Challenge May-2021
COVID-19 Testing

A solution with DT5GL by Jack Jansonius – 26 May 2021

At first glance a simple challenge, which becomes particularly interesting as more intelligence is built in.

The description of the challenge as stated on the website:

> We ask to submit a knowledge specification, through DM tables or otherwise, that describes the following triage procedure that determines for (a group of) patients who should undergo subsequent testing:
>
> 1. Patients present sneezing and/or coughing as symptoms; this is determined by what the patient him or herself details (i.e., anamnesis).
> 2. A patient's temperature is measured. The default fever threshold is 38°C; due to different methods of measurement for young children and their associated accuracy, for patients younger than 10 the fever threshold is considered to be 37.2°C.
> 3. Based on the first two steps, the number of presented symptoms is determined. If a patient presents at least two (2) of these symptoms, subsequent testing is performed, otherwise standard quarantine practices are advised.
>
> In this knowledge specification, pay special attention to the nature of the pandemic as an ongoing crisis with expertise being gained continuously. Specifically, take into account how the following scenarios impact your specification:
>
> 1. The possible emergence of additional symptoms such as loss of smell or loss of taste.
> 2. A different threshold of symptoms deemed present before referral to subsequent testing.

In this clear description, the rule as mentioned at step 3 raises a question: what should be done if both step 1 and step 2 produce no symptoms? What seems plausible to me is: no subsequent action. And that brings the number of possible outcomes of the procedure to 3: "subsequent testing", "quarantine practice" and "no follow-up action".

That does not make this challenge impressively complicated. It is notable, however, that Simon Vandevelde does not build any intelligence into his solution, stating "First, we should decide if a patient has a fever", while the challenge explicitly states that any fever is only determined at step 2.

Thus, when applying his model in an interactive environment, the user will first be asked questions about age and temperature for determining fever and only then be asked the questions associated with step 1.

This fully confirms the comments I made at the April 2021 challenge: as a result of a data-driven approach, Vandevelde's solution has no intelligence built into it.

What would the solution look like if intelligence did get built in?
Very general: determine if there is a fever at step 2 only if the outcome affects the final decision.

If we assume 4 possible symptoms at step 1, for example sneezing, coughing, loss of smell and loss of taste and a minimum of 4 symptoms to decide on PCR testing and additionally give the symptom fever at step 2 a weight of 2, this leads to the following scenarios:

1. Step 1 yields 4 symptoms; step 2 not relevant because 4(+2) => PCR testing
2. Step 1 yields 3 symptoms; step 2 is relevant because 3+2=5 => PCR testing
3. Step 1 yields 2 symptoms; step 2 is relevant because 2+2=4 => PCR testing
4. Step 1 yields 1 symptom; step 2 not relevant because 1(+2) => quarantine practice
5. Step 1 yields 0 symptoms; step 2 is relevant because 0+1 => quarantine practice

With this intelligence built in, the challenge immediately becomes a lot tougher!

On the following pages I first show an interactive version of the solution, in which the desired input variables are requested from the user.

With a slight modification to this interactive version, the desired input variables are then read from a SQLite database file and the results are then recorded in that same file.

Which proves once again:
intelligence requires a goal-driven approach, not a data-driven one.

