Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет
«Высшая школа экономики»

Факультет математики

Гончаренко Аркадий Александрович

Применение альтернативной модификации алгоритма Демпстера к снижению размерности данных

Курсовая работа студента 3 курса
образовательной программы бакалавриата «Математика»

Научный руководитель:
PhD,
Вайнберг Аллен Анна Львовна

Москва 2024

Введение.

В данной работе рассматривается реализация ускоренного алгоритма Демпстера из [1]. Его суть состоит в построении графа (и ковариационной матрицы) на основе многомерных нормальных данных для нахождения зависимостей между переменными. Алгоритм Демпстера строит такой граф $G(V, E)$, где вершины V отвечают осям координат, и ковариационную матрицу $\Sigma = (\sigma_{ij})$, что для любой пары разных элементов (i, j), не соединенных ребром, их коэффициент (σ^{ij}) в матрице $\Sigma^{-1} = (\sigma^{ij})$ равен 0, что означает, что эти две переменные частично независимы – независимы при фиксированных значениях всех остальных переменных.

Это позволяет отсеивать условные зависимости, созданные шумом, что может быть очень полезно при изучении связей между переменными.

Недостаток алгоритма Демпстера состоит в том, что подобный перебор требует много операций, так как матрицу с максимальным правдоподобием он находит методом, аналогичным методу градиентного спуска, а для выбора ребра для добавления в граф на каждом шаге приходится перебирать все рёбра. Поэтому в 2008 году Аллен [2] предложила ускоренный алгоритм Демпстера, который строил деревья. В данной работе я взял комбинацию этих двух подходов и протестировал её на разных данных.

Краткое описание алгоритма.

Алгоритм состоит из трех частей:

1. Выбор нового ребра для добавления в граф – это было предложено Аллен в [2] для ускорения алгоритма.

2. Поиск такой ковариационной матрицы $\hat{\Sigma} = (\hat{\sigma}_{ij})$ d-мерной выборки размера n, которая отвечает следующим условиям:

   (a) $\hat{\Sigma}$ совпадает с выборочной ковариационной матрицей $\overline{\Sigma} = \frac{1}{n-1}\sum_{i=1}^{d}(x_i - \bar{x})^\top(x_i - \bar{x})$ на диагонали и элементах, отвечающих ребрам графа G.

   (b) $\hat{\Sigma}^{-1} = (\hat{\sigma}^{ij})$ равна 0 во всех элементах, отвечающим парам вершин G, которые не соединены ребром.

3. Проверка значимости изменения матрицы за счет добавления последнего ребра – если изменение мало, то добавленное ребро может быть выбрано из-за шума несмотря на условную независимость соединенных этим ребром величин.

Выбор ребра.

Здесь я руководствуюсь [2]. На каждом шаге с помощью алгоритма Крускала я добавляю ребро с наибольшим модулем коэффициента корреляции Пирсона, которое не создает циклов в графе. Таким образом можно не перебирать все ребра. Отсутствие циклов позволяет избежать основной проблемы выбора нового ребра по коэффициентам корреляции – две условно независимые величины X_i и X_j, сильно коррелирующие с одинаковыми величинами, не будут связаны ребром, если не будет сильной погрешности в выборочных ковариациях, даже если коэффициент корреляции между X_i и X_j из-за этой косвенной связи окажется большим.

Обозначения.

Для удобства вычислений будем считать, что $\bar{x} = 0$. При другом значении $\bar{x}$ можно заменить в последующих вычислениях x на $x - \bar{x}$.

