Перспективы применения моделей с латентными переменными и вариационный байес для задач оптимизации

Л. С. Звягин
Финансовый университет при Правительстве Российской Федерации (Финуниверситет), Financial University
lszvyagin@fa.ru

***Аннотация*. В основе большинства инструментов интеллектуального анализа данных лежат две технологии-машинное обучение и визуализация (визуальное представление информации). Байесовские сети позволяют объединить в себе эти две технологии. С математической точки зрения байесовские сети – это модель для представления вероятностных зависимостей, а также отсутствия этих зависимостей. Разработка методов, позволяющих уменьшить вычислительную сложность при построении моделей сетей Байеса, является актуальной и востребованной.**

Ключевые слова: байесовский подход; оценка достоверности; апостериорное распределение; метод байесовских сетей

I. Введение

Весь байесовский подход базируется на одной единственной формуле или теореме. Теорема Байеса приведена в математической и концептуальной формах.

Ключевая идея. Предположим, есть какая-то неизвестная величина, которую мы бы хотели оценить по каким-то ее косвенным проявлениям. В данном случае неизвестная величина – θ, а ее косвенное проявление – y. Тогда можно воспользоваться теоремой Байеса, которая позволяет наше исходное незнание или знание о неизвестной величине, априорное знание, трансформировать в апостериорное после наблюдения некоторых косвенных характеристик, как-то косвенно характеризующих неизвестную величину θ.

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$$

Ключевая особенность формулы – в том, что на вход мы подаем априорное распределение, кодирующее наше незнание или нашу неопределенность о неизвестной величине, и что выходом также является распределение. Это крайне важный момент. Не точечная оценка, а некая сущность того же формата, что был на входе. Благодаря этому становится возможным, например, использовать результат байесовского вывода, апостериорное распределение, как априорное в какой-то новой вероятностной модели и, таким образом, охарактеризовать новую неизвестную величину с разных сторон путем анализа ее различных косвенных проявлений. Это первое достоинство, благодаря которому удается получить свойство расширяемости – или композицируемости – разных вероятностных моделей, когда мы можем из простых моделей строить более сложные [2].

Второе интересное свойство. Простейшее правило суммирования произведения вероятностей означает: если у нас есть вероятностная модель – а другими словами, совместное вероятностное распределение на все переменные, возникающие в нашей задаче, – то мы, как минимум в теории, всегда можем построить любой вероятностный прогноз, спрогнозировать интересующую нас переменную U, зная какие-то наблюдаемые переменные O. При этом есть переменная L, которую мы не знаем и она нас не интересует. По этой формуле они отлично исключаются из рассмотрения.

$$p(U|O) = \frac{\int p(U,O,L)d\,L}{\int p(U,O,L)d\,LdU}$$

Для любых сочетаний этих трех групп переменных мы всегда можем построить такое условное распределение, которое и укажет, как изменились наши представления об интересующих нас величинах U, если мы пронаблюдали величины O, предположительно связанные с U.

II. Сущность и свойства байесовского подхода

По сути, повсеместное применение формул Байеса дало рождение второму альтернативному подходу к теории вероятностей. Есть классический подход – на Западе нередко называемый частотным– и есть альтернативный байесовский подход. Естественно, они друг другу не противоречат. Они скорее друг друга дополняют. По этой табличке можно проследить, что у них общего и в чем различия.

Ключевое различие в том, что понимать под случайной величиной. В частотных терминах мы под случайной величиной понимаем величину, значение которой мы спрогнозировать не можем, не оценив какие-то статистические закономерности. Нужно что-то обладающее объективной неопределенностью, в то время как при байесовском подходе случайная величина интерпретируется просто в качестве детерминированного процесса. Он может быть полностью спрогнозирован.

Просто в данном детерминированном процессе мы не знаем часть факторов, влияющих на исход. Поскольку мы их не знаем, мы не можем спрогнозировать исход детерминированного процесса. Значит, для нас данный исход выглядит как случайная величина.

Простейший пример – подбрасывание монетки. Речь идет о классической случайной величине, но мы понимаем, что монетка подчиняется законам классической механики и, вообще-то, зная все начальные условия – силу, ускорение, коэффициент сопротивления среды и т.д. – мы могли бы точно сказать, как монетка упадет: орлом или решкой [5].

