КОНФИДЕНЦИАЛЬНОЕ ДВУСТОРОННЕЕ ВЫЧИСЛЕНИЕ ПАРАМЕТРОВ НЕЧЕТКОЙ ЛИНЕЙНОЙ РЕГРЕССИОННОЙ МОДЕЛИ

Волкова Е.С.[1], Гисин В.Б.[2]

Цель статьи: *представить алгоритм, исполнение которого позволяет двум его участникам найти коэффициенты при объясняющих переменных в рамках нечеткой линейной регрессионной модели при горизонтальном секционировании данных, сохраняя конфиденциальность данных.*

Метод: *при построении алгоритма используется трансформационный подход. Оптимизационные задачи каждого из двух участников трансформируются и соединяются в общую задачу, решение которой получает один из участников.*

Полученный результат. *Предложен протокол, при исполнении которого два пользователя получают модель нечеткой линейной регрессии на объединенных данных. Каждый из пользователей имеет набор данных о результатах наблюдений, содержащий значения объясняющих переменных и объясняемой переменной. Структура данных является общей: оба пользователя используют один и тот же набор регрессоров и общую критериальную переменную. Регрессионные коэффициенты ищутся в виде симметричных треугольных нечетких чисел путем решения соответствующей задачи линейного программирования. Предполагается, что оба пользователя являются полу-честными (честными, но любопытными, или пассивными и любопытными), т.е. они исполняют протокол, но могут попытаться извлечь информацию об исходных данных партнера, применяя к полученным данным произвольные методы обработки, не предусмотренные протоколом. Протокол описывает построение трансформированной задачи линейного программирования, решение которой находит один из пользователей. Число наблюдений каждого из пользователей известно обоим пользователям. Данные наблюдений остаются конфиденциальными. Доказана корректность протокола и обоснована его безопасность.*

Ключевые слова: *нечеткие числа, коллаборативное решение задачи линейного программирования, трансформационный подход, облачные вычисления, федеративное машинное обучение.*

DOI:10.21681/2311-3456-2021-3-11-19

1. Введение

Современные возможности по сбору, хранению и обработке данных выявили две тенденции. С одной стороны, большие объемы данных позволяют строить более точные и адекватные модели. С другой стороны, сбор и хранение данных может осуществлять большое число субъектов, и соответственно неимоверно выросло число хранилищ данных. Информация, необходимая для построения развитых моделей, оказывается фрагментированной и рассредоточенной по большому числу хранилищ. Желание владельцев данных сохранить конфиденциальность принадлежащей им информации, делает невозможным простое объединение данных. Сохранение конфиденциальности данных регулируется все более жесткими правовыми нормами.

Традиционная модель обработки данных, когда собранные пользователями данные передаются для обработки в некий общий центр, во многих случаях оказывается неработоспособной. Таким образом, в эпоху «больших данных» возникает проблема консолидированной обработки «малых данных», распределенных между конечными пользователями.

Одним из возможных решений возникшей проблемы служат вычисления, основанные на специальных криптографических протоколах. Подобные вычисления изучались с начала 80-х годов двадцатого столетия в рамках криптографии. Классическим примером может служить решение задачи о миллионерах. До последнего десятилетия конфиденциальные вычисления считались слишком неэффективными для широкого распространения и применялись для решения узкого класса специальных задач.

В последнее десятилетие прогресс в развитии вычислительной техники и совершенствование криптографических алгоритмов сделали безопасные вычисления применимыми для решения практических задач [1], в частности, задач регрессионного анализа с данными приемлемого объема [2].

1 Волкова Елена Сергеевна, кандидат физико-математических наук., доцент ФГОБУ ВО «Финансовый университет при Правительств Российской Федерации», Москва, Россия. E-mail: evolkova@fa.ru, ORCHID ID 0000-0001-9037-592X

2 Гисин Владимир Борисович, кандидат физико-математических наук., профессор ФГОБУ ВО «Финансовый университет при Правительств Российской Федерации», Москва, Россия. E-mail: vgisin@fa.ru, ORCHID ID 0000-0002-7269-0587

В настоящее время в разработке криптографических протоколов выделяются два направления: вычисления, сохраняющие конфиденциальность (PPC – privacy-preserving computations), и безопасные вычисления (SC - secure computations). Эти два направления различаются главным образом по тому, каковы угрозы конфиденциальности, целостности и доступности данных. Таким образом, механизмы обеспечения конфиденциальности в значительной степени определяются моделью угроз [3]. Участники протоколов конфиденциальных вычислений могут выступать в трех ролях: 1) владелец данных; 2) вычислитель; 3) потребитель результата.

По типу поведения участников принято относить к одной из двух групп: 1) полу-честные (пассивные, честные, но любопытные) участники; 2) злонамеренные (активные) противники. Полу-честные участники исполняют протокол, но стремятся извлечь из получаемых по протоколу данных дополнительную информацию конфиденциального характера. Злонамеренные противники могут отклоняться от исполнения протокола и, вообще, вести себя произвольным образом, в том числе, препятствуя исполнению протокола.

В настоящей работе мы рассматриваем протокол конфиденциального вычисления коэффициентов нечеткой линейной регрессии для двух полу-честных участников. Мы предполагаем, что участники I и II обладают данными наблюдений соответственно (X', y') и (X'', y''), где

$$X' = \left(x'_{ij}\right),\ x'_{i0} = 1,\ i = 1,\ldots,m_1,\ X'' = \left(x''_{ij}\right),\ x''_{i0} = 1,$$

$$i = 1,\ldots,m_2,\ j = 0,\ldots,n\ —$$

матрицы значений объясняющих переменных, а

$$y' = \left(y'_1,\ldots,y'_{m_1}\right)^T,\ y'' = \left(y''_1,\ldots,y''_{m_2}\right)^T,\ —$$

векторы значений зависимой переменной.

