DOI: 10.17323/2587-814X.2024.3.24.40

Контрфактуальные объяснения на основе генерации синтетических данных

Ю.А. Зеленков
E-mail: yzelenkov@hse.ru

Е.В. Лашкевич
E-mail: evlashkevich@hse.ru

Высшая школа бизнеса, Национальный исследовательский университет «Высшая школа экономики», Москва, Россия

Аннотация

Контрфактуальное объяснение – это генерация для заданного экземпляра множества объектов, которые принадлежат к противоположному классу, но находятся в пространстве признаков максимально близко к объясняемому фактуалу. Известные алгоритмы, решающие эту задачу, как правило, основаны на сложных моделях, требующих большого объема обучающих данных и значительных вычислительных затрат. В данной статье предлагается метод, который включает два этапа. На первом этапе на основе простых статистических моделей (гауссовская копула, последовательная модель на основе условных распределений, байесовская сеть и др.) генерируется синтетическое множество потенциальных контрфактуалов, на втором – производится отбор объектов, удовлетворяющих ограничениям правдоподобия, близости, разнообразия и т.д. Такая организация позволяет сделать процесс прозрачным, управляемым и повторно использовать модели генерации. Эксперименты на трех свободно распространяемых наборах данных показали, что предложенный метод позволяет добиться результатов, как минимум, сравнимых с известными алгоритмами контрфактуальных объяснений, а в ряде случаев их превосходит, особенно на малых наборах данных. Наиболее эффективной моделью генерации при этом является байесовская сеть.

Ключевые слова: контрфактуальные объяснения, генерация синтетических данных, моделирование мультимодальных распределений, байесовская сеть, кредитный скоринг

Цитирование: Зеленков Ю.А., Лашкевич Е.В. Контрфактуальные объяснения на основе генерации синтетических данных // Бизнес-информатика. 2024. Т. 18. № 3. С. 24–40.
DOI: 10.17323/2587-814X.2024.3.24.40

Введение

В последние годы стремительно растет интерес к интерпретируемому искусственному интеллекту (explainable AI, XAI), что продиктовано расширяющимся использованием алгоритмов машинного обучения в различных областях человеческой деятельности [1, 2]. Более того, многие национальные и международные регуляторы требуют обеспечить прозрачность решений, основанных на алгоритмах. В частности, Общий регламент ЕС по защите данных (GDPR) предусматривает право граждан запрашивать «содержательную информацию о задействованной логике, а также о значении и предполагаемых последствиях» автоматизированных решений[1], а кредитное законодательство США требует предоставлять потребителям обоснования неблагоприятных решений [3]. Центральный банк РФ также следует рекомендациям ОЭСР по использованию AI[2], согласно которым модели должны быть прозрачны и интерпретируемы для ограничения модельных рисков и обеспечения возможности независимой внешней, внутренней и регуляторной проверки.

Методы XAI можно разделить на две группы [4]. Первая включает модели, для которых интерпретируемость (interpretability) является базовым свойством (например, дерево решений или линейная регрессия). Ко второй группе относятся методы, рассматривающие модель как черный ящик. В отличие от моделей первой группы, они не обладают свойствами, которые обеспечивают осмысленную интерпретацию, поэтому необходимо предпринимать дополнительные действия для объяснения логики принятия решения постфактум (explainability). Во второй группе, в свою очередь, можно выделить методы объяснения модели, объяснения локального результата и инспекции черного ящика [5].

В данной работе рассматриваются методы контрфактуального объяснения [5–8]. Контрфактуальное объяснение (counterfactual explanation, CE) позволяет для заданного экземпляра найти множество объектов, которые принадлежат к противоположному классу, но находятся в пространстве признаков максимально близко к объясняемому экземпляру. В качестве примера в литературе обычно приводится заемщик, которому было отказано в кредите на основании решения алгоритма, используемого в банке. Задачей CE является генерация для данного заемщика такого профиля, чтобы его заявка была одобрена (например, уменьшение суммы запрашиваемого кредита). Очевидным ограничением при этом является реализуемость предлагаемых изменений, поэтому обязательным параметром, минимизируемым в задачах такого рода, является расстояние между образцом и контрфактуалом. Из данного примера следует, что, согласно приведенной выше классификации, CE относится к группе локальных методов объяснения постфактум, поскольку объясняет решение обученной модели, трактуемой как черный ящик, для конкретного образца.

Следует отметить, что в русском языке нет устоявшегося соответствия английскому термину “counterfactual”. В Большой российской энциклопедии (БРЭ)[3] можно встретить как «контрфактуáльный», так и «контрфактический». Мы выбрали первый вариант, поскольку он использован в переводе книги [9][4], впервые, насколько нам известно, представившем эту концепцию на русском языке. Кроме того, на наш взгляд, при использовании такой формы прослеживается связь с моделью потенциального результата Д. Рубина, которая противопоставляет действительно происшедшее событие (factual) и его альтернативу (counterfactual).

В философии контрфактуал определяется как условное утверждение, антецедент (предшествующее событие, помогающее понять настоящее) которого ложен, а консеквент (следствие) описывает, каким был бы мир, если бы антецедент имел место (ответ на вопрос «что-если»). Согласно БРЭ, контрфактуальное мышление – вид мышления, характеризующийся склонностью человека представлять возможные иные варианты уже произошедших событий, т.е. размышление вопреки фактам.

В то время как большинство методов XAI направлены на получение ответов на вопрос «почему» [4], контрфактуальные утверждения служат средством интерпретации, указывая на то, какие

[1] General Data Protection Regulation (https://gdpr-info.eu)

[2] Recommendation of the Council on Artificial Intelligence (https://legalinstruments.oecd.org/en/instruments/OECD-LEGAL-0449)

[3] https://bigenc.ru

[4] Мишра П. Объяснимые модели искусственного интеллекта на Python. ДМК Пресс, 2022.

изменения потребуются для достижения желаемой цели (предсказание), а не помогают понять, почему текущая ситуация имеет определенный прогнозируемый результат [8]. Поэтому многие авторы [5], констатируют что CE соответствует третьему уровню моделей причинности Перла [10], которые должны отвечать на вопросы, предполагающие ретроспективные рассуждения, например, «какова вероятность события y при x, если в действительности наблюдаются x' и y'». При этом CE также не накладывает ограничений на сложность модели и не требует раскрытия информации о модели [3].

Очевидно, что методы CE являются мощным инструментом поддержки принятия решений в различных областях, например, в финансах [11, 12] и медицине [13]. К настоящему времени уже известно несколько десятков алгоритмов CE (см. обзоры [5, 6, 8] и др.). Большинство из них основаны на оптимизации некоторой целевой функции, и эта задача решается каждый раз, когда необходимо вычислить множество контрфактуалов для заданного образца. Это накладывает ограничения на производительность и масштабируемость CE [6]. Возможной альтернативой является использование методов, которые позволяют моделировать совместное распределение признаков изучаемых объектов. В этом случае однократно обученная модель может генерировать контрфактуалы для различных образцов без значительных вычислительных затрат.

Отметим, что в такой постановке задачу можно рассматривать как генерацию синтетических табличных данных [14, 15]. Для создания таких моделей используются как статистические методы – копулы, байесовские сети, так и методы машинного обучения – вариационные автоэнкодеры, генеративные состязательные сети и т.д. [16]. Некоторые исследователи также адаптируют для этой цели методы оверсэмплинга, которые разработаны для генерации объектов минорного класса в случае несбалансированных данных [17].

Учитывая эти обстоятельства, в данной статье предлагается подход к CE, основанный на принципах генерации синтетических данных, который включает два этапа. На первом этапе генерируется множество потенциальных контрфактуалов, на втором – производится отбор тех из них, которые удовлетворяют ограничениям реализуемости, близости, стоимости и т.д. Такая организация позволяет сделать процесс CE прозрачным, управляемым, повторно использовать модели генерации и, тем самым, значительно сократить вычислительные затраты.

Оставшаяся часть работы организована следующим образом. После обзора литературы, в разделе 2 представлен предлагаемый метод. В разделах 3 и 4 представлены результаты эксперимента по сравнению предложенного метода с другими известными методами CE. В заключении обсуждаются ограничения предложенного метода, а также дальнейшие направления исследований.

1. Обзор литературы

1.1. Генерация контрфактуалов

CE базируется на нескольких неявных предположениях [3]:

- рекомендуемое изменение значений признаков однозначно реализуется в реальном мире;
- распределение значений признаков может быть восстановлено из доступных обучающих данных;
- предлагаемые изменения имеют отношение только к принимаемому решению и не затрагивают другие области;
- модель устойчива во времени, монотонна и ограничена бинарными исходами.

Как уже отмечалось выше, CE является активно развивающейся областью исследований. Сам термин «контрфактуальное объяснение» применительно к AI-системам впервые использован в [18], однако работы, использующие аналогичный подход, стали появляться с середины 2010 годов [5].

