DUJASE Vol. 6 (2) 82-93, 2021 (July) *https://doi.org/10.3329/dujase.v6i2.59223*

A Review on Probabilistic Graphical Models and Tools

Md. Samiullah[1,3*], David Albrecht[2], Ann E. Nicholson[3] and Chowdhury Farhan Ahmed[1]

[1] Department of Computer Science and Engineering, University of Dhaka
[2]Department of Digital Technologies, John Monash Science School
[3]Faculty of IT, Monash University

[*]E-mail: samiullah@du.ac.bd

Received on 09 June 2021, Accepted for publication on 21 September 2021

ABSTRACT

Our daily life is full of challenges, and the biggest challenge is the unpredictability of many of our significant life events. To deal with this unpredictability, analysing the probability of events has become very important. In particular, the theorem of English statistician *Thomas Bayes* has been revolutionary. Numerous theories and techniques have been proposed, and many tools have been developed to solve real-life problems based on the theorem, yet it is still very much an area of active research. It still attracts researchers dealing with cutting-edge technologies. One tool that has been used extensively in modelling probabilistic analysis for decades is the Probabilistic Graphical Model (PGM). PGMs have very challenging childhood but glorious youth. The vast applicability of the models in cutting-edge technologies attracts researchers, modellers and scientists of diversified fields. Hence there are numerous models with their respective features, merits and backlogs. To date, there have been very few surveys conducted among the wide range of models and their associated tools. More specifically, those few reviews are highly application and domain focused, and limited to three to four very popular and widely used models and their associated learning and inference algorithms. To the best of our knowledge, this paper is the first that presents the features, limitations, design and implementation platforms, research challenges and applicability of the models based on a common framework that consists of some essential attributes of the popular PGMs and tools for probabilistic analysis. The study helps deciding an appropriate tool as per the perspective of the application and feature of the tool. This paper concludes with future research scope and a non-exhaustive list of applications of PGMs.

Keywords: Probabilistic Graphical Model, Probabilistic Relational Model, Bayes Theorem, Bayesian Network, Object-Oriented Bayesian Network.

1. Introduction

Probabilistic Graphical Models (PGMs) are one of the classes of probabilistic models where a graph represents the structure of the conditional dependence between random variables. Various PGMs have been used for reasoning under uncertainty for decades [1, 2]. Reasoning is a critical process in real-life applications because of the complex nature and variety of the applications and multiple challenges of uncertainty associated with the events of the applications. PGMs are very effective in dealing with various decision-making challenges due to their ability to present causal relationships among random variables, incorporating dynamic information from a variety of areas with varying degrees of uncertainty, and to perform several kinds of reasoning (e.g., predictive, diagnostic, and intercausal). Moreover, the graphical visualization of PGMs helps specialists with diverse expertise to cooperate easily.

Among the many families of PGMs, one of the very widely exploited and demanding is the Bayesian network (BN) [3, 4]. A BN is a popular, widely used and special type of PGM that represents causal relationships using a DAG and which supports the reasoning for decision-making under uncertainty. In real-life applications, it is very crucial to have the ability of reasoningin presence of uncertainty for decision-making tasks. BN[5, 6] is amighty tool that is able to perform various reasoning in uncertain situations.A non-exhaustive list of the application areas is: monitoring and surveillance, prediction and estimation, classification and groupings of objects, clinical diagnosis, risk anticipation and making effective decisions. Its usefulness as an efficient and effective modelling technology is established by an exceptionally comprehensive range of application areas where they have been applied, including medicine [7], education [8], agriculture [9], ecology and environmental management [10], biosecurity [11], surveillance [12], the military [13], weather forecasting [14] and software engineering [15].

BNs (and PGMs more generally) enable us to document a problem and its current state of knowledge, characterizing the overall model of the problem, as well as behaving like a storehouse of knowledge [16]. The process of creating a BN helps clarify assumptions and identify uncertainties within the system [17]. As information expands, new data can be added to improve the model.

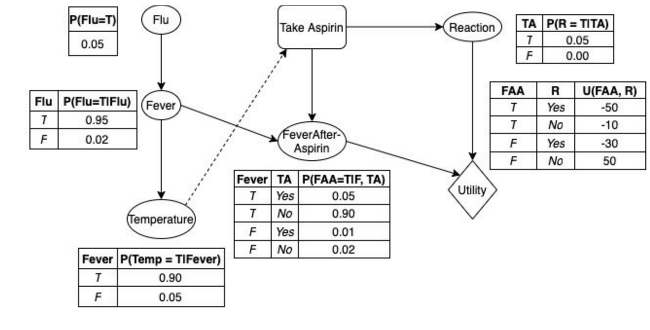
As an example, in Figure 1, a BN, actually a Bayesian Decision Network (BDN), is shown. It deals with a decision-making process during a flu causing fever. Now, to decide whether taking “Aspirin” is going to be worthful or not, the BDN can be used. With five chance nodes (containing conditional probability distribution of the relevant events in the form of Conditional Probability Table), a decision and a utility node (with a utility table representing the utility function to calculate maximum utility in order to make best move/decision). The bottom right table in Figure 1 shows the decisions with respect to no evidence, evidence of "normal temperature", evidence of "high temperature" and evidence of "high temperature with allergic reaction to aspirin", respectively, in its rows. A BDN (and a PGM more generally) can help making best possible decision using its underlying features such as inference, reasoning, probability propagation, and evidence propagation. Moreover, adding experts' opinion in the form of "intervention" is quite easy. As an example, if an expert suggest that the reaction can be reduced by consuming a supplementary, then another node with an outgoing edge from the node to "Reaction" node is enough to model the intervention. This is called knowledge engineering in BDN (i.e., PGM).

Researchers working in this field have started to develop various theories and approaches to overcome the challenges involved in knowledge engineering in PGMs. A non-exhausted list of such techniques is: object-oriented BNs [18, 19], generalized decision graph [20]; BN fragments [21]; and various techniques combining probabilistic relational models and objects (e.g., module networks) [22], probabilistic relational models [48] and plate models [23], multi-entity BNs (MEBNs) [24], Multiply Sectioned BNs (MSBNs) [25], Idioms [26], and Templates [27].

The objectives of the survey are as follows:

- Correlating the models and organizing them, for the first time ever in the literature, in a family tree.
- Analysing and comparing state-of-the-arts PGMs and tools in terms of basic and fundamental features, particularity, limitations, and platforms for design and development.
- Analysing the theories, applicability, scopes and limitations of the popular models.
- Enlisting the key challenges in solving various real-life complex problems using the models and overcoming techniques.