Если вдуматься, подавляющее большинство величин, которые мы привыкли считать случайными, на самом деле случайны именно в байесовском смысле. Это какие-то детерминированные процессы, просто мы не знаем часть факторов этих процессов.

Поскольку один может не знать одни факторы, а другой – другие, возникает понятие субъективной неопределенности или субъективного незнания.

Остальное – прямые различия этой интерпретации. Все величины в байесовском подходе можно трактовать как случайные. Аппарат теории вероятностей применяется к параметрам распределения случайной величины. Другими словами, то, что в классическом подходе бессмысленно, в байесовском подходе обретает смысл.

Оценки получаем не точечные, а вида апостериорного распределения, позволяющего нам комбинировать разные вероятностные модели. И в отличие от частотного подхода – теоретически обоснованного при больших n, а некоторые доказывают, например, при n, стремящихся к бесконечности, – Байесовский подход верен при любых объемах выборки, даже если n = 0. Просто в данном случае апостериорное распределение совпадет с априорным.

III. Возможности регуляризации

Другое достоинство байесовского подхода, уже применительно к машинному обучению, – регуляризация. Благодаря учету априорных предпочтений мы препятствуем излишней настройке наших параметров в ходе процедуры машинного обучения и тем самым способны справляться с эффектом переобучения. Какое-то время назад, когда алгоритмы начали обучать на огромных объемах данных, считалось, что проблема переобучения снята с повестки дня. Но дело было исключительно в том, что люди психологически боялись переходить к нейросетям гигантского размера. Все начинали с небольших нейросетей, и они, при гигантских обучающих выборках, действительно не переобучались. Но по мере того, как психологический страх исчезал, люди начинали использовать сети всё большего размера.

Стали очевидными две вещи. Для начала – чем больше сеть, тем она в принципе лучше. Большие сети работают лучше, чем маленькие. Но большие сети начинают переобучаться. Если у нас количество параметров – 100 млн, то 1 млрд объектов – не очень большая обучающая выборка, и нам необходимо регуляризовывать процедуру такого машинного обучения [3].

Байесовский подход как раз дает прекрасную возможность делать это за счет введения априорного распределения на те параметры, на те веса нейронной сети, которые настраиваются в ходе процедуры обучения.

В частности, оказалось, что такая популярная техника эвристической регуляризации, как drop out, является частным случаем, грубым приближением для байесовской регуляризации. На самом деле речь идет о попытке сделать байесовский вывод.

IV. Перспективы применения модели с латентными переменными

Наконец, третье преимущество – возможность построения модели с латентными переменными. О ней подробнее.

У нас есть мотивирующий пример – метод главных компонент. Метод очень простой – линейное уменьшение размерности. Берем выборку в пространстве с высокой размерностью, строим ковариационную матрицу, проецируем на главную ось с соответствующим самым большим значением. Вот что геометрически здесь показано. И уменьшили размерность пространства с 2 до 1, сохранив максимум дисперсии, содержавшейся в выборке.

Метод простой, допускает решение в явном виде. Но можно альтернативно сформулировать иначе, в терминах вероятностной модели.

Представим, что наши данные устроены так: для каждого объекта есть его скрытое представление в пространстве маленькой размерности. Здесь оно обозначено как z. А мы наблюдаем линейную функцию от этого скрытого представления в пространстве с более высокой размерностью. Мы взяли линейную функцию и дополнительно добавили нормальный шум. Тем самым мы получили x, высокоразмерные данные, по которым нам крайне желательно восстановить их низкоразмерное представление [6].

$$x \in \mathbb{R}^D, z \in \mathbb{R}^d, \text{при } D \gg d$$

Математически это выглядит так. Мы задали вероятностную модель, которая является совместным вероятностным распределением на наблюдаемые и скрытые компоненты, на x и z. Модель достаточно простая, выборка характеризуется произведением по объектам. Наблюдаемая компонента каждого объекта определяется скрытой компонентой, речь идет об априорном распределении на скрытой компоненте. И то и другое является нормальным распределением. Мы предполагаем, что в малоразмерном пространстве данные распределены априорно нормально, и наблюдаем линейную функцию этих данных, которая зашумлена нормальным шумом.

$$p(X,Z|\theta) = \prod_{i=1}^{n} p(x_i|z_i,\theta)p(z_i|\theta)$$
$$= \prod_{i=1}^{n} \mathrm{N}(x_i|Vz_i,\sigma^2 I)\mathrm{N}(z_i|0,I)$$

Нам задана выборка, представляющая собой высокоразмерное представление. Мы знаем x, мы не знаем z, и наша задача – найти параметр θ. θ – это матрица V, σ^2 и всё.

