Электронный научный журнал "Математическое моделирование, компьютерный и натурный эксперимент в естественных науках" http://mathmod.esrae.ru/
URL статьи: **mathmod.esrae.ru/50-216**
Ссылка для цитирования этой статьи:
Олифиренко А.А. Математическое моделирование выявления атак Data Poisoning на модели машинного обучения с применением алгоритма Local Outlier Factor // Математическое моделирование, компьютерный и натуральный эксперимент в естественных науках. 2025. №2

УДК 004.056.5 DOI:10.24412/2541-9269-2025-2-35-49

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ВЫЯВЛЕНИЯ АТАК DATA POISONING НА МОДЕЛИ МАШИННОГО ОБУЧЕНИЯ С ПРИМЕНЕНИЕМ АЛГОРИТМА LOCAL OUTLIER FACTOR

Олифиренко А.А.
Саратовский государственный технический университет имени Гагарина Ю.А.,
Россия, Саратов, artemolifirenko@yandex.ru

MATHEMATICAL MODELING FOR DETECTING DATA POISONING ATTACKS ON MACHINE LEARNING MODELS USING THE LOCAL OUTLIER FACTOR ALGORITM

Olifirenko A.A.
Yuri Gagarin State Technical University of Saratov, Russia,
Saratov, artemolifirenko@yandex.ru

Аннотация. В работе рассматривается проблема атак Data Poisoning на модели машинного обучения и предложен усовершенствованный подход к их выявлению на основе алгоритма Local Outlier Factor. Модификации включают адаптивное определение параметра contamination, медианное сглаживание шумов и использование расстояния Махаланобиса для учета корреляций признаков. Экспериментально подтверждена высокая эффективность предложенных методов в условиях целенаправленного внесения вредоносных данных.

Ключевые слова: атаки Data Poisoning, модели машинного обучения, алгоритм Local Outlier Factor, медианный фильтр, адаптивное contamination, расстояние Махаланобиса.

Abstract. The study addresses the issue of Data Poisoning attacks on machine learning models and proposes an improved approach for their detection using the Local Outlier Factor algorithm. The modifications include adaptive contamination parameter tuning, median noise filtering, and the use of Mahalanobis distance to account for feature correlations. Experimental results demonstrate the high effectiveness of the proposed methods under targeted data poisoning scenarios.

Keywords: Data Poisoning attacks, machine learning models, Local Outlier Factor algorithm, median filter, adaptive contamination, Mahalanobis distance.

Введение

Современные модели машинного обучения, широко применяемые в самых разных областях – от автономного транспорта и финансового сектора до анализа больших массивов данных в медицине, – нередко сталкиваются с проблемой так называемого «отравления» (Data Poisoning) [1], когда злоумышленник умышленно вносит в обучающую выборку искаженные или заведомо ложные примеры. Механизм подобной атаки заключается в том, что модель, обучающаяся в предположении, что все представленные ей данные достоверны, начинает подстраиваться под нетипичную, искусственно созданную информацию. В классическом понимании, если исходный набор содержит примеры, отражающие реальные закономерности, то после добавления отравляющих записей распределение признаков либо смещается, либо дополняется «шумовыми» точками, которые кажутся системе статистически правдоподобными, но на деле не имеют отношения к истинной задаче [2]. В результате итоговые параметры модели (θ) перестают соответствовать «чистому» решению, и качество ее предсказаний существенно падает на новых, не встречавшихся ранее данных [3]. Более того, если цель атакующего сводится не просто к снижению точности, а к изменению поведения модели для конкретных входов, то такая атака приобретает целенаправленный характер [4]: злоумышленник может добиться ошибочной классификации отдельного критически важного примера или группы примеров, используя минимальные и почти незаметные модификации.

Подобные действия особенно опасны в тех областях, где некорректная работа алгоритма способна привести к финансовым потерям, сбоям в работе инфраструктурных систем или угрозе безопасности [5]. К примеру, в распознавании уязвимостей кибербезопасности злоумышленник может добавлять в обучающий набор «легитимные» записи, которые на самом деле маскируют вредоносные паттерны, с тем чтобы впоследствии проходить проверку без детектирования. В системах фильтрации спама, если атакующие убедят модель считать множество спам-писем «безобидными», эффективность антиспам-фильтра стремительно падает, а пользователи подвергаются большему риску фишинговых атак. Похожие проблемы возникают и в компьютерном зрении: незначительное изменение набора пикселей на снимках дорожных знаков может привести к тому, что автономный автомобиль перестанет правильно распознавать ограничения скорости или приоритет движения, что чревато критическими последствиями для участников дорожного движения [6].