Дадим формальные определения. Рассмотрим классификатор h: $\mathcal{X} \to \mathcal{Y}$ обученный на наборе данных $\mathcal{D} = \{(x_1, y_1), ..., (x_n, y_n)\}$, $x_i \in \mathcal{X}$, $y_i \in \mathcal{Y}$, $\mathcal{X} \subset \mathbb{R}^m$ – пространство признаков, $\mathcal{Y}$ – пространство меток классов. Обычно полагается $\mathcal{Y} = \{0, 1\}$ но все предлагаемые определения легко обобщаются и на случай многоклассовой классификации. Каждый экземпляр x_i – это вектор m пар признаков, $x_i = \{(a_j, v_{ij})\}_{j=1}^{m}$, где a_j – это признак (атрибут), а v_{ij} – его значение из домена a_j. Атрибуты могут быть как категориальными, порядковыми, так и непрерывными.

Определение 1. Если классификатор h присваивает метку $y = h(x)$ экземпляру x, контрфактуальным объяснением x является экземпляр x^* такой, что метка x^* отлична от y, т.е. $h(x^*) \neq y$, при этом различие между x и x^* минимально. Концепция мини-

мального различия здесь не уточняется, поскольку она зависит от контекста решаемой задачи и будет рассмотрена позднее.

Определение 2. Контрфактуальная модель (counterfactual explainer) – это функция f_k, которая для набора данных $\mathcal{D}$, классификатора h и экземпляра x возвращает набор $C = f_k(h, \mathcal{D}, x)$ из $l \le k$ допустимых контрфактуальных примеров $C = \{x_1^*, ..., x_l^*\}$, где k – количество необходимых контрфактуалов.

Характеристики, которые позволяют оценить качество алгоритма генерации контрфактуалов:

1. Валидность (validity) измеряется отношением числа контрфактуалов, которые имеют требуемую метку класса, к общему числу сгенерированных объектов [11]:

$$V = |C_v| / |C|,$$

где C_v – множество валидных контрфактуалов, сгенерированных моделью f_k;

C – множество примеров, сгенерированных f_k, $C_v \subset C$.

Валидность сгенерированного примера определяется при помощи предиктивной модели h, для валидного примера должно выполняться условие $h(x^*) \neq h(x)$. Как следует из определения, максимальное значение валидности $V = 1$, значения меньше 1 сигнализируют о недостаточной эффективности модели.

2. Близость (proximity) – расстояние контрфактуала от объекта, для которого генерируется объяснение. Близость множества контрфактуалов оценивается через среднее значение на этом множестве [19]:

$$P = \frac{1}{|C_v|} \sum_{x^* \in C_v} dist\,(x^*, x).$$

Для измерения расстояния $dist(x^*, x)$ чаще всего используются L_0, L_1, L_2 и L_∞ – нормы, $L_k = \left(\sum_i |x_i|^k\right)^{1/k}$ и их взвешенные комбинации. Чем меньше значение P, тем ближе найденные объекты к объясняемому фактуалу.

3. Разреженность (sparsity) – это оценка того, сколько признаков нужно изменить, чтобы перейти в класс контрфактуалов. Желательно, чтобы контрфактуалы имели как можно меньше изменений в своих характеристиках. Это свойство позволяет получить более эффективные, понятные человеку и интерпретируемые контрфактуализации [18].

$$S = \frac{1}{|C_v|} \sum_{x^* \in C_v} K(x^*).$$

$K(x^*)$ – количество атрибутов контрфакутала x^*, значение которых изменяется по сравнению с фактуалом x. Таким образом, предпочтительнее модели с меньшим значением S.

4. Разнообразие (diversity). Поиск ближайших точек в соответствии с функцией расстояния может привести к очень похожим контрфактуальным кандидатам с небольшими различиями между ними. Разнообразие подразумевает, что процесс генерации контрфактуалов дает различающиеся объяснения для одного и того же экземпляра данных. Это приводит к тому, что объяснения становятся более интерпретируемыми и более понятными для пользователя. Авторы [19] в качестве меры разнообразия предлагают использовать среднее расстояние между всеми парами валидных контрфактуалов:

$$D = \frac{1}{|C_v|^2} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} dist(x_i^*, x_j^*),$$

где $dist(x_i^*, x_j^*)$ – мера расстояния между двумя контрфактуалами x_i^* и x_j^*. Чем выше разнообразие, тем эффективнее алгоритм CE.

5. Правдоподобность (plausibility). Это свойство подчеркивает, что генерируемые контрфактуалы должны быть легитимными, а процесс поиска должен обеспечивать логически обоснованные результаты. Это означает, в частности, что найденный контрфактуал никогда не должен изменять неизменяемые характеристики, такие как пол или раса. В литературе выделяются три категории правдоподобия [20]:

- согласованность с доменом, которая ограничивает диапазон допустимых значений признаков контрфактуала;
- согласованность распределения требует, чтобы вероятности конкретных значений признаков контрфактуала соответствовали (эмпирическому) распределению данных. Это свойство может быть измерено [6] как среднее расстояние до k ближайших соседей, например, локальный коэффициент выбросов (local outlier factor, LOF) [21], а также с помощью ядерных функций (kernel density estimation, KDE). В последнем случае оценивается плотность распределения каждого признака на основе KDE, а затем вычисляется вероятность принадлежности соответствующего атрибута контрфак-

туала этому распределению. Данный подход имеет очевидные ограничения – рассматривается каждый признак раздельно, и он применим только к непрерывным атрибутам. Способ на основе ближайших соседей таких ограничений не имеет;

- согласованность с прототипом выбирает контрфактуальные экземпляры, которые либо непосредственно присутствуют в наборе данных, либо близки к объясняемому объекту данных. Отметим, что данное свойство близко к определению близости (proximity), представленному выше.

В данной работе мы будем использовать измерение правдоподобности на основе значения LOF, т.е.

$$U = \frac{1}{|C_v|} \sum_{x^* \in C_v} LOF(x^*).$$

Отметим, что значения LOF трудно интерпретировать ввиду локальности метода. Значения около 1, говорит, что точка внутренняя, чем выше значение, тем больше вероятность того, что она является выбросом. Таким образом, с точки зрения оценки алгоритма CE предпочтительными являются значения, близкие к 1.

6. Осуществимость (actionability / feasibility). Поиск наиболее близкого контрфактуала для экземпляра данных не обязательно приводит к осуществимому изменению характеристик. Возможность изменения конкретной переменной описывается одной из трех категорий:

- изменение признака может быть осуществимо (actionable) и, соответственно, признак изменяем (mutable), например, данные бухгалтерского баланса;
- признак изменяем, но изменение не осуществимо (например, кредитный рейтинг);
- признак неизменяем (например, место рождения).

Отметим, что пользователь не может изменить значения переменных двух последних категорий, однако эти значения могут меняться в результате воздействия на их предков в причинно-следственной модели [20]. Некоторые авторы полагают, что удовлетворение требования осуществимости автоматически гарантирует правдоподобность контрфактуальной рекомендации [22], однако, несмотря на некоторое пересечение, это разные концепции [20]. Осуществимость ограничивает набор действий теми, что можно выполнить, правдоподобие требует, чтобы результирующий контрфактуал был реалистичным.

Авторы обзорных статей [5–8] используют различные таксономии методов CE. Здесь предлагается классификация на основе архитектуры используемых моделей (*рис. 1*).

Первая группа методов основана на решении задачи оптимизации, в которой часть перечисленных выше свойств рассматривается как целевая функция, а оставшиеся свойства – как ограничения. Например, в [18] в качестве цели используется расстояние $dist(x^*, x)$ с ограничением на метку контрфактуала $h(x^*) = y^*$. Данная задача может быть преобразована в проблему, описываемую дифференцируемой функций без ограничений:

$$x^* \in \arg\min_{x^*} \max_{\lambda} \lambda(h(x^*) - y^*)^2 + dist(x^*, x).$$

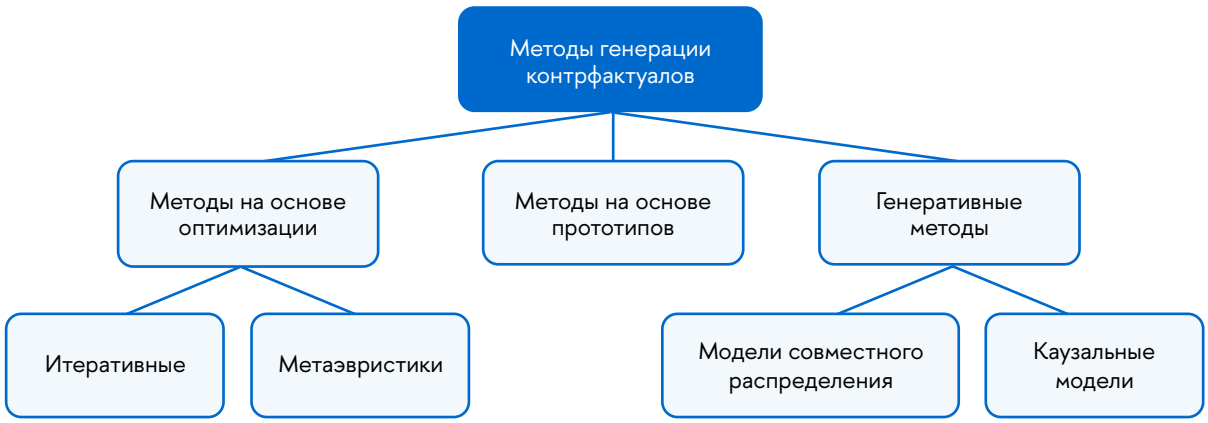

Рис. 1. Классификация алгоритмов генерации контрфактуалов.