This paper offers a historical perspective of probability and points to the works in the literature relevant to PGMs. It starts with a chronological account: of the necessity of probabilities, how the theory of probability evolved, and how such evolution has led to the development of models for probabilistic analysis. There have been various models proposed by researchers at different times to address different issues and with a particular goal in mind. The models differ in nature, policy, procedure and applicability. Each model has some specific advantages and also limitations. This paper provides a classification of the models in terms of their features, policy, nature, and behaviour. It then goes into the detail of some of the most relevant models. The final section discusses Knowledge Engineering in BN (KEBN) in association with how the PGM tools have been serving the purpose of science and technology from ancient times, how they play an essential role in probabilistic analysis and what roles they play in various sectors of science and technology.



Evidence	Belief(FAA=T)	EU(TA=yes)	EU(TA=no)	Decision
None	0.046	45.27	**45.29**	No
Temp = F	0.018	45.40	**48.40**	No
Temp = T	0.273	**44.12**	19.13	Yes
Temp = T & Reaction = T	0.033	-30.32	**0**	No

Figure 1: Fever Bayesian Decision Network [Korb & Nicholson] [40].

Related Surveys

To the best of our knowledge, there have been very few surveys conducted to date on probabilistic graphical models. The finger countable number of reviews are limited to the applications and analysis of two to three popular models, their associated inference and learning techniques, and their application areas. Among them, some are also research area base reviews of PGMs. Following is a brief discussion on some of such surveys of PGMs:

Authors in [28] promotes the applicability of PGMs and their exact and approximate inference and learning techniques including Expectation Maximization, Gibbs sampling, conditional modelling, and sum-of-product algorithm. They show the application of the aforementioned approaches in vision modelling. The article also contains the comparison of the behavior and performance of the latest inference and learning algorithms using a case study.

Koller, in a book published in 2009 [29] that can be considered as the best ever storehouse of probabilistic graphical models, includes exceptional lists of PGMs ranging from traditional to the then state-of-the-arts PGMs. However, further PGM theories and tools have been developed since then, and noteworthy advancement has taken place in the field.

In [30], the impacts of PGM in AI is highlighted. The authors started from early days of PGM when it was struggling to prove its suitability. Then the principle milestones that PGMs achieved including Pearl's discovery of causal network, prediction, inference and decision-making are explained. It is concluded with some future research challenges and applications of the models. Interesting fact is that the authors themselves didn't claim the work as a complete review of PGM.

A review of PGMs' applications [31] is available in solving complex real-life problems. It is actually a survey of evolutionary algorithms based on PGMs that includes comparison of different methods.

In [32], three widely used types of PGMs, namely Bayesian network, Markov network, and factor graph, are reviewed. The focus was mainly on the basic features of the models, learning and inference related theories, status of the contemporary researches relevant to the models, and the applications of the models.

The objectives of the survey (see Section I) serves the purpose of a more realistic literature survey and eradicate the limitations of the existing surveys. It also guides a researcher or modeler to choose the best potential model to solve a problem.

2. Methodology

2.1 Probabilistic Graphical Models

In order to perform probabilistic analysis, Probabilistic Models (PMs) are used as a prominent tool. While developing various tools to assist modellers with flexibility and ease of representation, researchers developed PMs, namely a representation of a random situation that is defined by its sample space, the associated events and the events' probabilities in mathematical format[33]. PMs can incorporate probability distributions with random variables, where random variables represent the potential outcomes of an uncertain event. The probability of getting any number by rolling a dice, say "1", before throwing the dice can be represented using a variable. Probability distributions assign probabilities to the potential outcomes of the associated events. If the dice is fair, then all six sides have the same probability, that is, 1/6 . Assigning this value to all six variables representing the probability of getting a particular dice face is an example of a probability distribution.

The role of PMs in decision making is to acknowledge the associated uncertainty of the inputs and outputs. That means, in some complex applications, that even the input-taking process and the generated outputs may have uncertainty associated with them. Using a PM allows us to be able to formulate a new model to be more relevant and more appropriate for the complicated situation. The key feature of a PM is that it incorporates uncertainty explicitly in order to understand and quantify risk and to make better management decisions. There are many PMs and different people classify them differently. A snapshot of the taxonomy of PMs is shown in Figure 2 based on summaries in [27, 34, 35].

In order to perform probabilistic analysis, a suitable model is chosen that best express the problem and for getting the expected outcome from the model, a special operation called `Inference' is performed. Inference in a PM calculates the final outcome for the application using the defined approach for the particular model.

A probabilistic model that expresses the dependencies of random variables (conditional dependency, especially) in a graphical structure is known as a probabilistic graphical model (PGM). It offers a framework that helps in visually representing causal dependencies between variables within a set of random variables in order to perform probabilistic analysis. Examples include the Bayesian network and the Markov network. PGMs use graphical representation to encode a complete distribution over multi-dimensional space. The graph represents a set of dependencies of a particular distribution in a compact form.

PGMs are mostly used in applications associated with probability theory, statistics (especially Bayesian statistics), data mining, data analysis and machine-learning. To deal with uncertainty and probability, PGMs seem to be very effective and have become increasingly popular because of their ability to represent the conditional independence between random variables, incorporate dynamic information, and perform both predictive and diagnostic reasoning. Moreover, real-world problems usually need a combination of knowledge from a variety of areas, and this makes PGMs an ideal choice. One of the distinct characteristics of PGMs is their graphical visualization capability, a facility that, helps specialists from different fields to cooperate more efficiently.

Probabilistic relational models[1] (PRMs) [36], first proposed in the dawn of 2000s inspired by the theory of relational database, relational logic programming and algebra. It is also significant that BNs were developed for data with the traditional 2D format in mind. Due to tremendous advancement in the technology, the data is becoming more complex with expanded dimensions and a number of associated attributes [37]. Ordinary BNs are not reliable in modelling applications to deal with such data.

PRMs offer reference slots to establish relationship between instances. These also allow accessing into the class components to facilitate modellers in building large and complex BN applications. However, according to some researchers, reference slots violate the data integrity by breaking encapsulation and data hiding mechanism, the most prominent OO features. Consequently, it throws challenges in decomposing large applications and make it hard to maintain a hefty application developed through a group of modellers.