Давайте введем обозначения: многомерное нормальное распределение можно представить как член экспоненциального семейства

$$f(x,\Phi) = exp(\phi + t(x) + \phi_1 t_1(x) + \phi_2 t_2(x) + \ldots + \phi_r t_r(x)), \quad (1)$$

где $r = \frac{d(d-1)}{2},\ t(x) = 0,\ \phi = \frac{1}{2}(-p\log(2\pi) - \log(|\Sigma|)),\ \Phi = (\phi_1, \ldots, \phi_r) =$
$= (\sigma^{11}, \ldots, \sigma^{1d}, \sigma^{22}, \ldots, \sigma^{2d}, \ldots, \sigma^{dd}),\ t_1 = -\frac{x_1^2}{2},\ t_2 = -x_1x_2,\ \ldots,\ t_d = -x_1x_d,$
$t_{d+1} = -\frac{x_2^2}{2},\ \ldots,\ t_r = -\frac{x_d^2}{2}.$

Введем также

$$\theta_i = \int t_i(x) f(x,\Phi) dx. \quad (2)$$

Тогда $\Theta = (\theta_1, \ldots, \theta_r) = (-\frac{\sigma_{11}}{2}, -\sigma_{12}, \ldots, -\frac{\sigma_{dd}}{2})$

Таким образом, между множествами возможных Σ, Θ и Φ многомерного нормального распределения существует биекции, отвечающие конкретному нормальному распределению (с точностью до среднего). Другими словами, Σ, Θ и Φ однозначно определяют друг друга.

Пусть $\Gamma = (\gamma_{ij})$, $\gamma_{ij} = \int (t_i(x) - \theta_i)(t_j(x) - \theta_j) f(x,\Phi) dx$.

Это равно ковариации $\mathrm{cov}(x_{i_1}x_{i_2}, x_{j_1}x_{j_2}) = \sigma_{i_1,j_1}\sigma_{i_2,j_2} + \sigma_{i_1,j_2}\sigma_{i_2,j_1}$, умноженной на $1/2$, если только $i_1 = i_2$, или только $j_1 = j_2$, и на $1/4$, если оба условия выполняются.

Лемма. $\gamma_{ij} = \partial\theta_i/\partial\phi_j$.

◀ Так как f – плотность вероятности, $\int f(x,\Phi)dx = 1$. Выразим $f(x,\Phi)$ через (1) и возьмем дифференциал обоих частей равенства по Φ. Получим

$$d\phi + \left(\int t_1(x) f(x,\Phi) dx\right) d\phi_1 + \ldots + \left(\int t_r(x) f(x,\Phi) dx\right) d\phi_r = 0 \quad (3)$$

Выразим $f(x,\Phi)$ в (2) через (1) и возьмем дифференциал обоих частей равенства по Φ. Получим

$$\left(\int t_i(x) f(x,\Phi) dx\right) d\phi + \left(\int t_i(x) t_1(x) f(x,\Phi) dx\right) d\phi_1 + \ldots + \\ + \left(\int t_i(x) t_r(x) f(x,\Phi) dx\right) d\phi_r = d\theta_i \quad (4)$$

Умножим обе части (3) на $\int t_i(x) f(x,\Phi) dx$ и вычтем их из (4). Благодаря (2) получим

$$\gamma_{i1} d\phi_1 + \ldots + \gamma_{ir} d\phi_r = d\theta_i \quad (5)$$

Из (5) следует $\gamma_{ij} = \partial\theta_i/\partial\phi_j$. ▶

Поиск ковариационной матрицы.

Демпстер в [1] доказал следующее утверждение:

Теорема. Пусть $\hat{\Sigma}$ – симметричная матрица, совпадающая с выборочной матрицей ковариаций на диагонали и парах коэффициентов, отвечающих ребрах графа, обратная $\hat{\Sigma}^{-1}$ к которой равна 0 на всех парах разных коэффициентов, не отвечающим ребрам графа. Пусть она положительно определена. Тогда

1. $\hat{\Sigma}$ существует и единственна, если существует положительно определенная симметричная матрица, совпадающая с выборочной матрицей ковариаций на диагонали и парах коэффициентов, отвечающих ребрах графа.

2. Среди матриц, совпадающих с выборочной матрицей ковариаций на диагонали и парах коэффициентов, отвечающих ребрах графа, $\hat{\Sigma}$ имеет максимальную энтропию.

3. Среди ковариационных матриц, обратные к которым равны 0 на всех парах разных коэффициентов, не отвечающим ребрам графа, $\hat{\Sigma}$ является максимально правдоподобной оценкой.

