

TDT4171 Artificial Intelligence Methods

Exercise 3

Introduction

My model in this exercise determines which drink I will buy/consume based on a set of variables. In this task I decided to include only 6 different drinks to decide between. 6 possible outcomes demonstrate that the problem works without making the workload unnecessarily large. This model contains three different nodes; A decisions node, chance nodes and utility(value)-nodes. All utility-nodes are denoted with a capital 'U' at the start of their name.

Model

What drink to buy – This is the decision node that determines which drink I will buy based on the input from the chance, and utility nodes.

UDrink – Sets a starting preference on certain drinks. For example; usually I like beer better than jeager shots, therefore beer gets a greater score from the beginning.

Exercise due tomorrow – Influences the decision based on whether an exercise is due tomorrow, or not. Is true/false.

Uexercise – Gives a weight of -5 if 'Exercise due tomorrow' is 'yes', and 0 if it is 'no'.

Tasted good – Is only observed after a decision is made. Is true/false. Has the child 'Ask for new drink.'

Ask for new drink – Is only observed after a decision is made. Is true/false. Has the parent 'Tasted good'. This node is a bit unrealistic since I rarely ask for a new drink, but I was running out of ideas for new nodes.

Has Eaten – Sets probabilities for having eaten on the day in question and sets a probability for buying each drink based on that. Contains the states 'little', 'medium' and 'a lot'.

Ueaten – Gives a weight of -5 if 'Has eaten' is 'little', -1 if it is 'medium' and 0 if it is 'a lot'.

Location – Sets probabilities for being at a specific location and sets a probability for buying each drink based on that. Contains the states 'Lyche', 'Wildside', 'Home', 'WorkWork' and 'Cellar'.

Ulocation – Gives a weight of -2 if 'Location' is 'Lyche', -4 if 'Wildside', 0 if 'Home', -3 if 'WorkWork' and 0 if 'Cellar'.

Hangover – Hangover determines whether I have a hangover on the day in question and sets a probability for buying each drink based on that. Contains the states 'no', 'little' and 'very'. This node is conditionally dependent on 'Out night before', 'Is tired', 'Money' and 'Have a job'.

Uhangover - Gives a weight of -5 if 'Hangover' is 'very', -2 if 'little' and 0 if 'no'.

Is tired – Determines whether I am tired on the day in question and sets a probability for buying each drink based on that. Is true/false. This node is conditionally dependent on 'Out night before', 'Hangover', 'Money' and 'Have a job'.

