

Графические вероятностные модели

Сергей Николенко

НИУ ВШЭ — Санкт-Петербург

25 апреля 2020 г.

Random facts:

- 25 апреля — День ДНК; именно 25 апреля 1953 г. в *Nature* вышли три статьи Джеймса Уотсона, Фрэнсиса Крика, Мориса Уилкинса, Розалинд Франклин и их коллег, а 25 апреля 2003 г. объявили о завершении проекта «Геном человека»
- 25 апреля 1792 г. разбойник Николя Пеллетье первым испробовал на себе гильотину
- 25 апреля 1945 г. советские и американские войска встретились на Эльбе
- 25 апреля 1960 г. субмарина *USS Triton* завершила первое подводное кругосветное плавание
- 25 апреля 1983 г. Юрий Андропов пригласил в Советский Союз Саманту Смит
- 25 апреля 1993 г. на референдуме в России большинство поддержали политику Бориса Ельцина, а 25 апреля 2007 г. прошли его похороны

Графические модели

В чём же проблема

- В предыдущих лекциях мы рассмотрели задачу байесовского вывода, ввели понятие сопряжённого априорного распределения, поняли, что наша основная задача – найти апостериорное распределение.
- Но если всё так просто – взяли интеграл, посчитали, всё получилось – о чём же здесь целая наука?
- Проблема заключается в том, что распределения, которые нас интересуют, обычно слишком сложные (слишком много переменных, сложные связи).
- Но, с другой стороны, в них есть дополнительная структура, которую можно использовать, структура в виде независимостей и условных независимостей некоторых переменных.

Графические модели с направленными графами

- Пример: рассмотрим распределение трёх переменных и запишем его по формуле полной вероятности:

$$p(x, y, z) = p(x \mid y, z)p(y \mid z)p(z).$$

- Теперь нарисуем граф, в котором стрелки указывают, какие условные вероятности заданы.
- Пока граф полносвязный, это нам ничего не даёт – любое распределение $p(x_1, \dots, x_n)$ так можно переписать.
- Но если некоторых связей *нет*, это даёт нам важную информацию и упрощает жизнь.

Графические модели с направленными графами

- Рассмотрим направленный ациклический граф на вершинах x_1, \dots, x_k и зададим в каждой вершине распределения $p(x_i \mid \text{pa}(x_i))$. Тогда будем говорить, что граф с этими локальными распределениями является графической моделью (байесовской сетью доверия) для совместного распределения вероятностей

$$p(x_1, \dots, x_k) = \prod_{i=1}^k p(x_i \mid \text{pa}(x_i)).$$

- Другими словами, если мы можем разложить большое совместное распределение в произведение локальных распределений, каждое из которых связывает мало переменных, это хорошо. :)

Графические модели с направленными графами

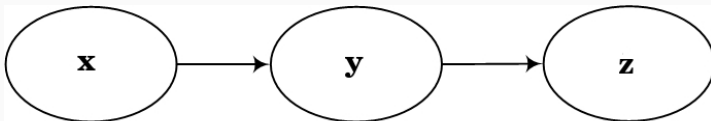
- Пример: обучение параметров распределения по нескольким экспериментам (плашки, можно нарисовать параметры явно):

$$p(x_1, \dots, x_n, \theta) = p(\theta) \prod_{i=1}^n p(x_i \mid \theta).$$

- Что можно сказать о (не)зависимости случайных величин x_i и x_j ?
- Задача вывода на графической модели: в некоторой части вершин значения наблюдаются, надо пересчитать распределения в других вершинах (подсчитать условные распределения). Например, из этой модели получатся и задача обучения параметров, и задача последующего предсказания.

Графические модели с направленными графами

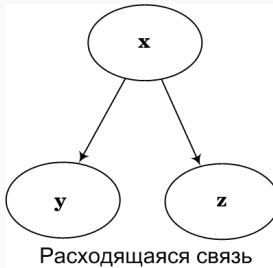
- d -разделимость – условная независимость, выраженная в структуре графа:
 - последовательная связь, $p(x, y, z) = p(x)p(y | x)p(z | y)$:
 - если y не наблюдается, то
$$p(x, z) = p(x) \int p(y | x)p(z | y)dy = p(x)p(z | x);$$
 - если y наблюдается, то
$$p(x, z | y) = \frac{p(x, y, z)}{p(y)} = \frac{p(x)p(y|x)p(z|y)}{p(y)} = p(x | y)p(z | y),$$
 получили условную независимость.