Таким образом $\hat{\Sigma}$ является оценкой максимального правдоподобия среди матриц с выбранными нами частичными независимостями.

Эта часть алгоритма занимается поиском $\hat{\Sigma}$.

Искать $\hat{\Sigma}$ мы будем с помощью метода Ньютона для нелинейных уравнений, аналогичному градиентному спуску в машинном обучении.

Разделим Φ на (Φ_A, Φ_B), где Φ_B – все компоненты, которые соответствуют парам разных вершин без ребер между ними, соответственно Φ_B приравняем нулю. Аналогично разделим Θ на (Θ_A, Θ_B), соответственно будем использовать алгоритм, который приближает Θ_A к значениям, соответствующим выборочной матрице. Аналогично разделим матрицу $\Gamma = (\gamma_{ij})$ на 4 подматрицы Γ_{AA}, Γ_{AB}, Γ_{BA} и Γ_{BB}:

$$\Gamma = \begin{bmatrix} \Gamma_{AA} & \Gamma_{AB} \\ \Gamma_{BA} & \Gamma_{BB} \end{bmatrix}$$

Пусть Σ_i – матрица на i-м шаге, Φ_A^i и Θ_A^i – соответствующие ей векторы, Γ^i – соответствующая ей матрица Γ. Пусть Θ_A^0 – значение Θ_A, соответствующее выборочной матрице. Пользуясь Леммой, разложим Θ_A в ряд Тейлора как функцию от Φ_A:

$$\Theta_A^0 = \Theta_A^i + \Gamma_{AA}^i(\Phi_A^0 - \Phi_A^i) + o(\Phi_A^0 - \Phi_A^i) \tag{6}$$

Применим метод Ньютона к отображению $\Theta_A(\Phi_A)$, чтобы найти точку, в которой $\Theta_A = \Theta_A^0$: $\Phi_A^i + s$ возьмем в качестве Φ_A^{i+1}, где s определяется уравнением $\Theta_A^0 - \Theta_A^i = \Gamma_{AA}^i s$. Пользуясь тем, что $\Phi^{i+1} = \Sigma^{-1}$ для соответствующего распределения, вычислим Θ_A^{i+1} и Γ_{AA}^{i+1}, соответствующие Φ^{i+1}.

Будем повторять процесс, пока изменение s не станет достаточно маленьким, что произойдет, когда Θ_A^i будет находиться в окрестности Θ_A^0. Получим приближение $\hat{\Sigma}$.

Проверка значимости.

Точных тестов значимости не существует, но в [1] предлагается χ^2- тест с одной степенью свободы для умноженного на удвоенный размер выборки изменения в логарифме правдоподобия. В [3] рекомендуется делить уровень значимости на количество ещё не добавленных ребер, включая последнее добавленное. Вычислим логарифм правдоподобия.

Предложение. Логарифм правдоподобия равен $-\frac{1}{2}(p\log(2\pi)+\log(|\Sigma|)+tr(\hat{\Sigma}\Sigma^{-1}))$

◀ $\log(f(x,\Phi))$ по (1) равно

$$\phi + t(x) + \phi_1 t_1(x) + \phi_2 t_2(x) + \ldots + \phi_r t_r(x) = \frac{1}{2}(-p\log(2\pi) - \log(|\Sigma|)) + \Phi t(x) =$$

$$= \frac{1}{2}(-p\log(2\pi) - \log(|\Sigma|)) - \frac{1}{2}\sum_{i,j=1}^{d}(\Sigma_{i,j}^{-1}\hat{\Sigma}_{i,j}) = -\frac{1}{2}(p\log(2\pi) + \log(|\Sigma|) + tr(\hat{\Sigma}\Sigma^{-1})) \blacktriangleright \tag{7}$$

Эксперименты и результаты.

К сожалению, с теоретической стороны вопроса алгоритм работает, лишь если распределние является многомерным нормальным распределением. Это значит, что необходимо проверять данные на многомерную нормальность. Для этого я руководствовался статьей [5]. Я выбрал BHEP-тест, как наиболее универсальный из описанных тестов.

