Introduction to Bayesian Networks

TJ Machine Learning

Ram Reddy Eric Feng

April 2021

1 Introduction

Bayesian networks are a type of probabilistic graphical model that shows conditional dependence and structure based on random variables. They are efficient and are good for taking an event and predicting what the cause is.

2 Bayesian Statistics

Bayesian statistics are the basics for understanding Bayesian networks. It is the theory in the field of statistics where probability is the expression of the belief of an event happening.

2.1 Bayesian Probability

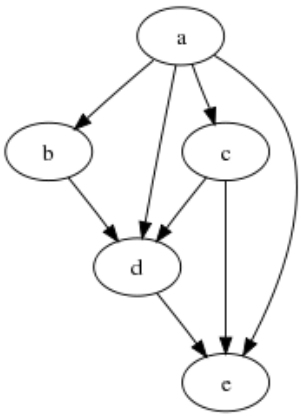
In Bayesian probability there are a couple of things that are useful to know.

- Each variable is denoted by a capital letter like A and B
- Variables can either be discrete or continuous. Discrete variables are something that can be countable like amount of money in a bank account, while continuous variables are something that can be measurable and also have an infinite number of possible values in an interval like the speed of a car.
- The number of possible values in the variable is denoted by |variable name| like $|A|$
- A set of variables is denoted by a bold uppercase letter like $\mathbf{X}$
- The number of variables in set is denoted by $|\mathbf{X}|$
- All variables in Bayesian network is denoted by $\mathbf{U}$
- P(A) is probability of A
- P(A, B) is joint probability of A and B or otherwise noted as P(A U B)

- P(A | B) is the conditional probability of A given B or otherwise noted as P(A ∩ B)/P(B)

2.2 Bayesian Graphs

A Bayesian graph or directed acyclic graph are graphs with no directed cycles. In this type of graph all edges travel down: topologically ordered, and form no closed loops. The graph is structurally specified in a way that each node is represented by a variable. Nodes can hold discrete (gender) or continuous (age) variables. They reflect the knowledge based on the probability distributions of the variables. Below is an example of a directed acyclic graph:



2.3 Bayes' Theorem

The Bayes' theorem is a probabilistic equation that is useful in Bayesian statistics.

$$P(y|X)=\frac{P(X|y)P(y)}{P(X)}$$

Bayes' Theorem can be generalized to find the probability of a class variable(y), given X as the parameters or features.

$$P(y|x_1,...,x_n)=\frac{P(x_1|y_P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

Bayes' Theorem is expanded using the chain rule to include multi-variable data sets.

$$P(y|x_1,...x_n) \propto P(y)\prod_{i=1}^{n} P(x_i|y)$$

Since the denominator is constant, a proportionality is introduced, as well as a Cartesian product in the numerator.

$$y=argmax_y P(y)\prod_{i=1}^{n} P(x_i|y)$$

argmax identifies the class, y, with the maximum probability, which yields the outcome. This equation accommodates multivariate classifications.

3 Implementing Bayes' Theorem

Bayes' Theorem can be used in a Bayesian network, which are less sensitive to small data sets, and are more suitable for environments that are constantly changing, or require time model reconstructions. However, it is more frequently used to generalize models for real-world situations. It can be used in anomaly detection, evaluating the correlation between a piece of data and the desired task, before analyzing it in a CNN or other type of model. It identifies corrupted data or incorrect inputs, identifies uncertainties, and decreases standard variation, while increasing a model's confidence.

3.1 Training

Bayesian networks do not have any weights or biases, so it is only constructed by the given data. There are two methods for construction.

1. **Manual Construction**

   Manual construction requires expert knowledge of the data, so a directed acyclic graph can be constructed from it. This is a knowledge driven procedure and can take a lot of time, however, the results will be very accurate.

2. **Automatic Construction**

   Automatic construction or Data-driven, doesn't require expert knowledge of the data or its causal relationships. It requires no missing data and estimates conditional probability distributions.

