Assignment 1: Compute Blood Types

AI-2 Systems Project (Summer Semester 2025)

Jan Frederik Schaefer

Friedrich-Alexander-Universität Erlangen-Nürnberg, Department Informatik

Topic:	Bayesian networks
Due on:	June 22, 2025
Version from:	May 9, 2025
Author:	Jan Frederik Schaefer

Make sure you sign up before working on this assignment.[a]

Using someone else's solution code, even as inspiration, is not allowed!

---

[a] You can still decide to postpone the assignment. Signing up includes an elligibility check, which avoids situations where you invest work into an assignment that you are not supposed to take.

1 Task summary

Using Bayesian networks, compute the probability that someone has a particular blood type given some test results for the blood types of their relatives. The assignment repository [AR] contains example problems for you to solve.

Didactic objectives

1. Learn how to model a problem as a Bayesian Network,
2. gain experience working with graphs,
3. use complex conditional probability tables,
4. find and use a suitable library for Bayesian inference (many exist and finding good libraries is an important skill),
5. get to know the JSON format (in case you haven't used it before).

Prerequisites and useful methods

1. Basics of representing graphs in code and working with them,
2. Bayesian networks (taught in the AI lecture).

2 Background: The ABO blood group system

The **ABO blood group system** [ABO] is used to describe the presence/absence of A and B antigens on the red blood cells in humans. We distinguish 4 different **(ABO) blood types**: **A** (if only the A antigens are present), **B** (if only the B antigens are present), **AB** (if both are present) and **O** (if neither are present). Knowing the blood type is in particular important for blood transfusions as a transfusion with an incompatible blood type can be lethal. Another important antigen for blood transfusions is Rh, which we will ignore in this assignment.

2.1 Alleles

The ABO blood type is determined by the **ABO gene**, which comes in three variants, called **alleles**: the **A allele**, the **B allele** and the **O allele**. Humans have chromosome pairs and therefore two versions of the **ABO gene** – one taken from each parent. The A allele and B allele are co-dominant: if at least one ABO gene has the A allele, the person has A antigens, and if at least one ABO gene has the B allele, the person has B antigens. Humans with both the A and B allele have both antigens and therefore blood type AB and only humans with two O alleles have neither antigens and therefore blood type O. In the following, we will sometimes write XY to denote that someone has one X allele and one Y allele.

2.2 Inheritance

A child randomly gets one allele from each parent. For example, if the father has an A allele and an O allele (short AO) and the mother has an A allele and a B allele (short AB), then the child could have one of the following allele combinations:

1. AA (blood type A)
2. OA (blood type A)
3. AB (blood type AB)
4. OB (blood type B)

Each of the four combinations is equally likely.

```
{
    "family-tree": [
        {"relation": "father-of", "subject": "Omar", "object": "Rory"},
        {"relation": "mother-of", "subject": "Samantha", "object": "Rory"}
    ],
    "country": "North␣Wumponia",
    "test-results": [{"type": "bloodtype-test", "person": "Samantha", "result": "O"}],
    "queries": [{"type": "bloodtype", "person": "Rory"}]
}
```

Listing 1: Example problem file.

Country	**A allele**	**B allele**	**O allele**
North Wumponia	50 %	25 %	25 %
South Wumponia	15 %	55 %	30 %

Figure 1: Prior probabilities that the ABO gene has a particular allele by country.

3 Detailed problem description

The assignment repository [AR] contains problem files and example solutions encoded in the JSON format. Listing 1 shows an example problem file. The `family-tree` field specifies the relationships between family members: Omar (`subject`) is the father (`relation`) of Rory (`object`) and Samantha is the mother of Rory. Every family member is uniquely identified by their name. The `country` field specifies the country where the family lives. This is relevant because the distribution of alleles depends on the country (see Section 3.2 for more details). The `test-results` field lists the results of different tests. In this case, we know that Samantha has blood type O. At last, the `queries` field specifies what we should query for. In this case, we are interested in the blood type of Rory. Section 3.4 describes the solution format.

3.1 Details on the family tree specification

As mentioned above, the family tree is described via relationhips between people. The following types of relationship are used: `mother-of`, `father-of`.

However, if a relation is not stated, you should not assume it. For example, if X is the parent of both Y and Z, you should not assume that Y and Z share the second parent.