Bayesian networks

A Bayesian network (BN) [3, 4] is a probabilistic graphical model that (a) compactly represents the joint distribution over a set of variables in the form of conditional probability tables (CPTs), each variable contains one, and (b) represents a set of conditional dependencies and independencies between the set of random variables in the

[1]Note that some authors classify PRMs as directed graphical models. In this paper, PRM has been classified separately because of the difference in the underlying principles of graphical and relational models.

form of a directed acyclic graph (DAG). In a BN, nodes and edges of the DAG represent random variables and their conditional dependencies, respectively. Nodes are connected with an edge or a path consisting of a set of edges representing conditional dependencies. If there is no such edge or path between two nodes, then the nodes are known to be conditionally independent. Also, each node has a CPT attached to it. More details on the BN and its classes are in documented in [54]. BNs can be used to perform reasoning under uncertainty: more specifically, there are techniques to calculate the posterior probability distributions over the states of a subset of the variables, given a set of evidence. A variable does not necessarily represent an event.

The BN, being a graph-based framework, represents the causal relationship among various events (cause to effect) using DAGs. This framework provides ease of understanding and enhances the expressiveness of the dependencies and conditionalization in calculating the likelihood of associated events. The calculation of the likelihood of some hypotheses (state of an event) with respect to a set of evidence can be represented more effectively using a BN.

BNs are very effective in dealing with various decision-making processes, due to their ability to represent causal relationships among random variables, incorporating dynamic information from a variety of areas with varying degrees of uncertainty, and performing both predictive and diagnostic reasoning. Moreover, the graphical visualization power of BNs helps specialists with diverse expertise to cooperate easily [38]. BNs also allow documenting of a problem and its current state of knowledge, characterizing the overall model of the problem, as well as behaving like a storehouse of knowledge [17, 39]. The process of creating a BN helps clarify assumptions and identify uncertainties within the system. As information improves, new data can be added to improve the model.

Reasoning with BNs

The process of thinking in a logical, sensible way to take a suitable action in a right time is known as reasoning. According to Charles Sander's Pierce [40], there are three basic irreducible and indispensable kinds of inference:

- **Deduction**: deriving logical conclusions from known premises
- **Induction**: deducing a universal conclusion from particular premises
- **Abduction**: the only kind of reasoning that helps in introducing new ideas

Handling an uncertain situation and calculating the likelihood of a consequence (conclusion) with logic, statistics, and probability (premises) is known as probabilistic reasoning. An example of probabilistic reasoning is using past situations and statistics to predict the outcome of a future event. Analogously in the case of BN, the reasoning is a special action or operation to draw a conclusion using the premises or evidence provided in the network.

The process of conditioning is performed through a flow of information over the network. The flow is not bound to be directed towards the direction of the arcs. In a BN "information flow" refers to the process of calculating the posterior probability distribution for some query nodes, in presence of some evidence (or observation) nodes values. There are four types of reasoning [41].

- **Diagnostic reasoning**: If the information flows from symptoms to causes. The direction of information flow is opposite of network arcs in this kind of reasoning.
- **Predictive reasoning**: When the information flow is from change in causes to the change in belief of effects. Here the direction of information flow follows the direction of arcs of the network.
- **Intercausal reasoning**: The reasoning that includes the mutual causes of a common effect. If multiple causes of a common effect are initially independent, but in the presence of evidence from any of the causes, other causes are *explained away* and intercausal reasoning takes place.
- **Combined reasoning**: In a BN, any node can become a query node, and any node may contain evidence. In such a situation, none of the aforementioned reasoning types fits well. Thus, a combined approach that fits well, can be used for reasoning. This approach is known as combined reasoning.

Bayesian decision networks

Bayesian networks can be used to make intelligent decisions under uncertainty. To perform such a crucial task, ordinary BNs are extended by adding decision and utility nodes. This extended BN is known as Bayesian decision network.

BDNs have been used for reasoning under uncertainty for decades. The reasoning is substantial in real-life applications because of their complex nature, large span and associated challenges such as the uncertainty of events.

To use the facilities of BDNs, constructing BDNs or modelling in BDNs is the first step. BDNs can be built in four ways [41]:

- Manually with human effort, using experts' opinion and guidance

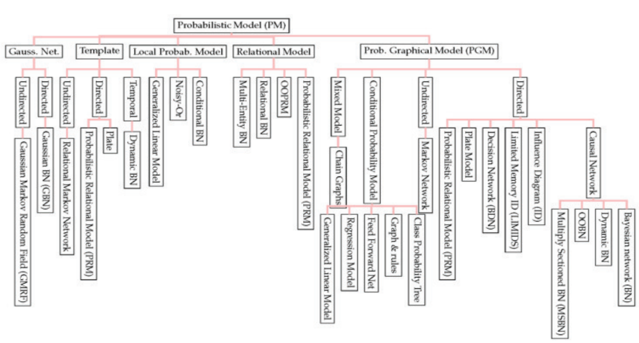


Figure 2: Taxonomy of Probabilistic Model

- From various models found in the existing literature
- By using algorithms to automatically learning from data, given enoughand appropriate data is available, or
- By various combinations of the aforementioned approaches.

Other Related Probabilistic Graphical Models

Researchers have been working to develop various principles and algorithms to scale up the process of BN modelling from long ago. Thus, there has been much effort to improve BNs in many ways over the years. The concept of BN and inference in BN have also been extended to suit particular fields to serve the purpose of probability analysis.

The approaches to extending BNs also include versions of well-known approaches those are for complexity handling, e.g., divide and conquer approach, where dividing the problem into subparts, then solving each sub problem by a BN model and finally merging the models for the sub problems; and reusing a BN segment that is developed previously and validated for some other application. These techniques include Dynamic BNs [42-45], BN fragments [21], generalised decision-graphs [20], OOBNs [46], PRM [36], OOPRM [47], varieties combining PRMs} and objects, such as module networks [22], PRMs and plate models [23], multi-entity BNs (MEBNs) [24], idioms [26], and template-based representations [27].

BNs do not represent temporal relationships between variables. The only way of capturing the temporal relationships between the value of the variables at different points in time (past, present and future) is to add extra variables of the same type but with different names. In real-life applications, it is important to model how the world changes with the change in time from a particular point of time.

Dynamic BNs (DBNs) are an extension of BNs that are capable of modelling changes of probabilities, actions and evidence with respect to time. According to Korb and Nicholson [41], for a BN with n nodes that model a domain of n variables $\boldsymbol{X} = \{X_1, \dots, X_n\}$, the DBN that models the change of values in the variables over time should contain one node for each X_i for each instance of time. For a current time instance t, the immediate past time instance and immediate future time instance are represented by t-1 and t+1, respectively. Hence, the nodes for DBN to model such a temporal system is:

- Current (in time t): $\{X_1^t, \dots, X_n^t\}$
- Immediate past (in time t-1): $\{X_1^{t-1}, \dots, X_n^{t-1}\}$
- Immediate future (in time t+1): $\{X_1^{t+1}, \dots, X_n^{t+1}\}$

Each time instance is called a "time-slice", and two new types of arcs are introduced, namely "Intra-slice arcs" and "Inter-slice or temporal arcs". As the names suggest, the arcs that represent a temporal relation between variables within a time-slice are intra-slice arcs and the arcs between the variables of successive time instances are inter-slice arcs. The latter type can be between the same variables or between different variables at successive times. These arcs deal with the effect of the change in one variable in time t on the other variable of time t+1.