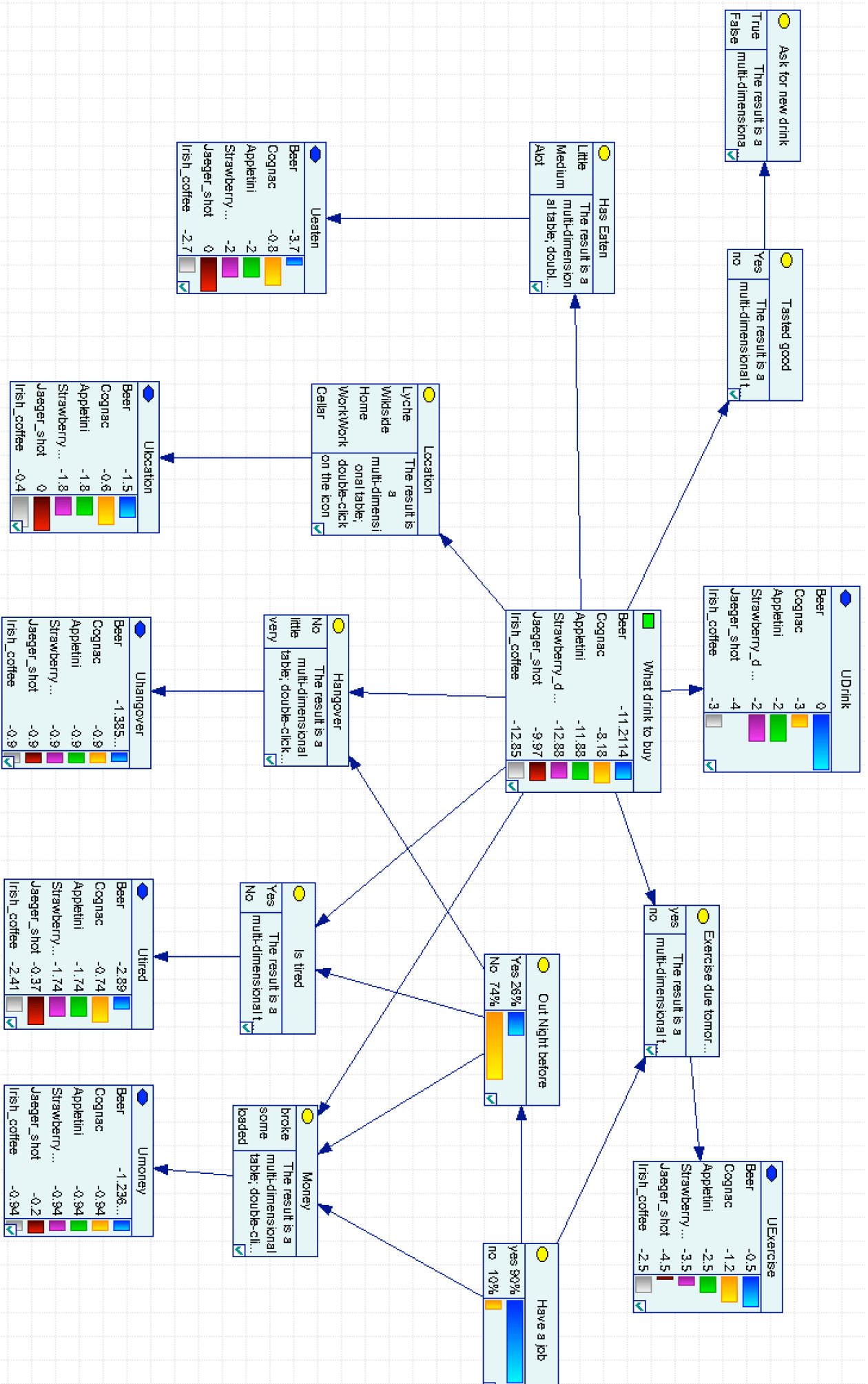
Utired - Gives a weight of -5 if 'Is tired' is 'yes' and 0 if it is 'no'.

Money - Determines how much money I have on the day in question and sets a probability for buying each drink based on that. Contains the states 'broke', 'some' and 'loaded'. This node is conditionally dependent on 'Out night before', 'Is tired', 'Hangover' and 'Have a job'.

Umoney - Gives a weight of -5 if 'Money' is 'broke', -2 if 'some' and 0 if 'loaded'.

Out night before – Determines if I was out the night before and sets a probability for buying each drink based on that. Is true/false. This node is parent to 'Hangover', 'Is tired' and 'Money'. Is a child of 'Have a job'.

Have a job – This node gives the probability for having a job at this current time. Is true/false. Has the children 'Exercise due tomorrow', 'Out night before' and 'Money'. In this model I always assume that this node affects whether or not I have money and if I was out the night the night before.