Положим

$$X = \begin{pmatrix} X' \\ X'' \end{pmatrix},\ y = \begin{pmatrix} y' \\ y'' \end{pmatrix},\ m = m_1 + m_2,$$

считая, что $x_{ij} = x'_{ij}$, $y_i = y'_i$ при $i = 1,\ldots,m_1$, и $x_{ij} = x''_{i-m_1 j}$, $y_i = y''_{i-m_1}$ при $i = m_1 + 1,\ldots,m$.

Задача нечеткой линейной регрессии состоит в подборе нечетких коэффициентов

$$\tilde{a}_0,\tilde{a}_1,\ldots,\tilde{a}_n,$$

так, чтобы расхождения между (четкими) значениями y_i и нечеткими значениями

$$\tilde{y}_i = \tilde{a}_0 + \tilde{a}_1 x_{i1} + \tilde{a}_2 x_{i2} + \ldots + \tilde{a}_n x_{in}$$

были приемлемо малыми.

Применение нечеткой линейной регрессии целесообразно в тех случаях, когда применение классической линейной регрессии оказывается недостаточно обоснованным [4]. Достаточно типичным примером является применение нечеткой линейной регрессии при оценке недвижимости, когда x_{ij} — значение фактора j на объекте i, а $\tilde{y}_i$ — цена объекта. Характерным для ситуаций, в которых используется модель нечеткой линейной регрессии, является сравнительно небольшое число наблюдений.

Наибольшее распространение и многочисленные применения нашел метод, при котором нечеткие коэффициенты ищутся в форме симметричных треугольных нечетких чисел [5].

В этом случае нечеткие коэффициенты $\tilde{a}_0,\tilde{a}_1,\ldots,\tilde{a}_n$, нужно подобрать так, чтобы для каждого i выполнялось условие

$$\mu_{\tilde{y}_i}\left(y_i\right) \geq h \qquad (1)$$

где $h \in [0,1]$ — некоторое заданное наперед пороговое значение, а $\mu_{\tilde{y}_i}$ — функция принадлежности нечеткой величины $\tilde{y}_i$, при условии, что суммарная нечеткость коэффициентов минимальна.

Если симметричное нечеткое число a_j задано своим центральным значением a_j и спрэдом α_j, условие (1) имеет следующий вид:

$$\sum_{j=0}^{n} x_{ij} a_j - (1-h)\sum_{j=0}^{n} |x_{ij}|\,\alpha_j \leq y_i \leq \sum_{j=0}^{n} x_{ij} a_j + (1-h)\sum_{j=0}^{n} |x_{ij}|\,\alpha_j.$$

Суммарная нечеткость по всем наблюдениям составляет

$$(1-h)\sum_{i=1}^{m}\sum_{j=0}^{n} |x_{ij}|\,\alpha_j.$$

Таким образом, построение нечеткой линейной регрессии сводится к следующей задаче линейного программирования:

$$I^T |X|\,\alpha \to min$$

при

$$\begin{pmatrix} X & -(1-h)|X| \\ -X & -(1-h)|X| \end{pmatrix}\begin{pmatrix} a \\ \alpha \end{pmatrix} \leq \begin{pmatrix} y \\ -y \end{pmatrix},\ \alpha \geq 0,$$

где

$$|X| = (|x_{ij}|),\ a = \left(a_0,\ldots,a_n\right)^T,\ \alpha = \left(\alpha_0,\ldots,\alpha_n\right)^T,$$

$$I = (1,1,\ldots 1)^T.$$

Имея в виду конфиденциальность, мы получаем задачу конфиденциального линейного программирования с данными, горизонтально разделенными между двумя участниками. С вычислительной точки зрения это принципиально отличает задачу построения нечеткой линейной регрессии от обычной линейной регрессии. Безопасное (и конфиденциальное) построение обычной линейной регрессии может рассматриваться как задача вертикального федеративного обучения [6], решаемая методом градиентного спуска. При таком подходе, конфиденциальность данных, принадлежащих участникам I и II, обеспечивается методами гомоморфного шифрования.

Конфиденциальное коллаборативное решение задачи линейного программирования также может быть выполнено с использованием криптографических протоколов гомоморфного шифрования. Как правило, криптографические алгоритмы используют симплекс-метод и целочисленные вычисления и имеют существенные для практических целей ограничения по числу итераций и объему данных [1]. Алгоритмы, в определенной степени свободные от этих ограничений, могут приводить

к существенной потере точности. Так или иначе, алгоритмы, основанные на криптографических протоколах безопасных вычислений, оказываются недостаточно эффективными.

Альтернативный путь конфиденциального решения задачи линейного программирования связан с так называемым трансформационным подходом. Трансформационный подход основан на том, что задача линейного программирования преобразуется так, чтобы замаскировать исходные данные, затем, после преобразования, задача решается стандартными методами, и из полученного решения извлекается решение исходной задачи. Этот подход был развит и проанализирован в серии работ. Проблемы с нарушениями конфиденциальности были в определенной степени решены в [7]. В [8] и [9] протоколы из цитированных работ были модифицированы, проведен анализ их безопасности и эффективности.

Заметим, что проблема коллаборативных конфиденциальных вычислений тесно связана с задачей сохранения конфиденциальных облачных вычислений. Трансформационный подход и методы безопасных криптографических вычислений в той или иной форме могут использоваться как в одном, так и в другом случае. В работах [10], [7], [11] представлены методы конфиденциального облачного решения задачи линейного программирования, основанные на трансформационном подходе. Обзоры работ этого направления представлены в [12] и [13]. Обзор [14] дает представление о современных подходах к аутсорсингу задач линейного программирования для облачных вычислений, основанному на криптографических протоколах. Эти подходы обеспечивают высокую безопасность, но пока еще недостаточно эффективны, если речь идет о решении задач со сравнительно невысокой ценой информации. К сходному заключению подводит и анализ проблемы аутсорсинга задач линейного программирования в [15].

2. Постановка задачи

Имея m результатов наблюдений $(\boldsymbol{x}_i, y_i)$, $i = 1,2,\ldots,m$, требуется оптимальным образом определить коэффициенты нечеткие $\tilde{a}_0, \tilde{a}_1, \supset, \tilde{a}_n$ так, чтобы расхождения между значениями y_i и $\tilde{\boldsymbol{a}}\boldsymbol{x}_i$ были приемлемо малыми. Последнее условие нуждается в уточнении и зависит от решаемой задачи.