Член $\lambda(h(x^*) - y^*)^2$ обеспечивает соответствие метки контрфактуала желаемому классу.

Подобный подход может быть расширен так, чтобы включать ограничения осуществимости (actionability), разреженности (sparsity), согласованности распределения (data manifold closeness) и др., например [6]:

$$x^* \in \arg\min_{x^* \in \mathcal{A}} \max_{\lambda} \lambda\left(h(x^*) - y^*\right)^2 + dist(x^*, x) + \\ + g(x^* - x) + l(x^*; \mathcal{X}).$$

Условие $x^* \in \mathcal{A}$ ограничивает список изменяемых атрибутов x^* теми, изменение которых может быть осуществимо, $g(x^* - x)$ – функция штрафа за различие между оригинальным экземпляром и контрфактуалом (например, L_0, L_1 норма), $l(x^*; \mathcal{X})$ – функция штрафа за отклонение от многообразия данных.

Авторы методов CE, основанных на оптимизации, фокусируются прежде всего на определении целевой функции, включающей различные метрики для перечисленных выше свойств, а затем на выборе алгоритма нахождения оптимума. Как правило, при этом невозможно гарантировать выпуклость целевой функции. Часто используются итерационные методы различных порядков, также широкое распространение получили метаэвристики (например, генетические алгоритмы). Однако, такой подход требует решения задачи оптимизации при генерации контрфактуалов для каждого нового экземпляра данных. Поэтому в [6] авторам таких работ рекомендуется приводить время вычислений как одну из характеристик алгоритма.

Вторая группа методов основана на поиске в $\mathcal{D}$ прототипов, которые будут использованы для генерации контрфактуалов [23]. Концептуально этот подход близок к методу рассуждения на основе прецедентов (Case Based Reasoning, CBR) [24], который включает четыре шага: (1) *retrieve* – извлечение кейса, имеющего отношение к решаемой проблеме, (2) *reuse* – сопоставление найденного решения с проблемой, (3) *revise* – тестирование решения и при необходимости его пересмотр, (4) *retain* – сохранение успешно адаптированного решения.

В частности, в [25] предложен алгоритм, согласно которому набор данных $\mathcal{D}$ рассматривается как множество пар (x, x^*), где (x, x^*) – наиболее близкие объекты, для которых $h(x^*) \neq h(x)$. Для заданного фактуала z находятся ближайший экземпляр x, принадлежащий к тому же классу, $h(z) = h(x)$. Значения атрибутов контрфактуала z^* инициализируются значениями из z, затем изменяются те атрибуты, которые различаются в x и x^*, пока не будет найден такой z^*, что $h(z^*) = h(x^*)$. Если данное условие не достигнуто, то используется следующая пара (x, x^*). Идея заключается в том, что z^* должен отличаться от z так же, как x^* отличается от x.

Третья группа методов CE (генеративные модели) основана на моделировании процесса генерации данных. В данной группе можно выделить два типа моделей: моделирование совместного распределения и каузальные модели.

Модель совместного распределения $P(\mathcal{X})$ обучается на основе наблюдений $\mathcal{D}$ и затем используется для поиска контрфактуалов. В качестве такой модели в CE чаще всего используются вариационные автоэнкодеры (variational autoencoder, VAE), которые состоят из двух частей – энкодера, отображающего распределение признаков $P(\mathcal{X})$ в пространстве $\mathbb{R}^m$ в распределение латентных переменных $P(\mathcal{Z})$ в пространстве меньшей размерности $\mathcal{Z} \subset \mathbb{R}^k$ $(k < m)$, и декодера, генерирующего значение x', соответствующее точке z' в $P(\mathcal{Z})$. Подход на основе VAE открывает интересную перспективу – проводить поиск контрфактуалов в латентном пространстве, в частности, некоторые авторы используют для этого градиентный спуск [26, 27], однако, как показано в [28], это связано с потенциальными проблемами.

Авторы методов CE на основе VAE вынуждены учитывать перечисленные выше требования к генерации контрфактуальных объяснений, поэтому они вносят дополнительные ограничения в модель латентных представлений. Так, в [29] адаптируется традиционная схема, при которой энкодер используется только для поиска $P(\mathcal{Z})$ и не участвует в генерации данных, и включают его в процесс генерации. С помощью энкодера находится точка z в латентном пространстве, соответствующая заданному фактуалу x, контрфактуал генерируется из точки $z^* = z + \delta$, где δ – малое возмущение. Это должно обеспечивать требование близости. Кроме того, авторы этой работы кластеризуют латентное пространство на основе Гауссовской смеси, чтобы получить условное распределение $P(\mathcal{Z}|\mathcal{J})$, где $\mathcal{J}$ – множество неизменяемых признаков.

Авторы работы [28] используют модель VAE, адаптированную для поиска латентных переменных, коррелированных с метками класса [30]. При этом латентное пространство разделяется на две части: одна предназначена для обучения представ-

лений, предсказывающих метки, а другая – для обучения остальных латентных представлений, необходимых для генерации данных. Это позволяет генерировать контрфактуалы, изменяя только релевантные латентные признаки. Сгенерированные примеры затем фильтруются в соответствии с причинно-следственными ограничениями (например, повышение уровня образования заемщика должно сопровождаться соответствующим увеличением его возраста).

Отметим, что помимо VAE могут использоваться и другие модели совместного распределения $P(\mathcal{X})$, в частности, статистические модели, такие как копулы и байесовские сети, однако, эти техники значительно реже применяются в задачах CE (см. обзоры алгоритмов в [5, 8]). Кроме того, в некоторых специфических случаях, например, в задачах анализа изображений, могут применяться генеративные состязательные сети [13].

Каузальная модель может быть представлена с как ориентированный ациклический граф (directed acyclic graph, DAG), что позволяет компактно и наглядно отобразить структуру исследуемой системы [10]. Способность DAG кодировать причинно-следственные связи основана на графическом критерии *d*-разбиения (*d*-separation), которое соответствует условной независимости переменных в наборе данных. Другими словами, для любых трех непересекающихся подмножеств переменных (X, Y, Z), если вершины X и Y условно независимы при наличии Z в совместном распределении $\mathcal{P}$, то они будут *d*-разделены в графе $\mathcal{G}$ (Марковское условие): $(X \perp\!\!\!\perp_{\mathcal{P}} Y) | Z \Rightarrow (X \perp\!\!\!\perp_{\mathcal{G}} Y) | Z$. Узлы DAG соответствуют переменным, ребра – связям между ними, а направление ребер – причинно-следственным отношениям.

DAG соответствует структурной модели $\mathcal{M}$:

$$\mathcal{M} = (\mathbf{S}, P_U), \quad \mathbf{S} = \left\{ X_j := f_j\left(X_{pa(j)}, U_j\right) \right\}_{j=1}^{m},$$
$$P_U = P_{U1} \times \ldots \times P_{Um}.$$

Здесь $\mathbf{S}$ – структурные уравнения, задающие правила генерации наблюдаемых переменных X_j в виде детерминированной функции их предков в каузальной модели $X_{pa(j)} \subseteq \mathrm{X} \backslash X_j$. Предположение о взаимной независимости шумов U_j (полная факторизация P_U) подразумевает отсутствие ненаблюдаемых конфаундеров – спутывающих переменных, влияющих на причину и следствие одновременно. Отметим, что во многих исследованиях полагается, что шум является аддитивным, т.е. $\mathbf{S} = \left\{ X_j := f_j\left(X_{pa(j)} + U_j\right) \right\}_{j=1}^{m}$, это позволяет построить эффективные алгоритмы идентификации модели по данным [31].

