An Evaluation of Bayesian Networks and Gaussian Processes for Statistical Analysis

Omar Ali
Advanced Artificial Intelligence, Department of Computer Science
University of Lincoln
Lincoln, United Kingdom
28587497@students.lincoln.ac.uk

Fig. 1. Output structure generated using Naive Bayes classification for the Diabetes data set

Introduction

This report is an exploration of different statistical methods to evaluate their effectiveness on a set of provided datasets. These datasets include a 'Diabetes' dataset, and a much larger 'Cardiovascular' dataset. Initially, an exploration of different Bayesian Network structures is done and some inference queries are made to test the outcome of these structures. Later, Gaussian Processes are explored where an analysis is done to compare their performance to the best Bayes Net structures.

I. Bayesian Networks

In this section, the goal is to infer probabilistic queries using Naive Bayes Classification (NBC) as well as a set of alternative Bayesian Network structures through an implementation of structure learning algorithms. This analysis opts to implement the PC-Stable[1] algorithm to generate these alternative structures.

Structure Generation

A. Naive Bayes Classification

NBC is a model that assumes all the feature variables are conditionally independent of each other given the desired target variable[2]. This assumptions is the reasoning behind the 'naive' name, however has been found to be very effective in it's own right and is commonly used in application such as language detection and thrives with smaller sample sizes [3]. Using this assumption, it can be said that:

$$P(Cause, Effect_1, ..., Effect_n) = P(Cause) \prod_i P(Effect_i|Cause)$$

The graph for what the right hand side of this equation looks like can be seen for the provided Diabetes data set in Fig. 1.

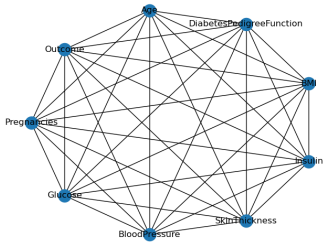

Fig. 2. PC-Stable initialisation: Fully Connected Nodes

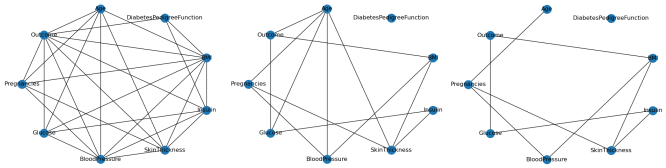

Fig. 3. PC Stable Skeleton Generations: Iterations 0, 1 and 2 respectively

B. PC Stable

The PC Stable algorithm implementation makes use of the 'Chisq'[4] and 'Gsq'[5] Conditional Independence (CI) tests to evaluate if a pair of nodes should remain connected in a given structure. Thresholds for CI tests were set as 0.05 and 0.01. Using theses tests and thresholds, the steps for the Algorithm were as followed:

1) Generated Fully connect nodes:
   Such as in Fig. 2
2) Skeleton generation:
   As described in "Algorithm 4.1 Step 1 of the PC-stable algorithm (oracle version)" in the paper "Order-Independent Constraint-Based Causal Structure Learning"[1], additionally cross referencing the section "Algorithm 1 Skeleton phase of the PC algorithm"[6]
   In Fig. 3 we can see how the edges are being removed per iteration. The implementation of the skeleton generation can be found in the 'pcstable.py', method 'evaluate_skeleton'.
3) Identify V-structures:
   Orientation was originally attempted by identifying collider nodes in the structure and generating di-

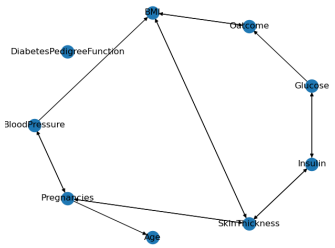

Fig. 4. PC-Stable orientation: Identification of the V-Structures

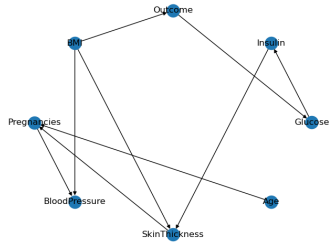

Fig. 5. PC-Stable orientation: Randomised Directional Acyclic Graph

rection through that. The code written in 'pc-stable.py', methods 'evaluate_immoralities' and 'create_directional_edge_using_immorality' were based on the discussion shared in the same paper in the section "Determination of the V-structures"[1] where Node V_k is considered a collider (or immoral) if:
- V_i and V_j have not got an edge
- V_i is dependent on V_j a shared adjacent node V_k
The output skeleton structure for this for this can be seen on Fig. 4. This produced structures with bi-directional edges which is not unexpected, however, the directions can not be interpreted causally and due to time constraints and complexity found through the debugging process, it was decided it would be best to move forward by generating random directional on the edges in the structure, and ensuring that no directional cycles were generated. This is done in 'pcstable.py', method 'randomised_directed_graph', where Fig. 5 is an example output.