In recent years, with the growing interest of relational pattern extraction, **Relational BNs** (RBNs) have gained much importance. These are an extension of ordinary BNs that allow relational data representation [37]. This model was called the "Probabilistic Relational Model" (PRM) and was first proposed by Koller and Pfeffer[48]. The concept of objects (as opposed to random variables and their attributes in BNs), objects' properties, and relationship between objects are at the core of this model. It has a similar different relation to BNs as relational logic has to propositional logic. The model specifies a template to represent the probability distribution for a particular database. The template comprises of a relational component and a probabilistic component where the former one is for describing the relational schema and the later is for describing the probabilistic dependencies [49].

Later, Nevile and Jensen [50] proposed the term "RBN" to refer to the BNs that can model relational data, i.e., the "PRMs" as proposed by Koller. They also proposed the use of the term "PRM" for the type of PGMs that mines statistical patterns from relational data rather than extended BNs that model relational databases. Note that there is another kind of RBN, proposed by Jaeger [51]. This is entirely different from the PRMs (renamed as RBNs) proposed by Koller. Jaeger's RBN is an extension of the BN using First-Order logic (FOL). For representing probabilistic relations, it is more powerful and expressive than the ordinary BN, as it uses a powerful FOL in contrast to propositions. Using FOL allows the adding of constraints on the equality of events, defining complex, nested functions and a recursive network.

Gaussian Bayesian networks (GBNs) [51] are a particular type of BN where all of the variables are continuous, and all of the CPDs for the variables are linear Gaussians. It is used to define a continuous joint distribution and provide an alternative representation for a multivariate Gaussian distribution class [36].

Multiply-sectioned BNs (MSBNs) [25] were proposed with the localisation of queries and evidence in mind. By localisation, the authors meant that at a particular point in time, queries are directed towards a part of the whole network. A point to note is that the original formulation of BNs (ordinary BNs) do not consider the structure in the domain and the whole network is treated as a homogeneous network of the variables under consideration. In such a system, probability propagation for inference is inefficient since, for localized evidence, the whole network needs to be updated. MSBN offers a localisation-preserving partition of a BN by allowing a set of separate Bayesian sub-networks. These sub-networks are transformed into a set of permanent JTs such that evidential reasoning can occur in any one of them at any time. This also ensures that the calculated marginal probabilities are the same as if they were calculated in the homogeneous network.

The **Multi-entity Bayesian network** (MEBN) [24] is a first-order language for modelling uncertainty using first-order logic language. It combines BNs with FOL to provide BNs with the power of first-order expressiveness and uses FOLs as the means of modelling probability. The MEBN specifies parameterized fragments of BNs (a.k.a. MFrags) to express probabilistic relationships and dependencies among a small collection of relevant and correlated hypotheses in order to form a probabilistic knowledge base. A set of instantiated, combined MFrags form a graphical probability models having an arbitrary degree of complexity. An MFrag can be instantiated any number of times, and that enables an MEBN to express complex graphical models with redundant structures. Hence, MEBN is a compact language capable of representing knowledge at a finer level of granularity. Like BNs, the MEBN also uses directed graphs to define joint probability distributions.

Fragments, proposed by Laskey and Mahoney [21], are large-scale BN construction schemes where knowledge is specified in larger and meaningful units, called "fragments". A fragment is a set of interrelated random variables that is constructed and used for reasoning separately from the others. The OO concepts are used to represent and manipulate fragments. In fragments, input variables specify interfaces and so-called "resident" variables encapsulate private data. Authors emphasize on network composition rather than network construction. In fact, any vast network can be constructed from non-decomposable small units. If any method can pre-compute and store them, then computation for a new but slightly different (perhaps larger) network construction need not be started from scratch. The framework allows for representing asymmetric independence and canonical intercausal interaction.

Idioms: Although fragments [21] provide the ability to solve real-world, large-scale problems by providing methods for defining component-level BNs and combining them into a consistent model, knowledge engineers still need a guide to adopting past inference solutions to current problems. Inspired by design patterns, Fenton [26] described a solution to these problems based on the notion of generally applicable "building blocks" that can be combined into objects: they named these "idioms".

By combining the idea of idioms, some large-scale problems may be addressed and solved. However, there are some common problems of patterns in software engineering and Idioms in BNs. These are:

- There is no guarantee that patterns are suitable enough to model all real-life applications.
- It is hard to find appropriate patterns/Idioms due to overlapping segments among patterns.
- Some complex real-life applications require more than one pattern which may lead to undesirable overheads.

Sub-networks: GeNIe[52] supports another specialised form of BNs, i.e., sub-networks. Sub-networks cannot be characterised as classes in OOBNs. It allows building a hierarchy of embedded networks and the hierarchy needs manual maintenance. In the case of any change in one sub-network at any level of the hierarchy, all other sub-networks in the lower levels connected to this sub-network need manual changes. Moreover, if a modeller embeds multiple copies of the same sub-network, they have to make changes in each of the instances.

Object Oriented BNs (OOBNs)[46], incorporates OO features in BNs to resolve the so-called "scalability" problem by ensuring reuse of existing and previously defined components. A segment of a BN is encapsulated into a class (a blueprint of an object) with an interface of input and output nodes. The segment can be used in other models and in larger models by making a copy (called an instance) and adding connections only to the interface node, via connections only to the input and output nodes. Such reuse helps save design time and reduce complexity. It also supports maintenance, as any changes to a class (for example, if the CPTs are updated using new data) can be automatically applied to all the instances and its subclasses. It is a great approach to utilize the OO features, such as encapsulation, abstraction, inheritance, and polymorphism, in BN arena to provide various facilities to the modellers. A more complete and formal presentation of OOBNs, is given in Chapter 2 of [54].In **Template**-based representation, a PGM specifies joint distributions over a specified group of random variables. This group of variables and the distribution can be used in many different applications and problems. For example, a student performance observation network can be applied to multiple students. Basically, all the students share the same structure -- the components in this structure can be viewed as attributes. Only the attributes' variables differ between students. Koller and Friedman [27] called this model "variable-based" because of the focus on the presentation on random variables.