Важным элементом каузального моделирования является аппарат do-вычислений (do-calculus) [10]. Например, интервенция, т.е. присвоение подмножеству переменных $\mathbf{X}_K$ ($K \subseteq |m|$) значений $\boldsymbol{\theta}$, описывается с помощью оператора $do(\mathbf{X}_K = \boldsymbol{\theta})$. Распределение оставшихся переменных $\mathbf{X}_{-k}$ может быть получено из системы $\mathbf{S}^{do(\mathbf{X}_K = \boldsymbol{\theta})}$, в которой уравнения для $\mathbf{X}_k$ заменены соответствующими значениями. Таким образом, каузальная модель может быть использована для нахождения контрфактуалов [20], для экземпляра x контрфактуал определяется как $x^* = \mathbf{X}(a)|x$, где $a = do(\mathbf{X}_K = \boldsymbol{\theta})$, $a \in A$, a – действие, A – множество допустимых действий.

Каузальные модели могут быть восстановлены из наблюдаемых данных или построены на основе экспертных знаний. Однако модель $\mathcal{M}$, обученная на данных, может быть несовершенной, например, из-за ограниченности выборки или, что более важно, из-за неправильной спецификации модели (т.е. принятия неверной параметрической формы структурных уравнений). С другой стороны, хотя во многих случаях экспертные знания позволяют построить причинно-следственную модель, но предположения о виде структурных уравнений, как правило, не поддаются проверке [32]. В результате контрфактуальные объяснения, вычисленные на основе неверно определенной каузальной модели, могут оказаться неточными и рекомендовать неоптимальные или, что еще хуже, неэффективные действия.

Чтобы преодолеть эти ограничения, авторы [20] предлагают два вероятностных подхода к выбору оптимальных действий при ограниченном знании причинно-следственных связей (например, когда известен только DAG). Первый из них применим к моделям с аддитивным гауссовским шумом и использует байесовское усреднение для оценки контрфактуального распределения. Во втором случае исключаются любые предположения о структурных уравнениях, а вместо этого вычисляется средний эффект действий на объекты, которые похожи на рассматриваемый фактуал.

1.2. Генерация синтетических табличных данных

Генерация синтетических данных (synthetic data generation, SDG) является ключевым элементом решения нескольких проблем машинного обучения: анонимизации данных, дополнения малых

наборов данных, выравнивания классов в случае сильного дисбаланса и т.д. [14].

Определение 3. Модель генерации синтетических данных – это функция $g \in \mathcal{G}$, которая для набора наблюдаемых данных $\mathcal{D} \sim \mathbb{P}$, возвращает набор данных $\mathcal{D}^S = g(\mathcal{D}, \theta)$ заданного размера, $\mathcal{D}^S \sim \mathbb{P}^S$, так что выполняется условие $\mathbb{P}^S \approx \mathbb{P}$, $x_i \neq x_j$, $\forall x_i \in \mathcal{D} \wedge \forall x_j \in \mathcal{D}^S$. Здесь θ – вектор гиперпараметров, определяющий политику генерации и $\mathcal{G}$ – семейство генеративных функций.

Математически это можно представить как задачу минимизации расстояния Кульбака-Лейблера:

$$\theta^* = \arg\min_{\theta} \sum_i \mathbb{P}(x_i) \log g(x_i, \theta).$$

Исходя из данного определения, ключевой метрикой эффективности генеративной модели является точность (fidelity) соответствия распределения синтетических данных $\mathbb{P}^S$ эмпирическому распределению $\mathbb{P}$. Кроме этого, могут вводиться дополнительные метрики [33], например, разнообразие (diversity) и обобщение (generalization). Согласно требованию разнообразия синтетические экземпляры должны охватывать весь диапазон изменений $\mathcal{D}$. Свойство обобщения требует, чтобы синтетические данные не были копиями реальных наблюдений.

В данном обзоре мы ограничимся рассмотрением генерации синтетических табличных (кросс-секционных) данных (tSDG). Можно выделить следующие классы методов tSDG:

- Модели рандомизации, основанные на перемешивании, интерполяции и геометрической трансформации исходных данных и добавлении случайного шума.
- Вероятностные алгоритмы, которые генерируют данные на основе многомерного распределения $\mathbb{P}^S$, моделирующего реальное распределение $\mathbb{P}$. Здесь можно выделить несколько подходов, а именно:
  - ◊ моделирование совместного распределения $\mathbb{P}$, например, на основе Гауссовской смеси или копул [15];
  - ◊ последовательная генерация атрибутов $\mathcal{D}$ на основе условных распределений $\mathbb{P}(x_i \mid \mathcal{D} \backslash \{x_1, ..., x_{i-1}\})$;
  - ◊ моделирование $\mathbb{P}$ с помощью факторизации на основе графической вероятностной модели (байесовской сети) [34].
- Модели, генерирующие данные из латентного пространства меньшей размерности.
- Моделирование сэмплирования на основе генеративных состязательных сетей (GAN).
- Модели, основанные на априорно известной каузальной структуре.

Отметим, что подход на основе моделей условных распределений синтезирует переменные x_i последовательно с помощью регрессионных моделей $x_i = f(x_1, ..., x_{i-1})$, которые могут быть построены как параметрическими (линейная регрессия), так и непараметрическими (дерево решений) методами [35, 36]. Таким образом, условные распределения $\mathbb{P}(x_i \mid \mathcal{D} \backslash \{x_1, ..., x_{i-1}\})$, из которых берутся синтетические значения x_i, определяются для каждой переменной отдельно и зависят от атрибутов $x_1, ..., x_{i-1}$, которые находятся раньше в последовательности синтеза. Значение самой первой переменной в последовательности генерируется на основе ее маргинального распределения.

Детальный анализ методов tSDG представлен в [14]. Ряд публикаций [16, 17] сравнивают некоторые из рассмотренных подходов на реальных наборах данных. Из представленных результатов можно сделать вывод, что не существует доминирующего метода и качество генерации зависит от конкретной задачи.

Можно также отметить, что концептуально методы синтетической генерации данных близки к алгоритмам контрфактуальных объяснений: и те, и другие базируются на моделировании распределения наблюдаемых данных, но различаются конечным результатом. Если цель CE – найти объект максимально близкий к исследуемому, но с противоположной меткой (см. Определение 1), то цель tSDG – сгенерировать множество объектов, которые принадлежат распределению наблюдаемых данных (Определение 3). Соответственно, они базируются на разных метриках эффективности.

2. Предлагаемый метод

Как следует из представленного выше обзора, известные алгоритмы CE обладают несколькими ограничениями. Методы, основанные на оптимизации, требуют повторного построения модели для каждого фактуала, подходы на основе прототипов требуют наличия пар «фактуал – контрфактуал» в обучающем наборе $\mathcal{D}$, подходы на основе генеративных моделей вводят дополнительные ограничения в алгоритм, что также усложняет вычисления. В то же время, как отмечено выше, методы синтетической генерации данных концептуально близки

к CE и отличаются лишь результатом и метриками его оценки.

Исходя из этих соображений, мы предлагаем двухэтапный метод генерации контрфактуальных объяснений (*рис. 2*). На первом этапе обучается модель $g(\mathcal{D}, \theta)$ генерации синтетических данных. Согласно Определению 3, данная модель эмулирует эмпирическое распределение $\mathbb{P}$ реальных данных. С помощью этой модели для данного фактуала x генерируется множество потенциальных контрфактуалов $\{x^*\}_g$.

На втором этапе с помощью модели отбора $s(R)$ из $\{x^*\}_g$ отбирается множество $\{x^*\}_s$, элементы которого удовлетворяют ограничениям R. Множество $\{x^*\}_s$, является решением задачи CE. Модель отбора может включать любые ограничения, сформулированные в виде неравенств вида $r(c) = m(c) \leq v(c)$, $r \in R$. Здесь c – требование к результату (например, валидность, близость, разреженность для CE или стоимость реализации), $m(c)$ – соответствующая метрика, $v(c)$ – граница допустимых значений. Отметим, что в число требований могут быть также включены ограничения конкретной предметной области.

Предложенный подход обладает следующими преимуществами:

- генеративная модель строится один раз и позволяет вычислять контрфактуалы для любых новых наблюдений без повторного обучения;
- разделение процесса на два этапа позволяет использовать достаточно простые, легко изменяемые правила отбора;
- модель отбора может включать не только требования задач CE, но и любые ограничения, специфичные для рассматриваемой предметной области.

3. Эксперимент

Для проверки предложенного метода прежде всего необходимо убедиться, что методы tSGD позволяют генерировать контрфактуалы, удовлетворяющие требованиям, перечисленным в разделе 1.1, а также сравнить результаты с известными методами CE.

Модели генерации $g(\mathcal{D}, \theta)$, которые будут использованы в эксперименте, представлены в *таблице 1*. Мы отобрали простейшие статистические модели, поскольку наша задача – предложить эффективный метод генерации контрфактуалов с небольшими вычислительными затратами. К таким моделям относятся Гауссовская копула (GC), последовательная непараметрическая модель на базе условных распределений (CD) и байесовская сеть (BN), которая моделирует распределение $\mathbb{P}$ как произведение условных распределений факторов (признаков). Для сравнения мы также включили модель, генерирующую данные на основе маргинальных распределений признаков (MD). Ее можно рассматривать как вырожденный случай BN, в котором связи между признаками не учитываются. Как отмечалось выше, такие простые модели практически не используются в задачах CE, однако мы предполагаем, что их потенциал может быть использован значительно эффективнее с помощью предлагаемого двухэтапного подхода.