Structure Evaluations

For the structures to be evaluated, Conditional Probability Tables (CPTs) were generated for each structure using the 'CPT_Generator.py'.

To effectively evaluate the metrics of each structure, Table 1 was generated to display how well each structure compared with each other, in this case the Diabetes dataset is being used.

The NBC structure had consistent performance, because the structure is fixed. The PC-Stable had randomness in the directionality of the edges, and so had lots of variation in the performance. The 'best structure' was determined by looking for the highest possible 'Balance Accuracy' over a

TABLE I
METRICS EVALUATION ON CONTINUOUS DIABETES DATASET

Metric	NBC	Chi-0.05	Chi-0.01	Gsq-0.05	Gsq-0.01
Bal Acc	0.78	0.79	0.70	0.78	0.76
F1	0.71	0.72	0.58	0.71	0.01
AUC	0.86	0.86	0.82	0.86	0.76
Brier	0.15	0.14	0.15	0.15	0.68
KL	32	35	39	32	0.84
Training (s)	0.0029	0.018	0.032	0.018	0.0034
Inference (s)	0.0009	0.0053	0.0093	0.0053	0.0056

TABLE II
INFERENCE OF QUERIES ON THE DIABETES DATA SET

	Enum		Rej 10^4		Rej 10^5		Rej 200000	
	Prob	Time	Prob	Time	Prob	Time	Prob	Time
Queries	Naive Bayes							
Q1	0.26	0.0033	1.00	0.53	0.29	5.21	0.38	10.59
Q2	0.89	0.0029	0.84	0.51	0.81	5.09	0.81	10.16
Q3	0.32	0.001	0.27	0.57	0.32	5.8	0.30	11.5
Q4	0.24	0.001	0.25	0.49	0.25	4.9	0.24	9.70
	PC Stable Chisq 0.05							
Q1	0.52	0.0029	0.50	0.03	0.90	0.36	0.68	0.70
Q2	0.83	0.0029	0.79	0.29	0.79	2.98	0.81	5.94
Q3	0.25	0.0044	0.25	2.54	0.27	24.7	0.24	49.4
Q4	0.27	0.0028	0.25	0.41	0.32	4.18	0.29	8.60
	PC Stable Gsq 0.05							
Q1	0.31	0.003	0.50	0.43	0.20	4.38	0.17	8.81
Q2	0.86	0.003	0.81	0.72	0.84	7.11	0.84	14.3
Q3	0.28	0.073	0.25	2.17	0.27	23.1	0.27	46.4
Q4	0.28	0.073	0.23	0.83	0.33	7.72	0.34	15.4

set of 10,000 structures, for the Diabetes dataset, per CI test per threshold. From now on, PC Stable Structure naming convention will follow that of 'Chi-0.05' if the test name is 'Chisq' and the Independence Threshold '0.05'.

Chi-0.05 performed the best compared to the remaining PC-Stable structures and had marginally outperformed NBC in all metrics apart from the Training and Inference times where it had a training and inference time of about 6x. Gsq-0.05 also performed quite well, in fact, almost identically to the NBC structure, but required 6x the training and inference time.

Probabilistic Inference

NBC, Chi-05 and Gsq-05 were selected as the structures for inference evaluation due to being the best performers. Inference by Enumeration (Enum) as well as Rejection Sampling (Rej) at samples set to $10,000$, $100,000$ and $200,000$ were used for inference. Additionally, the best evaluated structure methods for Diabetes had also been applied to the the Cardiovascular dataset, where 50 iterations, as opposed to $10,000$, had be run to evaluate the best structures. Based on that, the following queries were inferred:

Q1 - P(outcome=1|glucose=183, bmi=23.3 age=58)

Based on Rejection sampling at $100,000$ and $200,000$, using the NBC and Gsq-05 structure, the most likely result seems to be range between 0.170 to 0.375, however this is an uncertain result due to the other not remaining consistent with this. It is possible there was a large variation of variable properties that we uncharacteristic of each other making each

structure unable to consistently place it hence the unclear result.

Q2 - P(outcome=0|glucose=109, bmi=25.4, age=25)

This has a variance ranging between 0.79-0.89 where all the models are in agreement with one another about the results of the data.