Нечеткие коэффициенты $\tilde{a}_0, \tilde{a}_1, \supset, \tilde{a}_n$ нужно подобрать так, чтобы для каждой нечеткой величины

$$\tilde{y}_i = \tilde{\boldsymbol{a}}\boldsymbol{x}_i = \tilde{a}_0 + \tilde{a}_1 x_{i1} + \tilde{a}_2 x_{i2} + \ldots + \tilde{a}_n x_{in}, i = 1,2,\ldots,m \qquad (2)$$

выполнялись соотношения (1): $\mu_{\tilde{y}_i}(y_i) \geq h$, где h — некоторое заданное наперед пороговое значение, при условии, что возникающая нечеткость в определенном смысле минимальна.

Пусть коэффициенты $\tilde{a}_j$ представлены в виде симметричных треугольных нечетких чисел с центральным значением a_j и спрэдом α_j.

Задача построения нечеткой линейной регрессии сводится к следующей задаче линейного программирования:

$$\sum_{i=1}^{m}\sum_{j=0}^{n} \alpha_j \mid x_{ij} \mid \to \min;$$

$$y_i \geq \sum_{j=0}^{n} a_j x_{ij} - (1-h)\sum_{j=0}^{n} \alpha_j \mid x_{ij} \mid,\ i = 1,2,\ldots,m;$$

$$y_i \leq \sum_{j=0}^{n} a_j x_{ij} + (1-h)\sum_{j=0}^{n} \alpha_j \mid x_{ij} \mid,\ i = 1,2,\ldots,m;$$

$$\alpha_j \geq 0,\ j = 0,1,2,\ldots,n.$$

Положим

$$c_j = \sum_{i=1}^{m} \left| x_{ij} \right|,\ j = 0,1,\ldots,n,$$

и

$$c = (c_0, c_1, \ldots, c_n)^T.$$

Далее, пусть

$$X = (x_{ij}),\ |X| = (|x_{ij}|);\ i = 0,1,\ldots,m,\ j = 1,\ldots,n,\ x_{i0} = 1;$$

$$y = (y_1, \ldots, y_m)^T;$$

$$a = (a_0, a_1, \ldots, a_n)^T,\ \alpha = (\alpha_0, \alpha_1, \ldots, \alpha_n)^T.$$

Обозначим через $J_h(X)$ матрицу

$$\begin{pmatrix} X & -(1-h)|X| \\ -X & -(1-h)|X| \end{pmatrix}$$

порядка $2m \times 2(n+1)$.

Требуется найти минимальное значение целевой функции $c^T\alpha$ при следующих ограничениях:

$$J_h(X)\begin{pmatrix} a \\ \alpha \end{pmatrix} \leq \begin{pmatrix} y \\ -y \end{pmatrix},\ \alpha \geq 0.$$

Введем вектор балансовых переменных

$$\xi = (\xi_1, \ldots, \xi_{2m})$$

Система ограничений приобретает следующий вид:

$$(J_h(X) \quad I_{2m})\begin{pmatrix} a \\ \alpha \\ \xi \end{pmatrix} = \begin{pmatrix} y \\ -y \end{pmatrix},\ \begin{pmatrix} \alpha \\ \xi \end{pmatrix} \geq 0 \qquad (3)$$

Опираясь на содержательные соображения, можно выбрать постоянную $\bar{a}$ так, что $a + \bar{a}$ заведомо неотрицательно. Тогда задачу (3) можно заменить задачей

$$(J_h(X) \quad I_{2m})\begin{pmatrix} b \\ \alpha \\ \xi \end{pmatrix} = \begin{pmatrix} y + X\bar{a} \\ -y - X\bar{a} \end{pmatrix},\ \begin{pmatrix} b \\ \alpha \\ \xi \end{pmatrix} \geq 0 \qquad (4)$$

где $b = a + \bar{a}$.

Предположим теперь, что данные наблюдений разделены между двумя пользователями I и II так, что данные

$$(x'_{i0}, x'_{i1}, \ldots, x'_{in}; y'_i),\ i = 1,\ldots,m_1,$$

принадлежат пользователю I, а данные

$$(x''_{i0}, x''_{i1}, \ldots, x''_{in}; y''_i),\ i = 1,\ldots,m_2,$$

принадлежат пользователю II.

Положим

$$X' = (x'_{ij}),\ X'' = (x''_{ij}),$$

$$y' = (y'_1, \ldots, y'_{m_1}),\ y'' = (y''_1, \ldots, y''_{m_2}),$$

$$y = \left(y'^T, -y'^T, y''^T, -y''^T\right)^T$$

$$c'_j = \sum_{i=1}^{m_1} \left|x'_{ij}\right|,\ c''_j = \sum_{i=1}^{m_2} \left|x''_{ij}\right|,\ j = 0,1,\ldots,n,$$

$$c' = \left(c'_0, c'_1, \ldots, c'_n\right)^T,\ c'' = \left(c''_0, c''_1, \ldots, c''_n\right)^T,$$

Требуется решить следующую задачу линейного программирования:

$$\left(c' + c''\right)^T \alpha \ \to min,$$

$$\begin{pmatrix} J_h(X') & I_{2m_1} & 0_{2m_1 \times 2m_2} \\ J_h(X'') & 0_{2m_2 \times 2m_1} & I_{m_2} \end{pmatrix} \begin{pmatrix} a \\ \alpha \\ \xi' \\ \xi'' \end{pmatrix} = \begin{pmatrix} y' \\ -y' \\ y'' \\ -y'' \end{pmatrix} \qquad (5)$$

$$\left(\alpha^T, \xi'^T, \xi''^T\right)^T \geq 0.$$

Решение должно быть получено так, чтобы информация о результатах наблюдений осталась конфиденциальной. Пользователь I не должен знать матрицу X и векторы y, c, пользователь II не должен знать матрицу X и векторы y, c. Оптимальное решение a^*, α^* получают оба пользователя.