Рис. 2. Двухэтапный метод генерации контрфактуальных объяснений.

Таблица 1.

Методы генерации синтетических табличных данных

ID	Тип модели	Описание	Источник
GC	Совместное распределение $\mathbb{P}$	Гауссовская копула	[15][5]
CD	Условные распределения $\mathbb{P}(x_i \mid \mathcal{D}\backslash x_i)$	Непараметрический метод / дерево решений	[36][6]
BN	Факторизация $\mathbb{P} = \prod p(x_i)$	Байесовская сеть	[34][7]
MD	Маргинальные распределения x_i	Сэмплинг на основе маргинальных распределений	[38][8]
GAN	Глубокое обучение	Генеративная состязательная сеть	[37][9]

Кроме того, как следует из обзора литературы, большинство исследователей, использующих генеративные модели для решения задачи CE, фокусируются на сложных алгоритмах, основанных на глубоких нейронных сетях, поэтому мы также рассмотрели GAN. Мы также исследовали возможность применения VAE, но в наших экспериментах эти модели не позволили добиться устойчивой генерации $\{x^*\}_g$. Скорее всего, это объясняется недостаточным объемом данных для обучения (*таблица 2*).

Модель отбора $s(R)$ задана в виде правила

$$R: h(x^*) \neq h(x) \wedge x_i^* \in \left\{\overline{x}_i \mp 1{,}5 \cdot IQR(\mathcal{D})\right\} \wedge dist(x^*, x) \rightarrow \\ \rightarrow \min \wedge \left|\{x^*\}_s\right| = k.$$

Это означает, что для данного x из сгенерированного множества $\{x^*\}_g$ будут отбираться k экземпляров, метка которых $h(x^*)$ не равна метке $h(x)$, значения атрибутов x^* находятся в диапазоне трех межквартильных интервалов $IQR(\mathcal{D})$ относительно среднего x_i (граница Тьюки), и расстояние между x и x^* минимально.

Таблица 2 представляет три набора данных, использованных в экспериментах, их общие характеристики и классификацию признаков с точки зрения осуществимости изменений (признак изменяем, изменение не осуществимо, признак не изменяем). Эти общедоступные наборы данных широко используются в работах по машинному обучению и, в частности, исследованиях, касающихся CE. Использование открытых наборов данных обеспечивает воспроизводимость результатов.

В *таблице 2* также представлены результаты обучения классификатора h, который используется в процессе нахождения CE: метрика ROC AUC, полученная с помощью 10-кратной кросс-валидации и модель, показавшая наилучшие результаты. В одном случае это Random Forest (RF), в остальных случаях – CatBoost (CB).

Одной из наиболее популярных библиотек, реализующих методы CE, является DiCE13 [19], которая поддерживает три способа поиска контрфактуалов. Помимо случайного поиска, это оптимизация на основе генетических алгоритмов и метод поиска и последующей адаптации прототипов в обучающей выборке [23]. Мы использовали эти модели для сравнительной оценки полученных результатов. Для каждого набора данных обучались все три типа моделей и выбиралась лучшая. Отметим, что подход на основе прототипов не позволил найти контрфактуалы ни для одного набора данных. Очевидно, это связано с ограничением, отмеченным выше: в $\mathcal{D}$ должен присутствовать набор пар (x, x^*) для широкого диапазона фактуалов.

Для оценки результатов вычисления контрфактуалов будем использовать метрики валидности (V), близости (P), разреженности (S), разнообразия (D) и правдоподобия (U), описанные выше. Указание признаков, изменение которых возможно, осуществляется на уровне модели генерации $g(\mathcal{D}, \theta)$.

[5] https://sdv.dev
[6] https://www.synthpop.org.uk/
[7] https://github.com/DataResponsibly/DataSynthesizer
[8] https://github.com/vanderschaarlab/synthcity
[9] https://github.com/NextBrain-ai/nbsynthetic

Таблица 2.

Наборы данных

$\mathcal{D}$	$n \times m$	Признаки			Классификатор		Описание
		Неизменяем	Изменение не реализуемо	Изменяем	Модель $h(x)$	AUC	
German Credit[10]	1000 × 20	3	14	3	RF	0,79 (0,03)	700 одобренных и 300 заблокированных кредитных заявок
Adult[11]	48842 × 14	8	3	3	CB	0,93 (0,002)	Уровень дохода в зависимости от данных переписи населения
Loan Default[12]	255347 × 16	8	3	5	CB	0,76 (0,002)	29653 плохих и 225694 одобренных заявок на кредит

4. Анализ результатов эксперимента

Рассмотрим процесс применения предложенного метода на примере набора данных German Credit. Этот датасет содержит записи о 1000 заявках на кредит, 700 из которых были одобрены. В числе атрибутов сумма и срок кредита, а также показатели социального и финансового положения заемщика (кредитный рейтинг, срок работы на одном месте, доля платежей по кредиту в общем доходе заемщика и т.д.). Большинство атрибутов являются либо категориальными, либо порядковыми.

Задачей CE в данном случае является генерация контрфактуалов для заемщиков, которым было отказано в кредите. Анализ данных показывает, что изменяемыми атрибутами являются *laufzeit* – срок кредита в месяцах, *hoehe* – сумма кредита и *buerge* – наличие созаемщика или поручителя. Все остальные атрибуты либо не изменяемы (пол, гражданство), либо не могут быть изменены прямым воздействием (кредитный рейтинг).

Процедура вычислений выполнена в соответствии с методом, представленным на *рис. 2*. На первом этапе обучена модель генерации $g(\mathcal{D}, \theta)$, с помощью которой для исследуемого фактуала генерируется 200 синтетических экземпляров. Из этого набора отбираются экземпляры в соответствии с правилом, заданным уравнением (1).

В *таблице 3* представлен пример данных, сгенерированных для отклоненной заявки на сумму 2348 DM на срок 36 месяцев. Как следует из представлен-

Таблица 3.

Пример генерируемых данных (атрибуты объяснены в тексте)

	laufzeit	*hoehe*	*buerge*	Метка класса
Фактуал	36	2384	1	0
Контрфактуалы	8	1956	1	1
	14	2234	2	1
	26	4276	3	1

[10] South German Credit. UCI Machine Learning Repository. 2019. https://doi.org/10.24432/C5X89F.
[11] Becker,B., Kohavi,R. Adult. UCI Machine Learning Repository. 1996. https://doi.org/10.24432/C5XW20.
[12] Loan Default Dataset. https://www.kaggle.com/datasets/nikhil1e9/loan-default/data
[13] http://interpret.ml/DiCE/index.html

ных данных, кредит для данного заемщика может быть одобрен при снижении срока до 8 месяцев и сумы до 1956 марок. Если заемщик представит второе ответственное лицо, которое будет участвовать в погашении кредита (*buerge* = 2), то сумма может быть увеличена до 2234 DM сроком на 14 месяцев. При наличии поручителя (*buerge* = 3) кредит может составить 4276 DM на срок 26 месяцев. Таким образом, даже три представленных контрфактуала позволяют описать ситуацию для конкретного заемщика и предложить ему действия, которые помогут добиться поставленной цели.

Таблица 4 представляет средние значения и стандартные отклонения метрик качества для рассматриваемых методов, рассчитанные по всем трем наборам данных, а *рис. 3* – распределения метрик по датасетам. Лучшие значения метрик в *таблице 4* выделено жирным шрифтом.

Таблица 4.

