CLIPS

CLIPS is one of the most widely used expert system tools. It is a rule-based, object-oriented and procedural expert system. In CLIPS, I defined template of facts, rules and methods by deftemplate, defrule and defmethod commands respectively.

Computation of probability

We are given P(H), P(E|H) and P(E|~H). The project requires to find which disease a person is most likely to have. In order to find this disease, I calculated the probability of having each disease, then took the one with the maximum probability. Assuming that the symptoms (evidences) are independent, the probability of having a disease is:

$P(H|E_1E_2\ldots E_n) = P(E_1|H)P(E_2|H)\ldots P(E_n|H)P(H)/P(E_1E_2\ldots E_n)$ (1)

ILLNESS.txt file contains P(H), $P(E_n|H)$ and $P(E_n|\sim H)$, however, does not contain $P(E_n)$.

We know that

$P(H|E_1E_2\ldots E_n) + P(\sim H|E_1E_2\ldots E_n) = 1$ (2)

and

$P(\sim H|E_1E_2\ldots E_n) = P(E_1|\sim H)P(E_2|\sim H)\ldots P(E_n|\sim H)P(\sim H)/P(E_1E_2\ldots E_n)$. (3)

The denominator of (1) and (3) are the same. If we refer the numerator of (1) “a”, numerator of (3) “b” and their common denominator “c”, then equation (2) becomes

(a / c) + (b / c) = 1. (4)

Hence,

c = a + b. (5)

If we substitute the denominator in (1) by (5) then (1) becomes

$P(H|E_1E_2\ldots E_n) = a / (a + b)$. (6)

Since we are given all the probabilities in “a” and “b”, we can easily compute $P(H|E_1E_2\ldots E_n)$ using equation (6).

Knowledge representation

I created four templates:

1. illness: contains information read from ILLNESS.txt and is modified as symptoms are added to facts. Slots:
    - illness-name: used for printing the name of the illness to output file and as key.
    - illness-no: used as key.
    - prob: refers to “a” or “b” in the **Computation of probability** . Initialized as P(H) or P(~H), then multiplied by P(E|H) or P(E|~H) as new evidences appear.
    - is-positive: if “y”, then prob is “a” else (“n”) prob is “b”.
    - input-no: refers to input number in INPUTS.txt which is from 1 to 40.
2. final-illness: contains $P(H|E_1E_2...E_n)$ which is refered as a / (a + b) in the **Computation of probability** .
    - illness-name: used for printing the name of the illness to output file and as key.
    - prob: a / (a + b)
    - input-no: refers to input number in INPUTS.txt which is from 1 to 40.
3. illness-cond-prob: contains P(E|H) and P(E|~H)
    - illness-no: used as key.
    - illness-name: used for printing the name of the illness to output file and as key.
    - symptom-no: eg. 1 (Are you sneezing a lot ?)
    - prob-yes: P(E|H)
    - prob-no: P(E|~H)
4. input: contains information read from INPUTS.txt
    - input-no: refers to input number in INPUTS.txt which is from 1 to 40.
    - answer: if yes then “y”, else (no) “n”.
    - symptom-no: eg. 1 (Are you sneezing a lot ?)
    - illness-no: used as key.

Structure of the code

First, I defined all templates and methods. Then I read ILLNESS.txt and INPUTS.txt by readline command. Then I calculated “a” and “b”, and the final value, “a / (a + b)”. Then I took the maximum of probabilities for each input and wrote to OUTPUTS.txt using printout command. Please note that the result in OUTPUTS.txt is not in the ascending order of input numbers.

How to run

I run the project by basically copy-pasting each command in enclosing parentheses in Assignment2.clp. All input files should be in the same directory as Assignment2.clp. OUTPUTS.txt is generated in the same directory.