

Разработка системы стохастического моделирования на основе вероятностных графовых моделей

И. Н. Манякин

МСистемы, Санкт-Петербург

ilya@mbayes.ru

М. В. Бахмутский

МСистемы, Санкт-Петербург

michael@mbayes.ru

Д. А. Иванов

МСистемы, Санкт-Петербург

danil@mbayes.ru

Аннотация. Статья содержит краткое введение в тему вероятностных графовых моделей и описание разработанной системы вероятностного моделирования MBayes. Система интегрирует графический редактор для создания и редактирования байесовских сетей, а также набор алгоритмов байесовского вывода, позволяющих производить апостериорные оценки параметров моделей на основе данных. Разработка нацелена на упрощение создания вероятностных моделей и применения байесовских методов в прикладных исследованиях.

Ключевые слова: вероятностные графовые модели; байесовские сети; байесовский вывод; интерпретируемый искусственный интеллект

I. ВВЕДЕНИЕ

Теорема Байеса и методы Байесовского вывода существуют уже почти 300 лет и предлагают обоснованный подход к работе с различными типами и источниками неопределенности в научных, технологических и социотехнических областях. Исторически применение байесовских методов было ограничено недостатком вычислительных мощностей и высоким порогом для их успешного и корректного использования. И если первая проблема, благодаря растущей доступности вычислительных мощностей, в значительной степени решена, то вторая проблема остаётся актуальной по сей день. Вероятно, что одна из причин этому является то, что, байесовские методы традиционно требуют от пользователя не только понимания своей предметной области, но и знаний в области байесовской статистики, методов оценки параметров распределений, алгоритмов байесовского вывода, а также навыков их программной реализации. В совокупности это ограничивает применение байесовского методов к относительно небольшому количеству пользователей.

Одновременно, за последние десятилетия мы видим развитие и распространение простых в использовании языков программирования. Так, например Python обладает богатой экосистемой библиотек для научных вычислений (NumPy, SciPy и т.п.). Эти языки и экосистемы радикально упростили написание пайплайнов анализа данных для многих задач. Однако с ростом скорости разработки кода возрастают и риски ошибок как в реализации, так и в семантике этих пайплайнов, включая неверные статистические допущения, переобучение, «р-hacking» и, как следствие, увеличение общего уровня ложноположительных результатов исследований. Следовательно, упрощение процесса написания кода для анализа данных

необязательно ведет к росту количества качественных, воспроизводимых исследований. Для этого необходимы инструменты для разработки пайплайнов анализа, основанных на надежных процедурах статистического вывода, для чего байесовские методы являются одним из самых перспективных направлений. Это создает потребность в создании инструментов для проведения байесовского анализа и построения байесовских моделей. Такие системы должны быть адаптируемыми к широкому кругу различных задач, одновременно быть простыми в применении, а также позволять легко масштабировать вычисления для обработки больших объемов данных.

В данной работе мы предлагаем подход к решению вышеописанных задач на основе вероятностных графовых моделей (ВГМ) и использования ВГМ для создания аналогов CAD/CAM систем, применяемых для проектирования инженерных изделий. Ключевая идея состоит в том, чтобы собирать из более маленьких, хорошо определенных блоков, модели процессов, которые в дальнейшем решаются численными методами. В CAD/CAM системах это принимает форму дифференциальных уравнений, а в рамках анализа и обработки информации это принимает форму решения задач статистического (байесовского) вывода.

В качестве первого шага в этом направлении мы реализовали MBayes — интегрированную среду разработки (IDE) для построения гибридных байесовских сетей, содержащих как дискретные и непрерывные случайные переменные, так и детерминированные и функциональные зависимости. Разработанная система позволяет пользователям создавать, редактировать и обучать гибридные байесовских сетей, запускать алгоритмы байесовского вывода и визуализировать результаты. В основе предложенной платформы лежит концепция «модельно-ориентированного машинного обучения» [1], которая разделяет процесс спецификации модели и задачу вычисления вероятностных запросов.

Такое разделение позволяет специалистам в прикладных науках разрабатывать модели процессов без требования их реализации в программах, а программистам-математикам работать над алгоритмами вывода, что упрощает и ускоряет процесс разработки. Похожий подход применяется в языках программирования, где высокоуровневые языки программирования трансформируются компилятором в машинный код, обеспечивая переносимость кода между различными архитектурами ЭВМ и ускоряя процесс

разработки программ за счёт снижения количества ошибок в логике реализаций и повышения интерпретируемости. Подход модельно-ориентированного машинного обучения основан на использовании похожей абстракции для моделей (программы) и алгоритмов вычислений (ЭВМ), где роль компилятора играет MBayes.