Прилагаю реализацию BHEP-теста присутствует в [7]. К сожалению, многие данные этот тест не прошли, например, данные по продажам авокадо, опубликованные в [6], имели 8 потенциально нормальных переменных, но тест не проходили, пока я не оставил всего 4 переменных, что я счел слишком малым для оценки эффективности алгоритма. К тому же невозможно оценить эффективность алгоритма, не зная, какие пары переменных на самом деле частично независимы относительно остальных переменных.

Поэтому я взял матрицу ковариаций шести переменных из [2]:

$$\Gamma = \begin{bmatrix} 1 & 0.3966 & 0.3688 & 0.1764 & -0.4632 & 0.2939 \\ 0.3966 & 1 & 0.0232 & -0.0854 & 0.0193 & 0.2191 \\ 0.3688 & 0.0232 & 1 & 0.0494 & -0.1350 & -0.2376 \\ 0.1764 & -0.0854 & 0.0494 & 1 & -0.4671 & 0.1135 \\ -0.4632 & 0.0193 & -0.1350 & -0.4671 & 1 & -0.3656 \\ 0.2939 & 0.2191 & -0.2376 & 0.1135 & -0.3656 & 1 \end{bmatrix}$$

После этого я создал программу(выложена в [7]), которая с выбранной вероятностью p_0 обнуляет элементы матрицы (каждое обнуление независимо), генерирует по получившейся матрице выборку(я выбрал размер выборки, равный 720) и строит граф по алгоритму Демпстера. Затем я провел эту процедуру для каждого значения p_0.Прилагаю таблицу результатов:

p_0	I	II	Error	Full Tree
0.01	137	109	5	19
0.02	74	100	0	16
0.05	21	135	2	1
0.1	12	118	0	2
0.2	0.005	0.124	0	1
0.5	0.001	0.113	0	1
0.9	0	122	0	1

I – Количество ребер, не прощедших проверку значимости несмотря на отсутствие условной независимости

II– Количество ошибок второго рода проверки значимости

Error – Количество ошибок реализации алгоритма

Full Tree – Количество раз алгоритм создал дерево, связывающее все вершины. **Выводы** Как видно из стобца II, алгоритм имеет приблизительно 0.1 вероятность включить ребро, где его не должно быть, и останавливается раньше нужного в небольшом числе случаев, причем чем больше условно независимых пар переменных, тем менее вероятно это произойдет.

Из этого можно сделать вывод, что если ожидается, что есть много условных независимостей, повышение строгости критерия независимости позволит уменьшить количество неверно включенных ребер.

Направления для будущих исследований. С момента изобретения изначального алгоритма прошло 50 лет. За это время возможности компьютеров

выросли многократно. Поэтому можно написать изначальный алгоритм Демпстера, перебирающий все ребра, например, и оценить его эффективность и практичность в сравнении с этим алгоритмом.

С теоретической стороны вопроса основной проблемой является необходимость многомерной нормальности данных. Однако многие, но не все, утверждения Демпстера в его статье применимы и к произвольному экспоненциальному семейству, так что основным направлением исследований я бы назвал расширение результатов Демпстера на другие распределения экспоненциального семейства.

Также можно заняться изучением того, как алгоритм Демпстера будет работать вне многомерного нормального распределения, например, если все переменные нормальны, но совместного нормального распределения не образуют.

Приложение 1.Псевдокод

Это сокращенный код, полный код указан в [7].

Algorithm 1. Поиск ребра – Алгоритм Крускала

$S = (s_{ij}) = \frac{1}{n-1} \sum_{k=1}^{n} (x_{i,k} - \bar{x}_i)(x_{j,k} - \bar{x}_j)$
$A = \emptyset$
for all $i \in \{1, 2. \ldots, d\}$ **do** A.append(i,i)
while $P \neq 0$ **do**
    $P = 0$
    $corr_{max} = -1$
    for all $i, j \in \{1, 2, \ldots, d\}$ **do**
        if вершины i и j не принадлежат одной компоненте связности графа A и $|corr_{i,j}| > corr_{max}$ **then**
            $P = (i, j)$
            $corr_{max} = |corr_{i,j}|$
    if $P \neq 0$ **then**
        A.append(P)
        Поиск ковариационной матрицы
        Проверка значимости