Последовательная связь

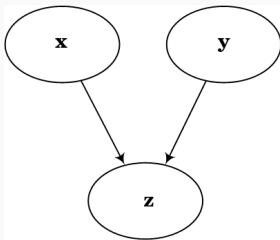
Графические модели с направленными графами

- расходящаяся связь, $p(x, y, z) = p(x)p(y | x)p(z | x)$, – так же:
 - если y не наблюдается, то
$$p(x, z) = p(x)p(z | x) \int p(y | x) dy = p(x)p(z | x);$$
 - если y наблюдается, то
$$p(x, z | y) = \frac{p(x, y, z)}{p(y)} = \frac{p(x)p(y|x)p(z|x)}{p(y)} = p(x | y)p(z | y),$$
 получили условную независимость.



Графические модели с направленными графами

- Интересный случай – сходящаяся связь, $p(x, y, z) = p(x)p(y)p(z | x, y)$:
 - если z не наблюдается, то $p(x, y) = p(x)p(y)$, независимость есть;
 - если z наблюдается, то $p(x, y | z) = \frac{p(x, y, z)}{p(z)} = \frac{p(x)p(y)p(z|x, y)}{p(z)}$, и условной независимости нету.



Сходящаяся связь

Обобщение: если наблюдается хотя бы один из потомков z , уже может не быть независимости между x и y .

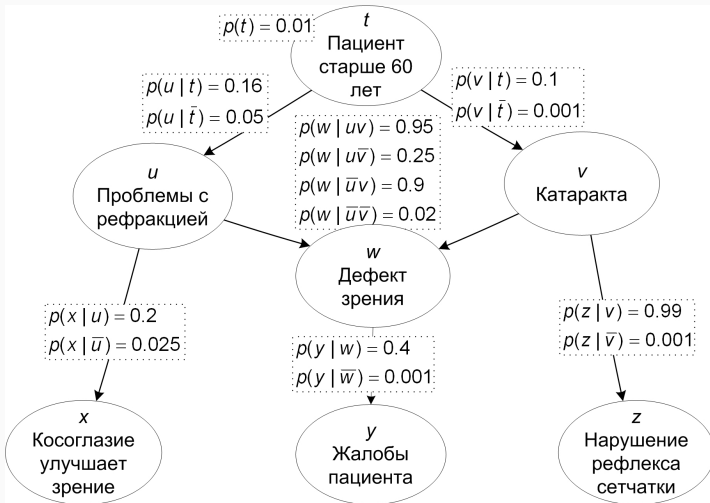
- Можно сформулировать, как структура графа соотносится с условной независимостью: в графе, где вершины из множества Z получили означивания (evidence), две ещё не означенные вершины x и y условно независимы при условии множества означенных вершин Z , если любой (ненаправленный) путь между x и y :
 - либо проходит через означенную вершину $z \in Z$ с последовательной или расходящейся связью;
 - либо проходит через вершину со сходящейся связью, в которой ни она, ни её потомки не получили означиваний.

Графические модели с направленными графами

- Можно сказать, что граф задаёт некоторое семейство распределений – не все распределения на вершинах графа будут соответствовать тем ограничениям по условной независимости, которые накладывает структура графа.
- Теорема (без доказательства): это семейство распределений в точности совпадает с семейством тех распределений, которые можно разложить в произведение

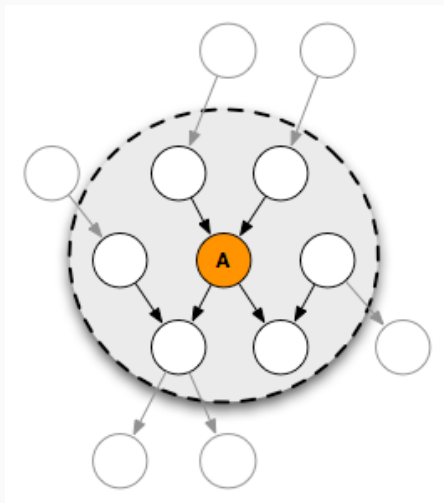
$$p(x_1, \dots, x_k) = \prod_{i=1}^k p(x_i \mid \text{pa}(x_i)).$$

Пример байесовской сети



- Интересный вопрос: какие вершины нужно означить, чтобы наверняка «отрезать» одну вершину (Markov blanket)?
- Иначе говоря, для какого минимального множества вершин X $p(x_i \mid x_{j \neq i}) = p(x_i \mid X)$?