Средние значения и стандартные отклонения метрик качества моделей по трем наборам данных

Модель	*V*	*P*	*S*	*D*	*U*
BN	**1,000 (0,000)**	**1,230 (0,805)**	3,790 (1,251)	1,310 (0,340)	**2,053 (0,855)**
CD	**1,000 (0,000)**	1,400 (1,155)	3,303 (1,036)	1,135 (0,228)	2,128 (0,836)
DiCE	0,869 (0,273)	3,138 (1,519)	**1,962 (0,556)**	1,159 (0,206)	2,797 (0,734)
GAN	0,966 (0,186)	1,984 (1,045)	4,190 (1,640)	1,285 (0,372)	2,298 (0,742)
GC	0,967 (0,183)	1,333 (0,851)	3,887 (1,521)	1,284 (0,402)	2,089 (0,828)
MD	1,000 (0,000)	3,198 (2,094)	4,177 (1,506)	**1,314 (0,299)**	2,851 (0,760)

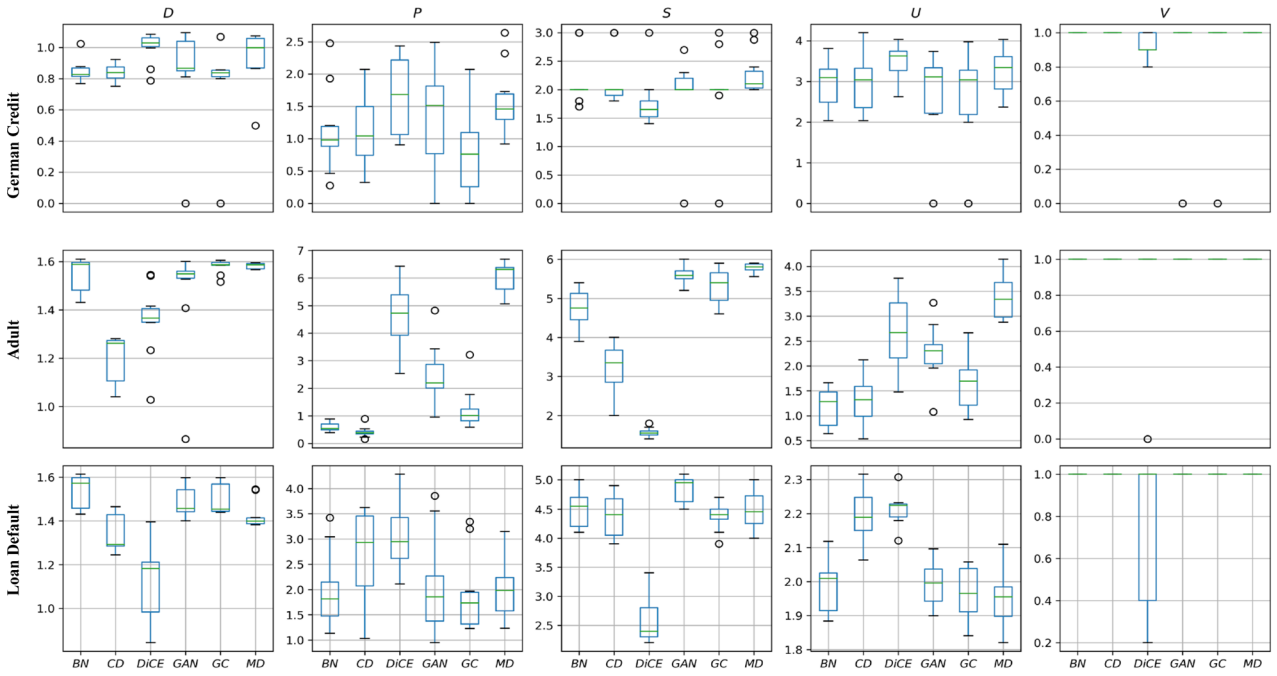

Рис. 3. Метрики качества рассматриваемых моделей по наборам данных.

Отметим, что по большинству метрик (валидность, близость и правдоподобие) лучшие результаты демонстрирует модель на основе байесовской сети (BN), генерирующая выборки на основе условных распределений признаков, т.е. с учетом зависимостей между ними. Если учесть, что по разнообразию эта модель лишь немногим уступает MD, то выбор BN для CE представляется весьма обоснованным. Высокое разнообразие контрфактуалов, генерируемых MD, объясняется тем, что эта модель рассматривает только маргинальные распределения признаков и не учитывает связи между ними. Эта модель должна хорошо работать в случае некоррелированных признаков, но может порождать проблемы, когда такие корреляции присутствуют (см. распределение *D* для набора данных Loan Default на *рис. 3*). Напротив, BN в случае таких данных показывает лучшие результаты по разнообразию среди всех моделей.

В наших экспериментах самая сложная модель GAN уступила другим моделям, возможно, потому, что было недостаточно данных для обучения, хотя авторы использованной нами реализации [37] подчеркивают, что она ориентирована именно на малые обучающие выборки. *Рисунок 3* показывает, что с возрастанием выборки результаты GAN улучшаются, но не превосходят другие модели.

Методы, разработанные непосредственно для решения задачи CE (DiCE), стали лучшими по метрике разреженности (*табл. 4*), но *рисунок 3* показывает, что это достигнуто за счет высоких результатов на самом большом наборе данных (Loan Default). На меньших данных этот метод уступает более простым моделям, в частности GC и BN. Кроме того, следует отметить, что DiCE на всех наборах данных не находит требуемое количество контрфактуалов ($V < 1$) и в этом смысле является наихудшим из рассмотренных методов.

Заключение

Таким образом, можно сделать вывод, что предложенный метод поиска контрфактуалов на основе генерации синтетических данных позволяет добиться результатов, как минимум, сравнимых со «стандартными» методами CE, а в ряде случаев их превосходит, особенно на малых наборах данных. Согласно нашим результатам, наиболее очевидным выбором при этом является модель генерации на основе байесовской сети, которая учитывает связи между атрибутами.

Этот результат открывает новые возможные направления исследований. Байесовская сеть является статистической моделью, поскольку строится на ассоциациях, измеряемых с помощью корреляций. Поэтому представляет интерес изучение каузальных моделей, которые отражают причинно-следственные связи в наборе данных.

При этом следует отметить, что, насколько нам известно, направление, связанное с использованием каузальных моделей для CE, только начинает исследоваться [20], а работы, посвященные их применению для генерации синтетических данных, отсутствуют. ■

Литература

1. Samek W., Muller K.-R. Towards explainable artificial intelligence // Explainable AI: Interpreting, Explaining and Visualizing Deep Learning. Lecture Notes in Computer Science. 2019. Vol. 11700. P. 5–22. https://doi.org/10.1007/978-3-030-28954-6_1
2. Giuste F., Shi W., Zhu Y., Naren T., Isgut M., Sha Y., Tong L., Gupte M., Wang M.D. Explainable artificial intelligence methods in combating pandemics: A systematic review // IEEE Reviews in Biomedical Engineering. 2023. Vol. 16. P. 5–21. https://doi.org/10.1109/RBME.2022.3185953
3. Barocas S., Selbst A. D., Raghavan M. The hidden assumptions behind counterfactual explanations and principal reasons // Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (FAT* '20). 2020. P. 80–89. https://doi.org/10.1145/3351095.3372830
4. Murdoch W.J., Singh C., Kumbier K., Abbasi-Asl R., Yu B. Definitions, methods, and applications in interpretable machine learning // National Academy of Sciences. 2019. Vol. 116(44). P. 22071–22080. https://doi.org/10.1073/pnas.1900654116
5. Guidotti R. Counterfactual explanations and how to find them: Literature review and benchmarking // Data Mining and Knowledge Discovery. 2022. https://doi.org/10.1007/s10618-022-00831-6
6. Verma S., Boonsanong V., Hoang M., Hines K. E., Dickerson J. P., Shah C. Counterfactual explanations and algorithmic recourses for machine learning: A review // arXiv:2010.10596. 2020. https://doi.org/10.4550/arxiv.2010.10596
7. Stepin I., Alonso J.M., Catala A., Pereira-Fariña M. A survey of contrastive and counterfactual explanation generation methods for explainable artificial intelligence // IEEE Access. 2021. Vol. 9. P. 11974–12001. https://doi.org/10.1109/ACCESS.2021.3051315
8. Chou Y.L., Moreira C., Bruza P., Ouyang C., Jorge J. Counterfactuals and causability in explainable artificial intelligence: Theory, algorithms, and applications // Information Fusion. 2022. Vol. 81. P. 59–83. https://doi.org/10.1016/j.inffus.2021.11.003

9. Mishra P. Practical explainable AI using Python: Artificial Intelligence model explanations using python-based libraries, extensions, and frameworks. Apress, 2022.

10. Pearl J. Causality: models, reasoning, and inference. 2nd ed. New York: Cambridge University Press, 2009.

11. Cho S.H., Shin K.S. Feature-weighted counterfactual-based explanation for bankruptcy prediction // Expert Systems with Applications. 2023. Vol. 216. Article 119390. https://doi.org/10.1016/j.eswa.2022.119390

12. Wang D., Chen Z., Florescu I., Wen B. A sparsity algorithm for finding optimal counterfactual explanations: Application to corporate credit rating // Research in International Business and Finance. 2023. Vol. 64. Article 101869. https://doi.org/10.1016/j.ribaf.2022.101869

13. Mertes S., Huber T., Weitz K., Heimerl A., André E. Ganterfactual – counterfactual explanations for medical non-experts using generative adversarial learning // Frontiers in Artificial Intelligence. 2022. Vol. 5. Article 825565. https://doi.org/10.3389/frai.2022.825565

14. Fonseca J., Bacao F. Tabular and latent space synthetic data generation: A literature review // Journal of Big Data. 2023. Vol. 10(1). Article 115. https://doi.org/10.1186/s40537-023-00792-7

15. Patki N., Wedge R., Veeramachaneni K. The synthetic data vault // Proceedings of the 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA). 2016. P. 399–410. https://doi.org/10.1109/DSAA.2016.49

16. Dankar F., Ibrahim M., Ismail L. A multi-dimensional evaluation of synthetic data generators // IEEE Access. 2022. Vol. 10. P. 11147–11158. https://doi.org/10.1109/ACCESS.2022.3144765

17. Endres M., Mannarapotta Venugopal A., Tran T.S. Synthetic data generation: A comparative study // Proceedings of the 26th International Database Engineered Applications Symposium. 2022. P. 94–102. https://doi.org/10.1145/3548785.3548793

18. Wachter S., Mittelstadt B., Russell C. Counterfactual explanations without opening the black box: Automated decisions and the GDPR // Harvard Journal of Law & Technology (Harvard JOLT). 2017. Vol. 31. Article 841.

