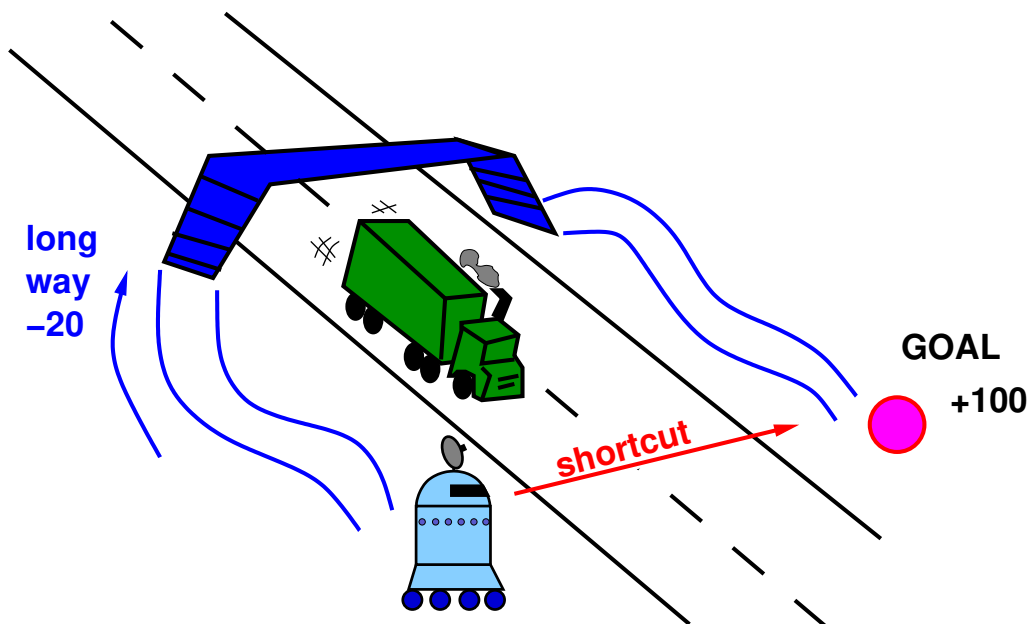
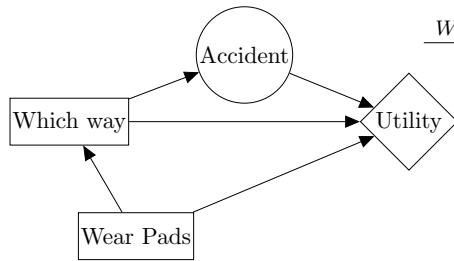


# Decision Network Examples

Jesse Hoey  
David R. Cheriton School of Computer Science  
University of Waterloo  
Waterloo, Ontario, CANADA, N2L3G1  
jhoey@cs.uwaterloo.ca

To get to its goal, a robot can go one of two ways: a long, safe route and a shortcut. The shortcut crosses a busy road on which the robot gets into an accident with probability 0.2. The probability of an accident on the long, safe route is 0.01. The goal is worth 100, but the robot has to pay 20 to take the long route. Having an accident costs 97 on the short route (leaving the robot with only 3) and 80 on the long route (leaving the robot with 0). However, the robot can put on a set of pads before it starts, but putting on the pads costs 5. Getting into an accident while wearing pads only costs 60 on the short route (leaving the robot with 35) and 45 on the long route (leaving the robot with 30).





WhichWay	$P(\text{Accident} = \text{true}   \text{WhichWay})$
long	0.01
short	0.2

WhichWay	Accident	WearPads	$U(\text{Accident}, \text{WhichWay}, \text{WearPads})$
long	true	true	30
long	false	true	75
long	true	false	0
long	false	false	80
short	true	true	35
short	false	true	95
short	true	false	3
short	false	false	100

The expected value for the robot is:

$$\mathcal{E}(\text{which\_way}, \text{wear\_pads}) = \sum_{\text{accident}} P(\text{accident} | \text{which\_way}) U(\text{which\_way}, \text{accident}, \text{wear\_pads})$$

which can be computed for each possible decision as:

which_way	wear_pads	$\mathcal{E}(\text{which\_way}, \text{wear\_pads})$
short	true	$0.2 * 35 + 0.8 * 95 = 83$
long	true	$0.01 * 30 + 0.99 * 75 = 74.6$
short	false	$0.2 * 3 + 0.8 * 100 = 80.6$
long	false	$0.01 * 0 + 0.99 * 80 = 79.2$

And so, the optimal decision is to put on the pads and go the short way (with expected value of 83).