   There are three parts to automatic construction:

   - **Inferring unobserved variables**

     The network can update knowledge of the state of its variables when new evidence is observed. New evidence variables, as well as its likelihood updates the prior distribution using Bayes' theorem to create a updated Posterior distribution. This process is called probabilistic inference.

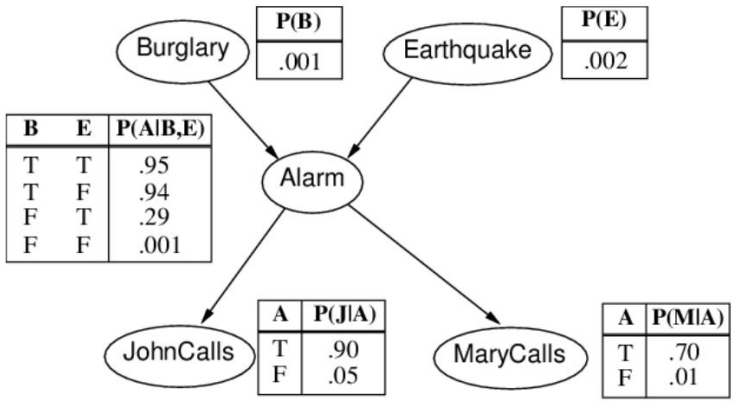
   - **Parameter learning**

     When the parameters (the possible inputs for a variable) are not known it is possible to estimate them. To estimate the parameters, we use the known posterior distribution and try to approximate what the most likely values could be. A common approach to do this is called maximum likelihood estimation.

- **Structure learning**
  Structure learning is the process by which the model evaluates the data and determines the structure and parameters of the local distribution. For simple BNs, the structure is inputted by the expert, but as they grow in complexity, manual specification of causal relationships becomes impossible. Thus, machine learning and other algorithms are used to systematically determine the orientation of causal arrows that link the nodes within the skeleton of a Bayesian network, while maintaining the topographical order.

One example of its application is using text-mining to find causal relationships between nouns within research articles or journals.

3.2 Structure



Bayesian Convolutional Neural Networks are essentially the same as a vanilla CNN, except for the last Dense Layer.

3.3 Implementation

Bayes' theorem can be introduced into any neural network, as stated before, by generalizing models and identifying corrupted data and incorrect inputs. In addition, it can also serve to regularize weights and adjustments in the model, decreasing standard variation and increasing confidence in the model's decision making. This process is called "Bayes by Backprop", introduced by Blundell et al (2015), it replaces Backpropogation in a normal network. Instead of a point estimate of weights, that is, updating weights to a single value to calculate loss, Bayes by Backprop approximates a Gaussian distribution, back-propagating by two hyper-parameters: mean and standard deviation. This allows the model to calculate uncertainty estimates of actions updating the weights and biases, thus

generalizing it. It can be implemented by applying a Gaussian prior, which calculates the probability of an event occurring before the data is collected. This is then used to update the weights through back-propagating.

Here is a code snippet:

```
import numpy as np

LOG2PI = np.log(2.0 * np.pi)

def log_gaussian(x, mu, sigma): # Using mean and standard deviation
    return -0.5 * LOG2PI - nd.log(sigma) - (x - mu) ** 2 / (2 * sigma ** 2)

def gaussian_prior(x):
    sigma_p = nd.array([config['sigma_p']], ctx=ctx)

    return nd.sum(log_gaussian(x, 0., sigma_p))
```

4 Summary

Bayesian Networks are a great way to represent random variables with possible relationships. The network is made up of nodes which are the random variables and these nodes are connected by edges that represent possible relationships. Bayesian Networks attempt to find the posterior conditional probability of variables based on new evidence using concepts from Bayesian statistics like the Bayes' theorem. They can be trained and constructed manually or automatically. Once constructed they can be useful in scenarios like finding the probabilities of diseases based on symptoms.

5 Sources

Click to view sources.