Актуальность противодействия этим атакам определяет особый интерес к методам обнаружения и фильтрации аномальных данных, способных вовремя распознать «подозрительные» примеры и предотвратить их негативное влияние на процесс обучения [7].

Алгоритм LOF и обоснование его модификации

Одним из таких методов является алгоритм Local Outlier Factor (LOF), который базируется на сравнении локальной плотности вокруг каждой точки с плотностью ее ближайших соседей [8]. Предполагается, что каждое наблюдение окружено рядом похожих (или статистически близких) точек, образующих локальную структуру данных; если же точка оказывается «слишком далекой» от своего окружения, ее можно считать потенциальным выбросом [5].

Одним из ключевых выражений, используемых для определения степеней аномальности, служит формула, задающая локальный коэффициент выброса:

$$LOFk(p) = \frac{1}{|N_k(p)|}\sum o \in N_k(p)\frac{lrd_k(o)}{lrd_k(p)}$$

$N_k(p)$ – множество k-ближайших соседей точки p, а $lrd_k(\cdot)$ – оценка локальной плотности (Local Reachability Density – далее LRD), отражающая инверсию среднего расстояния достижения между точкой и ее соседями с учетом структуры локального окружения.

Для количественной оценки вводится специальная величина $LOFk(p)$, где p – интересующий нас объект, а k – число ближайших соседей, чья локальная плотность сравнивается с плотностью точки p. При вычислении используется понятие расстояния достижения (reachability distance – мера удалённости между объектами, сглаженная с учётом плотности их окружения), а также локальной плотности (Local Reachability Density, LRD – обратная величина среднего расстояния достижения между точкой и её ближайшими соседями, отражающая плотность окружения) [3].

Формально алгоритм сводится к тому, что для каждого объекта мы вычисляем отношение между «локальными плотностями» соседей [2] и его собственной плотностью, а затем усредняем эти отношения. Если результат близок к единице, объект не воспринимается как аномальный; если же он существенно превышает единицу, точка считается выбросом.

Однако при всей привлекательности базовой идеи LOF остается уязвимым к ряду факторов, связанных с тем, что злоумышленник может целенаправленно добавлять в набор не просто редкие точки, но и шумовые примеры, которые сбивают оценку локальной плотности или затрудняют определение оптимального числа ближайших соседей. Более того, стандартный алгоритм часто предполагает, что исследователь заранее знает долю выбросов, чтобы в соответствии с этим порогом отсеивать аномальные объекты [8]. В условиях же атаки Data Poisoning достоверные сведения о масштабах «отравления» отсутствуют: злоумышленник может добавить минимальное число точек, но так искусно «вписать» их в структуру данных, что алгоритм с жестко зафиксированными параметрами не заметит ничего подозрительного. Также существенной проблемой оказывается предполагаемое использование евклидова расстояния при вычислении близости, ведь в реальных наборах признаков нередко встречаются корреляции, и вредоносные примеры могут быть

специально выбраны с учетом этих зависимостей, обманывая простой евклидов критерий.

Чтобы преодолеть перечисленные недостатки, в рамках предлагаемого подхода мы модифицируем LOF по нескольким ключевым направлениям. Во-первых, отказываемся от фиксированной доли выбросов и используем адаптивный порог, зависящий от квантильных характеристик самого распределения показателей $LOFk$. В частности, рассчитывается эмпирическое распределение значений локального коэффициента для всех точек обучающего набора, а затем в качестве границы аномальности берется, например, верхняя пятерка или один процент наблюдений, имеющих максимальный LOF. Такой динамический метод позволяет эффективно реагировать на изменения в данных: если в результате атаки внезапно появляются целые кластеры точек с сильно возросшим $LOFk$, адаптивная граница «сместится» и «захватит» их. Во-вторых, используются методы сглаживания, в частности медианная фильтрация локальных оценок, призванная подавить единичные всплески, не отражающие систематической проблемы, но могущие приводить к ложным тревогам. Таким образом, сохраняются существенные аномалии, а мелкие шумовые искажения сглаживаются, повышая точность детектирования именно вредоносных, а не случайных точек. Наконец, переход от евклидовой метрики к расстоянию Махаланобиса помогает учесть корреляции между признаками и выявлять неочевидные выбросы, расположенные в направлениях, где дисперсия (с учетом ковариации признаков) минимальна. Так реализуется более тонкое исследование многомерного пространства, что особенно важно, когда атакующий целенаправленно «подбирает» координаты добавляемых записей с учетом скрытых зависимостей между признаками.