We can also compute the minimum probability of accident ( $p$ ) at which the robot will put on pads by setting the values to be equal. For the short route, this is where

$$p * 35 + (1 - p) * 95 = p * 3 + (1 - p) * 100$$

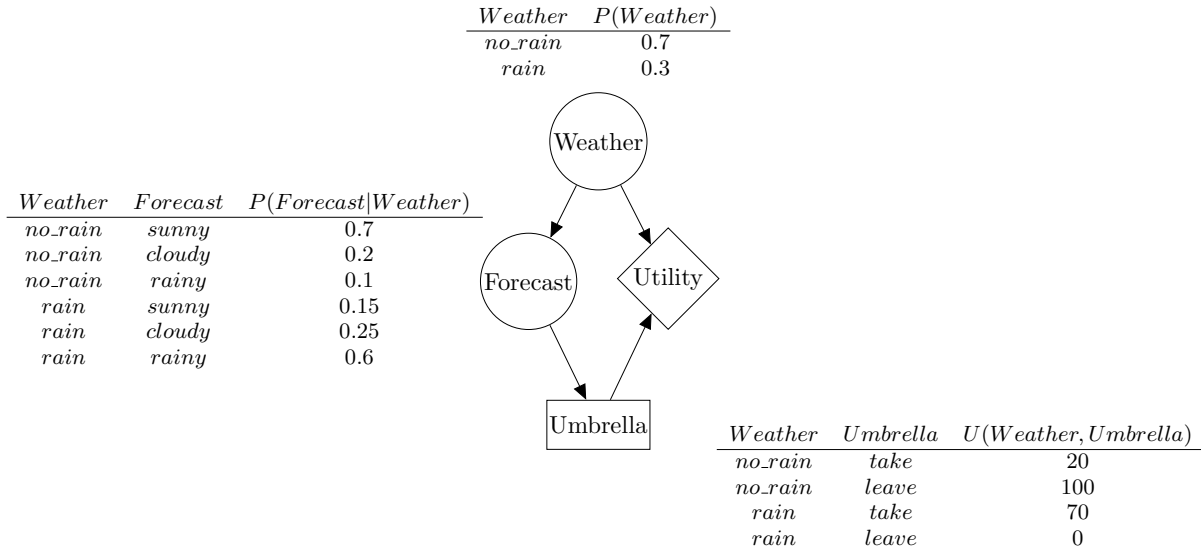
solving for  $p$  gives 0.135. Thus, if  $p > 0.135$ , the robot will put on the pads if going the short route.

For the long route, call the probability of accident  $q$ , we get

$$q * 30 + (1 - q) * 75 = q * 0 + (1 - q) * 80$$

gives  $q = 0.142$ .

*When going out, one must decide whether to take an umbrella or not. This decision is not based on the forthcoming weather, but rather on the weather forecast (which is sometimes wrong). Taking an umbrella on a sunny day sucks because it needs to be carried around all day. Not taking an umbrella on a rainy day sucks even more. The network is as shown below.*



Factors are:

$$f_0(\text{Weather}, \text{Umbrella}) = U(\text{Weather}, \text{Umbrella})$$

$$f_1(\text{Weather}) = P(\text{Weather})$$

$$f_2(\text{Weather}, \text{Forecast}) = P(\text{Forecast}|\text{Weather})$$

Overall, the calculation will be one of

$$\mathcal{E}(\text{Umbrella}) = \sum_{\text{Forecast}, \text{Weather}} P(\text{Weather}, \text{Forecast}) U(\text{Weather}, \text{Umbrella})$$