Указанную задачу можно сформулировать на байесовском языке как задачу обучения с латентными переменными. Чтобы применить обычный метод максимального правдоподобия, не хватает знания увеличения z. Оказывается, для этой техники существует стандартный подход, основанный на EM-алгоритме и различных модификациях. Можно запустить итерационный процесс. На EM-шаге приведены формулы, описывающие, что мы делаем. Можно теоретически показать, что процесс монотонный и гарантированно сходится к локальному экстремуму, но все-таки.

Возникает вопрос: а зачем нам применять итерационный процесс, когда мы знаем, что задача решается в явном виде?

Ответ простой. Для начала – алгоритмическая сложность. Сложность аналитического решения – $O(nD^2)$, в то время как сложность одной итерации EM-алгоритма – O(nDd).

Если мы мысленно представим, что проецируем пространство размерностью 1 млн в пространство размерностью 10, и EM-алгоритм сходится итераций за сто, то наша итерационная схема будет работать в 1000 раз быстрее, чем решение в явном виде [5].

Кроме того, важное достоинство: теперь мы нашу базовую модель, метод главных компонент, можем различными способами расширять в зависимости от специфики конкретной задачи. Например, мы можем ввести понятие смеси методов главных компонент, и сказать, что наши данные живут не в одном пространстве, линейном подпространстве меньшей размерности, а в нескольких. И мы не знаем, из какого подпространства пришел каждый конкретный объект.

Возникает смесь методов главных компонент. Формально модель записывается так. Просто ввели дополнительную номенклатуру скрытых дискретных переменных t. И опять же, EM-алгоритм позволяет нам найти решение задачи почти по тем же формулам, хотя исходный метод главных компонент нам не позволял производить никакие модификации.

Еще одно – возможность работать в ситуациях, когда, допустим, для нашей выборки часть компонент x неизвестны, данные пропущены.

Несколько более экзотическая ситуация – когда нам известны скрытые представления части объектов, низкоразмерное представление, целиком или частично. Но опять же, такая ситуация возможна.

Всходная модель метода главных компонент не способна учесть ни того, ни другого, в то время как вероятностная модель, сформулированная на байесовском языке, и то, и другое учитывает элементарно – простой модификацией EM-алгоритма [1]. Мы просто немного меняем номенклатуру наблюдаемых скрытых переменных.

Тем самым мы можем решать задачи обработки данных, когда какие-то произвольные фрагменты этих данных могут отсутствовать, не то что в стандартном машинном обучении, когда выделяется номенклатура наблюдаемых и как бы скрытых, целевых переменных и перемешиваться они никак не могут. Здесь возникает дополнительная гибкость.

V. Масштабирование байесовского метода

Наконец, до недавнего времени считалось, что существенное ограничение байесовских методов состоит в том, что они, обладая высокой вычислительной сложностью, применимы к небольшим выборкам данных и не переносятся на большие данные. Результаты последних нескольких лет показывают, что это не так. Человечество наконец-то научилось обеспечивать масштабируемость байесовских методов. И люди тут же начали скрещивать байесовские методы с глубинными нейронными сетями.

Теорема Байеса. Мы хотим по известным X, M сказать что-то про Z, который как-то связан с X. Мы применяем теорему Байеса. Вопрос: какое здесь самое уязвимое место, какой самый тяжелый фрагмент- это интеграл.

$$p(Z|X) = \frac{p(X,Z)}{p(X)} = \frac{p(X,Z)p(Z)}{\int p(X|Z)p(Z)dZ}$$

В тех редких случаях, когда интеграл берется аналитически, все хорошо. Другое дело, если он аналитически не берется – а ведь мы представляем, что говорим о высокоразмерных данных, и тут интеграл в пространстве размерностью не 1 или 2, а десятки и сотни тысяч.

С другой стороны, давайте запишем цепочку: $\int p(X|Z)\, p(Z)\, dZ$. Поскольку $\log P(X)\, dZ$ не зависит, это просто интеграл по всему Z, равный 1.

$$\log_p X = \int q(Z)\log_p(X)dZ = \int q(Z)\log_p \frac{p(X,Z)}{p(Z|X)}dZ$$

Второе действие. Выразили p(X), по этой формуле перенесли влево, p(Z|X) в знаменатель, записали под интегралом [6].