Koller and Friedman [27]proposed a general framework that defines templates for fragments of the model for probabilistic analysis. Templates can be reused both within a single or across multiple models having divergent structures. The two template-based representation languages that can be applied to the theory of OOBNs are Plate models and PRMs [23].

Object Oriented Probabilistic Relational Model (OOPRM):

An alternative to OOBNs is Probabilistic relational model (PRM) [36].It was first proposed in the early 2000s, motivated by the theory of relational databases, relational logic programming, and relational algebra. The OOPRM [47] lengthens the PRM framework (a relational model without OO-features) by introducing the principles of Object-Orientation(that is, interfaces, inheritance and polymorphism). Compared with the PRM, it has another special feature, i.e., the inverse reference slot, which helps in more efficiently accessing the attributes of classes. However, an interesting fact is that the OOPRM has an issue same as the PRM, i.e., reference slots. While OOPRM has an inverse reference slots also that violate encapsulation, and thus OOPRM becomes hard to extend, alteration, decomposition, and decoupling. Moreover, in an OOPRM, the chain of reference slots and inverse reference slots make the process of changing as per the requirement and reuse of existing components far more complicated because the model developer needs to know the details of how a class is embedded and how it impacts in the whole system when making even a trivial modification. In addition, while the PRM and OOPRM frameworks allow the effective representation of relationships and dependencies between classes that are instantiated with multiple instances, they do not allow the utility and decision nodes that facilitate BNs (BDNs and OOBNs) to be used for decision-making under uncertainty and utility computation to make the best available decision.

Finally, while the OOPRM has been implemented as a part of a research software tool, AGrUM [53], it is not available yet in any commercial modelling tool or more specifically not openly and widely available, and there seem to be very few real-world OOPRM models described in the literature to the best of our knowledge. *Table 1* demonstrates a comparative study of the probabilistic models described so far, in terms of their significant features, limitations and a suggestive model to overcome the limitations.

Table 1: Comparing popular Probabilistic Models

	Features	Limitations	Overcoming Techniques
Ordinary BN	Represents knowledge using propositions and probability Distributions	Scalability	OOBN, Templates, Fragments, Idioms, PRM, OOPRM
		Time-slice representation	Dynamic BN
		Repeated structure	OOBN, Template, Idioms, Fragments
		Dealing with localisation of query	MSBN
		Expressiveness	MEBN, Relational BN
Dynamic BN	Modelling change in uncertainty with respect to time	Requires extra nodes to deal explicitly with time steps	OOBN, OOPRM
		Scalability	OOBN, OOPRM
Relational BN	Extends BN by adding First Order Logic, facilitates defining nested and complex functions	Same as BNs except expressiveness	OOBN, Templates,Fragments Idioms, PRM, OOPRM
MSBN	(1) Facilitates localized query (2) Avoids probability propagation throughoutwhole BN	(1) Does not provide maximum reusability such as OO definitions (2) Same as BN except scalability	OOBN, OOPRM
MEBN	(1) Provides a First-order language for modelling application under uncertainty (2) Defines MFrag to allow repeated structure	Does not provide maximum reusability like OO definitions Same as BN except for expressiveness Decision and Utilities are not supported	OOBN, OOPRM
Fragments	(1) Offers fragment (non-decomposable unit)as a set of variables (2) Preliminary Idea of OOBNs	Not a complete OO-system andpast inference resultscannot be used	OOBN, Idioms
Idioms	Offers compositional probability modelling A large system can be clustered into Idioms and represented by a setof Idioms to solve it by combining the solutions	(1) All problems may not bedecomposableby Idioms (2) May lead to undesired overheads	OOBN
Templates	Offers a special set of random variables, called template. Template is a general form of an analogous segments in a model and helps in defining a common solution	All the features of OO-paradigm such as inheritance, encapsulation, and polymorphism are not defined	OOBN
OOBN	Introduces OO features (inheritance, encapsulation, polymorphism) to BNs	Recursive definition not allowed	OOPRM
PRM	(1) Extends BNs by introducing classes and defining the relations of attributes by reference slots (2) Models relational data by relational logic	Reference slots violate encapsulation	Pure OO-notion
		Dependency added by reference slots makes extension, maintenance and decomposition difficult	
		Only random variables are supported	
OOPRM	(1) Introduces OO features (inheritance, encapsulation, polymorphism) toPRMs (2) Inverse reference slot and interface	Complex representation	OOBN
		Only random variables are supported	OOBN
		Reference and inverse slots violate encapsulation	Pure OO-notion
		Maintenance, decomposition and extension is difficult	Pure Encapsulation and abstraction

2.2 Software for Probabilistic Models

There are numerous implementations of various Probabilistic Graphical Models. Each has some specialty and a list of features. Some are free, some are open source, and some are commercial. Some support BNs (directed models), while some support relational or undirected models. They also differ in underlying methodologies, mechanisms to perform inference, learning, and making decisions. Some of the tools support particular programming languages and are compatible with various operating systems. Each serves a particular purpose and has limitations for some features as opposed to others. Some of the tools support GUI+API, some offer only API, or only programming languages, and some offer query languages.

Table 2 compares the critical features of some popular BN tools, while *Table 3* does the same for some popular relational modelling tools.

3. Applications and Research Directions

Applications: As we have seen in Section I, how knowledge engineering using PGM can be helpful. The example is a simple one with lots of real-life factors suppressed. However, in practice, PGM are more powerful and capable of dealing with lot more complicated scenarios with thousands of parameters and events. The models and tools can be used to develop various intelligent applications and deploy in critical sectors of our country where decision making is very tough, full of challenges, uncertainty associated, risky and crucial. Some of the sectorsare fraud detection in Banking sectors, gamblers detection in Stock exchange, risk anticipation in investment on a particular project or business, diagnosis in Health sector, crime investigation in law and order, cybercrime detection and prevention, national defense and intelligence, weather forecasting, disaster prediction, and risk anticipation. The scope is briefly explained in the following points:

- Health sector is another very important sector where making a decision is very crucial and harder. It involves numerous factors interlinked and overlapping where conditional probability plays vital role. PGMs are great tools to deal with such scenarios.
- Another very important sector for any country is its defense. In order to anticipate potential threats of its security, lots of closely related factors need to be considered simultaneously. Quite a lot of computation is required to compute risk and risk factors where probabilistic graphical models are really helpful as experts from various fields can easily collaborate and share their knowledge in modelling a solution using the models. The expert elicitation and documentation facility help greatly.
- Crime and cybercrime investigation, analysing potential sources and epicenters of crimes, diminishing crime by destroying its root cause/s, monitoring law and order situation, and computing risks of various actions taken by the Law-and-order agencies are quite sensitive and vital for a country and nation. Sometimes these factors impact international relations too. PGM and its tools are very handy for such analysis.
- A terrifically uncertain and very hard to predict area is weather especially for coastal areas. Large number of factors are associated with it and there are numerous hidden factors making this sector more vulnerable. Hence, sensitivity analysis, hidden factor and hidden variable finding, incorporating experts' opinion and adding interventions dynamically are required. These are all together available in PGMs.
- Furthermore, predicting natural disasters, risk analysis in terms of loses, damages and life threats can be predicted using PGMs. It is beyond the scope of this paper to list all possible applicable fields where PGMs can be used. Nevertheless, it can be a part and parcel of the development plan for our country like other modern countries have already deployed it extensively in as many fields as possible.

Research Directions: Though PGMs are matured modelling tools, no single class of PGM is self-contained. They have their own features and specific limitations. Hence, one easy research direction would be to eradicating the limitations of a particular modelling method or tool. To be more specific, numerous works have been done to address the scalability issue of Bayesian Networks, such as OOBN [18], PRM [36], and OOPRM [24]. These frameworks are not complete and lots of scopes available there to contribute. Automated BN structure learning is another very important research topic. Though several landmarks are added, such as IC algorithm [55] and CaMML [56], in this field but they are not sufficient and automated structure learning for OOBNs and Dynamic BNs are yet to be discovered. Efficient compilation technique for BNs, OOBNs, for BNs with continuous variables, for BDNs are still very demanding fields of research in this field. Knowledge engineering in BNs (KEBNs) [40] and OOBNs are not studied extensively yet. A API type library for BN and OOBN repositories are the urges of time. Because, build once and reuse repeatedly is the moto of knowledge engineering in BNs which is yet to be explored extensively.

Other PGMs such as OOPRMs and PRMs need special care in developing better explanation mechanisms to make them easy to understand and visualise by researchers of other domains. To gain wider acceptability of the models, the reference slots and inverse-reference slots need alternate representations or alternatives. This can be a very interesting research problem. MSBNs, Idioms and Fragments can be better utilized to perform KEBN processes. MSBNs and its underlying techniques can be used to develop better compilation techniques for OOBNs and OOPRMs.

In conclusion, probability and probabilistic analysis are very important in cutting edge technologies. From cloud computing to quantum computing, from neural algorithms to genetic algorithms, and from pattern mining to classification and clustering tasks, probabilistic analysis and probability calculus play very important role. Introducing PGMs with such tasks results in greater solutions. Numerous examples are available in the arena of Information Technology, Engineering, Mathematics and Statistics, Science and Applied Science. Another set of directions and overview is available in the surveys on PGM done in [57-59].

Table 2: Comparison of popular BN tools

Tool name	Commercial / Open source	GUI / Programming	OO-features Supported	Inference Feature	Learning Feature	Continuous Data Support	Languages supportedfor API	Well Documented in English	OtherFeatures
Hugin	Commercial	GUI	Yes	Exact and Approximate	Learning BNs with missing data	Yes	C, C++, Java, .NET, ActiveXserver	Yes	Sensitivity Analysis
Netica	Commercial	GUI	No	Various Inference	Learning with missing data	Discretized Continuous Data	C, C++, C# Visual Basic, MatLab, CLisp	Yes	Sensitivity Analysis
BayesiaLab	Commercial	GUI	No	Exact and approximate	Parameter learning by maximum likelihood. Structurelearning	Discretized Continuous Data	Java	Yes	Supervised and unsupervised learning
ProbaYes/ ProBT	Commercial	Structured Programming Language	No	Exact and approximate	Parameter and Structure with missing value	Yes	C++, C#, Java, Python, Excel plugin	Yes	DBN, HMM,
Genie	Commercial	GUI	Yes~	Exact and approximate	Parameter and Structure	Yes	C++, Python, Java, .NET, MS Excel	Yes	WebbrowserandMobiledevice
UnBBayes	Open source	GUI	Yes	Exact Inference	Learning: K2, B, CBL-A, CBL-B, and Incremental	No	Java	Mostly in Portuguese	supports MSBN and HBN
BayesNet for Matlab	Open source	API only	No	Exact and approximate	Parameter and structure learning with missing data MCMC, IC, PC are there	API only	Matlab, C	Yes	Both Directed and Undirected PGM
BNLearn in R	Open source	No	No	Exact and approximate	Parameter and Structure	Yes	R	Yes	Random data generation, TAN
OpenMarkov	Open source	GUI	No	Exact Inference	PC and Hill climbing Parameter by Laplace-correction	No evidence	Java	Yes	Sensitivity Analysis
Elvira	Open source	Both	No	Exact Inference	Structure Learning	No	Java	In Spanish	Decision Making
BN Tool in Java	Open source	API	No		Structure Learning	Continuous State	Java	Yes	ID and DBN

Table 3: Comparative study of popular Relation Models

Tool name	Commercial /Open source	GUI / Programming	OO-features Supported	Inference Feature	Learning Feature	Continuous Data Support	Languagessupportedfor API	Well Documented in English	OtherFeatures
ProbReM	Free and Open source	Language to describe the relationship	No	Inference for DAPER models by MCMC	Parameter learning by ML (max. likelihood)	No	Python XML based Datarepresentation	Yes	Directed Graphical Model
Alchemy	Open source	Programming	No	Logic inference in Markov logic net	statistical relational learning, structure learning	Yes	C++	Yes	Lifted Belief Propagation Sampling

Primula	Open source	GUI	No	Exact and Approximate Inference for RBN	parameter learning for RBNs, Complex and nested models	No evidence	Java	Yes	Supports Inheritance and Nesting and DBN
BLOG	Open source	Probabilistic Modelling Language	No	Default Inference	No evidence	Yes Limited to static continuous	Java .NET	Yes	Provides a Query language, Dynamic
UnBBayes	Open source	GUI	No	Inference	No evidence	No	Java	Mostly in Portuguese	Supports MEBN
Proximity	Open source	Programming and QGraph	No	Inference	Learn from relational data but not forRBNs	Yes	QGraph visual query language	No evidence	Supports highly expressive domains
AGrUM	Open source	Programming API	Yes	Lifted prob. inference	Learning graphical models	Yes	C++	Yes	Decision Trees
IBAL	Open source	Programming	No	Approximate and Exact Inference	Parameter Learning in the formof Bayesian parameter estimation	No	Objective CAML	Yes	Strongly typed built-in extensibility