3. Описание протокола

Применим трансформационный подход, использующий идеи из [8]. Опишем протокол, в результате исполнения которого пользователь I получает трансформированную задачу ЛП, решает эту задачу и затем пользователи I и II находят значения исходных переменных.

Протокол распадается на несколько суб-протоколов:

- протокол установки общих параметров;
- формирование коэффициентов уравнений;
- формирование правых частей уравнений;
- формирование коэффициентов неравенств;
- формирование целевой функции;
- формирование решения задачи ЛП.

3.0. Протокол установки общих параметров

Пользователи I и II передают друг другу значения m_1 и m_2, договариваются о параметре секретности — целом числе q, для которого 2^q достаточно велико. Далее совместно выбирают подходящий вектор $\bar{a}$ так, чтобы заведомо выполнялось соотношение $b = a + \bar{a} \geq 0$. Соответственно

$$z' = y' + X'\bar{a},\ z'' = y'' + X''\bar{a}.$$

Целевая функция при этом остается прежней.

3.1. Установка ключевых параметров, общих для всех протоколов

Пользователь *I* генерирует:

- случайный вектор $\theta \geq 0$ размерности $2m_1$;
- обратимые матрицы P и Q порядка $2m_1$.

Пользователь *II* генерирует:

- случайную обратимую матрицу K порядка $4m_1 + 2m_2$;
- случайную обратимую матрица M порядка $2(n+1) + 4m_1 + 2m_2$;
- случайную положительную квадратную матрицу B порядка $2(n+1) + 4m_1 + 2m_2$, которая содержит в каждой строке ровно один ненулевой элемент.

3.2. Формирование коэффициентов уравнений

Пользователь *I*

- генерирует случайные матрицы $H'_1, \ldots, H'_{q-1}$ размера $4m_1 \times \left(2(n+1) + 4m_1\right)$;
- вычисляет матрицу H'_q так, что

$$\sum_{t=1}^{q} H'_t = \begin{pmatrix} J_h(X') & I_{2m_1} & P \\ 0 & 0 & Q \end{pmatrix};$$

- генерирует случайные матрицы $G'_{t,1-\omega_t}$, размера $4m_1 \times \left(2(n+1) + 4m_1\right)$, $t = 1, \ldots, q$;
- генерирует случайное целое число ω, имеющее двоичное представление $\omega = \omega_1 \ldots \omega_q$;
- формирует и передает пользователю II массив матриц $\left(G'_{t,s}\right)$, в котором

$$G'_{t,\omega_t} = H'_t.$$

Пользователь *II*

- генерирует случайные матрицы $V''_1, \ldots, V''_{q-1}$ размера $2m_2 \times 2(n+1)$ и матрицу V''_q такие, что

$$\sum_{t=1}^{q} V''_t = J_h(X'');$$

- генерирует случайные матрицы R_t размерности $\left(4m_1 + 2m_2\right) \times \left(2(n+1) + 4m_1 + 2m_2\right)$;
- генерирует случайные квадратные матрицы R_t порядка $2(n+1) + 4m_1 + 2m_2$;
- для всех $t = 1, \ldots, q$, $s = 0,1$ формирует матрицы

$$U'_{t,s} = \begin{pmatrix} G'_{t,s} & 0 \\ 0 & 0 \end{pmatrix}$$

и матрицы

$$U''_t = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ V''_t & 0 & 0 & I_{2m_2}/q \end{pmatrix}$$

размерности $\left(4m_1 + 2m_2\right) \times \left(2(n+1) + 4m_1 + 2m_2\right)$.

- Для всех $t = 1, \ldots, q$ и для всех t, s пользователь II вычисляет матрицы

$$Z_{t,s} = K\left(U'_{t,s} + U''_t\right)M + R_t.$$

Матрицы $Z_{t,s}$ и сумма $R = \sum_{t=1}^{q} R_t$ передаются пользователю I.

Пользователь *I*

Вычисляет матрицу коэффициентов

$$\left[\left[X\right]\right] = \sum_{t=1} Z_{t,\omega_t} - R.$$

3.3. Формирование правых частей уравнений

Пользователь *I*

вычисляет и передает пользователю II вектор

$$v = \begin{pmatrix} \begin{pmatrix} z' \\ -z' \end{pmatrix} + P\theta \\ Q\theta \end{pmatrix}.$$

Пользователь *II*

генерирует случайные векторы r_t размерности $4m_1 + 2m_2$;

для всех t формирует векторы

$$z_t = K\begin{pmatrix} q^{-1}\begin{pmatrix} v \\ z'' \\ -z'' \end{pmatrix} \end{pmatrix} + r_t$$

и передает пользователю I эти векторы и сумму $r = \sum_{t=1}^{q} r_t$.

Пользователь *I*

вычисляет вектор

$$[[z]] = \sum_{t=1}^{q} z_t - r.$$

3.4. Формирование коэффициентов неравенств

Пользователь *I*

генерирует случайные матрицы $H_1^{'},\ldots,H_{q-1}^{'}$ размера $4m_1 \times \left(2(n+1)+4m_1\right)$ и вычисляет матрицу $H_q^{'}$ так, что

$$\sum_{t=1}^{q} H_t^{'} = \begin{pmatrix} J_h(X') & I_{2m_1} & P \\ 0 & 0 & Q \end{pmatrix};$$

- генерирует случайные матрицы $G_{t,1-\omega_t}^{'}$, размера $4m_1 \times \left(2(n+1)+4m_1\right)$, $t=1,\ldots,q$;
- генерирует случайное целое число ω, имеющее двоичное представление $\omega = \omega_1 \ldots \omega_q$;
- формирует и передает пользователю II массив матриц $\left(G_{t,s}^{'}\right)$, в котором

$$G_{t,\omega_t}^{'} = H_t^{'}.$$

Пользователь *II*

- генерирует случайные матрицы $V_1^{''},\ldots,V_{q-1}^{''}$ размера $2m_2 \times 2(n+1)$ и матрицу $V_q^{''}$ такие, что

$$\sum_{t=1}^{q} V_t^{''} = J_h(X^{''});$$

- генерирует случайные матрицы R_t размерности $(4m_1+2m_2)\times(2(n+1)+4m_1+2m_2)$;
- генерирует случайные квадратные матрицы R_t порядка $2(n+1)+4m_1+2m_2$;
- для всех $t=1,\ldots,q$, $s=0,1$ формирует матрицы

$$U_{t,s}^{'} = \begin{pmatrix} G_{t,s}^{'} & 0 \\ 0 & 0 \end{pmatrix}$$

и матрицы

$$U_t^{''} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ V_t^{''} & 0 & 0 & I_{2m_2}/q \end{pmatrix}$$