В совокупности такие модификации позволяют добиться более надежной защиты от атак Data Poisoning. Адаптивная настройка порога сводит к минимуму риск пропустить новую партию вредоносных объектов, медианное сглаживание сокращает количество ложных срабатываний и удерживает уровень отказоустойчивости на приемлемом уровне, а учет корреляций в данных затрудняет маскировку атакующих примеров под статистически нормальные. Все это делает усовершенствованный алгоритм Local Outlier Factor гибким и эффективным инструментом для задачи фильтрации потенциально опасных записей, причем не только в лабораторных условиях, но и в реальных производственных системах, где набор данных может сильно меняться со временем, а угрозы со стороны злоумышленников остаются постоянным фактором риска.

Чтобы подтвердить эффективность предложенных улучшений и продемонстрировать их применимость в реальном сценарии, был проведен эксперимент, включающий этапы подготовки данных и моделирования возможной атаки Data Poisoning.

Математическое моделирование атаки «Data poisoning»

Был выбран датасет Credit Card Fraud, опубликованный на платформе Kaggle. Данный набор данных содержит анонимизированные записи транзакций по кредитным картам, среди которых небольшая часть относится к мошенническим операциям, а остальные являются нормальными. Поскольку основная задача сводится к бинарной классификации – определению, является ли транзакция мошеннической или нет, – рассмотренный датасет позволяет наглядно оценить, насколько «отравляющие» примеры могут влиять на итоговую способность модели обнаруживать аномалии и сохранять высокую точность предсказаний. В общей сложности в исходном наборе представлено тридцать признаков, часть которых содержит зашифрованные компоненты (из соображений анонимности), а два признака – «Время» (Time) и «Сумма» (Amount) – наиболее понятны и существенно влияют на интерпретацию подозрительности транзакции. Перед началом эксперимента эти и другие признаки были нормализованы (использовалась стандартизация), что позволило снизить влияние разницы в масштабах и единицах измерения, а также улучшило условия для корректного вычисления расстояний между точками. Вдобавок к этому был проведен первичный анализ распределения, призванный выявить «естественные» выбросы, не связанные с потенциальной атакой. Исключение заведомо некорректных данных, возникших по иным причинам (например, техническим ошибкам), необходимо для того, чтобы в дальнейшем приписывать обнаруженные аномалии именно атаке Data Poisoning, а не случайному шуму.

Чтобы сымитировать сценарий, в котором злоумышленник старается в массовом порядке дестабилизировать систему, не имея при этом детальной информации о каждом конкретном примере, была использована методика случайного инвертирования меток класса. Идея заключается в следующем: берется 20% транзакций из обучающей выборки, после чего их истинная метка «переворачивается». Если исходно транзакция считалась нормальной, ей присваивается статус мошеннической, и наоборот. Тем самым создается ложная статистическая картина, где часть действительных мошеннических операций выглядит «безобидной», а некоторые корректные транзакции подаются модели как подозрительные. Формально такую операцию можно представить в виде

$$y_{poisoned}(i) = 1 - y(i) \text{ для тех } i,$$

которые попали в «зону инверсии». Поскольку процесс выбора этих индексов происходит равновероятно среди всей обучающей выборки, атакующий как будто бы не располагает никакими сведениями о самом датасете и лишь хочет в целом ухудшить качество предсказаний. Данная методика позволяет воспроизвести реальный сценарий, в котором злоумышленник, обладая ограниченными ресурсами и знаниями, вносит шум сразу в большое количество записей, не заботясь о точечном воздействии.

