# CS686 – assignment 2

Name: Erlie Shang

Student#: 20696176

1. see the code attached

2.



|  |
| --- |
| Pr(OC) |
| 0.8 |

|  |
| --- |
| Pr(Trav) |
| 0.05 |

|  |  |
| --- | --- |
| Trav | Pr(Fraud|Trav) |
| True | 0.01 |
| False | 0.004 |

|  |  |  |
| --- | --- | --- |
| Trav | Fraud | Pr(FP|Trav, Fraud) |
| False | True | 0.1 |
| False | False | 0.01 |
| True | True | 0.9 |
| True | False | 0.9 |

|  |  |  |
| --- | --- | --- |
| OC | Fraud | Pr(IP|OC, Fraud) |
| True | False | 0.1 |
| True | True | 0.15 |
| False | False | 0.001 |
| False | True | 0.051 |

|  |  |
| --- | --- |
| OC | Pr(CRP|OC) |
| True | 0.1 |
| False | 0.01 |

1. Run the program that I provide, and the results of each step will be calculated automatically and shown on the terminal.

All the following derivation will be based on the factors list below:

f1(OC) = Pr(OC)

f2(Trav) = Pr(Trav)

f3(Fraud, Trac) = Pr(Fraud|Trav)

f4(Fp, Trav, Fraud) = Pr(Fp|Trav, Fraud)

f5(Ip, OC, Fraud) = Pr(Ip|OC, Fraud)

f6(Crp, OC) = Pr(Crp|OC)

Pr(Fraud)

No restriction because no evidence.

Step 1: add f7(Fp, Fraud) = ΣTravf2(Trav)f3(Fraud, Trav)f4(Fp, Trav, Fraud)

Remove: f2(Trav)f3(Fraud, Trav)f4(Fp, Trav, Fraud)

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | |
|  |  |  |
|  |  |  | |

Step 2:

Pr(Fraud|fp, ~ip, crp)

= aPr(fp, ~ip, crp|Fraud)Pr(Fraud)

= aΣTrav, OC Pr(fp, ~ip, crp, OC, Trav|Fraud)Pr(Fraud)

= aΣOC Pr(~ip|OC, Fraud)Pr(crp|OC)Pr(OC) ΣTravPr(fp|Trav, Fraud)Pr(Trav)Pr(Fraud)

= aΣOC f1(OC, Fraud)f2(OC)f3(OC) ΣTravf4(Trav, Fraud)f5(Trav)f6(Fraud)

= af7(Fraud) f8(Fraud)

= [0.01430783, 0.98569217]

Factor:

f1(OC, Fraud) = Pr(~ip|OC, Fraud)

f2(OC) = Pr(crp|OC)

f3(OC) = Pr(OC)

f4(Trav, Fraud) = Pr(fp|Trav, Fraud)

f5(Trav) = Pr(Trav)

f6(Fraud) = Pr(Fraud)

f7(Fraud) = ΣOC f1(OC, Fraud)f2(OC)f3(OC)

f8(Fraud) = ΣTravf4(Trav, Fraud)f5(Trav)f6(Fraud)

a is the normalizing constant

1. Pr(Fraud|fp, ~ip, crp, trav)

= aPr(fp, ~ip, crp, trav, OC|Fraud)Pr(Fraud)

= aΣOC Pr(fp, ~ip, crp, trav, OC|Fraud)Pr(Fraud)

= aΣOC Pr(~ip|OC, Fraud)Pr(crp|OC)Pr(OC)Pr(fp|trav, Fraud)Pr(trav)Pr(Fraud)

= aΣOC f1(OC, Fraud)f2(OC)f3(OC) f4(Fraud) f5() f6(Fraud)

= af7(Fraud) f8(Fraud)

= [0.00945117, 0.99054883]

Factor:

f1(OC, Fraud) = Pr(~ip|OC, Fraud)

f2(OC) = Pr(crp|OC)

f3(OC) = Pr(OC)

f4(Fraud) = Pr(fp|trav, Fraud)

f5() = Pr(trav) = 0.05

f6(Fraud) = Pr(Fraud)

f7(Fraud) = ΣOC f1(OC, Fraud)f2(OC)f3(OC)

f8(Fraud) = f4(Fraud)f5()f6(Fraud)

a is the normalizing constant

1. If I were the thief, I would prefer to purchase a computer before I made the internet purchase. In this way, the probability of rejection would decrease.

Precisely, the calculation is shown as below:

Pr(Fraud|ip)

= aΣOCPr(ip, OC|Fraud)Pr(Fraud)

= aΣOCPr(ip|OC, Fraud)Pr(Fraud)

= aΣOCf1(OC, Fraud)f2(Fraud)

= [0.00696213, 0.99303787]

Pr(Fraud|ip, crp)

= aPr(ip, crp|Fraud)Pr(Fraud)

= aΣOCPr(ip|OC, Fraud)Pr(crp|OC)Pr(Fraud)

= aΣOCf1(OC, Fraud)f3(OC)f2(Fraud)

= [0.0064889, 0.9935111]

Factor:

f1(OC, Fraud) = Pr(ip|OC, Fraud)

f2(Fraud) = Pr(Fraud)

f3(OC) = Pr(crp|OC)

a is the normalizing constant

Therefore, the probability decreased by about 0.0005

3.

1. No, D and G are connected
2. No, same as above
3. Yes, B is not in the evidence so B blocks A and G
4. No, there are undirected paths between C and G, and B cannot block the path between A and C because B is in the evidence, hence they are dependent
5. Yes, C blocks the path between B and G, so B and G are independent, hence A and G are independent
6. No, there is an undirected path between C and G (C, E, F, G), and B cannot block the path between A and C because B is in the evidence, hence they are dependent
7. No, C and G are dependent because E is in the evidence, and A and C are dependent given B, therefore A and G are dependent
8. C is in the query so it is relevant. And D, the parent of C, is relevant. A is not a descendent of relevant elements, so A is not relevant. E is a descendent, E and all its parents are relevant. So the relevant elements are {C, D, E, F}