Using the variable elimination algorithm, we first sum out *Weather*:

$$f_3(\text{Forecast}, \text{Umbrella}) = \sum_{\text{Weather}} f_0(\text{Weather}, \text{Umbrella}) f_2(\text{Weather}, \text{Forecast}) f_1(\text{Weather})$$

Forecast	Umbrella	$f_3(\text{Forecast}, \text{Umbrella})$
sunny	take	$0.7 * 0.7 * 20 + 0.3 * 0.15 * 70 = 12.95$
sunny	leave	$0.7 * 0.7 * 100 + 0.3 * 0.15 * 0 = 49$
cloudy	take	$0.7 * 0.2 * 20 + 0.3 * 0.25 * 70 = 8.05$
cloudy	leave	$0.7 * 0.2 * 100 + 0.3 * 0.25 * 0 = 14$
rainy	take	$0.7 * 0.1 * 20 + 0.3 * 0.6 * 70 = 14$
rainy	leave	$0.7 * 0.1 * 100 + 0.3 * 0.6 * 0 = 7$

Now, we max out *Umbrella*

$$f_4(\text{Forecast}) = \max_{\text{Umbrella}} f_3(\text{Forecast}, \text{Umbrella})$$

Forecast	$f_4(\text{Forecast})$	policy(Forecast)
sunny	49	leave
cloudy	14	leave
rainy	14	take

Finally, sum out Forecast to get the  $f_5() = 77$  which is the value of the optimal policy: take the umbrella if the forecast is rainy otherwise leave it behind. The number 77 gives the return, on average, that this agent will expect to get from following this policy over a number of days.

Now, suppose we observe  $Forecast = cloudy$ . We already know the policy is to leave the umbrella, but what is the expected value? Its not 14 as in  $f_4(Forecast)$  above, as this is the portion of the overall expected value allocated to situations where  $Forecast = cloudy$  (which is not very likely). To answer this query, consider that we have to restrict  $f_2$  to be only over  $Forecast = cloudy$ .

<i>Weather</i>	$f_2(Weather)$
<i>no_rain</i>	0.2
<i>rain</i>	0.25

but this is not normalized, so we need to make it a proper probability distribution first, by noting that we are now trying to find

$$\mathcal{E}(Umbrella, Weather) = \sum_{Weather} P(Weather|Forecast = cloudy)U(Weather, Umbrella)$$

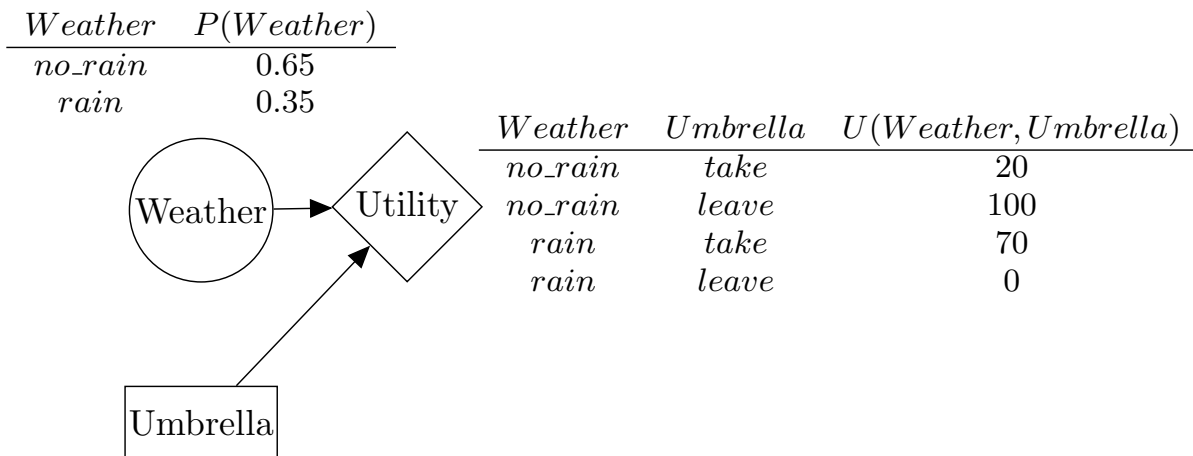
so we have to find the factor  $f_6(Weather)$  corresponding to

$$P(Weather|Forecast = cloudy) = \frac{P(Forecast = cloudy|Weather)P(Weather)}{P(Forecast)}$$

which is

<i>Weather</i>	$f_6(Weather)$
<i>no_rain</i>	$0.2 * 0.7 / (0.2 * 0.7 + 0.25 * 0.3) = 0.65$
<i>rain</i>	$0.25 * 0.3 / (0.2 * 0.7 + 0.25 * 0.3) = 0.35$