Markov blanket

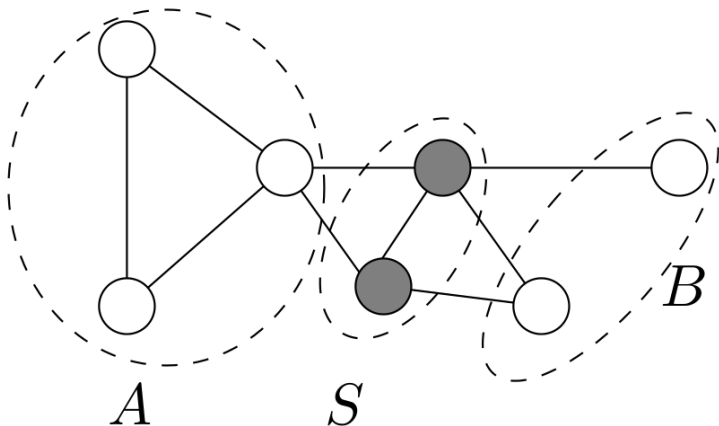


Другие графические модели

Графические модели с ненаправленными графами

- Можно сделать и так, чтобы условие независимости было (более) локальным.
- Для этого нужно задавать модели ненаправленными графами. В них условие совсем естественное: множество вершин X условно независимо от множества вершин Y при условии множества вершин Z , если любой путь от X к Y проходит через Z .
- В частности, очевидно, $p(x_i, x_j \mid x_{k \neq i, j}) = p(x_i \mid x_{k \neq i, j})p(x_j \mid x_{k \neq i, j})$ тогда и только тогда, когда x_i и x_j не соединены ребром.
- Такие модели называются *марковскими сетями* (Markov random fields).

Условная независимость в ненаправленных моделях



Графические модели с ненаправленными графами

- Поэтому в ненаправленных моделях локальные распределения соответствуют кликам в графе, и факторизация получается в виде

$$p(x_1, \dots, x_k) = \frac{1}{Z} \prod \psi_C(x_C),$$

где C – максимальные клики, ψ_C – неотрицательные функции (*номенциалы*), а Z – нормировочная константа (partition function).

- Поскольку $\psi_C \geq 0$, их обычно представляют как экспоненты:

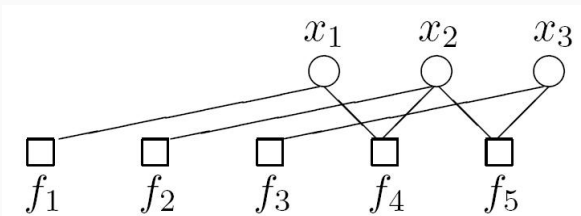
$$\psi_C(x_C) = \exp(-E_C(x_C)),$$

E_C – функции энергии, они суммируются в полную энергию системы (это всё похоже на статистическую физику, отсюда и терминология).

- Интересный факт: назовём *идеальной картой* (perfect map) распределения D графическую модель G , если все условные независимости, присутствующие в D , отображены в G , и наоборот (ничего лишнего). Тогда идеальные карты в виде направленных моделей существуют не у всех распределений, в виде ненаправленных тоже не у всех, и эти множества существенно различаются (бывают распределения, которые нельзя идеально выразить направленной моделью, но можно ненаправленной, и наоборот).

