**Part I [50 points]: Learning Conditional Probability Tables**

**Task 1 [25 points]**

Run the main method of ‘solution’, with the first parameter as the filename stored under ./data/ and the second parameter as “Task1”. The output file is located under: ./solutions/.

**Task 2 [10 points]**

Run the main method of ‘solution’, with the first parameter as the filename stored under ./data / and the second parameter as “Task2”. This outputs the likelihood and log-likelihood to standard error.

**Task 3 [15 points]** - Comparison of the likelihood and log-likelihood results.

*Please run the program you've written for Task-1 and Task-2 on each training data set.*

Files are stored under: ./solutions/.

*Please write the likelihood and log-likelihood of the CPT for each training data set.*

|  |  |  |
| --- | --- | --- |
| **Filename** | **Likelihood** | **Log Likelihood** |
| cpt-CPTNoMissingData-d1 | 2.8712103624861472 E -11 | -24.27370235297644 |
| cpt-CPTNoMissingData-d2 | 8.626060535385458 E -58 | -131.39514747767186 |
| cpt-CPTNoMissingData-d3 | 0.0 | -Infinity |

*Please explain how the likelihood and log-likelihood measure of the Bayesian Network differs as the number of training data set increases.*

*Please explain how the likelihood and log-likelihood measure of the Bayesian Network differs as the number of variables (nodes) increases.*

*Please write a short discussion on how the likelihood and log-likelihood measure will differ when the possible values of each variable increases.*

**Part II [50 points]: Learning Structure and Conditional Probability Tables**

**Task 4 [25 points]**

Run the main method of ‘solution’, with the first parameter as the filename stored under ./data/ and the second parameter as “Task4”. The output file is located under: ./solutions/.

**Task 5 [7 points]**

*Please experiment with the scoring function by changing the constant parameter. For each parameter, please run the program you’ve written for Task-4 on each data set. For each data set and each parameter, please write the score function of the final Bayesian Network. Please explain how the final Bayesian Network changes as the parameter increases/decreases.*

By using a negative constant in the network, its score will increase proportionately to its complexity. This is because the constant is serving the opposite of its original purpose, and adding the linearly magnified result of the constant and complexity rather than subtracting it. There probably won’t be many situations when a negative constant is used.

A constant of 0 (obviously) results in a score equal to the log likelihood.

Try C-values: 0.01, 0.1, 1, 10, 100, 1000

Well if you have a negative constant you will value complexity very highly. The formula for the score is:

score=log-likelihood−*C*∗*E*

So if you have a negative C you will actually end up increasing the score as the network gets more complex.

**Task 6 [8 points]**

Please implement “no edge” and “random chain” to initialize the structure. Please run the Bayesian Network generation program (Task-4) with these two initialization methods on each data set and compare the final Bayesian Network (in terms of the scoring function and structural complexity) after 3 minutes searching time.

**Task 7 [10 points]**

Please implement the “best tree network” to initialize the structure, and compare the final Bayesian Networks results with the Bayesian Networks generated with “no edge” and “random chain” initialization method (Task-6), in a similar manner as in Task-6.