После формирования «чистого» и «отравленного» наборов данных была обучена модель, выбранная в качестве базового классификатора: многослойный персептрон (Multi-Layer Perceptron, MLP – разновидность нейросети, состоящей из нескольких слоёв взаимосвязанных узлов-нейронов), имеющий три скрытых

слоя, содержащих последовательно 64, 32 и 16 нейронов. В качестве функции активации применялась ReLU (Rectified Linear Unit – функция, пропускающая положительные значения без изменений и обнуляющая отрицательные, что ускоряет обучение и предотвращает насыщение градиентов).

Эта архитектура демонстрирует достаточную гибкость, чтобы улавливать сложные взаимосвязи между признаками, но в то же время её вычислительная сложность не чрезмерна, позволяя быстро производить многократные прогоны эксперимента. Обучение модели осуществлялось с помощью оптимизатора Adam (адаптивный алгоритм градиентного спуска, эффективно подбирающий скорость обучения для каждого параметра), а в качестве функции потерь была выбрана бинарная кроссэнтропия (Binary Cross-Entropy – мера ошибки для задач бинарной классификации, рассчитывающая расхождение между предсказанными вероятностями и реальными метками классов).

Для удобства оценки результатов обучение модели проводилось в течение 20 эпох (эпоха – полный проход по всему обучающему набору данных), при этом данные подавались в модель небольшими порциями по 32 объекта (батчами, то есть мини-наборами данных, обрабатываемыми за одну итерацию).

После каждой эпохи фиксировались ключевые метрики: точность (accuracy – доля правильно классифицированных примеров среди всех), точность положительного класса (precision – доля правильно выявленных мошеннических транзакций среди всех предсказанных как мошеннические), полнота (recall – доля правильно распознанных мошеннических операций среди всех реально имевших место), а также F1-мера (F1-score – гармоническое среднее между precision и recall, отражающее баланс между ними).

Поскольку задача сводилась к обнаружению мошеннических операций, особый интерес представляли именно показатели precision и recall, которые отражают, насколько верно модель выделяет действительно опасные транзакции и не пропускает ли реальные угрозы.

Оценка итоговой производительности показала, что при обучении на «чистом» наборе данных модель успешно справлялась с задачей выявления мошенничества, демонстрируя высокие значения метрик и сбалансированную F1-меру. В частности, доля верно обнаруженных мошеннических примеров (recall) оставалась на достаточно высоком уровне, а число ложных срабатываний (false positives – обычных транзакций, ошибочно распознанных как мошеннические) было относительно невелико.

Однако в сценарии, когда в обучающую выборку были внесены отравляющие метки, наблюдалось значительное падение recall. Это указывает на то, что часть настоящих мошеннических транзакций перестала распознаваться моделью и слилась с нормальными. Подобная деградация объясняется тем, что «перевёрнутые» метки (нормальные записи, отмеченные как мошеннические, и наоборот) искажали общую структуру данных, сбивая обучающий алгоритм с корректного распознавания. Формула для вычисления recall при двухклассовой классификации, как известно, задается соотношением

$$Recall = \frac{TP}{TP+FN},$$

где TP – количество верно классифицированных мошеннических транзакций, а FN – число мошеннических операций, ошибочно принятых моделью за нормальные. Снижение этой величины указывает на уязвимость системы к тому типу атак, при котором даже случайное вмешательство в метки способно в достаточной мере исказить статистическую структуру выборки. В результате часть действительных случаев мошенничества «теряется» моделью, а значит, общий уровень безопасности снижается. Итоги эксперимента подтверждают тезис о том, что Data Poisoning существенно влияет на способность модели правильно классифицировать аномалии и свидетельствуют в пользу использования дополнительных механизмов фильтрации и обнаружения вредоносных примеров. В частности, предложенные ранее расширения алгоритма Local Outlier Factor позволяют проводить детектирование подобных отравляющих точек, даже если злоумышленник генерирует их случайным образом, и сохранять высокий уровень чувствительности к истинным аномалиям.