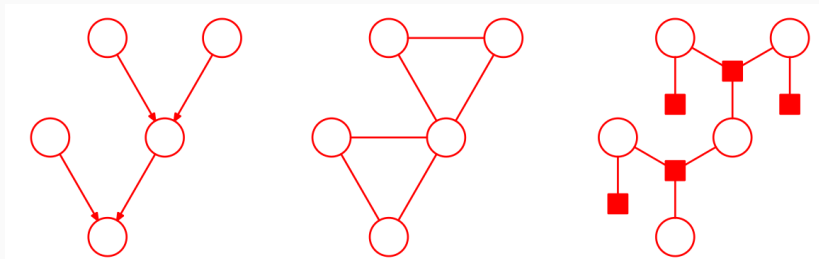
Фактор-графы

- Важная для вывода модификация – *фактор-граф* (можно построить и по направленной модели, и по ненаправленной).
- Фактор-граф – двудольный граф функций и переменных.
- Функция, соответствующая графу, – произведение всех входящих в него функций (т.е. то самое разложение и есть).
- Пример: $p(x_1, x_2, x_3) = f_1(x_1)f_2(x_2)f_3(x_3)f_4(x_1, x_2)f_5(x_2, x_3)$.



Алгоритм передачи сообщений

Три представления



Функция в общем виде

- Чтобы поставить задачу в общем виде, рассмотрим функцию

$$p^*(X) = \prod_{j=1}^m f_j(X_j),$$

где $X = \{x_i\}_{i=1}^n$, $X_j \subseteq X$.

- Т.е. мы рассматриваем функцию, которая раскладывается в произведение нескольких других функций.

- Задача нормализации: найти $Z = \sum_X \prod_{j=1}^m f_j(X_j)$.
- Задача маргинализации: найти

$$p_i^*(x_i) = \sum_{k \neq i} p^*(X).$$

Также может понадобиться, например, $p_{i_1 i_2}$, но реже.

- Поиск гипотезы максимального правдоподобия:

$$\mathbf{x}^* = \arg \max_X p(X).$$

- Все эти задачи NP-трудные.
- То есть, если мир не рухнет, сложность их решения в худшем случае возрастает экспоненциально.
- Но можно решить некоторые частные случаи.

- Давайте начнём с графа в виде (ненаправленной) цепи:

$$p(x_1, \dots, x_n) = \frac{1}{Z} \psi_{1,2}(x_1, x_2) \dots \psi_{n-1,n}(x_{n-1}, x_n).$$

- Мы хотим найти

$$p(x_k) = \sum_{x_1} \dots \sum_{x_{k-1}} \sum_{x_{k+1}} \dots \sum_{x_n} p(x_1, \dots, x_n).$$

Пример

- Очевидно, тут можно много чего упростить; например, справа налево:

$$\begin{aligned}\sum_{x_n} p(x_1, \dots, x_n) &= \\ &= \frac{1}{Z} \psi_{1,2}(x_1, x_2) \dots \psi_{n-2,n-1}(x_{n-2}, x_{n-1}) \sum_{x_n} \psi_{n-1,n}(x_{n-1}, x_n).\end{aligned}$$

- Эту сумму можно вычислить отдельно и продолжать в том же духе справа налево, потом аналогично слева направо.

Пример

- В итоге процесс сойдётся на узле x_k , куда придут два «сообщения»: слева

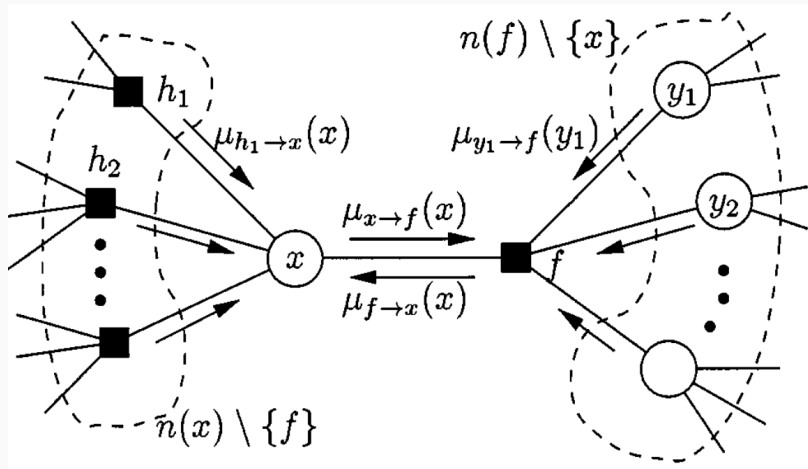
$$\mu_{\alpha}(x_k) = \sum_{x_{k-1}} \psi_{k-1,k}(x_{k-1}, x_k) \left[\dots \sum_{x_2} \psi_{2,3}(x_2, x_3) \left[\sum_{x_1} \psi_{1,2}(x_1, x_2) \right] \dots \right],$$

справа

$$\mu_{\beta}(x_k) = \sum_{x_{k+1}} \psi_{k,k+1}(x_k, x_{k+1}) \left[\dots \left[\sum_{x_n} \psi_{n-1,n}(x_{n-1}, x_n) \right] \dots \right].$$

- Каждую частичную сумму можно рассматривать как «сообщение» от узла к своему соседу, причём это сообщение – функция от соседа.