Implementation of the decision tables in DT5GL; interactive version:

```
Table 0:
If:                                          | 0| 1| 2| 3| 4| 5| 6|
'Continue'                                   | Y| Y| Y| Y| Y| Y| N|
#Step1_symptoms >= Threshold                 | Y| N| N| N| N| N| -|
#Step1_symptoms >= Threshold - Weight_step2  | -| Y| Y| N| N| N| -|
#Step1_symptoms = 0                          | -| -| -| Y| Y| N| -|
#Step2_symptom  > 0                          | -| Y| N| Y| N| -| -|
Then:
Action is finished                           |  |  |  |  |  |  | X|
Action is PCR-testing                        | X| X|  |  |  |  |  |
Action is standard_quarantine_practice       |  |  | X| X|  | X|  |
Action is not_required                       |  |  |  |  | X|  |  |
# .......


Proposition: 'Continue'
Askable_using: "Next client?"

Attribute: #Step1_symptoms
Equals: Sneezing + Coughing + Loss_of_Smell + Loss_of_Taste

Attribute: Sneezing                          Type: Integer
Askable_using: "Sneezing (0=False, 1=True)?"
Attribute: Coughing                          Type: Integer
Askable_using: "Coughing (0=False, 1=True)?"
Attribute: Loss_of_Smell                     Type: Integer
Askable_using: "Loss of Smell(0=False, 1=True)?"
Attribute: Loss_of_Taste                     Type: Integer
Askable_using: "Loss of Taste (0=False, 1=True)?"

Attribute: Threshold                         Type: Integer
Equals: 4
Attribute: Weight_step2                      Type: Integer
Equals: 2

Attribute: #Step2_symptom
Equals: Fever * Weight_step2

Table 1:
If:                           | 0| 1| 2| 3|
Age >= 10                     | Y| Y| N| N|
Temperature >= 38.0           | Y| N| -| -|
Temperature >= 37.2           | -| -| Y| N|
Then:
Fever = 1                     | X|  | X|  |
Fever = 0                     |  | X|  | X|
# .......

Attribute: Age
Askable_using: "What is the age?"
Attribute: Temperature
Askable_using: "What is the temperature?"
```

```
GoalAttribute: Action
Repeat_until: finished

Case: finished
Print: "Finished!"

Case: PCR-testing
Print: "Subsequent PCR testing is recommended!"
Print: "Step 1 symptoms: %s. "          #Step1_symptoms
Print: "Weight step2 symptom: %s. "    #Step2_symptom.getvalue

Case: standard_quarantine_practice
Print: "Standard quarantine practice is recommended."
Print: "Step 1 symptoms: %s. "          #Step1_symptoms
Print: "Weight step2 symptom: %s. "    #Step2_symptom.getvalue

Case: not_required
Print: "No action is required."
Print: "Step 1 symptoms: %s. "          #Step1_symptoms
Print: "Weight step2 symptom: %s. "    #Step2_symptom.getvalue
```

Some test cases:

```
"Next client? (y/n)? > y
"Sneezing (0=False, 1=True)?"
(int)> 1

"Coughing (0=False, 1=True)?"
(int)> 1

"Loss of Smell(0=False, 1=True)?"
(int)> 1

"Loss of Taste (0=False, 1=True)?"
(int)> 1

Subsequent PCR testing is recommended!
Step 1 symptoms: 4.
Weight step2 symptom: None.

"Next client? (y/n)? > y
"Sneezing (0=False, 1=True)?"
(int)> 1

"Coughing (0=False, 1=True)?"
(int)> 0

"Loss of Smell(0=False, 1=True)?"
(int)> 1

"Loss of Taste (0=False, 1=True)?"
(int)> 0

"What is the age?"
(int)> 9

"What is the temperature?"
(real)> 37.1

Standard quarantine practice is recommended.
Step 1 symptoms: 2.
Weight step2 symptom: 0.

"Next client? (y/n)? > n
Finished!
```