размерности $(4m_1+2m_2)\times(2(n+1)+4m_1+2m_2)$;

- генерирует случайную квадратную матрицу L порядка $4m_1 \times 2m_2$ такую, что

$$LK\begin{pmatrix} v \\ z'' \\ -z'' \end{pmatrix} = 0;$$

- генерирует случайные квадратные матрицы R_t порядка $2(n+1)+4m_1+2m_2$.
- Для всех s,t пользователь II формирует матрицы

$$Z_{t,s} = \left(q^{-1}BM - LK\left(U_{t,s}^{'} + U_t^{''}\right)M\right) + R_t$$

и передает пользователю I эти матрицы и сумму $R = \sum_{t=1}^{q} R_t$.

Пользователь *I*

Вычисляет матрицу коэффициентов неравенств

$$[[F]] = \sum_{t=1}^{q} Z_{t,\omega_t} - R.$$

3.5. Формирование целевой функции

Пользователь *I*

- генерирует случайное целое число ω, имеющее двоичное представление $\omega = \omega_1 \ldots \omega_q$,
- генерирует случайные векторы $u_1^{'},\ldots,u_{q-1}^{'}$ размерности $n+1$ и вычисляет вектор u_q такой, что

$$\sum_{t=1}^{q} u_t^{'} = c'.$$

- для всех $t=1,\ldots,q$, пользователь I генерирует случайные векторы $g_{t,1-\omega_t}$, размерности $n+1$, полагает $g_{t,\omega_t} = u_t^{'}$ для всех t.

Формирует матрицу $\left(g_{t,s}\right)$, и высылает эту матрицу пользователю II.

Пользователь *II*

- генерирует случайные векторы $v_1,\ldots,v_{q-1}$ размерности $n+1$ и вычисляет вектор v_q такой, что

$$\sum_{t=1}^{q} u_t^{''} = c''.$$

- генерирует случайные векторы r_t размерности $2(n+1)+4m_1+2m_2$;
- генерирует случайное положительное число γ;
- для каждого вектора $g_{t,s}$ формирует вектор

$$h_{t,s}^{'} = \left(0_{n+1}^T, g_{t,s}^T, 0_{2m_1}^T, 0_{2m_1}^T, 0_{2m_2}^T\right)^T;$$

- для всех $t=1,\ldots,q$ строит векторы

$$h_t^{''} = \left(0_{n+1}^T, u_t^{''T}, 0_{2m_1}^T, 0_{2m_1}^T, 0_{2m_2}^T\right)^T;$$

- для всех $t=1,\ldots,q$, $s=0,1$ вычисляет векторы

$$z_{t,s} = \gamma\left(h_{t,s}^{'} + h_t^{''}\right)M^T + r_t.$$

Векторы $z_{t,s}$ и сумма $r = \sum_{t=1}^{q} r_t$ пересылаются пользователю I.

Пользователь *I*

Вычисляет вектор

$$[[c]] = \sum_{t=1}^{q} z_{t,\omega_t} - r.$$

3.6. Формирование решения задачи ЛП

Пользователь *I*

Имеет следующую задачу линейного программирования:

$$[[c]]^T w \to min;$$

$$[[X]]w = [[z]] \qquad (6)$$

$$[[F]]w \geq 0.$$

Пользователь I решает эту задачу и пересылает решение w^* пользователю II.

Пользователь *II*

Вычисляет решение задачи линейного программирования (5):

$$\begin{pmatrix} a^* \\ \alpha^* \end{pmatrix} = \left(I_{2(n+1)} \, 0_{2(n+1)\times(4m_1+2m_2)} \right) Mw^* - \begin{pmatrix} \bar{a} \\ 0 \end{pmatrix}.$$

4. Корректность протокола

Переменные a, α, ξ', ξ'' в задаче (5) и переменная w в задаче (6) связаны соотношениями

$$Mw = \begin{pmatrix} b \\ \alpha \\ \xi' \\ \beta \\ \xi'' \end{pmatrix}, \beta = \theta.$$

С учетом этого получаем следующую цепочку равносильных уравнений:

$$[[X]]w = [[z]];$$

$$\left(\sum_{t=1}^{q} Z_{t,\omega_t} - R \right) w = \sum_{t=1}^{q} z_t - r;$$

$$\left(\sum_{t=1}^{q} K\left(U'_{t,\omega_t} + U''_t \right) M + R_t - R \right) w = \sum_{t=1}^{q} \left(K \begin{pmatrix} v \\ q^{-1} z'' \\ -z'' \end{pmatrix} + r_t \right) - r$$

$$\left(\sum_{t=1}^{q} K \left(\begin{pmatrix} G'_{t,\omega_t} & 0 \\ 0 & 0 \end{pmatrix} + U''_t \right) M \right) w = \frac{K}{q} \sum_{t=1}^{q} \begin{pmatrix} v \\ z'' \\ -z'' \end{pmatrix};$$

$$K \left(\sum_{t=1}^{q} \begin{pmatrix} H'_t & 0 \\ 0 & 0 \end{pmatrix} + U''_t \right) Mw = K \begin{pmatrix} v \\ z'' \\ -z'' \end{pmatrix};$$

$$\left(\begin{pmatrix} J_h(X') & I_{2m_1} & P & 0 \\ 0 & 0 & Q & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ J_h(X'') & 0 & 0 & I_{2m_2} \end{pmatrix} \right) Mw = \begin{pmatrix} v \\ z'' \\ -z'' \end{pmatrix};$$

