cars might be approaching, believe very weakly that some object might be falling from the sky. In other words, we consider the first occurrence to be very probable; the second, extremely improbable.

[5] We shall see later that one more factor enters the explanation.

We shall take the notion of **probability** as intuitively understood (just as we did with the notion of truth). But it is important to emphasize and agree on at least two facts about it:

- **Probability is agent- and context-dependent**. A coin is tossed, comes down heads, and is quickly hidden from view. Alice sees that it landed heads-up. Bob instead doesn't manage to see the outcome and has no clue. Alice considers the sentence '`Coin came down heads`' to be `true`, that is, to have `100% probability`. Bob considers the same sentence to have `50%` probability.

  Note how Alice and Bob assign two very different probabilities to the same sentence; yet both assignments are

completely rational. If Bob assigned 100% to 'heads', we would suspect that he had seen the outcome after all; if he assigned 0% to 'heads', we would consider the assignment groundless and silly. We would be baffled if Alice assigned 50% to 'heads', because she saw the outcome; we would hypothesize that she feels unsure about what she saw.

- **Probability is not a physical property**. Whether a tossed coin lands heads up or tails up is fully determined by the initial conditions (position, orientation, momentum, rotational momentum) of the toss and the boundary conditions (air velocity and pressure) during the flight. The same is true for all macroscopic engineering phenomena. (Even quantum phenomena have not been proved to be non-deterministic, and there are deterministic and experimentally consistent mathematical presentations of quantum theory.)

These two facts are not just matter of principle; they have real practical consequences. A data engineer who does not carefully assess the context of a probability will design a system with non-optimal performance[6] – or even cause deaths. The same is true of a data engineer who does not carefully take advantage, when possible, of the physics involved in the engineering problem.

Reading material

Dynamical Bias in the Coin Toss

[6] This fact can be mathematically proven.

5.2 Making room for uncertainty:Plausibility, credibility, degree of belief, probability

5.3 Inferences with uncertainty: the probability calculus

THE FUNDAMENTAL RULES OF INFERENCE

Rule for "not" $\neg$

$$\mathrm{P}(\neg a \mid D) + \mathrm{P}(a \mid D) = 1 \tag{5.1}$$

Rule for "and" $\wedge$

$$\mathrm{P}(a \wedge b \mid D) = \mathrm{P}(a \mid b \wedge D) \cdot \mathrm{P}(b \mid D) = \mathrm{P}(b \mid a \wedge D) \cdot \mathrm{P}(a \mid D) \tag{5.2}$$

Rule for "or" $\vee$

$$\mathrm{P}(a \vee b \mid D) = \mathrm{P}(a \mid D) + \mathrm{P}(b \mid D) - \mathrm{P}(a \wedge b \mid D) \tag{5.3}$$

Rule of self-consistency

$$\mathrm{P}(a \mid a \wedge D) = 1 \tag{5.4}$$

5.3.1 The Three Fundamental Laws of inference

- Exercise: Monty-Hall problem & variations
- Exercise: clinical test & diagnosis

5.3.2 Bayes's theorem

5.4 Common points of certain and uncertain inference

No premises? No conclusions!

6 Data and information

6.1 Kinds of data

6.1.1 Binary

6.1.2 Nominal

6.1.3 Ordinal

6.1.4 Continuous

- unbounded
- bounded
- censored

6.1.5 Complex data

2D, 3D, images, graphs, etc.

6.1.6 "Soft" data

- orders of magnitude
- physical bounds

6.2 Data transformations

- log
- probit
- logit

7 Allocation of uncertainty among possible data values: probability distributions

7.1 The difference between Statistics and Probability Theory

Statistics is the study of collective properties of collections of data. It does not imply that there is any uncertainty.

Probability theory is the quantification and propagation of uncertainty. It does not imply that we have collections of data.

7.2 What's "distributed"?

Difference between distribution of probability and distribution of (a collection of) data.

7.3 Distributions of probability

7.3.1 Representations

- Density function
- Histogram
- Scatter plot

Behaviour of representations under transformations of data.

7.4 Summaries of distributions of probability

7.4.1 Location

Median, mean

7.4.2 Dispersion or range

Quantiles & quartiles, interquartile range, median absolute deviation, standard deviation, half-range

7.4.3 Resolution

Differential entropy

7.4.4 Behaviour of summaries under transformations of data and errors in data

7.5 Outliers and out-of-population data

(Warnings against tail-cutting and similar nonsense-practices)

7.6 Marginal and conditional distributions of probability

7.7 Collecting and sampling data

7.7.1 "Representative" samples

Size of minimal representative sample = (2^entropy)/precision

- Exercise: data with 14 binary variates, 10000 samples

7.7.2 Unavoidable sampling biases

In high dimensions, all datasets are outliers.

Data splits and cross-validation cannot correct sampling biases

7.8 Quirks and warnings about high-dimensional data

8 Making decisions

8.1 Decisions, possible situations, and consequences

8.2 Gains and losses: utilities

8.2.1 Factors that enter utility quantification

Utilities can rarely be assigned a priori.

8.3 Making decisions under uncertainty: maximization of expected utility

9 The most general inference problem