Учитывая значительное влияние атак Data Poisoning, продемонстрированное на этапе экспериментов с датасетом кредитных транзакций, была поставлена задача детектировать вредоносные записи до момента обучения модели. Для этой цели использовался усовершенствованный алгоритм LOF, характеризующийся такими особенностями, как автоматическая подстройка параметра, медианная фильтрация шума и переход к расстоянию Махаланобиса. Подобная модификация призвана повысить надежность идентификации аномальных точек даже в условиях, когда атакующий добавляет значительное количество «отравляющих» данных или вносит систематический шум.

Прежде чем применить алгоритм, все исходные признаки были нормализованы для выравнивания масштаба (стандартизация по среднему значению и стандартному отклонению, то есть Z-преобразование), а затем к ним последовательно применялась процедура PCA (Principal Component Analysis – метод главных компонент, позволяющий уменьшить размерность признакового пространства с сохранением наибольшей дисперсии), позволившая сократить исходное многомерное пространство до десяти главных компонент. Такой подход не только упрощает вычисления, но и снижает риск «маскировки» выбросов в высокомерных пространствах, куда злоумышленник может целенаправленно помещать искусственно измененные объекты. Одновременно с этим была реализована динамическая адаптация параметра contamination (ожидаемая доля выбросов, используемая в алгоритме LOF для отделения аномалий от нормальных точек), обычно отвечающего за долю предполагаемых аномалий в данных. Вместо его фиксированного задания за основу бралась статистическая оценка распределения значений LOF: вычислялась эмпирическая плотность для набора оценок и определялась такая граница, при которой

оставшиеся «хвостовые» наблюдения можно считать выбросами с высокой вероятностью. Подобная стратегия эффективна в сценариях, когда доля отравленных примеров заранее неизвестна, а их размещение в пространстве признаков может варьироваться по мере эскалации атаки.

Обнаружение аномалий с использованием улучшенного LOF

Алгоритм Local Outlier Factor в своей исходной версии предназначен для оценки степени аномальности объектов на основе сравнения локальной плотности рассматриваемой точки и ее ближайших соседей. Однако стандартная реализация LOF существенно зависит от выбора числа ближайших соседей k и параметра $contamination$, задающего ожидаемую долю выбросов. Именно поэтому в ходе данного исследования была предложена серия усовершенствований, позволяющих повысить робастность и точность обнаружения вредоносных записей.

В качестве первого шага была проведена предобработка данных, включающая нормализацию признаков с помощью класса StandardScaler и снижение размерности методом главных компонент (PCA). Подобный подход позволяет привести все признаки к одному масштабу и избежать чрезмерного доминирования отдельных компонент. Формально нормализацию можно представить в виде:

$$X_{scaled} = \frac{X - \mu}{\sigma}$$

X – исходные данные, μ – среднее значение признака, σ – его стандартное отклонение.

После этого к результату нормализации применялся PCA, позволяющий выделить наиболее информативные направления в многомерном пространстве и тем самым сузить задачу до ограниченного набора компонент. Такой шаг, помимо повышения вычислительной эффективности, снижает чувствительность алгоритма к шумовым измерениям.

Следующим нововведением стала автоматическая адаптация параметра $contamination$. В обычных условиях этот параметр задается вручную, что может приводить к ошибочному определению аномалий, особенно если атакующий добавляет в набор непредсказуемое число вредоносных точек. В рассматриваемом подходе, напротив, граница между нормальными примерами и выбросами находилась путем анализа плотности распределения данных, что позволило корректно выделять «естественные» выбросы и при этом выявлять потенциальные аномалии, вызванные атакой. При этом устранялась необходимость заранее указывать процент аномальных наблюдений, что особенно критично для сценариев с неизвестной степенью «загрязненности» набора.

Для уменьшения количества ложных тревог, связанных с шумовыми всплесками, применялся медианный фильтр с окном шириной три наблюдения. Сглаживая единичные экстремальные значения, этот фильтр сохранял

структурные аномалии, более характерные для вредоносных воздействий. Такая процедура обеспечила дополнительный контроль над «шумом», который нередко сознательно добавляется злоумышленниками в отравляемые данные в попытке замаскировать атакующие записи.

Наконец, в качестве ключевого изменения была предложена замена евклидовой метрики на расстояние Махаланобиса:

$$d_M(x) = \sqrt{(x-\mu)^T \Sigma^{-1}}(x-\mu)$$

μ – вектор средних значений признаков, а Σ^{-1} – обратная ковариационная матрица.