Algorithm 2. Поиск ковариационной матрицы

for all $a \in A$ **do**
  if $a = (i, i)$ **then**
    Θ^0_A.append$(-\frac{1}{2}S_a)$
  else
    Θ^0_A.append$(-S_a)$
$\Phi^i_A = []$
for all $a \in A$ **do**
  Φ^i_A.append(Σ_a^{-1})
for all $i, j \in \{1, \dots d\}$ **do**
  $\Sigma^{i-1}_{i,j} = I_A \Sigma_{i,j}$
$\Sigma^i = \Sigma(\Phi^i_A, 0)$
for all $a \in A$ **do**
  $\Theta^i_A(a) = -\Sigma^i_a$
  if $a_1 = a_2$ **then**
    $\Theta^i_A(a) /= 2$
$\delta = 100$
$\epsilon = 0.0001$
while $\delta > \epsilon$ **do**
  for all $a \in A$ **do**
    for all $b \in A$ **do**
      $\Gamma^i_{AA}(a, b) = \Sigma^i_{a_1 b_1} \Sigma^i_{a_2 b_2} + \Sigma^i_{a_1 b_2} \Sigma^i_{a_2 b_1}$
      if $a_1 = a_2$ **then**
        $\Gamma^i_{AA}(a, b) /= 2$
      if $b_1 = b_2$ **then**
        $\Gamma^i_{AA}(a, b) /= 2$
  $a = 0.1$
  $s = a{\Gamma^i_{AA}}^{-1}(\Theta_{A_0} - \Theta^i_A)$
  $\delta = \sum\limits_{a \in A} s_a^2$
  $\Phi^i_A += s$
  $\Sigma^i = \Sigma(\Phi^i_A, 0)$
  for all $a \in A$ **do**
    $\Theta^i_A(a) = -\Sigma^i_a$
    if $a_1 = a_2$ **then**
      $\Theta^i_A(a) /= 2$
Проверка значимости

Algorithm 3. Проверка значимости

$\delta_k = n(\log(|\Sigma^o|) - \log(|\Sigma^i|)) + (n-1)(S((\Sigma^o)^{-1}) - ((\Sigma^i - 1)^{-1})$
$\alpha = 0.05$
if $\mathbb{P}(\chi_1^2 > \delta_k) < \frac{\alpha}{d(d+1)/2-(|A|-1)}$ **then**
    $\Sigma^o = \Sigma^i$
    Продолжать искать новые ребра для добавления
else
    Убрать из графа ребро, добавленное последним, остановить модифицированный алгоритм Демпстера

Список литературы.

[1] Dempster, A. P. (1972). Covariance Selection. Biometrics, 28(1), 157–175.

[2] Weinberg Allen, Anna, 2008. "Graphical Models of Structural Relations between Variables and their Application to Russian Regions (Part One),"Applied Econometrics, Russian Presidential Academy of National Economy and Public Administration (RANEPA), vol. 10(2), pages 44-64, Weinberg Allen, Anna, 2008. "Graphical Methods of Structural Relations between Variables and their Application to Russian Regions (Part Two),"Applied Econometrics, Russian Presidential Academy of National Economy and Public Administration (RANEPA), vol. 12(4), pages 42-70.

[3] Miller R. G. J. Simultaneous statistical inference. Second Ed. Berlin, New York: Springer-Verlag, 1981.

[4] Baringhaus, L., Henze, N. A consistent test for multivariate normality based on the empirical characteristic function. Metrika 35, 339–348 (1988). https://doi.org/10.1007/BF02613322

[5] Norbert Henze (2002). Invariant tests for multivariate normality: a critical review. , 43(4), 467–506. doi:10.1007/s00362-002-0119-6

[6] https://www.kaggle.com/datasets/neuromusic/avocado-prices/data

[7] https://github.com/arkadgoncharenko/Dempster