$$\begin{pmatrix} J_h(X') & I_{2m_1} & P & 0 \\ 0 & 0 & Q & 0 \\ J_h(X'') & 0 & 0 & I_{2m_2} \end{pmatrix} \begin{pmatrix} b \\ \alpha \\ \xi' \\ \beta \\ \xi'' \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} z' \\ -z' \end{pmatrix} + P\theta \\ Q\theta \\ z'' \\ -z'' \end{pmatrix};$$

$$\begin{pmatrix} J_h(X') & I_{2m_1} & 0_{2m_1\times 2m_2} \\ J_h(X'') & 0_{2m_2\times 2m_1} & I_{m_2} \end{pmatrix} \begin{pmatrix} a \\ \alpha \\ \xi' \\ \xi'' \end{pmatrix} = \begin{pmatrix} y' \\ -y' \\ y'' \\ -y'' \end{pmatrix};$$

$$\begin{pmatrix} J_h(X') & I_{2m_1} & 0_{2m_1\times 2m_2} \\ J_h(X'') & 0_{2m_2\times 2m_1} & I_{m_2} \end{pmatrix} \left(\begin{pmatrix} a \\ \alpha \\ \xi' \\ \xi'' \end{pmatrix} + \begin{pmatrix} \bar{a} \\ 0 \\ 0 \\ 0 \end{pmatrix} \right) = \begin{pmatrix} y' \\ -y' \\ y'' \\ -y'' \end{pmatrix} + \begin{pmatrix} X'\bar{a} \\ -X'\bar{a} \\ X''\bar{a} \\ -X''\bar{a} \end{pmatrix};$$

Аналогичным образом для неравенств имеем:

$$[[F]]w \geq 0$$

$$\left(\sum_{t=1}^{q} Z_{t,\omega_t} - R \right) w \geq 0$$

$$\left(\sum_{t=1}^{q} \left(q^{-1} BM - LK\left(U'_{t,\omega_t} + U''_t \right) M + R_t \right) - R \right) w \geq 0$$

$$BMw - LK \sum_{t=1}^{q} \left(U'_{t,\omega_t} + U''_t \right) Mw \geq 0$$

$$B \begin{pmatrix} b \\ \alpha \\ \xi' \\ \beta \\ \xi'' \end{pmatrix} - LK \begin{pmatrix} J_h(X') & I_{2m_1} & P & 0 \\ 0 & 0 & Q & 0 \\ J_h(X'') & 0 & 0 & I_{2m_2} \end{pmatrix} \begin{pmatrix} b \\ \alpha \\ \xi' \\ \beta \\ \xi'' \end{pmatrix} \geq 0$$

$$B \begin{pmatrix} b \\ \alpha \\ \xi' \\ \beta \\ \xi'' \end{pmatrix} - LK \begin{pmatrix} \begin{pmatrix} \begin{pmatrix} z' \\ -z' \end{pmatrix} + P\theta \end{pmatrix} \\ Q\theta \\ z'' \\ -z'' \end{pmatrix} \geq 0$$

$$B \begin{pmatrix} b \\ \alpha \\ \xi' \\ \beta \\ \xi'' \end{pmatrix} - LK \begin{pmatrix} v \\ z'' \\ -z'' \end{pmatrix} \geq 0 \qquad B \begin{pmatrix} b \\ \alpha \\ \xi' \\ \beta \\ \xi'' \end{pmatrix} \geq 0 \qquad \begin{pmatrix} b \\ \alpha \\ \xi' \\ \beta \\ \xi'' \end{pmatrix} \geq 0$$

Наконец, рассмотрим целевую функцию. Имеем:

$$[[c]]^T = \sum_{t=1}^{q} z_{t,\omega_t}^T - r^T = \sum_{t=1}^{q} \left(\gamma \left(h'_{t,\omega_t} + h''_t \right) M^T + r_t \right)^T - r^T =$$

$$= \gamma \left(0, \sum_{t=1}^{q} \left(g_{t,\omega_t}^T + u_t''^T \right), 0, 0, 0 \right) M = \gamma \left(0, \sum_{t=1}^{q} u'_t + \sum_{t=1}^{q} u_t''^T, 0, 0, 0 \right) M =$$

$$= \gamma \left(0, c'^T + c''^T, 0, 0, 0 \right) M.$$

Таким образом,

$$[[c]]^T w = \gamma \left(0, (c' + c'')^T, 0, 0, 0 \right) \begin{pmatrix} b \\ \alpha \\ \xi' \\ \beta \\ \xi'' \end{pmatrix} = \gamma (c' + c'')^T \alpha \quad ,$$

так что значения целевой функции в задачах (5) и (6) различаются лишь положительным множителем.

5. Анализ конфиденциальности

Трансформационный подход к коллаборативному решению задач линейного программирования использовался и анализировался в целой серии работ, см. [13].

Общая идея состоит в том, чтобы трансформировать систему ограничений в задаче, представленной в канонической форме, умножив ее слева на обратимую матрицу и заменив переменные с помощью аффинного преобразования. В задаче построения нечеткой линейной регрессии представление ограничений в виде неравенств является принципиальным. Введение балансовых переменных может раскрыть матрицу K. С целью ее маскировки в ряде работ используется положительная мономиальная матрица M и вектор сдвига ρ:

$$\begin{pmatrix} 0^T & c^T & 0^T \end{pmatrix} Mw \to min,$$
$$K\begin{pmatrix} J(X) & I \end{pmatrix} Mw = K\begin{pmatrix} y \\ -y \end{pmatrix} + K\begin{pmatrix} J(X) & I \end{pmatrix}\rho \qquad (7)$$
$$w \geq 0$$

Новые переменные связаны со старыми соотношением

$$w = M^{-1}\left(\begin{pmatrix} a \\ \alpha \\ \xi \end{pmatrix} + \rho\right).$$

Использование положительной мономиальной матрицы M не обеспечивает достаточно конфиденциальности [16]. Умножение на матрицу M сводится к калибровке и перестановке столбцов. Номера переменных в w, соответсвующих балансовым переменным могут быть определены. Это в свою очередь позволяет определить, как переставлялись столбцы матрицы K, а, затем, и коэффициенты калибровки: вероятность того, что два случайных целых чисел окажутся взаимно простыми, асимтотически приближается к $6/\pi^2$, так что вероятность найти общий множитель у двух столбцов оказывается достаточно высокой. Так или иначе, нарушение конфиденциальности при описанном подходе, обусловлено необходимостью использовать балансовые переменные.

