# How to learn the Bayesian structure?



Probabilistic Graphical models





**Prevalent Hypertension change over age distribution**

```python
hc = HillClimbSearch(df, scoring_method=BicScore(df))
best_model = hc.estimate()

edges = list(best_model.edges())
model = BayesianModel(edges)

# Fitting the data to the model using Maximum Likelihood Estimator
model.fit(df, estimator=MaximumLikelihoodEstimator)

# Doing exact inference using Variable Elimination
infer = VariableElimination(model)
```

```python
print(infer.query(variables=['prevalentHyp'], evidence={'age': bin_age[0]}))
print(infer.query(variables=['prevalentHyp'], evidence={'age': bin_age[1]}))
print(infer.query(variables=['prevalentHyp'], evidence={'age': bin_age[2]}))
```

```
Finding Elimination Order: : 100%|
Eliminating: BPMeds: 100%|
Finding Elimination Order: : 100%|
Eliminating: BPMeds: 100%|
Finding Elimination Order: : 100%|
Eliminating: BPMeds: 100%|
```

| prevalentHyp | phi(prevalentHyp) |
|---|---|
| prevalentHyp(0) | 0.8445 |
| prevalentHyp(1) | 0.1555 |

| prevalentHyp | phi(prevalentHyp) |
|---|---|
| prevalentHyp(0) | 0.6649 |
| prevalentHyp(1) | 0.3351 |

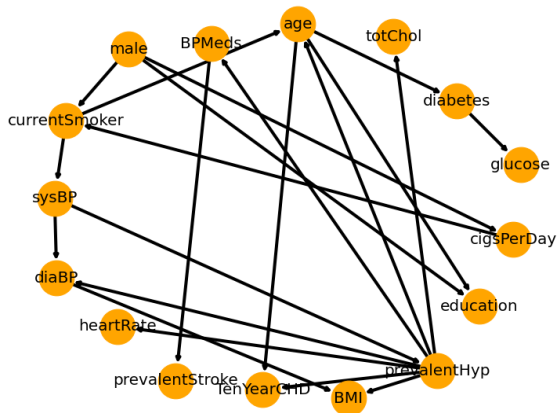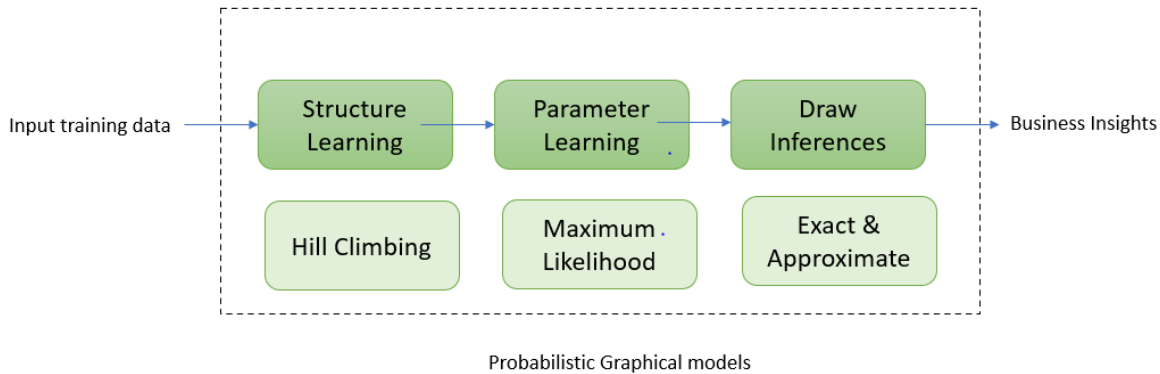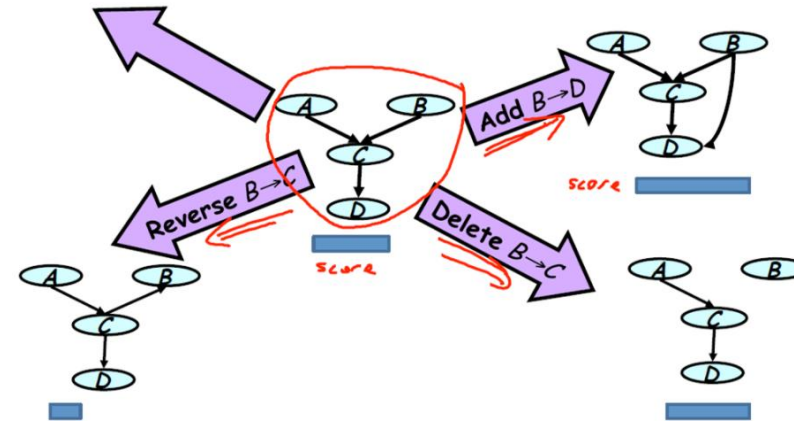| prevalentHyp | phi(prevalentHyp) |
|---|---|
| prevalentHyp(0) | 0.4938 |
| prevalentHyp(1) | 0.5062 |

- Network constructed using pgmpy

- Pre-requisite: discrete features

- Binning Methods: EqualFrequencyDiscretiser (divides continuous numerical variables into intervals that contain approximately the same proportion of observations)

  - Other options: EqualWidthDiscretiser (creates equidistant intervals) and KBinsDiscretizer