19. Mothilal R.K., Sharma A., Tan C. Explaining machine learning classifiers through diverse counterfactual explanations // Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (FAT* '20). 2020. P. 607–617. https://doi.org/10.1145/3351095.3372850

20. Karimi A.H., Barthe G., Schölkopf B., Valera I. A survey of algorithmic recourse: Contrastive explanations and consequential recommendations // ACM Computing Surveys. 2023. Vol. 55(5). Article 95. https://doi.org/10.1145/3527848

21. Breunig M.M., Kriegel H.-P., Ng R.T., Sander J. LOF: identifying density-based local outliers // Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data (ICDM). 2000. P. 93–104. https://doi.org/10.1145/335191.335388

22. Poyiadzi K., Sokol K., Santos-Rodriguez R., De Bie T., Flach P. FACE: feasible and actionable counterfactual explanations // Proceedings of the 2020 AAAI/ACM Conference on AI, Ethics, and Society (AIES 2020). 2020. P. 344–350. https://doi.org/10.1145/3351095.3372850

23. van Looveren A., Klaise J. Interpretable counterfactual explanations guided by prototypes // Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD 2021). 2021. P. 650–665. https://doi.org/10.1007/978-3-030-86520-7_40

24. Aamodt A., Plaza E. Case-based reasoning: Foundational issues, methodological variations, and system approaches // Artificial Intelligence Communications. 1994. Vol. 7(1). P. 39–59.

25. Keane M.T., Smyth B. Good counterfactuals and where to find them: A case-based technique for generating counterfactuals for explainable AI (XAI) // Proceedings of the 28th International Conference on Case-Based Reasoning Research and Development (ICCBR). 2020. P. 163–178. https://doi.org/10.1007/978-3-030-58342-2_11

26. Joshi S., Koyejo O., Vijitbenjaronk W., Kim B., Ghosh J. Towards realistic individual recourse and actionable explanations in black-box decision making systems // arXiv:1907.09615. 2019. https://doi.org/10.48550/arXiv.1907.09615

27. Guyomard V., Fessant F., Bouadi T., Guyet T. Post-hoc counterfactual generation with supervised autoencoder // Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD 2021). 2021. P. 105–114. https://doi.org/10.1007/978-3-030-93736-2_10

28. Downs M., Chu J.L., Yacoby Y., Doshi-Velez F., Pan W. CRUDS: Counterfactual recourse using disentangled subspaces // Proceedings of the 2020 ICML Workshop on Human Interpretability in Machine Learning (WHI 2020). 2020. P. 1–23.

29. Pawelczyk M., Broelemann K., Kasneci G. Learning model-agnostic counterfactual explanations for tabular data // Proceedings of the Web Conference 2020 (WWW'20). 2020. P. 3126–3132. https://doi.org/10.1145/3366423.3380087

30. Klys J., Snell J., Zemel R. Learning latent subspaces in variational autoencoders // Advances in Neural Information Processing Systems 31 (NeurIPS 2018). 2018.

31. Hoyer P., Janzing D., Mooij J.M., Peters J., Schölkopf B. Nonlinear causal discovery with additive noise models // Advances in Neural Information Processing Systems 21 (NIPS 2008). 2008.

32. Peters J., Janzing D., Schölkopf B. Elements of causal inference: foundations and learning algorithms. MIT press, 2017.

33. Alaa A., van Breugel B., Saveliev E.S., van der Schaar M. How faithful is your synthetic data? Sample-level metrics for evaluating and auditing generative models // Proceedings of the 39th International Conference on Machine Learning. 2022. P. 290–306.

34. Ping P., Stoyanovich J., Howe D. DataSynthesizer: Privacy-preserving synthetic datasets // Proceedings of the 29th International Conference on Scientific and Statistical Database Management (SSDBM'17). 2017. https://doi.org/10.1145/3085504.3091117

35. Drechsler J., Reiter J.P. An empirical evaluation of easily implemented nonparametric methods for generating synthetic datasets // Computational Statistics & Data Analysis. 2011. Vol. 55(12). P. 3232–3243. https://doi.org/10.1016/j.csda.2011.06.006

36. Nowok B., Raab G.M., Dibben C. Synthpop: Bespoke creation of synthetic data in R // Journal of Statistical Software. 2016. Vol. 74. P. 1–26. https://doi.org/10.18637/jss.v074.i11

37. Marin J. Evaluating synthetically generated data from small sample sizes: An experimental study // arXiv:2211.10760. 2022. https://doi.org/10.48550/arXiv.2211.10760

38. Qian Z., Cebere B.C., van der Schaar M. Synthcity: facilitating innovative use cases of synthetic data in different data modalities // arXiv:2301.07573. 2023. https://doi.org/10.48550/arxiv.2301.07573

Об авторах

Зеленков Юрий Александрович

д.т.н.;

профессор, департамент бизнес-информатики, Высшая школа бизнеса, Национальный исследовательский университет «Высшая школа экономики», Россия, 119049, г. Москва, ул. Шаболовка, д. 28/11, стр. 4;

E-mail: yzelenkov@hse.ru

ORCID: 0000-0002-2248-1023

Лашкевич Елизавета Витальевна

аспирант, департамент бизнес-информатики, Высшая школа бизнеса, Национальный исследовательский университет «Высшая школа экономики», Россия, 119049, г. Москва, ул. Шаболовка, д. 28/11, стр. 4;

E-mail: evlashkevich@hse.ru

ORCID: 0000-0002-3241-2291

Counterfactual explanations based on synthetic data generation

Yuri A. Zelenkov
E-mail: yzelenkov@hse.ru

Elizaveta V. Lashkevich
E-mail: evlashkevich@hse.ru

Graduate School of Business, HSE University, Moscow, Russia

Abstract

A counterfactual explanation is the generation for a particular sample of a set of instances that belong to the opposite class but are as close as possible in the feature space to the factual being explained. Existing algorithms that solve this problem are usually based on complicated models that require a large amount of training data and significant computational cost. We suggest here a method that involves two stages. First, a synthetic set of potential counterfactuals is generated based on simple statistical models (Gaussian copula, sequential model based on conditional distributions, Bayesian network, etc.), and second, instances satisfying constraints on probability, proximity, diversity, etc. are selected. Such an

approach enables us to make the process transparent, manageable and to reuse the generative models. Experiments on three public datasets have demonstrated that the proposed method provides results at least comparable to known algorithms of counterfactual explanations, and superior to them in some cases, especially on low-sized datasets. The most effective generation model is a Bayesian network in this case.

Keywords: counterfactual explanations, synthetic data generation, multimodal distribution modelling, Bayesian network, credit scoring

Citation: Zelenkov Yu.A., Lashkevich E.V. (2024) Counterfactual explanations based on synthetic data generation. *Business Informatics*, vol. 18, no. 3, pp. 24–40. DOI: 10.17323/2587-814X.2024.3.24.40