Благодаря этому все потенциальные корреляции между признаками (нередко задействуемые атакующими) принимались во внимание, повышая точность определения уровня «странности» каждой точки. В сочетании с динамической адаптацией параметра $contamination$ и медианной фильтрацией такая метрика сводит к минимуму риск того, что вредоносные объекты будут нераспознаны алгоритмом LOF из-за взаимозависимостей в исходных данных.

В упрощенном виде итоговую схему алгоритма можно сформулировать следующим образом. Для точки p сначала определяется локальная плотность (LRD), обратно пропорциональная среднему расстоянию достижения p до ее k-ближайших соседей, где расстояния рассчитываются по метрике Махаланобиса. Далее итоговый коэффициент выброса $LOF(p)$ вычисляется путем усреднения отношений

$$\frac{lrd(o)}{lrd(p)}$$

o принадлежит множеству $N_k(p)$. Формально это выражается формулой

$$LOF(p) = \frac{1}{|N_k(p)|} \sum_{o \in N_k(p)} \frac{lrd(o)}{lrd(p)}$$

После того как все точки получают свою LOF-оценку, производится медианное сглаживание, нивелирующее случайные пики. На финальном шаге по распределению $LOF(p)$ эмпирически определяется порог, разделяющий нормальные и аномальные объекты. Благодаря автоматической адаптации $contamination$ этот порог «подстраивается» к реалиям текущего набора данных, не требуя заранее знать точную долю отравляющих примеров.

Для визуализации полученных результатов и проверки корректности работы предложенных улучшений были построены два графика, наглядно иллюстрирующие распределение значений LOF и расположение аномальных точек в пространстве ключевых признаков. На первом графике (рис. 1) показано, как распределяются оценки LOF среди всех объектов выборки.

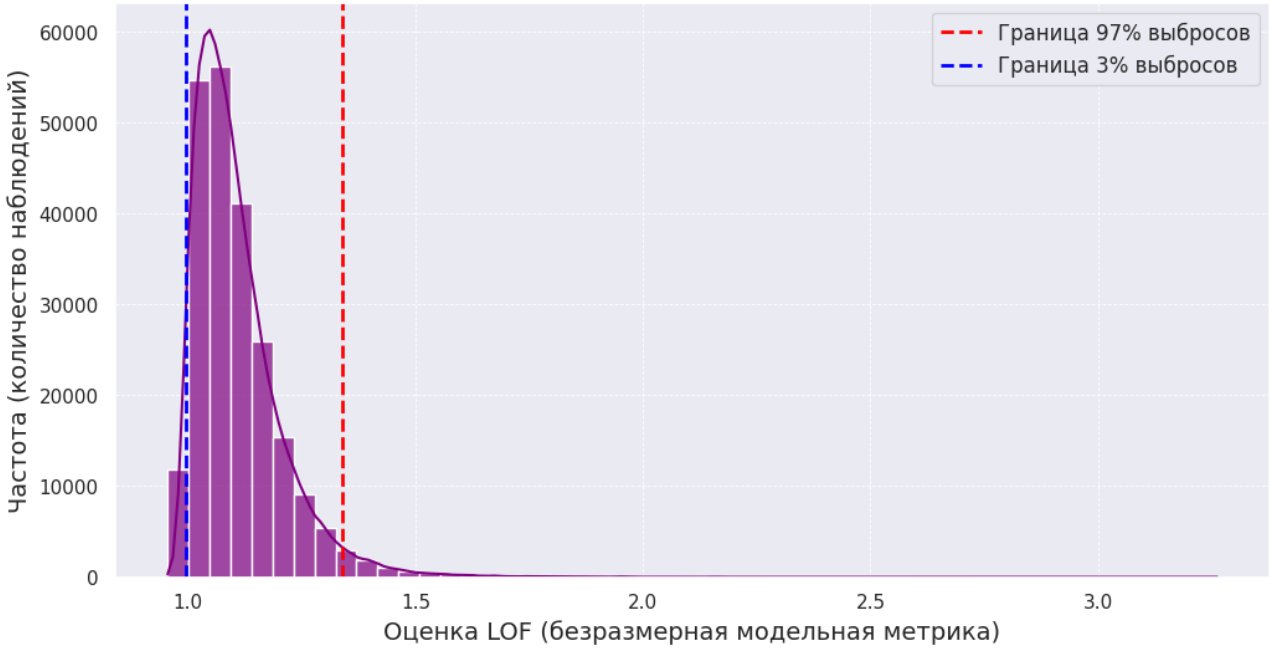

Рис. 1. Распределение LOF-оценок с границами выбросов

По оси X отложены значения оценки LOF (безразмерная модельная метрика), по оси Y – частота (количество наблюдений).