3.2 Distribution of ABO alleles

To solve the assignment, you need to know the distribution of the ABO alleles in the population, which actually varies significantly across the world. In this assignment, we use imaginary countries with different allele distributions. Figure 1 shows the corresponding distributions. We also assume that the alleles of a person are independent, so e.g. the prior probability of having blood type AB in North Wumponia is $2 \cdot 50\,\% \cdot 25\,\% = 25\,\%$ (both the AB and the BA allele pair result in the AB blood type).

3.3 Details on test results

You get some information about the family from tests. There are different types of tests:

1. **bloodtype-test**: This test tells you the blood type of a family member (see Listing 1 for an example) and is always correct.
2. **mixed-bloodtype-test**: In mixed tests, the blood of two people is mixed and the test center only tells you the blood type of the mixture. For example, if one person has blood type A and the other has blood type B, the test center will tell you that the mixture has blood type AB (both antigens are present). If both people have the same blood type, the test center will tell you that the mixture has that blood type.
3. **pair-bloodtype-test**: In the pair test, blood samples of two people are submitted together. The test center then tells you the blood type for each person individually. However, there is a 20 % chance that the test center mixes up the blood samples and returns the blood type of the other person instead. So, if everything goes well, the blood type specified in `result-1` is for `person-1` and the blood type specified in `result-2` is for `person-2`. But there is a 20 % chance that the blood type specified in `result-1` is instead for `person-2` and the blood type specified in `result-2` is for `person-1`.

3.4 Solution format

The solution to a problem file should also be stored in a JSON file. It should list answers to all the queries. For example, the solution to the example problem in Listing 1 should be represented as

```
[
    {
        "type": "bloodtype",
```

```
        "person": "Rory",
        "distribution": {
            "O": 0.25,
            "A": 0.5,
            "B": 0.25,
            "AB": 0.0
        }
    }
]
```

meaning that e.g. Rory has blood type O with a probability of 25 %.

The assignment repository contains more example problems and solutions that you can use for comparison. Furthermore, it provides a script for comparing your solutions to the example problem with the provided solutions.

4 What to submit

Your solution should be submitted to your team's repository. It should contain:

1. all your code,
2. a `README.md` file explaining
   - i. dependencies (programming language, version, external libraries and how to get them),
   - ii. how to run your code to solve other problems,
   - iii. the repository structure,
   - iv. anything else we should know,
3. a solution summary (see [SoS] for more details – it should describe the main ideas, not document the code),
4. a file `solution-[LETTER]-[NUMBER].json` for every `problem-[LETTER]-[NUMBER].json` that you managed to solve.

5 Points

You get 1 point for every correctly solved problem file. The problem files vary in their difficulty (Figure 2 provides an overview). As there are 80 problem files, that means that you can get up to 80 points for your solutions. Assuming you have at least a partial solution, you

Problems	**Family relations**	**Test types**	**Countries**
`problem-a-∗.json`	mother-of father-of	bloodtype-test	North Wumponia
`problem-b-∗.json`	mother-of father-of	bloodtype-test	North Wumponia South Wumponia
`problem-c-∗.json`	mother-of father-of	bloodtype-test mixed-bloodtype-test	North Wumponia
`problem-d-∗.json`	mother-of father-of	bloodtype-test pair-bloodtype-test	North Wumponia
`problem-e-∗.json`	mother-of father-of	bloodtype-test mixed-bloodtype-test pair-bloodtype-test	North Wumponia South Wumponia

Figure 2: Overview of the different problem files. The first problems (`problem-a-∗.json`) only use a minimal family (two parents and their child).

can additionally get up to 20 points for the quality of the submission (README, explanation, ...). The maximum number of points is therefore 100. If the grading scheme doesn't seem to work well, we might adjust it later on (likely in your favor).

References

[ABO] *ABO blood group system*. URL: https://en.wikipedia.org/wiki/ABO_blood_group_system (visited on 06/01/2022).

[AR] *Repository for Assignment 1: Compute Blood Types*. URL: https://gitlab.rrze.fau.de/wrv/AISysProj/ss25/a2.1-compute-blood-types/assignment.

[SoS] *Solution Summary*. URL: https://gitlab.rrze.fau.de/wrv/AISysProj/admin/general/-/blob/main/solution-summary.md.