В следующем разделе мы представим краткое введение в вероятностные графовые модели и байесовские сети. В третьем разделе дано описание разработанной системы MBayes. В заключение описаны планы дальнейшего развития и перспективные направления исследований.

II. ВЕРОЯТНОСТНЫЕ ГРАФОВЫЕ МОДЕЛИ И БАЙЕСОВСКИЕ СЕТИ

Вероятностная графовая модель (ВГМ) представляет собой множество функций над подмножествами переменных, которые описывают вероятностную, детерминированную или полезностную информацию и чья структура зависимостей задается графом $G = (V, E)$, где $V = \{v_1, \dots, v_m\}$ множество узлов, а $E = \{(v_i, v_j) \mid v_i, v_j \in V\}$ – множество связей. В графовых моделях структура графа описывает свойства независимости между различными параметрами или группами параметров модели, что используется для интерпретации информации, а также, для реализации алгоритмов обработки, включая логический и вероятностный вывод, на основе свойств независимости. Среди множества типов вероятностных графовых моделей часто выделяются байесовские сети, марковские сети и фактор-графы.

Байесовские сети основаны на направленном ациклическом графе (DAG) с направленными связями $E = \{(v_i, v_j) = v_i \rightarrow v_j \mid v_i, v_j \in V\}$. К каждому узлу v_i присвоена случайная величина X_i из множества $X = \{X_1, \dots, X_m\}$, для которой в рамках модели задается условная вероятность $P(X_i | \text{parents}(X_i))$. По свойству умножения вероятностей, байесовская сеть определяет эквивалентное ей совместное распределение как:

$$P(X_1, \dots, X_m) = \prod_{i=1}^m P(X_i | \text{parents}(X_i)).$$

Здесь при наличии связи $v_i \rightarrow v_j$ узел v_i принято называть узлом "родителем" узла v_j , а узел v_j – "ребенком" узла v_i . Функцией $\text{parents}(X_i) = \{X_j \mid v_i \rightarrow v_j \in E\}$ обозначают множество случайных переменных, от которых может зависеть распределение X_i . Так как каждому узлу v_i присвоена случайная величина X_i , то в дальнейшем мы не будем различать узлы графа байесовской сети и случайные величины модели.

Важно отметить, что направленные связи хоть и допускают причинно-следственную интерпретацию, но они представляют собой лишь дополнительный семантический слой поверх совместного распределения, представленного байесовской сетью. Также следует отметить, что ключевым фактором при создании байесовских сетей является разреженность графа, то есть количество связей невелико по сравнению с максимально возможным числом связей – $|E| \ll O(|V|^2)$.

Это предположение о разреженности позволяет реализовывать эффективные алгоритмы вывода. Модели байесовских сетей далее подразделяются по типу моделируемых ими случайных величин на три основных класса представленных в табл. 1.

ТАБЛИЦА I. КЛАССЫ БАЙЕСОВСКИХ СЕТЕЙ

Тип	Используемые узлы
Непрерывные	только непрерывные
Дискретные	только дискретные
Гибридные	дискретные и непрерывные

Дополнительные подклассы включают статические и динамические сети, где последние позволяют моделировать временные процессы. Байесовскую сеть можно обобщить за счет включения узлов «решения» и «полезности», которые моделируют действия и вознаграждения, формируя так называемые «диаграммы влияния».

A. Моделирование, используя Байесовские сети

Процесс работы с байесовской сетью включает три основных этапа:

- поиск структуры модели [2];
- обучение условных распределений [3];
- вычисление вероятностных запросов [4].

Первый этап состоит в определении соответствующих случайных переменных модели и их типов (непрерывных, дискретных, детерминированных), а также взаимосвязей между ними и их направлений. Этот этап может выполняться вручную, на основе экспертного мнения или автоматически/полуавтоматически с использованием различных алгоритмов «структурного обучения» [2]. Следует отметить, что причинно-следственная интерпретация байесовской сети может быть полезна при построении графа на основе экспертных оценок, так как позволяет легче интерпретировать допущения, заложенные в модели. Более того, возможность обосновать вычисления, выполняемые внутри модели, и проводить аудит модели обосновывает применение понятия «интерпретируемый искусственный интеллект» к байесовским сетям.