What we have done is essentially convert the original decision network into a new one that is restricted for  $Forecast = cloudy$ , so the variable *Weather* now has the interpretation of the weather *given* that the forecast is cloudy.



Now we can sum out *Weather* as before, but with only two factors:

$$\begin{aligned}
 f_7(\textit{Umbrella}) &= \sum_{\textit{Weather}} f_0(\textit{Weather}, \textit{Umbrella}) f_6(\textit{Weather}) \\
 &= \frac{\textit{Umbrella}}{\begin{array}{l} \textit{take} \\ \textit{leave} \end{array}} \frac{f_3(\textit{Forecast}, \textit{Umbrella})}{\begin{array}{l} 0.65 * 20 + 0.35 * 70 = 37.5 \\ 0.65 * 100 + 0.35 * 0 = 65 \end{array}}
 \end{aligned}$$

maxing out *Umbrella* gives us

$$f_8() = \max_{\textit{Umbrella}} f_7(\textit{Umbrella}) = 65$$

and the policy is to leave the umbrella behind and the expected value is 65. This is the actual expected value for all days in which the *Forecast* was cloudy.

If we don't do the normalization first, and simply run the same variable elimination algorithm, we get:

$$\begin{aligned}
 f_8(\textit{Umbrella}) &= \sum_{\textit{Weather}} f_0(\textit{Weather}, \textit{Umbrella}) f_6(\textit{Weather}) f_1(\textit{Weather}) \\
 &= \frac{\textit{Umbrella}}{\begin{array}{l} \textit{take} \\ \textit{leave} \end{array}} \frac{f_8(\textit{Umbrella})}{\begin{array}{l} 0.7 * 0.2 * 20 + 0.3 * 0.25 * 70 = 8.05 \\ 0.7 * 0.2 * 100 + 0.3 * 0.25 * 0 = 14 \end{array}}
 \end{aligned}$$

Now, we max out *Umbrella* to get  $f_9() = 14$  and the policy is to leave the umbrella. To get the expected value, we divide by the  $P(\textit{evidence})$  which is

$$P(\textit{evidence}) = P(\textit{Forecast} = \textit{cloudy})$$

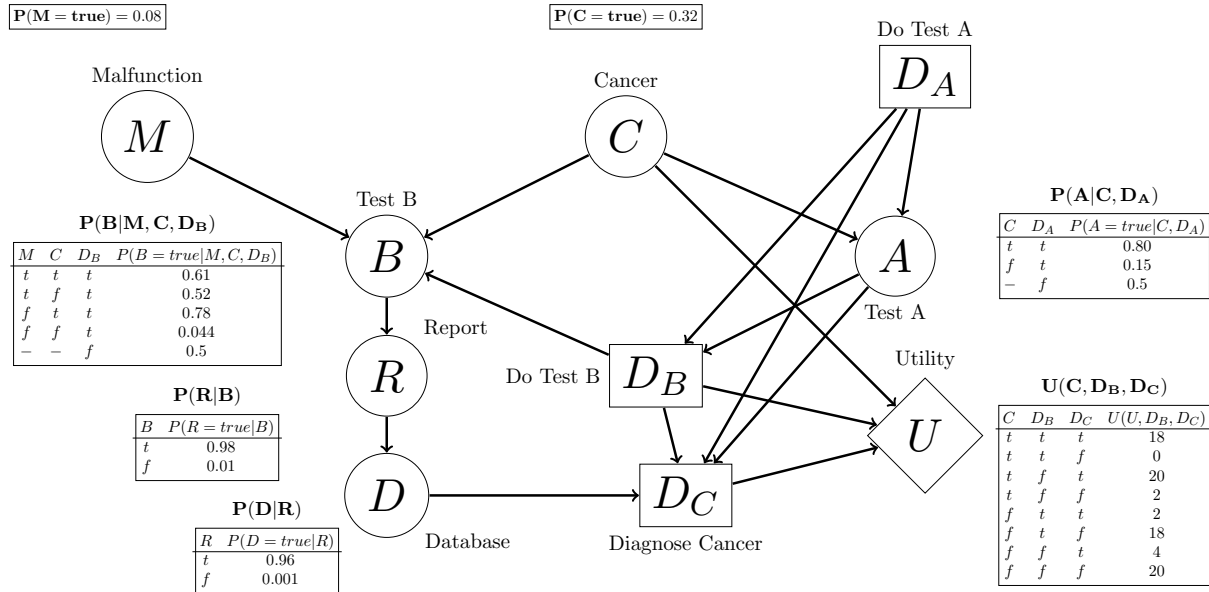
$$= \sum_{\textit{Weather}} P(\textit{Forecast} = \textit{cloudy} | \textit{Weather}) P(\textit{Weather}) = 0.2 * 0.7 + 0.25 * 0.3 = 0.215$$