4. Conclusions

Probabilistic graphical models are a special kind of tools dealing with reasoning and decision making under uncertainty. The stochastic world makes everything challenging by incorporating uncertainty in every sphere. PGMs are being used as the handiest tools in dealing with such challenges for decades. This paper presents an overview of the popular and widely used models, their family tree, features, special capabilities, limitations, applicability, and domains. Then it also enlists the software tools those implement a particular PGM and facilitates modelling using the concepts of those PGMs. We divided them into two classes BNs and non-BNs (or non-relational and relational models). The comparative tables show a list of software, their features, platforms, best working environment, and public availability. We then focus on some real-life applications where we can develop and deploy applications using the PGMs to get resolve issues occurring due to associated uncertainty. Researchers have been contributing to developing models with fewer limitations and better features. Hence, it is also a field of state-of-the-arts research where there is scope of contribution in cutting-edge technology.

As concluding remarks, this paper, to the best of our knowledge, articulates highest number of popular, widely used and state-of-the-arts PGMs and their associated tools, as well as for the first time ever in the literature it introduces a family tree of the models based on their characteristics, features, similarities, differences and principles. In fact, it is a guide for newbies in choosing the right model for their applications and problems, a handy referencefor associated challenges, limitations and overcoming techniques of their chosen models.

References

1. E. Charniak, “Bayesian networks without tears”, *AI Magazine*, vol. 12, no. 4, pp. 50–50, 1991.
2. D. Nikovski, “Constructing Bayesian networks for medical diagnosis from incomplete and partially correct statistics,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 12, no. 4, pp. 509–516, 2000.
3. J. Pearl, Probabilistic reasoning in intelligent systems: networks of plausible inference. *Elsevier*, 2014.
4. J. Pearl, “Causality: models, reasoning and inference,” Econometric Theory, vol. 19, no. 675685, p. 46, 2003.
5. J. Pearle, Probabilistic reasoning in intelligent systems.Morgan kaufmann, 1988.
6. F. V. Jensen and T. D. Nielsen, *Bayesian Networks and Decision Graphs.* (2nd ed.), New York, Springer Verlag, 2007.
7. T. Charitos, L. van der Gaag, S. Visscher, K. Schurink, and P. Lucas, “A dynamic Bayesian network for diagnosing ventilator-associated pneumonia in ICU patients,” *Expert Systems with Applications*, vol. 36, no. 2, pp. 1249–258, 2009.
8. C. Conati, A. Gertner, K. VanLehn, and M. Druzdzel, “On-Line Student Modeling for Coached Problem Solving Using Bayesian Networks,” Proc. of the 6th Int’l Conf. on User Modeling, pp. 231–242, 1997.
9. K. Kristensen and I. Rasmussen, “The use of a Bayesian network in the design of a decision support system for growing malting barley without use of pesticides,” *Computers and Electronics in Agriculture*, vol. 33, no. 3, pp. 197–217, 2002.
10. P. A. Aguilera, A. Fernández, R. Fernández, R. Rumí, and A. Salmerón, “Bayesian Networks in Environmental Modelling,” *Environmental Modelling & Software*, vol. 26, no. 12, pp. 1376–1388, 2011.
11. R. Baker, A. Battisti, J. Bremmer, M. Kenis, J. Mumford, F. Petter, G. Schrader, S. Bacher, P. DeBarro, P. Hulme, O. Karadjova, A. Lansink, O. Pruvost, P. Pysek, A. Roques, Y. Baranchikov, and J.-H. Sun, “ PRATIQUE: a research project to enhance pest risk analysis techniques in the European Union,” *EPPO Bulletin*, vol. 39, no. 1, pp. 87–93, 2009.
12. S. Mascaro, K. Korb, and A. Nicholson, “Anomaly Detection in Vessel Tracks using Bayesian Networks.” *Int’l Journal of Approximate Reasoning*, Elsevier Science, vol. 55, no. 1, pp. 84–96, 2011.