В задаче линейного программирования, связанной с построением нечеткой линейной регрессии, ограничения представлены неравенствами, так что без балансовых переменных не обойтись. В настоящей работе балансовые переменные сначала маскируются введением случайных матриц P и Q и случайного вектора θ, а затем применяется идея из [13] для трансформации задачи линейного программирования в канонической форме.

Покажем, что благодаря этому обеспечивается необходимая конфиденциальность.

В самом деле, матрица

$$\begin{pmatrix} J_h(X') & I_{2m_1} & P & 0 \\ 0 & 0 & Q & 0 \\ J_h(X'') & 0 & 0 & I_{2m_2} \end{pmatrix}$$

имеет ранг $4m_1 + 2m_2$, равный числу строк. Умножением слева и справа на обратимые матрицы из нее может быть получена произвольная матрица $[[X]]$ ранга $4m_1 + 2m_2$.

Таким образом, из коэффициентов системы уравнений пользователь I может извлечь лишь информацию о рангах.

Точно так же, произвольным может быть вектор

$$v = \begin{pmatrix} \begin{pmatrix} z' \\ -z' \end{pmatrix} + P\theta \\ Q\theta \end{pmatrix},$$

что защищает данные пользователя I.

Сохранение конфиденциальности при формировании остальных компонентов задачи (6) может быть обосновано так же, как в [7].

7. Заключение

В статье представлен протокол, исполняя который два пользователя могут провести коллаборативный расчет параметров нечеткой линейной регрессионной модели. Проведено обоснование корректности протокола. Конфиденциальность обеспечивается на основе трансформационного подхода. Применение протокола позволяет использовать при построении модели данные обоих пользователей, при этом каждый из пользователей получает информацию только о размерностных параметрах данных партнера.

Литература

1. Hastings M. Hemenway, B., Noble, D., & Zdancewic, S. Sok: General purpose compilers for secure multi-party computation // 2019 IEEE Symposium on Security and Privacy (SP). IEEE, 2019. С. 1220-1237.
2. Gascón A., Schoppmann, P., Balle, B., Raykova, M., Doerner, J., Zahur, S., & Evans, D. Privacy-preserving distributed linear regression on high-dimensional data // Proceedings on Privacy Enhancing Technologies. 2017. Т. 2017. №. 4. С. 345-364.
3. Запечников С. В. Модели и алгоритмы конфиденциального машинного обучения // Безопасность информационных технологий. 2020. Т. 27. №. 1. С. 51-67.
4. Pandit P., Dey P., Krishnamurthy K. N. Comparative assessment of multiple linear regression and fuzzy linear regression models // Springer Nature Computer Science. 2021. Т. 76. №. 2. С. 1-8. https://doi.org/10.1007/s42979-021-00473-3
5. Zeng W., Feng Q., Li J. Fuzzy least absolute linear regression //Applied Soft Computing. 2017. Т. 52. С. 1009-1019. https://doi.org/10.1016/j.asoc.2016.09.029
6. Yang Q., Liu, Y., Chen, T., & Tong, Y. Federated machine learning: Concept and applications //ACM Transactions on Intelligent Systems and Technology (TIST). 2019. Т. 10. №. 2. С. 1-19.
7. Wang C., Ren K., Wang J. Secure optimization computation outsourcing in cloud computing: A case study of linear programming // IEEE transactions on computers. 2016. Т. 65. №. 1. С. 216-229.

8. Wang Z., Yang L. I. U. Secure Outsourcing of Large-scale Linear Programming // In: 2017 2nd International Conference on Wireless Communication and Network Engineering (WCNE 2017) DEStech Transactions on Computer Science and Engineering. 2017. WCNE 2017. Pp.185-190 DOI:10.12783/dtcse/wcne2017/19821
9. Hong Y. Vaidya, J., Rizzo, N., & Liu, Q. . Privacy-preserving linear programming // World scientific reference on innovation: Volume 4: Innovation in Information Security. 2018. С. 71-93. https://doi.org/10.1142/9789813149106_0004
10. Ahire P., Abraham J. Addition of fake variable to enrich secure linear programming computation outsourcing in the cloud //2016 International Conference on Computing, Analytics and Security Trends (CAST). IEEE, 2016. С. 477-482.
11. Kumar R., Pravin A. Data protection and outsourcing in cloud with Linear programming and image based OTP //2017 International Conference on Information Communication and Embedded Systems (ICICES). IEEE, 2017. С. 1-6.
12. Mohammed N. M., Lomte S. S. Secure computations outsourcing of mathematical optimization and linear algebra tasks: Survey // International Journal for Research in Engineering Application and Management. 2019. С. 6-11. DOI : 10.18231/2454-9150.2018.0860
13. Shan Z. Ren, K., Blanton, M., & Wang, C. Practical secure computation outsourcing: A survey //ACM Computing Surveys (CSUR). 2018. Т. 51. №. 2. С. 1-40. https://doi.org/10.1145/3158363
14. Singh S., Sharma P., Arora D. Secure Outsourcing of Linear Programming in Cloud Computing Environment: A Review Int. Journal of Engineering Research and Application Vol. 7, Issue 4, (Part -6) April 2017. С.64-68 DOI: 10.9790/9622-0704066468
15. Phatangare S., Bhandari G. Secure Outsourcing of Linear Programming Solver in Cloud Computing: A Survey //Asian Journal for Convergence in Technology (AJCT). 2019. С. 1-5
16. Liu L., Liu Y. A Note on One Outsourcing Scheme for Large-scale Convex Separable Programming //International Journal of Electronics and Information Engineering. 2020. Т. 12. №. 4. С. 155-161. DOI: 10.6636/IJEIE.202012 12(4).02)