- Чтобы обобщить, удобно рассмотреть опять фактор-граф.
- Предположим, что фактор-граф – дерево (если не дерево, так просто не сработает).
- Алгоритм передачи сообщений решает задачу маргинализации для функции вида $p(x_1, \dots, x_n) = \prod_s f_s(X_s)$, заданной в виде фактор-графа.
- Передаём сообщения по направлению к нужному узлу от переменных к функциям и наоборот.



Алгоритм передачи сообщений

- Чтобы найти $p(x_k)$, запишем $p(x_1, \dots, x_n) = \prod_{s \in \text{ne}(x_k)} F_s(x_k, X_s)$, где X_s – переменные из поддерева с корнем в f_s . Тогда

$$\begin{aligned} p(x_k) &= \sum_{x_i \neq k} p(x_1, \dots, x_n) = \prod_{s \in \text{ne}(x_k)} \left[\sum_{X_s} F_s(x_k, X_s) \right] = \\ &= \prod_{s \in \text{ne}(x_k)} \mu_{f_s \rightarrow x_k}(x_k), \end{aligned}$$

где $\mu_{f_s \rightarrow x_k}(x_k)$ – сообщения от соседних функций к переменной x_k .

Алгоритм передачи сообщений

- Чтобы найти $\mu_{f_s \rightarrow x_k}(x_k)$, заметим, что $F_s(x_k, X_s)$ тоже можно разложить по соответствующему подграфу:

$$F_s(x_k, X_s) = f_s(x_k, Y_s) \prod_{y \in Y_s} G_y(y, X_{s,y}),$$

где Y_s – переменные, непосредственно связанные с f_s (кроме x_k), $X_{s,y}$ – соответствующие поддеревья.

- Итого получаем

$$\begin{aligned} \mu_{f_s \rightarrow x_k}(x_k) &= \sum_{Y_s} f_s(x_k, Y_s) \prod_{y \in Y_s} \left(\sum_{X_{s,y}} G_y(y, X_{s,y}) \right) = \\ &= \sum_{Y_s} f_s(x_k, Y_s) \prod_{y \in Y_s} \mu_{y \rightarrow f_s}(y). \end{aligned}$$

- Можно аналогично подсчитать, что

$$\mu_{y \rightarrow f_s}(y) = \prod_{f \in \text{ne}(y) \setminus f_s} \mu_{f \rightarrow y}(y).$$

Алгоритм передачи сообщений

- Итак, получился простой и понятный алгоритм:
 - как только узел получил сообщения от всех соседей, кроме одного, он сам начинает передавать сообщение в этого соседа;
 - сообщение по ребру между функцией и переменной является функцией от этой переменной;
 - узел-переменная x передаёт сообщение

$$\mu_{x \rightarrow f}(x) = \prod_{g \in \text{ne}(x) \setminus f} \mu_{g \rightarrow x}(x);$$

- узел-функция $f(x, Y)$ передаёт сообщение

$$\mu_{f \rightarrow x}(x) = \sum_{y \in Y} f(x, y) \prod_{y \in Y} \mu_{y \rightarrow f}(y);$$

- начальные сообщения в листьях $\mu_{x \rightarrow f}(x) = 1$, $\mu_{f \rightarrow x}(x) = f(x)$.