and so  $f_{10}() = 14 / 0.215 = 65$

Cancer is a disease which occurs with probability 0.32 in a certain population (e.g. older long-time smokers). A Decision network is used to diagnose the presence of cancer ( $C$ ) in such patients by using two binary tests, test A ( $A$ ) and test B ( $B$ ) which report the presence or absence of cancer. Test A is a simple test the doctor can perform directly on the patient and see the results immediately. Test A has a true positive rate of 0.8 and a false positive rate of 0.15. That is, when cancer is present, test A detects it 80% of the time, and when cancer is not present, test A reports it 15% of the time. Test A does not cost anything to administer. Test B, on the other hand, has greater precision (true positive rate 0.78 and false positive rate 0.044), but requires a complicated machine, which malfunctions ( $M$ ) with probability 0.08. When the machine malfunctions, test B's true positive rate drops to 0.61 and its false positive rate rises to 0.52. Test B costs 2 (arbitrary units) to administer. Further, test B's results are not directly available to the doctor. Instead, they are read by a technician who writes them down as a report ( $R$ ) in a logbook, which is then passed to a data entry person who enters the result in a database ( $D$ ). The doctor reads the result from the database. The technician and the data entry person sometimes make mistakes, however. The technician's true and false positive rates are 0.98 and 0.01, respectively, while the data entry person's rates are 0.96 and 0.001 (for true and false positive rates, respectively).

The cost function is defined in terms of a maximum reward of 20 (arbitrary units). The doctor receives 20 for diagnosing cancer correctly (whether cancer is actually present or not). Test B costs 2 to administer, and a false positive diagnosis (saying cancer is there when it is not) costs 16. A false negative diagnosis (saying cancer is not there when it is) costs 18.

The decision network and all conditional probability tables and the utility function are shown here:



We have the following factors:

$$f_0(C) = P(C) = \frac{\begin{array}{c|c} C & f_0(C) \\ \hline t & 0.32 \\ f & 0.68 \\ \hline \end{array}}$$

$$f_1(M) = P(M) = \frac{\begin{array}{c|c} M & P(M) \\ \hline t & 0.08 \\ f & 0.92 \\ \hline \end{array}}$$

$$f_2(B, M, C, D_B) = P(B|M, C, D_B) = \frac{\begin{array}{c|cccc|c} M & C & B & D_B & f_1(B, M, C, D_B) \\ \hline t & t & t & t & 0.61 \\ t & t & f & t & 0.39 \\ t & f & t & t & 0.52 \\ t & f & f & t & 0.48 \\ f & t & t & t & 0.78 \\ f & t & f & t & 0.22 \\ f & f & t & t & 0.044 \\ f & f & f & t & 0.956 \\ \hline - & - & - & f & 0.5 \\ \hline \end{array}}$$