Второй этап включает идентификацию и калибровку параметров распределения каждого узла в зависимости от доступной информации. В случае применения параметрических распределений выбор распределения может быть сделан на основе статистических тестов (например, критерий Колмогорова-Смирнова), а калибровка параметров – с помощью метода максимального правдоподобия [3] или байесовской апостериорной оценкой. Также возможно создать непараметрическую форму условного распределения узла, моделируя функцию распределения случайной величины на основе гладких аппроксимаций, чем могут служить, например, нейросети. В рамках данного этапа также можно аппроксимировать распределения, используя смеси более простых распределений или дискретизируя непрерывные переменные, получая на выходе дискретные аппроксимации — такие трансформации могут быть выполнены как пользователем, так и в полуавтоматическом/автоматическом режиме.

Третий этап состоит из вычисления вероятностных запросов на основе модели. Основные классы вероятностных запросов описаны в табл. 2 [4], где множество переменных модели, X , разделяются на три непересекающихся подмножества $X = Q \cup E \cup M$, где Q – множество узлов, по которым формулируется запрос, E – множество случайных величин, значения которых известны, и M – множество скрытых узлов, по которым требуется выполнить маргинализацию.

ТАБЛИЦА 2. КЛАССЫ ВЕРОЯТНОСТНЫХ ЗАПРОСОВ

Тип запроса	Формула
Вероятность свидетельства (PE)	$P(E = e)$
Маргинальный (Априорный MAR)	$P(Q) = \sum_M P(Q \cup M)$
Маргинальный (Апостериорный MAR)	$P(Q E = e) = \sum_M P(Q \cup M E = e)$
Максимум вероятности (MPE или MAP)	$q^* = \operatorname{argmax}_q P(Q E = e)$
Маргинальный максимум вероятности (MMPE)	$q^* = \operatorname{argmax}_q \sum_M P(Q \cup M E = e)$

Новые данные могут быть введены в узлы как «свидетельства», затем могут быть вычислены апостериорные оценки с учетом этих данных путем выполнения алгоритма точного или приближенного вывода. Если свидетельства не заданы, то байесовская сеть может использоваться для оценки априорных вероятностей, чаще всего используемых в создании прогнозов. Если же данные введены в узел, то на их основе выполняется апостериорная оценка неизмеряемых случайных величин, которая может быть использована как для диагностики, так и для адаптации прогнозов при поступлении данных. В случае, если сеть состоит только из дискретных узлов или условных линейных гауссовых узлов, то такие запросы могут быть вычислены точно с помощью алгоритмов распространения доверия [5] или точного вывода по

методу описанному в [6]. Однако в случае, когда форма зависимостей более сложная, например нелинейная, либо форма условных распределений не гауссовская, требуется использовать алгоритмы приближенного вывода, которые включают методы выборки по значимости [7], сэмплирование по Гиббсу, Монте-Карло с марковскими цепями (MCMC), динамическую дискретизацию [8] и другие [9,10]. Следует отметить, что алгоритм отделен от модели, и его выбор определяется требуемыми характеристиками вычисления: скорость, точность и тип узлов модели. Модель, в свою очередь, перед вычислениями может быть трансформирована (например, через дискретизацию непрерывных узлов), чтобы упростить вычисления или повысить точность аппроксимации [8].

III. СИСТЕМА МОДЕЛИРОВАНИЯ Mbayes

Была разработана система Mbayes для создания, обучения и работы с графовыми моделями. Система интегрирует low-code редактор (рис. 1) для создания моделей и позволяет создавать гибридные байесовские сети, содержащие четыре типа узлов: категориальные, непрерывные, дискретные и детерминированные, и комбинировать отдельные узлы в многоуровневые иерархические модели. Как и в других средах для разработки байесовских сетей (например, GENIE [11] и HUGIN [12]), пользователь может создавать категориальные и непрерывные гауссовские узлы и запустить алгоритмы байесовского вывода на основе точного инференса, используя алгоритмы дерева сочленений, что в свою очередь позволяет вычислять запросы класса PE. Система также позволяет визуализировать распределения, вычислять моменты и метрики зависимости распределений, что позволяет пользователям без навыков программирования создавать имитационные модели, которые можно симулировать интегрированными в систему алгоритмами Монте-Карло.