PRIVACY-PRESERVING TWO-PARTY COMPUTATION OF PARAMETERS OF A FUZZY LINEAR REGRESSION

Volkova E.S.[3], Gisin V.B.[4]

Purpose: *describe two-party computation of fuzzy linear regression with horizontal partitioning of data, while maintaining data confidentiality.*

Methods: *the computation is designed using a transformational approach. The optimization problems of each of the two participants are transformed and combined into a common problem. The solution to this problem can be found by one of the participants.*

Results: *A protocol is proposed that allows two users to obtain a fuzzy linear regression model based on the combined data. Each of the users has a set of data about the results of observations, containing the values of the explanatory variables and the values of the response variable. The data structure is shared: both users use the same set of explanatory variables and a common criterion. Regression coefficients are searched for as symmetric triangular fuzzy numbers by solving the corresponding linear programming problem. It is assumed that both users are semi-honest (honest but curious, or passive and curious), i.e. they execute the protocol, but can try to extract information about the source data of the partner by applying arbitrary processing methods to the received data that are not provided for by the protocol. The protocol describes the transformed linear programming problem. The solution of this problem can be found by one of the users. The number of observations of each user is known to both users. The observation data remains confidential. The correctness of the protocol is proved and its security is justified.*

Keywords: *fuzzy numbers, collaborative solution of a linear programming problem, two-way computation, transformational approach, cloud computing, federated machine learning.*

References

1. Hastings M. Hemenway, B., Noble, D., & Zdancewic, S. Sok: General purpose compilers for secure multi-party computation // 2019 IEEE Symposium on Security and Privacy (SP). IEEE, 2019. Pp. 1220-1237.
2. Gascón A., Schoppmann, P., Balle, B., Raykova, M., Doerner, J., Zahur, S., & Evans, D. Privacy-preserving distributed linear regression on high-dimensional data // Proceedings on Privacy Enhancing Technologies. 2017. v. 2017. №. 4. Pp. 345-364.
3. Zapechnikov S. V. Modeli i algoritmy konfidencial›nogo mashinnogo obucheniya // Bezopasnost› informacionnyh tekhnologij. 2020. Т. 27. №. 1. S. 51-67.

3 Elena Volkova, Ph.D., Associate Professor of the Department of Data Analysis and machine Learning, Financial University under the Government of the Russian Federation, Moscow, Russia. E-mail: evolkova@fa.ru, ORCHID ID 0000-0001-9037-592X

4 Vladimir Gisin, Ph.D., Professor, Head of the Department of Information Security, Financial University under the Government of the Russian Federation, Moscow, Russia. E-mail: vgisin@fa.ru, ORCHID ID 0000-0002-7269-0587

4. Pandit P., Dey P., Krishnamurthy K. N. Comparative assessment of multiple linear regression and fuzzy linear regression models // Springer Nature Computer Science. 2021. v. 76. №. 2. Pp. 1-8. https://doi.org/10.1007/s42979-021-00473-3
5. Zeng W., Feng Q., Li J. Fuzzy least absolute linear regression // Applied Soft Computing. 2017. v. 52. Pp. 1009-1019. https://doi.org/10.1016/j.asoc.2016.09.029
6. Yang Q., Liu, Y., Chen, T., & Tong, Y. Federated machine learning: Concept and applications //ACM Transactions on Intelligent Systems and Technology (TIST). 2019. v. 10. №. 2. Pp. 1-19.
7. Wang C., Ren K., Wang J. Secure optimization computation outsourcing in cloud computing: A case study of linear programming // IEEE transactions on computers. 2016. v. 65. №. 1. Pp. 216-229.
8. Wang Z., Yang L. I. U. Secure Outsourcing of Large-scale Linear Programming // In: 2017 2nd International Conference on Wireless Communication and Network Engineering (WCNE 2017) DEStech Transactions on Computer Science and Engineering. 2017. №. WCNE 2017. Pp. 185-190 DOI:10.12783/dtcse/wcne2017/19821
9. Hong Y. Vaidya, J., Rizzo, N., & Liu, Q. . Privacy-preserving linear programming // World scientific reference on innovation: Volume 4: Innovation in Information Security. 2018. Pp. 71-93. https://doi.org/10.1142/9789813149106_0004
10. Ahire P., Abraham J. Addition of fake variable to enrich secure linear programming computation outsourcing in the cloud // 2016 International Conference on Computing, Analytics and Security Trends (CAST). IEEE, 2016. Pp. 477-482.
11. Kumar R., Pravin A. Data protection and outsourcing in cloud with Linear programming and image based OTP //2017 International Conference on Information Communication and Embedded Systems (ICICES). IEEE, 2017. pp. 1-6.
12. Mohammed N. M., Lomte S. S. Secure computations outsourcing of mathematical optimization and linear algebra tasks: Survey // International Journal for Research in Engineering Application and Management. 2019. Pp. 6-11. DOI : 10.18231/2454-9150.2018.0860
13. Shan Z. Ren, K., Blanton, M., & Wang, C. Practical secure computation outsourcing: A survey //ACM Computing Surveys (CSUR). 2018. v. 51. №. 2. Pp.1-40. https://doi.org/10.1145/3158363
14. Singh S., Sharma P., Arora D. Secure Outsourcing of Linear Programming in Cloud Computing Environment: A Review Int. Journal of Engineering Research and Application v. 7, Issue 4, (Part 6) April 2017. Pp. 64-68 DOI: 10.9790/9622-0704066468
15. Phatangare S., Bhandari G. Secure Outsourcing of Linear Programming Solver in Cloud Computing: A Survey //Asian Journal for Convergence in Technology (AJCT). 2019. pp. 1-5
16. Liu L., Liu Y. A Note on One Outsourcing Scheme for Large-scale Convex Separable Programming //International Journal of Electronics and Information Engineering. 2020. v. 12. №. 4. Pp. 155-161. DOI: 10.6636/IJEIE.202012 12(4).02)