Saúl Abimael Gómez Martínez A01700790

Lab 5: Decision trees

This laboratory was all about decision trees where we made a program which would receive the name of the node alongside with its different paths, after this it'll receive the datasets.

For my case I could not finish my code since I got a bit of problems when calculating the information gain of the nodes. Somehow it didn't give me the values I needed for realizing the comparison and obtaining the correct expansion. Although my code wasn't right, I could at least grasp the most important characteristics of both my algorithm and WEKA.

If my program had function correctly, it could solve some problem by showing in the output the way it expands each node depending of each path it has, without any problems. WEKA, in the other hand has some problems displaying the decision tree when there are too little columns or attributes, I think that this happens because it is unable to do something when there are a lot of uncertainty. Basically, my code could avoid this sort of problem of uncertainty.

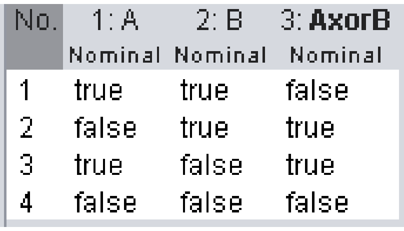
Even though WEKA is not very useful for short problems, it is helpful when dealing with large datasets. This would be a problem for my code which I doubt may commit a lot of mistakes when evaluating a lot of datasets.

Another good thing about my code is that you can change or modify the attributes and datasets directly while WEKA can only modify or add new dataset but not new attributes for this it needs to modify the archive it is using by the time.

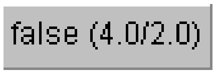
I'd say that the disadvantage of my code against WEKA is the fact that is too specific with this I mean that it needs the input to have a certain format otherwise it might not reach the answer, while WEKA can use different formats or at least types of archives like arff, cvs, etc.

For testing WEKA, I used similar examples as the ones in the test so I could see how it behaves. When I say similar, I mean with almost the same number of nodes or columns, and number of paths. The datasets I used are the next ones with their respective decision tree:

Example 1

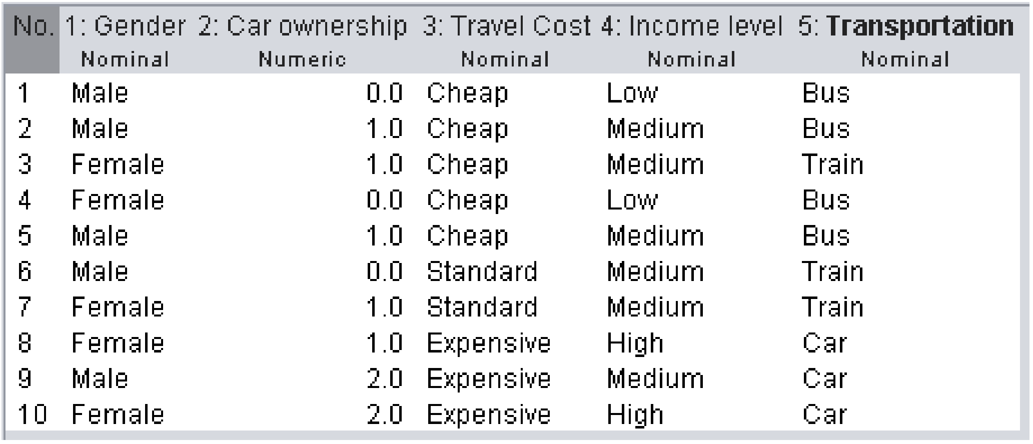


No.	1: A	2: B	3: **AxorB**
	Nominal	Nominal	Nominal
1	true	true	false
2	false	true	true
3	true	false	true
4	false	false	false

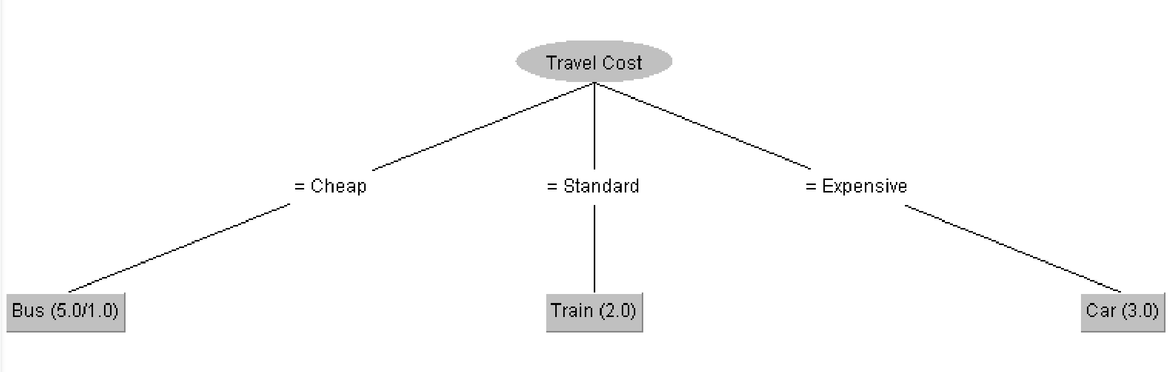


false (4.0/2.0)

Example 2



No.	1: Gender	2: Car ownership	3: Travel Cost	4: Income level	5: **Transportation**
	Nominal	Numeric	Nominal	Nominal	Nominal
1	Male	0.0	Cheap	Low	Bus
2	Male	1.0	Cheap	Medium	Bus
3	Female	1.0	Cheap	Medium	Train
4	Female	0.0	Cheap	Low	Bus
5	Male	1.0	Cheap	Medium	Bus
6	Male	0.0	Standard	Medium	Train
7	Female	1.0	Standard	Medium	Train
8	Female	1.0	Expensive	High	Car
9	Male	2.0	Expensive	Medium	Car
10	Female	2.0	Expensive	High	Car



The difference between this decision trees and what my code would print is very different because as I said before WEKA has some problems dealing with a large uncertainty it stops it from expanding more than it can do, while my code should be able to deal with it.

Another observation of decision tree is that they are almost like the Karnaugh maps for automation which do mostly the same but more tedious since it is based in bits.

Decision trees are a very useful tool for simplifying a problem and defining the most optimal solution, or at the very least only focus on those variables that affect the result. This could be applied in management for evaluating risks or choosing a project.