Алгоритм передачи сообщений

- Когда сообщения придут из всех соседей в какую-то переменную x_k , можно будет подсчитать

$$p(x_k) = \prod_{f \in \text{ne}(x_k)} \mu_{f \rightarrow x_k}(x_k).$$

- Когда сообщения придут из всех соседей в какой-то фактор $f_s(X_s)$, можно будет подсчитать совместное распределение

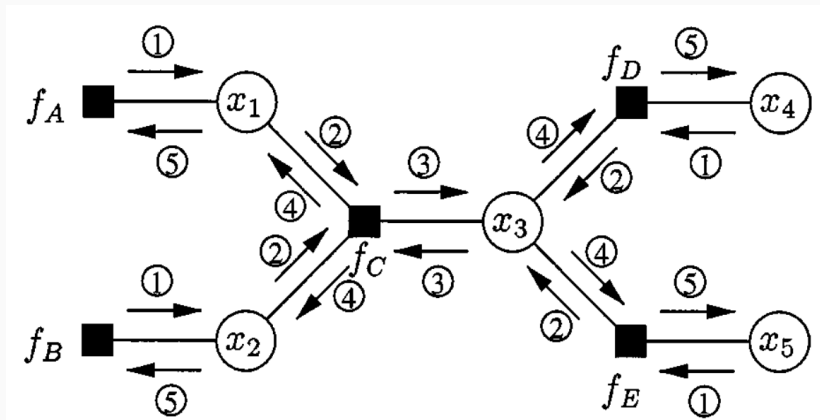
$$p(X_s) = f_s(X_s) \prod_{y \in \text{ne}(f_s)} \mu_{y \rightarrow f_s}(y).$$

- За два прохода (по каждому ребру туда и обратно) можно будет подсчитать маргиналы во всех узлах.

- Это называется алгоритм sum-product, потому что сообщение вычисляется как

$$\mu_{f \rightarrow x}(x) = \sum_{y \in Y} f(x, y) \prod_{y \in Y} \mu_{y \rightarrow f}(y).$$

- Задача максимизации $\arg \max_x p(x_1, \dots, x_n)$ решается так же, но алгоритмом max-sum: сумма заменяется на максимум, а произведение на сумму.



Так что же делать с байесовской сетью?

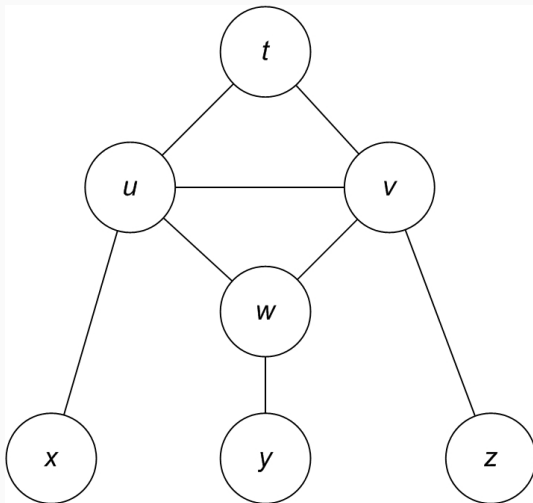
Для модели не в виде фактор-графа надо просто представить её в виде фактор-графа тем или иным способом.

Для байесовской сети это может означать, что надо сначала сделать морализацию, а потом добавить факторы в явном виде.

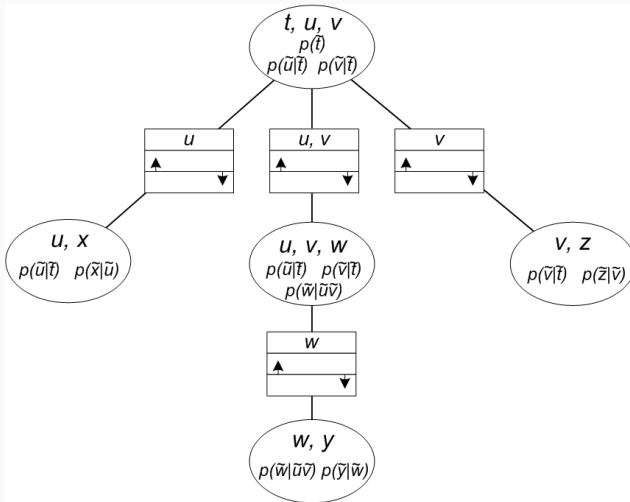
Так что же делать с байесовской сетью?



Так что же делать с байесовской сетью?



Так что же делать с байесовской сетью?



Спасибо!

Спасибо за внимание!