References

1. Samek W., Muller K.-R. (2019) Towards explainable artificial intelligence. *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning. Lecture Notes in Computer Science*, vol. 11700, pp. 5–22. https://doi.org/10.1007/978-3-030-28954-6_1
2. Giuste F., Shi W., Zhu Y., Naren T., Isgut M., Sha Y., Tong L., Gupte M., Wang M.D. (2023) Explainable artificial intelligence methods in combating pandemics: A systematic review. *IEEE Reviews in Biomedical Engineering*, vol. 16, pp. 5–21. https://doi.org/10.1109/RBME.2022.3185953
3. Barocas S., Selbst A. D., Raghavan M. (2020) The hidden assumptions behind counterfactual explanations and principal reasons. Proceedings of the *2020 Conference on Fairness, Accountability, and Transparency (FAT* '20)*, pp. 80–89. https://doi.org/10.1145/3351095.3372830
4. Murdoch W.J., Singh C., Kumbier K., Abbasi-Asl R., Yu B. (2019) Definitions, methods, and applications in interpretable machine learning. *National Academy of Sciences*, vol. 116(44), pp. 22071–22080. https://doi.org/10.1073/pnas.1900654116
5. Guidotti R. (2022) Counterfactual explanations and how to find them: Literature review and benchmarking. *Data Mining and Knowledge Discovery*. https://doi.org/10.1007/s10618-022-00831-6
6. Verma S., Boonsanong V., Hoang M., Hines K. E., Dickerson J. P., Shah C. (2020) Counterfactual explanations and algorithmic recourses for machine learning: A review. *arXiv:2010.10596*. https://doi.org/10.4550/arxiv.2010.10596
7. Stepin I., Alonso J.M., Catala A., Pereira-Fariña M. (2021) A survey of contrastive and counterfactual explanation generation methods for explainable artificial intelligence. *IEEE Access*, vol. 9, pp. 11974–12001. https://doi.org/10.1109/ACCESS.2021.3051315
8. Chou Y.L., Moreira C., Bruza P., Ouyang C., Jorge J. (2022) Counterfactuals and causability in explainable artificial intelligence: Theory, algorithms, and applications. *Information Fusion*, vol. 81, pp. 59–83. https://doi.org/10.1016/j.inffus.2021.11.003
9. Mishra P. (2022) *Practical explainable AI using Python: Artificial Intelligence model explanations using python-based libraries, extensions, and frameworks*. Apress.
10. Pearl J. (2009) *Causality: models, reasoning, and inference. 2nd ed*. New York: Cambridge University Press.
11. Cho S.H., Shin K.S. (2023) Feature-weighted counterfactual-based explanation for bankruptcy prediction. *Expert Systems with Applications*, vol. 216, article 119390. https://doi.org/10.1016/j.eswa.2022.119390
12. Wang D., Chen Z., Florescu I., Wen B. (2023) A sparsity algorithm for finding optimal counterfactual explanations: Application to corporate credit rating. *Research in International Business and Finance*, vol. 64, article 101869. https://doi.org/10.1016/j.ribaf.2022.101869
13. Mertes S., Huber T., Weitz K., Heimerl A., André E. (2022) Ganterfactual—counterfactual explanations for medical non-experts using generative adversarial learning. *Frontiers in Artificial Intelligence*, vol. 5, article 825565. https://doi.org/10.3389/frai.2022.825565
14. Fonseca J., Bacao F. (2023) Tabular and latent space synthetic data generation: A literature review. *Journal of Big Data*, vol. 10(1), article 115. https://doi.org/10.1186/s40537-023-00792-7
15. Patki N., Wedge R., Veeramachaneni K. (2016) The synthetic data vault. Proceedings of the *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 399–410. https://doi.org/10.1109/DSAA.2016.49
16. Dankar F., Ibrahim M., Ismail L. (2022) A multi-dimensional evaluation of synthetic data generators. *IEEE Access*, vol. 10, pp. 11147–11158. https://doi.org/10.1109/ACCESS.2022.3144765
17. Endres M., Mannarapotta Venugopal A., Tran T.S. (2022) Synthetic data generation: A comparative study. Proceedings of the *26th International Database Engineered Applications Symposium*, pp. 94–102. https://doi.org/10.1145/3548785.3548793
18. Wachter S., Mittelstadt B., Russell C. (2017) Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harvard Journal of Law & Technology (Harvard JOLT)*, vol. 31, article 841.
19. Mothilal R.K., Sharma A., Tan C. (2020) Explaining machine learning classifiers through diverse counterfactual explanations. Proceedings of the *2020 Conference on Fairness, Accountability, and Transparency (FAT* '20)*, pp. 607–617. https://doi.org/10.1145/3351095.3372850
20. Karimi A.H., Barthe G., Schölkopf B., Valera I. (2023) A survey of algorithmic recourse: Contrastive explanations and consequential recommendations. *ACM Computing Surveys*, vol. 55(5), article 95. https://doi.org/10.1145/3527848

21. Breunig M.M., Kriegel H.-P., Ng R.T., Sander J. (2000) LOF: identifying density-based local outliers. Proceedings of the *2000 ACM SIGMOD International Conference on Management of Data (ICDM)*, pp. 93–104. https://doi.org/10.1145/335191.335388

22. Poyiadzi K., Sokol K., Santos-Rodriguez R., De Bie T., Flach P. (2020) FACE: feasible and actionable counterfactual explanations. Proceedings of the *2020 AAAI/ACM Conference on AI, Ethics, and Society (AIES 2020)*, pp. 344–350. https://doi.org/10.1145/3351095.3372850

23. van Looveren A., Klaise J. (2021) Interpretable counterfactual explanations guided by prototypes. Proceedings of the *Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD 2021)*, pp. 650–665. https://doi.org/10.1007/978-3-030-86520-7_40

24. Aamodt A., Plaza E. (1994) Case-based reasoning: Foundational issues, methodological variations, and system approaches. *Artificial Intelligence Communications*, vol. 7(1), pp. 39–59.

25. Keane M.T., Smyth B. (2020) Good counterfactuals and where to find them: A case-based technique for generating counterfactuals for explainable AI (XAI). Proceedings of the *28th International Conference on Case-Based Reasoning Research and Development (ICCBR)*, pp. 163–178. https://doi.org/10.1007/978-3-030-58342-2_11

26. Joshi S., Koyejo O., Vijitbenjaronk W., Kim B., Ghosh J. (2019) Towards realistic individual recourse and actionable explanations in black-box decision making systems. *arXiv:1907.09615*. https://doi.org/10.48550/arXiv.1907.09615

27. Guyomard V., Fessant F., Bouadi T., Guyet T. (2021) Post-hoc counterfactual generation with supervised autoencoder. Proceedings of the *Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD 2021)*, pp. 105–114. https://doi.org/10.1007/978-3-030-93736-2_10

28. Downs M., Chu J.L., Yacoby Y., Doshi-Velez F., Pan W. (2020) CRUDS: Counterfactual recourse using disentangled subspaces. Proceedings of the *2020 ICML Workshop on Human Interpretability in Machine Learning (WHI 2020)*, pp. 1–23.

29. Pawelczyk M., Broelemann K., Kasneci G. (2020) Learning model-agnostic counterfactual explanations for tabular data. Proceedings of the *Web Conference 2020 (WWW'20)*, pp. 3126–3132. https://doi.org/10.1145/3366423.3380087

30. Klys J., Snell J., Zemel R. (2018) Learning latent subspaces in variational autoencoders. *Advances in Neural Information Processing Systems 31 (NeurIPS 2018)*.

31. Hoyer P., Janzing D., Mooij J.M., Peters J., Schölkopf B. (2008) Nonlinear causal discovery with additive noise models. *Advances in Neural Information Processing Systems 21 (NIPS 2008)*.

32. Peters J., Janzing D., Schölkopf B. (2017) *Elements of causal inference: foundations and learning algorithms*. MIT press.

33. Alaa A., van Breugel B., Saveliev E.S., van der Schaar M. (2022) How faithful is your synthetic data? Sample-level metrics for evaluating and auditing generative models. Proceedings of the *39th International Conference on Machine Learning*, pp. 290–306.

34. Ping P., Stoyanovich J., Howe D. (2017) DataSynthesizer: Privacy-preserving synthetic datasets. Proceedings of the *29th International Conference on Scientific and Statistical Database Management (SSDBM'17)*. https://doi.org/10.1145/3085504.3091117

35. Drechsler J., Reiter J.P. (2011) An empirical evaluation of easily implemented nonparametric methods for generating synthetic datasets. *Computational Statistics & Data Analysis*, vol. 55(12), pp. 3232–3243. https://doi.org/10.1016/j.csda.2011.06.006

36. Nowok B., Raab G.M., Dibben C. (2016) Synthpop: Bespoke creation of synthetic data in R. *Journal of Statistical Software*, vol. 74, pp. 1–26. https://doi.org/10.18637/jss.v074.i11

37. Marin J. (2022) Evaluating synthetically generated data from small sample sizes: An experimental study. *arXiv:2211.10760*. https://doi.org/10.48550/arXiv.2211.10760

38. Qian Z., Cebere B.C., van der Schaar M. (2023) Synthcity: facilitating innovative use cases of synthetic data in different data modalities. *arXiv:2301.07573*. https://doi.org/10.48550/arxiv.2301.07573

About the authors

Yuri A. Zelenkov

Dr. Sci. (Tech.);

Professor, Department of Business Informatics, Graduate School of Business, HSE University, 28/11, Shabolovka Str., Moscow 119049, Russia;

E-mail: yzelenkov@hse.ru

ORCID: 0000-0002-2248-1023

Elizaveta V. Lashkevich

Doctoral Student, Department of Business Informatics, Graduate School of Business, HSE University, 28/11, Shabolovka Str., Moscow 119049, Russia;

E-mail: evlashkevich@hse.ru

ORCID: 0000-0002-3241-2291