$$f_3(R, B) = P(R|B) = \frac{\begin{array}{c|cc|c} R & B & f_2(R, B) \\ \hline t & t & 0.98 \\ t & f & 0.01 \\ f & t & 0.02 \\ f & f & 0.99 \\ \hline \end{array}}$$

$$f_4(D, R) = P(D|R) = \frac{\begin{array}{c|cc|c} D & R & f_3(D, R) \\ \hline t & t & 0.96 \\ t & f & 0.001 \\ f & t & 0.04 \\ f & f & 0.999 \\ \hline \end{array}}$$

$$f_5(A, C, D_A) = P(A|C, D_A) = \frac{\begin{array}{c|ccc|c} C & A & D_A & f_5(C, A, D_A) \\ \hline t & t & t & 0.8 \\ t & f & t & 0.2 \\ f & t & t & 0.15 \\ f & f & t & 0.85 \\ \hline - & - & f & 0.5 \\ \hline \end{array}}$$

$$f_6(C, D_B, D_C) = U(C, D_B, D_C) = \frac{\begin{array}{c|ccc|c} C & D_B & D_C & U(C, D_B, D_C) \\ \hline t & t & t & 18 \\ t & t & f & 0 \\ t & f & t & 20 \\ t & f & f & 2 \\ f & t & t & 2 \\ f & t & f & 18 \\ f & f & t & 4 \\ f & f & f & 20 \\ \hline \end{array}}$$

Note that for  $f_2$  and  $f_5$  we have written a short-hand version since if the decision is not made (e.g. if test B or test A is not administered, resp.), then the conditional probability becomes uninformative. The same effect could be achieved by adding a third value “*not done*” to variables  $A$  and  $B$ , and then having that value have probability 1 when  $D_A = \textit{false}$  or  $D_B = \textit{false}$ , respectively. In this case, the tables for  $R$  and  $D$  would also need to be changed.

Applying variable elimination to this example, we have three decisions to optimize:  $D_A, D_B$  and  $D_C$ . We first sum out all variables that are not parents of a remaining decision node:  $R, B, M$  and  $C$ . We get the following sequence of factors:

$$\begin{aligned} f_7(B, C, D_B) &= \sum_M f_1(M) f_2(B, M, C, D_B) \\ f_8(R, C, D_B) &= \sum_B f_3(R, B) f_7(B, C, D_B) \\ f_9(D, C, D_B) &= \sum_R f_4(D, R) f_8(R, C, D_B) \\ f_{10}(D, A, D_A, D_B, D_C) &= \sum_C f_9(D, C, D_B) f_0(C) f_5(A, C, D_A) f_6(C, D_B, D_C) \end{aligned}$$

Now, we max out the “last” decision,  $D_C$ , giving

$$f_{11}(D, A, D_A, D_B) = \max_{D_C} f_{10}(D, A, D_A, D_B, D_C)$$

We get our first policy here as  $\delta_{D_C} = \arg \max_{D_C} f_{10}(D, A, D_A, D_B, D_C)$ .

This is a policy for  $D_C$  over  $D, A, D_A, D_B$ .

Then sum out  $D$  since it no longer is a parent of any remaining decision variable, giving

$$f_{12}(A, D_A, D_B) = \sum_D f_{11}(D, A, D_A, D_B)$$

Then, we max out the next decision,  $D_B$ , giving

$$f_{13}(A, D_A) = \max_{D_B} f_{12}(A, D_A, D_B)$$

This gives our second policy as  $\delta_{D_B} = \arg \max_{D_B} f_{12}(A, D_A, D_B)$ .

This is a policy for  $D_B$  over  $A, D_A$ .

Now we can sum out  $A$ , giving

$$f_{14}(D_A) = \sum_A f_{13}(A, D_A)$$

and finally, we max out  $D_A$  to get the final value:

$$f_{15}() = \max_{D_A} f_{14}(D_A)$$

and our third policy is  $\delta_{D_A} = \arg \max_{D_A} f_{14}(D_A)$ . This is a policy for  $D_A$ .



The final policies is then:

```
do test A
if Test A=positive,
    Do test B:
        Diagnose according to Database
else (Test A=negative)
    Diagnose no Cancer
```

and the final expected value is 15.9