HW2: Directed Graphical Models

Addis Ababa University
Addis Ababa Institute of Technology
School of Information Technology and Engineering

Course Title: Probabilistic Graphical Models (ITSC-1051)

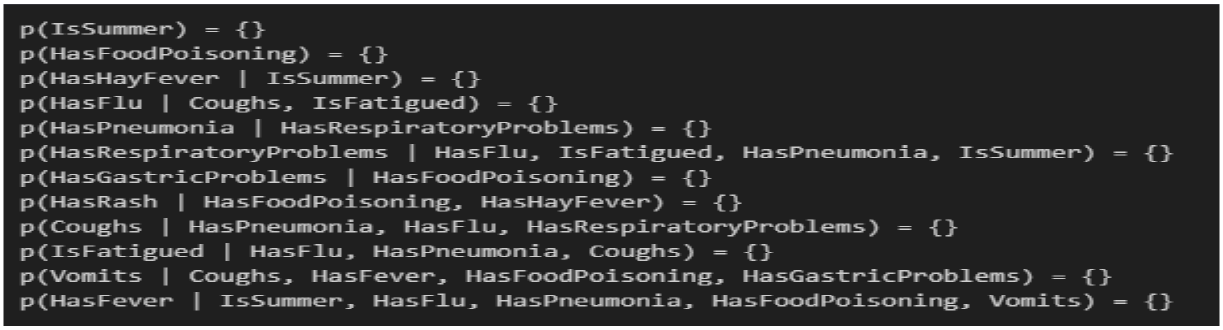
Reported To: Beakal Gizachew (PhD)
Reported By: Mahlet Nigussie

APRIL 21, 2024
Addis Ababa,Ethiopi

INTRODUCTION: This report details the exploration on building a DGM to model relationships between various medical conditions. The model captured these relationships through parent-child connections. It utilizes Conditional Probability Tables (CPTs) to estimate the probability of each condition given its parent states. hen estimates the probability distribution of these conditions and uses the model to answer queries about specific scenarios.

Code Implements: Defining Variables and Relationships, Loading Data, Building the DGM Model, Estimating Conditional Probabilities (CPTs), Model Verification, Computing Joint Probability Distribution, Evaluating Model Accuracy, and Querying the Model. Necessary visualizations are included.

GRAPHICAL MODEL DESCRIPTION: Bayesian Network Model type which represents the relationships between medical variables are connected._Twelve variables are used “IsSummer", "HasFlu", "HasFoodPoisoning", "HasHayFever", "HasPneumonia", "HasRespiratoryProblems", "HasGastricProblems", "HasRash", "Coughs", "IsFatigued", "Vomits", and "HasFever", with True or False possible values and from 0 upto 11 index respectively.The relationships between variables are stored in the parent_map attribute of the DGM class. The conditional probability equations for various variables in the DGM, highlighting the relationships between them:



```
p(IsSummer) = {}
p(HasFoodPoisoning) = {}
p(HasHayFever | IsSummer) = {}
p(HasFlu | Coughs, IsFatigued) = {}
p(HasPneumonia | HasRespiratoryProblems) = {}
p(HasRespiratoryProblems | HasFlu, IsFatigued, HasPneumonia, IsSummer) = {}
p(HasGastricProblems | HasFoodPoisoning) = {}
p(HasRash | HasFoodPoisoning, HasHayFever) = {}
p(Coughs | HasPneumonia, HasFlu, HasRespiratoryProblems) = {}
p(IsFatigued | HasFlu, HasPneumonia, Coughs) = {}
p(Vomits | Coughs, HasFever, HasFoodPoisoning, HasGastricProblems) = {}
p(HasFever | IsSummer, HasFlu, HasPneumonia, HasFoodPoisoning, Vomits) = {}
```

Figure 1: Model structure before calculating CPTs

Here are the brief explanation for having to chose this structure: *IsSummer* and *HasFoodPoisoning*: have no parents, *HasHayFever*: Influenced by IsSummer., *HasFlu*: Coughs and IsFatigued are the causes., *HasPneumonia*: is caused by HasRespiratoryProblems, *HasRespiratoryProblems:* Influenced by HasFlu, HasPneumonia, IsFatigued and IsSummer, *HasGastricProblems:* Caused by HasFoodPoisoning., *HasRash:* Symptoms of

HasFoodPoisoning and HasHayFever., *Coughs:* Can be a symptom of HasPneumonia, HasFlu, or HasRespiratoryProblems., *IsFatigued:* Can be a symptom of HasFlu or HasPneumonia. Or Coughs, *Vomits:* HasFoodPoisoning, HasGastricProblems, Coughs and, HasFever could be cause for vomiting, and *HasFever:* Can be a symptom of HasFlu, HasPneumonia, HasFoodPoisoning, Vomits or IsSummer.