Теперь у нас числитель и знаменатель зависят от Z, хотя их частное дает p(X), то есть не зависит от Z. Тождество.

Дальше умножили то, что стоит под log, умножили на 1 и поделили на q(Z).

И последнее – разбили интеграл на две части.

$$\int q(Z)\log_p \frac{p(X,Z)q(Z)}{q(Z)p(Z|X)}dZ$$
$$= \int q(Z)\log_p \frac{p(X,Z)}{q(Z)}dZ + \int q(Z)\log_p \frac{q(Z)}{p(Z|X)}dZ = \mathcal{L}(q) + KL(q(Z)\|p(Z|X))$$

Вторая часть – хорошо известная в теории вероятностей дивергенция Кульбака-Лейблера, величина неотрицательная и равная нулю тогда и только тогда, когда эти два распределения совпадают между собой. В каком-то смысле это аналог расстояния между распределениями. А напоминаю, наша задача оценить p(Z|X) хотя бы приближенно, выполнить байесовский вывод.

Эту величину мы посчитать не можем, здесь фигурирует p(Z|X). Зато отлично можем посчитать первое слагаемое. Каждое из слагаемых зависит от q(Z), но их сумма от q(Z) не зависит, потому что она равна log p(X). Возникает идея: а давайте мы будем первое слагаемое максимизировать по q(Z), по распределению.

Это означает, что, максимизируя первое слагаемое, мы минимизируем второе – которое показывает степень приближения q(Z) к истинному апостериорному распределению. И вот, тем самым, ключевая идея: мы свели задачу байесовского вывода, включающую в себя интегрирование в пространстве высокой размерности, к задаче оптимизации.

VI. Вариационный байес для задачи оптимизации

Задачу оптимизации человечество умеет решать хорошо даже при гигантских объемах данных. Такой подход называется вариационный Байес. Ключевая идея – что вывод становится оптимизацией.

Какие у нас есть универсальные анализаторы, аппроксиматоры? Особенно пригодные для работы с большими данными? Это нейронные сети.

Здесь уже был сделан следующий шаг – обучение по неполной разметке. Дано Xtr, Z мы не знаем, и нам бы оптимизировать этот функционал по θ. Мы даже не можем его в явном виде посчитать, но зато, когда мы заменили его на вариационную нижнюю оценку L, появилась еще и зависимость от θ – поскольку мы хотим дополнительно левую часть оптимизировать по θ.

Мы можем оптимизировать L одновременно по Q(Z) и по θ. Оптимизация по θ позволяет нам всё лучше описывать обучающую выборку, а оптимизация по Q(Z) позволяет всё точнее проводить байесовский вывод над скрытыми переменными Z.

$$\log_p(X_{tr}|\theta) \geq \int q(Z)\log\frac{p(X_{tr},Z|\theta)}{q(Z)}dZ = \mathcal{L}(q,\theta)$$
$$\frac{\partial\mathcal{L}(q,\theta)}{\partial\theta} \approx \frac{\partial\log_p(x_i,z_i|\theta)}{\partial\theta}$$

где $z_i \sim q_i(z_i)$ и $q(Z) = \prod_{i=1}^{N} q_i(z_i)$

Среди прочего оказывается, что такая модель допускает эффективную оптимизацию с помощью методов стохастической оптимизации. Они позволяют оптимизировать функцию, которую мы, может, даже не сможем посчитать ни в одной точке. Нам достаточно уметь считать для нее стохастический градиент и, быть может, какие-то дополнительные характеристики для дальнейшего ускорения сходимости [1].

Здесь оказывается, что никаких проблем нет. Стохастический градиент для вариационной нижней оценки может быть выражен по такой формуле, где мы сгенерировали Z из текущего Q(Z). Ну и всё, дальше можно дифференцировать.

Нейробайесовский подход родился в тот момент, когда мы поняли, как эту дисперсию можно уменьшить, – потому что затем стало возможным применять байесовские техники в глубинном обучении и, как следствие, успешно решать новый спектр задач.

Остановимся на одной из них. Речь идет о вариационном автокодировщике – он представляет собой сравнительно прямолинейное обобщение методов главных компонент в его байесовской интерпретации. Вспоминаем: наверху формула для методов главных компонент. Есть скрытые переменные, априорно имеющие нормальное распределение в пространстве маленькой размерности, и есть наблюдаемые компоненты, которые просто являются линейной функцией от скрытых компонент. Плюс нормальный шум.