13. L. Falzon, “Using Bayesian network analysis to support centre of gravity analysis in military planning,” *European Journal of Operational Research*, vol. 170, no. 2, pp. 629–643, 2006.
14. T. Boneh, G. Weymouth, P. Newham, R. Potts, J. Bally, A. Nicholson, and K. Korb, “Fog forecasting for Melbourne Airport using a Bayesian network.” *Weather and Forecasting*, vol. 30, no. 5, pp. 1218–1233, 2015.
15. A. Tang, A. Nicholson, Y. Jin, and J. Han, “Using Bayesian belief networks for change impact analysis in architecture design,” *Journal of Systems and Software*, vol. 80, no. 1, pp. 127–148, 2007.
16. M. Horny, “Bayesian networks,” *School of Public Health, Department of Health Policy and Management, Boston University*, Tech. Rep. 5, 2014
17. T. Boneh, “Ontology and Bayesian Decision Networks for Supporting the Meteorological Forecasting Process,” Ph.D. dissertation, Clayton School of Information Technology, Monash University, 2010.
18. D. Koller and A. Pfeffer, “Object-Oriented Bayesian Networks,” *Proc. of the 13th Conf.* on Uncertainty in Artificial Intelligence (UAI), USA, pp. 302–313, 1997.
19. O. Bangsø, M. J. Flores, and F. V. Jensen, “Plug & Play Object Oriented Bayesian Networks,” *Proc. Of the 10th Conf.* of the Spanish Association for Artificial Intelligence, CAEPIA 2003, pp. 457–467, 2003.
20. C. Meek and D. Heckerman, “Structure and Parameter Learning for Causal Independence and Causal Interaction Models,” *Proc. of the 13th Conf.* onUncertainty in Artificial Intelligence (UAI), USA, pp. 366–375, 1997.
21. K. B. Laskey and S. M. Mahoney, “Network Fragments: Representing Knowledge for Constructing Probabilistic Models,” *Proc. of the 13th Conf.* on Uncertainty in Artificial Intelligence (UAI), USA, pp. 334–341, 1997.
22. D. Heckerman, C. Meek, and D. Koller, “Probabilistic entity-relationship models, PRMs, and plate models,” *Proc. ofIntroduction to statistical relational learning*, pp. 201–238, 2007.
23. K. B. Laskey, “MEBN: A language for first-order Bayesian knowledge bases,” *Artif. Intell.*, vol. 172, no. 2-3, pp. 140–178, 2008
24. D. P. Xiang Yang and M. P. Beddoes., “Multiply Sectioned Bayesian Networks and Junction Forests for Large Knowledge-Based Systems,” *Computational Intelligence*, vol. 9, no. 2, pp. 171–220, 1993.
25. N. Fenton and M. Neil, “Building large-scale Bayesian networks,” *The Knowledge Engineering Review*, vol. 15, no. 3, pp. 257–284, 2000.
26. D. Koller and N. Friedman, Probabilistic Graphical Models - Principles and Techniques. *MIT Press*, 2009.
27. V. J. Easton and J. H. McColl. Statistics Glossary, 1997. [Online; accessed 15-July-2019]. [Online]. Available: http://www.stats.gla.ac.uk/steps/glossary/
28. B. J. Frey, andN. Jojic,“A comparison of algorithms for inference and learning in probabilistic graphical models”, *IEEE Transactions on pattern analysis and machine intelligence*, Vol. 27, No. 9, pp. 1392-1416, 2005.
29. P. Larrañaga,andS. Moral, “Probabilistic Graphical Models in Artificial Intelligence”, *Applied Soft Computing*, Vol. 11, No. 2, pp. 1511-1528, 2011.
30. P. Larrañaga,H. Karshenas,C. Bielza, andR. Santana,“A review on probabilistic graphical models in evolutionary computation”, *Journal of Heuristics*, Vol. 18, No. 5, pp. 795-819, 2012.
31. L. Hongmei, H. Wenning, G. Wenyan, andC. Gang,“Survey of probabilistic graphical models”, *Proc. of the 10thWeb Information System and Application Conference*, pp. 275-280, 2013.
32. W. L. Buntine, “Operations for Learning with Graphical Models,” CoRR, vol. abs/1105.2519, 2011. [Online]. Available: http://arxiv.org/abs/1105.2519
33. R. G. Cowell, A. P. Dawid, and D. J. Spiegelhalter, “Sequential Model Criticism in Probabilistic ExpertSystems,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 3, pp. 209–219, 1993.
34. D. Koller, “Probabilistic Relational Models,” *Proc. ofInt’l Conf. on Inductive Logic Programming, Springer*, pp. 3–13, 1999.
35. M. B. Ishak, “Probabilistic relational models: learning and evaluation. (Les modèles probabilistes relationnels : apprentissage et évaluation),” Ph.D. dissertation, University of Nantes, France, 2015. [Online]. Available: https://tel.archives-ouvertes. fr/tel-01179501
36. M. Cossalter, O. Mengshoel, and T. Selker, “Visualizing and understanding large-scale Bayesian networks,” in Workshops at the 25thAAAI Conference on Artificial Intelligence, 2011.
37. M. Horny, “Bayesian networks,” *School of Public Health, Department of Health Policy and Management*, Boston University, Tech. Rep. 5, 4 2014
38. S. Psillos, “An Explorer upon Untrodden Ground: Peirce on Abduction”, *Handbook of the History of Logic*, Vol. 10, pp. 117-151,2011
39. K. B. Korb and A. E. Nicholson, Bayesian Artificial Intelligence. *CRC Press*, 2010.
40. P. Dagum, A. Galper, and E. Horvitz, “Dynamic network models for forecasting,” *Uncertainty in Artificial Intelligence*. Elsevier, pp. 41–48, 1992.
41. A. E. Nicholson, “Monitoring discrete environments using dynamic belief networks (robotics),” Ph.D. dissertation, PhD thesis, University of Oxford (United Kingdom), 1992.
42. U. Kjærulff, “A computational scheme for dynamic Bayesian networks,” 1993.
43. T. L. Dean and K. Kanazawa, “Probabilistic Temporal Reasoning.” *in AAAI*, pp. 524–529, 1988.
44. D. Koller and A. Pfeffer, “Object-oriented Bayesian networks,” *Proc. of the 13th conf. on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, pp. 302–313, 1997.
45. L. Torti, P.-H. Wuillemin, and C. Gonzales, “Reinforcing the Object-Oriented aspect of probabilistic relational models,” *Proc. of European Workshop on Probabilistic Graphical Models*, pp. 273–280, 2010.

46. D. Koller, "Probabilistic Relational Models," *Proc. ofInductive Logic Programming, 9th International Workshop*, pp. 3–13, 1999.

47. L. Getoor, N. Friedman, D. Koller, A. Pfeffer, and B. Taskar, "Probabilistic relational models", *Introduction to statistical relational learning*, vol. 8, 2007

48. J. Neville and D. D. Jensen, "Dependency Networks for Relational Data," *Proc. of the 4th IEEE International Conference on Data Mining (ICDM 2004)*, pp. 170–177, 2004.

49. M. Jaeger, "Relational Bayesian Networks," *Proc. of the 13th Conference on Uncertainty in Artificial Intelligence*, USA, 1997, pp. 266–273.

50. M. Grzegorczyk, "An introduction to Gaussian Bayesian networks," *Systems Biology in Drug Discovery and Development*, Springer, pp. 121–147, 2010.

51. BayesFusion.com. GeNIe Modeler: Complete Modeling Freedom, 2018. [Online; accessed 19-February-2019]. [Online]. Available: https://www.bay esfusion.com/genie/

52. C. Gonzales, L. Torti, M. Chopin, and P.-H. Wuillemin, "aGrUM: A GRaphical Universal Modeler," https://forge.lip6.fr/projects/agrum, [Online; accessed 27-June-2017].

53. M. Samiullah, iOOBN: An Object-Oriented Bayesian Network Modelling Framework with Inheritance. *Doctoral dissertation*, Monash University, 2020.

54. J. Pearl andT. S. Verma, "A statistical semantics for causation", *Statistics and Computing*, Vol. 2, No. 2, pp. 91-95, 1992.

55. R. T. O'Donnell,A. E. Nicholson, B. Han, K.B. Korb, M.J. Alam and L. R. Hope,"Causal discovery with prior information", *Proc. Australasian Joint Conference on Artificial Intelligence*, Springer, Berlin, Heidelberg, pp. 1162-1167, 2006.

56. P.Larrañaga, H. Karshenas, C. Bielza andR. Santana, "A review on probabilistic graphical models in evolutionary computation", *Journal of Heuristics*, 18(5), 795-819, 2012.

57. E. M.Airoldi,"Getting started in probabilistic graphical models", *PLoS Comput Biol*, 3(12), e252, 2007.

58. T. A. Jilani, and S. A. R. Naqvi,"A review of probabilistic graph models for feature selection with applications in Economic and Financial time series forecasting VFAST", *Transactions on Software Engineering*, 2(1), 21-28, 2014.