Assumptions

Conditional independence

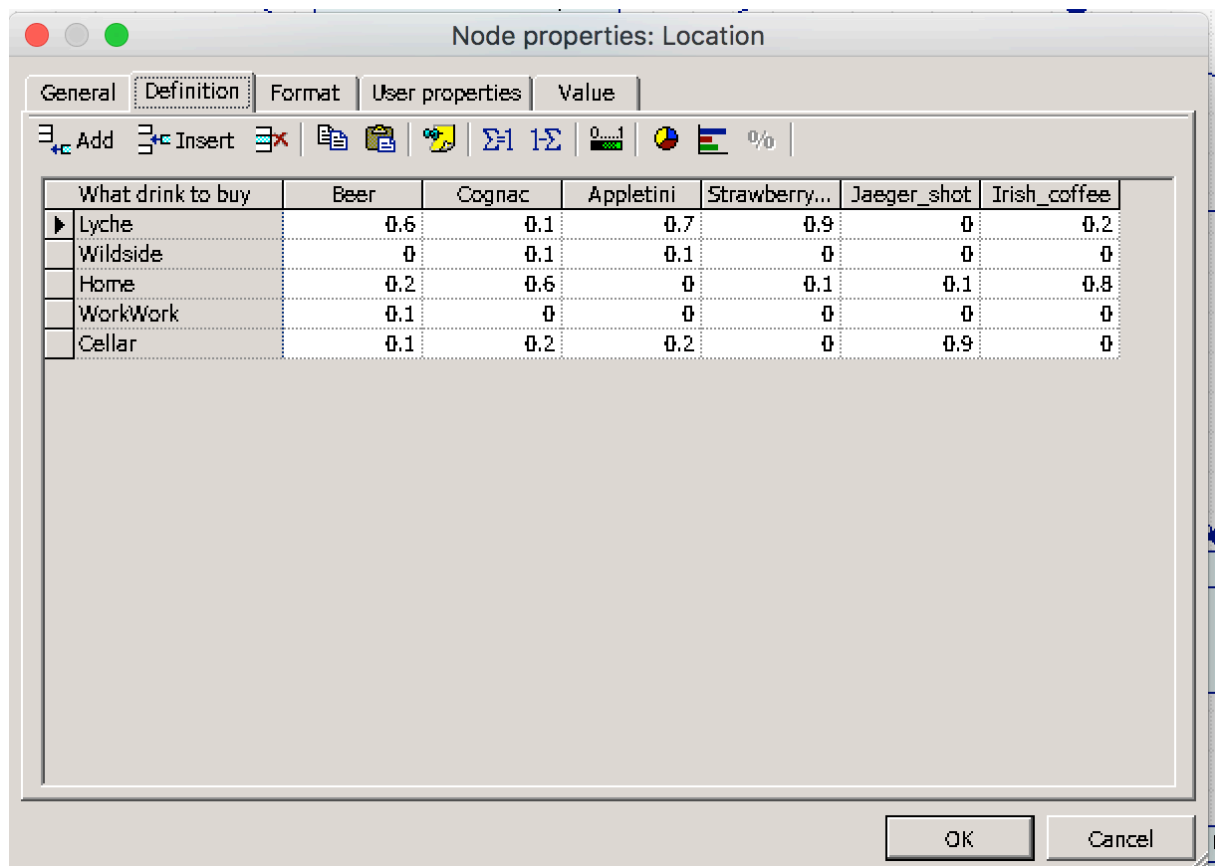
There are some nodes that could be dependant on each other in the real world, such as: 'Has eaten' could be dependant on 'Hangover'. I have chosen not to include this relation since it's hard to come up with a correct ratio at which one affects the other. Some times a hangover leads to eating a lot the day after and sometimes it leads to not eating at all.

The location may depend upon how much money I have, but as of now it is mostly never an issue when we decide where to go.

Structure

I have chosen to have singular utility nodes for the chance nodes. This is mostly just to avoid getting very large multidimensional matrices which would take a long time to edit. I also choose to add initial preferences to each drink with the 'Udrink' node. This was a simple way to say that I, for example, enjoy beer the most in most situation.