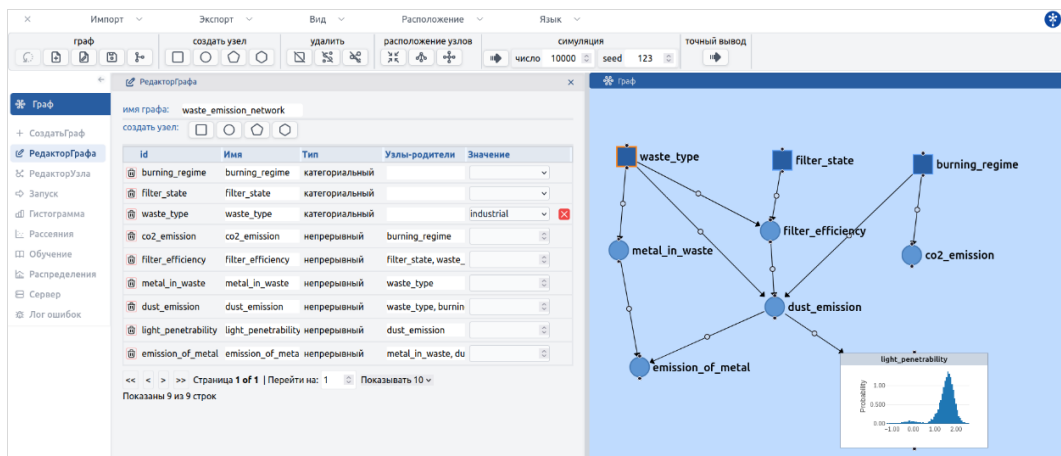


Рис. 1. Редактор системы Mbayes с примером гибридной модели WASTE [14]

ключевое отличие от других библиотек по работе с байесовскими сетями [12, 13] – это возможность задавать нелинейные зависимости в условных распределениях через ввод формул в таблицы условных вероятностей узлов, а также реализация распределений с ограниченным носителем, которые часто возникают в практических применениях. В системе также реализована возможность задавать детерминированные

узлы, используя язык Python, что позволяет реализовывать широкий класс детерминированных зависимостей, например if-elif-else, и позволяет интегрировать внешние программные модули в модель, без существенного повышения сложности и без введения нового синтаксиса. Для экспорта моделей из системы используется формат JSON, который позволяет пользователям с навыками программирования создавать

модели как в Python, так и других языках и проводить вычисления, используя REST API вычислительных ядер системы. Система также интегрирует алгоритмы поиска структуры графов на основе алгоритмов класса Питер-Кларк. Калибровка параметров распределений использует методы максимального правдоподобия, что вместе с методами структурного обучения и алгоритмами вычислений вероятностных запросов позволяет пройти все этапы моделирования от создания графа до вероятностных оценок. Для решения запросов класса MAR и PE в гибридных моделях, где невозможно применить алгоритмы точного вывода на основе дерева сочленений, используются алгоритмы выборки по значимости, которые позволяют легко масштабировать вычисления через параллелизацию сэмплирования и служат основной для дальнейшей реализации алгоритмов последовательного Монте-Карло [SMC].

Ключевое отличие от других библиотек по работе с байесовскими сетями [12, 13] – это возможность задавать нелинейные зависимости в условных распределениях через ввод формул в таблицы условных вероятностей узлов, а также реализация распределений с ограниченным носителем, которые часто возникают в практических применениях. В системе также реализована возможность задавать детерминированные узлы, используя язык Python, что позволяет реализовывать широкий класс детерминированных зависимостей, например if-elif-else, и позволяет интегрировать внешние программные модули в модель, без существенного повышения сложности и без введения нового синтаксиса. Для экспорта моделей из системы используется формат JSON, который позволяет пользователям с навыками программирования создавать модели как в Python, так и других языках и проводить вычисления, используя REST API вычислительных ядер системы. Система также интегрирует алгоритмы поиска структуры графов на основе алгоритмов класса Питер-Кларк. Калибровка параметров распределений использует методы максимального правдоподобия, что вместе с методами структурного обучения и алгоритмами вычислений вероятностных запросов позволяет пройти все этапы моделирования от создания графа до вероятностных оценок. Для решения запросов класса MAR и PE в гибридных моделях, где невозможно применить алгоритмы точного вывода на основе дерева сочленений, используются алгоритмы выборки по значимости, которые позволяют легко масштабировать вычисления через параллелизацию сэмплирования и служат основной для дальнейшей реализации алгоритмов последовательного Монте-Карло [10].