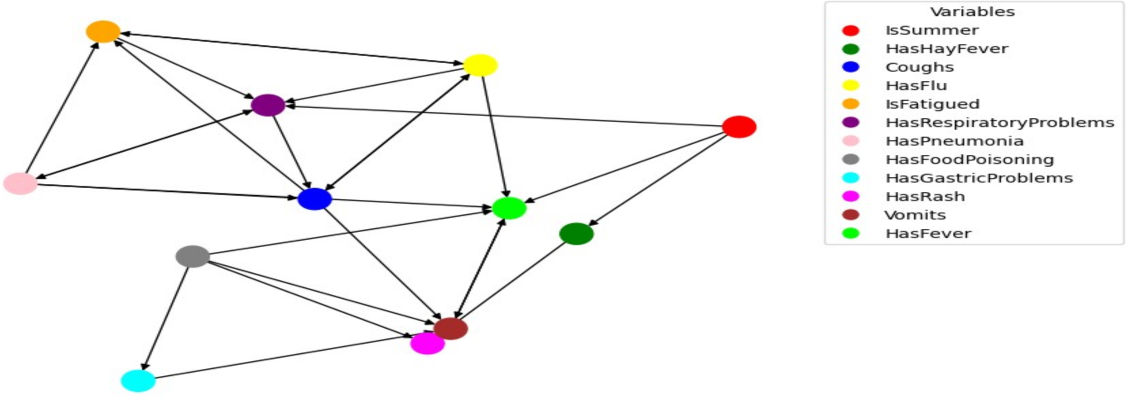
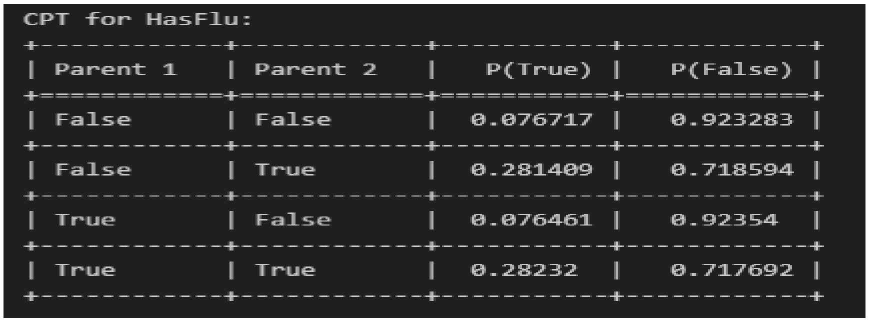


Figure 2: DGM plot

CONDITIONAL PROBABILITY TABLES: estimate_cpts, builds the core probability tables (CPTs) of a DGM model using Pre-defined dataset joint.dat the data set with data shape of (4096, 2). The function iterates through the training data, counting how often specific combinations of parent variable values (True/False) appear with the child variable being True or False. These counts are then normalized to become probabilities for each child variable given its parent settings. This process allows the model to estimate the likelihood of different variable combinations based on the observed data.   After Experimenting I have added smoothing constant to prevent zero probabilities for unseen combinations.



```
CPT for HasFlu:
+------------+------------+-----------+------------+
| Parent 1   | Parent 2   |   P(True) |   P(False) |
+============+============+===========+============+
| False      | False      |  0.076717 |   0.923283 |
+------------+------------+-----------+------------+
| False      | True       |  0.281409 |   0.718594 |
+------------+------------+-----------+------------+
| True       | False      |  0.076461 |   0.92354  |
+------------+------------+-----------+------------+
| True       | True       |  0.28232  |   0.717692 |
+------------+------------+-----------+------------+
```

Figure 3: CPTs for HasFlu

MODEL EVALUATION: After estimating the joint probability distributions for both the true model and the estimated DGM, two metrics to assess accuracy are used: L1 distance and KL divergence. These calculations involve iterating through corresponding probability values in both distributions. L1 distance sums the absolute differences, while KL divergence focuses on the information gain when using the estimated distribution to approximate the true one. In this Experiment splitting the data into half to observe the metrics was done.

Results: L1 distance: 0.5882 and KL divergence: 0.4066 this results indicate that DGM model captures some, but not all, of the complex relationships between the medical conditions in the data. While there are some discrepancies between the estimated and true probabilities (average difference of 0.5882 for L1), the model still reveals some underlying connections (moderate information gain for KL divergence). The values for splitting data are similar for both evaluation metrics.

QUERYING THE MODEL: The query function takes observed variables and query variables as input. It iterates through all possible assignments and considers only those that are consistent with the observed variable values. For each consistent assignment, it calculates the probability using the CPTs. then accumulates probabilities for the query variables across all consistent assignments. Finally, it normalizes the accumulated probabilities to get the probability distribution for the query variables given the observed variables.

- Flu and Symptoms, Pneumonia Symptoms, and Vomiting in Summer explored in this experiment, sometimes yielded unexpected results, This due to limitations in model complexity capturing real-world nuances, the training data's size or quality, or the inherent variability of medical conditions. Besides when the structure is changes to real world symptoms it enhanced.

CONCLUSION: A well-defined structure, Laplace smoothing, and around 70 parameters helped my DGM model relationships between medical conditions, queries sometimes yielded unexpected results. This could be due to limitations in model complexity capturing real-world nuances, the training data's size or quality, or the inherent variability of medical conditions. Future exploration using model selection, improved training data, and Laplace smoothing parameter tuning can potentially enhance the model's accuracy.