Implementation of the decision tables in DT5GL; database version:

```
SQLite_database: "Database/Covidtest.sqlite3"

Table 0:
If:                                        | 0| 1| 2| 3| 4| 5| 6|
'Next client'                              | Y| Y| Y| Y| Y| Y| N|
#Step1_symptoms >= Threshold              | Y| N| N| N| N| N| -|
#Step1_symptoms >= Threshold - Weight_step2 | -| Y| Y| N| N| N| -|
#Step1_symptoms = 0                        | -| -| -| Y| Y| N| -|
#Step2_symptom  > 0                        | -| Y| N| Y| N| -| -|
Then:
Action is finished                         |  |  |  |  |  |  | X|
Action is PCR-testing                      | X| X|  |  |  |  |  |
Action is standard_quarantine_practice     |  |  | X| X|  | X|  |
Action is not_required                     |  |  |  |  | X|  |  |
# .......


Proposition: 'Next client'
Obtain_instance_from_database_view: client

Attribute: #Step1_symptoms
Equals: client.sneezing + client.coughing + client.loss_of_smell    \
                                    + client.loss_of_taste

Attribute: Threshold        Type: Integer
Equals: 4
Attribute: Weight_step2    Type: Integer
Equals: 2

Attribute: #Step2_symptom
Equals: Fever * Weight_step2

Table 1:
If:                                        | 0| 1| 2| 3|
client.age >= 10                           | Y| Y| N| N|
client.temperature >= 38.0                 | Y| N| -| -|
client.temperature >= 37.2                 | -| -| Y| N|
Then:
Fever = 1                                  | X|  | X|  |
Fever = 0                                  |  | X|  | X|
# .......


Database_view: client
With_attributes:
personid,name,age,temperature,sneezing,coughing,loss_of_smell,loss_of_taste
Query:
SELECT *
  FROM client
 LIMIT 1 OFFSET %s
With_arguments: client.auto_index


Attribute: client.sneezing        Type: Integer
Attribute: client.coughing        Type: Integer
Attribute: client.loss_of_smell   Type: Integer
Attribute: client.loss_of_taste   Type: Integer
```

```
GoalAttribute: Action
Repeat_until: finished

Case: finished
Print: "Finished!"

Case: PCR-testing
Print: "Result for %s: Subsequent PCR testing."        client.name
>SQL:  "UPDATE client "
-SQL:  "   SET result = 'PCR test', "
-SQL:  "       fever = %s, "                            Fever.getvalue
-SQL:  "       feverweight = %s "                       #Step2_symptom.getvalue
<SQL:  " WHERE Name = '%s' "                            client.name

Case: standard_quarantine_practice
Print: "Result for %s: Standard quarantine practice." client.name
>SQL:  "UPDATE client "
-SQL:  "   SET result = 'Standard quarantine practice', "
-SQL:  "       fever = %s, "                            Fever.getvalue
-SQL:  "       feverweight = %s "                       #Step2_symptom.getvalue
<SQL:  " WHERE Name = '%s' "                            client.name

Case: not_required
Print: "Result for %s: No action required."            client.name
>SQL:  "UPDATE client "
-SQL:  "   SET result = 'No action required', "
-SQL:  "       fever = %s, "                            Fever.getvalue
-SQL:  "       feverweight = %s "                       #Step2_symptom.getvalue
<SQL:  " WHERE Name = '%s' "                            client.name



Initial_database_setup: delete_clients
Query:
    DELETE FROM client
End_Query

Initial_database_setup: insert_new_clients
Query:
    INSERT INTO client
    (personid,name, age,temperature,sneezing,coughing,loss_of_smell,loss_of_taste)
    VALUES
    (1,      'TC0',    33,      Null,     1,     1,          1,              1),
    (2,      'TC1a',   10,      38.0,     0,     1,          1,              1),
    (3,      'TC1b',    9,      37.2,     1,     0,          0,              1),
    (4,      'TC2a',   10,      37.9,     0,     1,          1,              1),
    (5,      'TC2b',    9,      37.1,     1,     0,          0,              1),
    (6,      'TC3',    10,      38.0,     0,     0,          0,              0),
    (7,      'TC4',     9,      37.1,     0,     0,          0,              0),
    (8,      'TC5',  Null,      Null,     0,     1,          0,              0)
End_Query
```

Testrun for the database-version:

```
Result for TC0: Subsequent PCR testing.
Result for TC1a: Subsequent PCR testing.
Result for TC1b: Subsequent PCR testing.
Result for TC2a: Standard quarantine practice.
Result for TC2b: Standard quarantine practice.
Result for TC3: Standard quarantine practice.
Result for TC4: No action required.
Result for TC5: Standard quarantine practice.
Finished!
```

	personid	name	age	temperatu	sneezing	coughing	loss of sn	loss of ta	fever	feverweig	result
1	1	TC0	33	NULL	1	1	1	1	NULL	NULL	PCR test
2	2	TC1a	10	38	0	1	1	1	1	2	PCR test
3	3	TC1b	9	37.2	1	0	0	1	1	2	PCR test
4	4	TC2a	10	37.9	0	1	1	1	0	0	Standard quarantine practice
5	5	TC2b	9	37.1	1	0	0	1	0	0	Standard quarantine practice
6	6	TC3	10	38	0	0	0	0	1	2	Standard quarantine practice
7	7	TC4	9	37.1	0	0	0	0	0	0	No action required
8	8	TC5	NULL	NULL	0	1	0	0	NULL	NULL	Standard quarantine practice