Казалось бы, давайте сделаем всё то же самое, только пусть будет не линейная функция, а что-нибудь более хитрое. Здесь так и сделано, μ и σ – теперь не линейные функции от латентного представления данного объекта. Пусть это будет выходом нейронной сети. Есть нейронная сеть, которая на вход получает Zi и выдает значение μ и σ, которое показывает распределение соответствующего Xi. Данная нейронная сеть имеет веса θ. В остальном модель та же самая. Единственное, мы вместо линейной функции поставили сюда нелинейную, определяемую нейронной сетью.

$$\mathcal{L}(\phi,\theta) = \int q(Z|X,\phi)\log\frac{p(X,Z|\theta)}{q(Z|X,\phi)}dZ$$

Проблема в следующем: в этой задаче строгий байесовский вывод, в отличие от метода главных компонент, уже сделать нельзя. Поэтому здесь приходится использовать ту вариационную технику, о которой я рассказал раньше. Мы переходим к вариационной оценке L, вводим распределение Q(Z), которое параметризуется через φ, и пусть это будет другая нейронная сеть. Чем богаче семейство распределений, где мы проводим оптимизацию, тем лучше. Самое главное, что мы умеем оптимизировать на сегодняшний день, – нейронные сети, обладающие достаточной гибкостью [3].

Добавим вспомогательную нейронную сеть. У нее будут свои параметры φ. Сеть, которая принимает Z, возвращает X. Эта сеть – наоборот, принимает X и возвращает распределение на Z. Такая техника очень

похожа на то, что в нейронных сетях известно, как модель автокодировщика. Только здесь все сформулировано на вероятностном языке, поэтому ее назвали вариационным автокодировщиком. Для начала оказалось, что такая модель может быть обучена крайне эффективно. Значит, ее можно обучать на гигантских выборках данных. Чем больше выборки, тем выше качество аппроксимации и тем лучше мы решаем задачу уменьшения размерности, поиска низкоразмерных латентных представлений, выучивания представлений. В глубинном представлении их называют representations.

VII. Заключение

Байесовский подход был достаточно популярен в 90-е и нулевые годы – до того, как началась глубокая революция, вызванная триумфальным шествием глубинных нейронных сетей. Какое-то время казалось: зачем все эти байесовские методы нужны, у нас нейросети и так прекрасно работают. Но как часто бывает, в какой-то момент выяснилось, что можно объединить преимущества нейросетевого и байесовского подходов. В первую очередь – благодаря тому, что появились техники вариационного байесовского вывода, и эти модели не противоречат друг другу, а наоборот, прекрасно дополняют, взаимно усиливая друг друга. В каком-то смысле можно воспринимать это как направление дальнейшего развития современного машинного обучения и глубинного обучения. Важно понимать, что нейронные сети не являются панацеей. Они – всего лишь важный шаг в правильном направлении, но далеко не последний шаг.

Список литературы

[1] Bańbura M., Giannone D., Reichlin L. Large Bayesian vector auto regressions. Journal of Applied Econometrics, 2010, vol. 25 (1), p. 71–92.

[2] Carriero A., Kapetanios G., Marcellino M. Forecasting government bond yields with large Bayesian vector autoregressions. Journal of Banking & Finance, 2012, vol. 36 (7), p. 2026–2047.

[3] De Mol C., Giannone D., Reichlin L. Forecasting using a large number of predictors: Is Bayesian shrinkage a valid alternative to principal components? Journal of Econometrics, 2008, vol. 146 (2), p. 318–328.

[4] Koop G., Korobilis D. Bayesian Multivariate Time Series Methods for Empirical Macroeconomics. Foundations and Trends (R) in Econometrics, 2010, vol. 3 (4), p. 267–358.

[5] Litterman R. Forecasting with Bayesian vector autoregressions – five years of experience. Journal of Business & Economic Statistics, 1986, vol. 4 (1), p. 25–38.

[6] Shevelev A. A. Bayesian Approach to Evaluate the Impact of External Shocks on Russian Macroeconomics Indicators. World of Economics and Management, 2017, vol. 17, no. 1, p. 26–40.

[7] Sims C. A., Zha T. Bayesian methods for dynamic multivariate models. International Economic Review, 1998, vol. 39 (4)