IV. ЗАКЛЮЧЕНИЕ И ПЕРСПЕКТИВЫ

В рамках данной статьи был рассмотрен подход к моделированию на основе вероятностных графовых моделей, и была представлена разработанная система MBayes. Цель системы – облегчение доступа широкого круга специалистов из различных отраслей к методам вероятностного моделирования с использованием байесовских сетей и ускорение создания и внедрения прикладных решений с использованием этих технологий. Система позволяет создавать модели и запускать байесовский вывод с помощью графического редактора без необходимости программирования. Для

тех, кто уже знаком и применяет байесовские методы в собственных разработках, система может представлять интерес в качестве инструмента быстрого прототипирования и тестирования. Кроме того, визуализация моделей с использованием графического редактора способствует предотвращению ошибок в процессе моделирования и облегчает коммуникацию между членами междисциплинарных команд. Разделение модели и вычислительных алгоритмов, в свою очередь, ускоряет разработку прикладных моделей за счет создания уровня абстракции, который отделяет логику работы конкретной модели от реализации вычислительных алгоритмов, работающих со случайными величинами.

Существует ряд перспективных направлений для дальнейшего развития системы, включая упрощение моделирования динамических байесовских сетей, методы автоматического синтеза кода из моделей и реализацию более продвинутых алгоритмов байесовского вывода, а также методы сравнения моделей на основе оценок маргинального правдоподобия.

СПИСОК ЛИТЕРАТУРЫ

- [1] Bishop C.M. Model-based machine learning / Christopher Michael Bishop // *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*. 2013. Т. 371, № 1984. – URL: <https://doi.org/10.1098/rsta.2012.0222> (дата обращения: 22.03.2026).
- [2] Kitson N.K. A survey of Bayesian Network structure learning / Neville Kenneth Kitson, Anthony C. Constantinou, Zhigao Guo, Yang Liu, Kiattikun Chobtham // *Artificial Intelligence Review*. 2023. Т. 56, №8. С. 8721–8814.
- [3] Ji Z. A Review of Parameter Learning Methods in Bayesian Network / Zhiwei Ji, Qibiao Xia, Guanmin Meng // *International Conference on Intelligent Computing*. 2015. С. 3–12.
- [4] Darwiche A. Modeling and reasoning with Bayesian networks / Adnan Darwiche. – Cambridge: Cambridge university press, 2009. 548 с.
- [5] Pearl J. Probabilistic reasoning in intelligent systems: networks of plausible inference / Judea Pearl. San Francisco: Morgan Kaufmann, 1988. 552 с.
- [6] Lauritzen S.L. Stable local computation with conditional Gaussian distributions. / Steffen L. Lauritzen, Frank Jensen // *Statistics and Computing*. 2001. Т. 11, № 2. С. 191–203.
- [7] Cheng, J. AIS-BN: An Adaptive Importance Sampling Algorithm for Evidential Reasoning in Large Bayesian Networks / Jian Cheng, Marek J. Druzdzel // *Journal of Artificial Intelligence Research*. 2000. Т. 13. С. 155–188.
- [8] Neil M. Inference in hybrid Bayesian networks using dynamic discretization / Martin Neil, Manesh Tailor, David Marquez // *Statistics and Computing*. 2007. Т. 17, № 3. С. 219–233.
- [9] Minka T.P. Expectation propagation for approximate Bayesian inference // *arXiv*. 2013. – URL: <https://arxiv.org/abs/1301.2294> (дата обращения: 22.03.2026).
- [10] Doucet A., Freitas N., Gordon N. (eds.) Sequential Monte Carlo methods in practice / A. Doucet, N. Freitas, N. Gordon. New York: Springer, 2001. 582 с.
- [11] Druzdzel M.J. SMILE: Structural Modeling, Inference, and Learning Engine and GeNIe: A Development Environment for Graphical Decision-Theoretic Models // *AAAI/IAAI*. 1999. С. 902–903.
- [12] Madsen A.L. The Hugin Tool for Probabilistic Graphical Models / Anders L. Madsen, Frank Jensen, Uffe B. Kjaerulff, Michael Lang // *International Journal on Artificial Intelligence Tools*. 2005. Т. 14, № 03. С. 507–543.
- [13] Ankan A. pgmpy: Probabilistic Graphical Models using Python / Ankur Ankan, Abinash Panda // *SciPy*. 2015. С. 6–11.
- [14] Lauritzen S.L. Graphical models. Т. 17. Oxford: Clarendon Press, 1996. 312 с. (Oxford Statistical Science Series).