P(target=0|height=164,weight=70,ap_hi=130,ap_lo=90, glucose=1,smoke=0,alcohol=0)

This has a variance ranging between 0.236-0.3175 where all the models are in agreement with one another about the results of the data.

P(target=1|height=164,weight=64,ap_hi=180,ap_lo=90, glucose=1,smoke=1,alcohol=0)

This has a variance ranging between 0 and 0.3372, even with this larger variance, all the models remain in agreement with each another about the results of the data.

II. Gaussian Processes

Gaussian Processes (GP) can be used for Regression and classification tasks as they are a type of probabilistic model that define a prior over functions and are converted into a posterior over functions once we begin to see data being added [7]. In GPs, a parwise evaluation, using a predetermined kernel function, is used in order to generate the covariant matrix and the mean of the provided data. The kernel effectively measures the similarity between these points in order to produce the covariance between the data points. The choice of the kernel will directly affect the covariance matrix and thus leading to a different overall distributions.

In this analysis, two Kernels have been used for evaluation. 1) Radial Basis Function 2) Rational Quadratic [8].

Evaluation

It can be seen in the Diabetes metrics evaluation, the GPs have produced slightly better results than that the PC-Stable structures apart from their Training Times, however, the GP, with both kernels, still performed worse than the NBC with significant differences in the training and inference timing. It is suspected this is due to the smaller dataset. When this is compared against the results of the cardiovascular dataset, the data set was so large that it needed to be reduced to 1/10 of the size to be trained for the GP, however, even with the limited data, there was a much more competitive performance against the Bayes Net structures. The time to evaluate the hyper parameters took quite long, at approximately 20 minutes to determine $l = 1866, sigma_f = 23, noise = 0.4$, where l is the length of scale, $sigma_f$ is the vertical variation of the kernel, and $noise$ which is the suggested noise level and type in the data. However, the overall balanced accuracy returned 0.72 which is is very impressive. This implies that, although it might take longer to train GPs, it is possible to generate very competitive predictions with significantly less data on hand which can be very valuable when data is very limited such as in medical practice applications.

TABLE III
Metrics evaluation on continuous Cardiovascular dataset

	Structure			GP	
Metric	NBC	Chi 0.05	Gsq 0.05	RBF	RQ
Balanced Accuracy	0.64	0.64	0.73	0.72	0.72
F1 Score	0.62	0.61	0.71	0.72	0.71
Area Under Curve	0.7	0.7	0.79	0.78	0.78
Brier Score	0.22	0.22	0.19	0.22	0.21
KL Divergence	4563	4696	3869	6017	5205
Training Time	715	199	273	1232	1128

TABLE IV
Inference of Queries on the Diabetes data set

	Structure			GP	
Metric	NBC	Chi 0.05	Gsq 0.05	RBF	RQ
Balanced Accuracy	0.75	0.71	0.7	0.73	0.72
F1 Score	0.67	0.61	0.57	0..63	0.63
Area Under Curve	0.84	0.82	0.84	0.87	0.86
Brier Score	0.15	0.15	0.15	0.14	0.14
KL Divergence	38.6	41.3	44	45.3	43.3
Training Time	0.8	2.38	1.49	3.21	7.73

In this case, it doesn't seem as though the implemented kernels have had much of a difference in impact. Further investigations would include expanding the training data set to incorporate more data as well as proceeding with an exploration of how changing the hyperparameters would affect the kernels as well as the overall accuracy of the predictions.

References

[1] M. H. M. D Colombo, "Order-independent constraint-based causal structure learning," *Journal of Machine Learning Research*, vol. 15, pp. 3921–3962, 2014. Skeleton phase of the PC algorithm.

[2] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall, 3 ed., 2010. Section 13.5 Bayes' Rule and its use.

[3] S. Raschka, "Naive bayes and text classification I - introduction and theory," *CoRR*, vol. abs/1410.5329, 2014.

[4] "causal-learn: Chi-square test," 2021. Causal learn, documentation.

[5] "causal-learn: G-square test," 2021. Causal learn, documentation.

[6] M. Tsagris, "Bayesian network learning with the pc algorithm: An improved and correct variation," *Applied Artificial Intelligence*, vol. 33, no. 2, pp. 101–123, 2019. Skeleton phase of the PC algorithm.

[7] O. D. J. Görtler, R. Kehlbeck, "Visual exploration of gaussian processes." https://distill.pub/2019/visual-exploration-gaussian-processes/, 2019. Posterior Distribution Section.

[8] Y. Natsume, "Gaussian process kernels." https://towardsdatascience.com/gaussian-process-kernels-96bafb4dd63e, 2023. Imlpemenation of different kernels for Gaussian Processes.