Quantifications of uncertainties



Node properties: Location

General Definition Format User properties Value

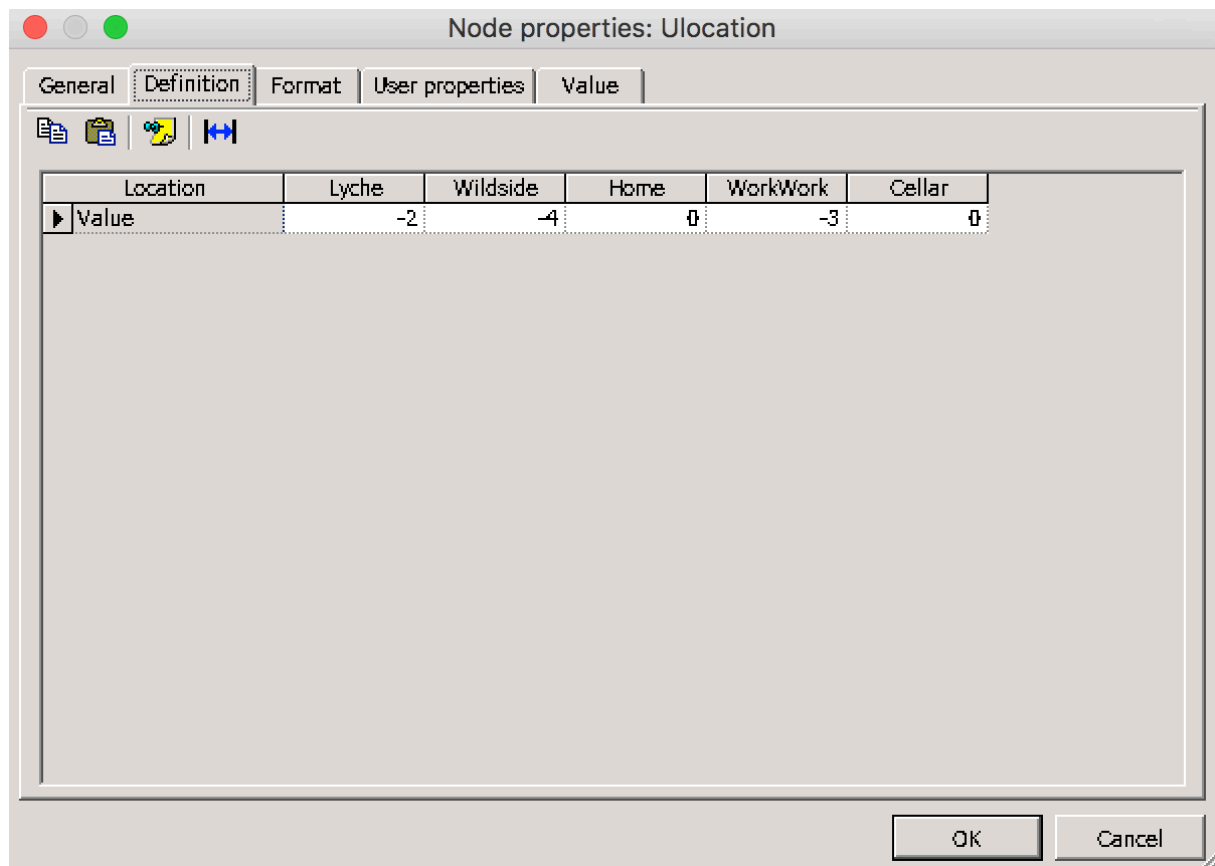
Add Insert X Copy Paste Undo Redo Sum Product 0...1 %

What drink to buy	Beer	Cognac	Appletini	Strawberry...	Jaeger_shot	Irish_coffee
▶ Lyche	0.6	0.1	0.7	0.9	0	0.2
Wildside	0	0.1	0.1	0	0	0
Home	0.2	0.6	0	0.1	0.1	0.8
WorkWork	0.1	0	0	0	0	0
Cellar	0.1	0.2	0.2	0	0.9	0

OK Cancel

The above image is from the 'Location' node. As you can see, the probabilities for each drink are based on the location.

Quantifications of utilities



The screenshot shows a dialog box titled "Node properties: Ulocation". It has five tabs: "General", "Definition", "Format", "User properties", and "Value". The "Definition" tab is selected. Below the tabs is a toolbar with icons for a document, a folder, a magnifying glass, and a double-headed arrow. Below the toolbar is a table with the following data:

Location	Lyche	Wildside	Home	WorkWork	Cellar
Value	-2	-4	0	-3	0

At the bottom right of the dialog are "OK" and "Cancel" buttons.

The above image is from the 'Ulocation' node. The values are based on the taste and cost of drinks at each location. As you can see I have chosen to add negative values starting at 0 and going downwards instead of starting at 0 and going upwards with positive values. This is consistent throughout the model. The choice of the model will be the same as long as the values are correct relative to each other.