В качестве ориентиров выделены две вертикальные линии, соответствующие 3-му и 97-му процентилям распределения – они служат эмпирическими границами между «характерными» точками и выбросами. Значения, превышающие верхнюю границу, трактуются как аномальные, а значения ниже нижней границы формируют зону эталонной плотности. Расслоение, наблюдаемое на гистограмме, подтверждает, что алгоритм LOF с корректно адаптированным параметром contamination способен эффективно выделять подозрительные записи даже при отсутствии априорной информации о числе выбросов.

На втором графике (рис. 2), представляющем двумерное распределение точек, показано положение транзакций в пространстве двух стандартизованных признаков: «Время» и «Сумма» транзакции. Оба признака были предварительно нормализованы с помощью Z-преобразования (стандартизация), поэтому значения отображаются в отклонениях от среднего (Z-оценка) и не имеют физических единиц измерения.

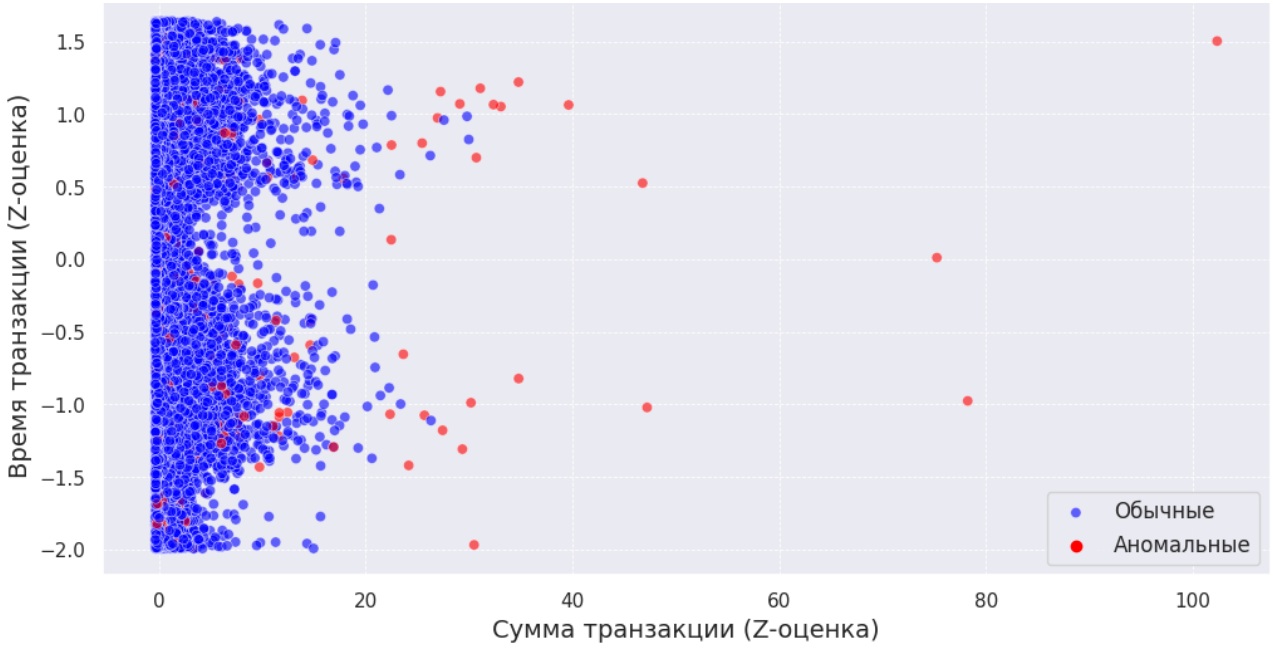

Рис. 2. Детализированное выявление аномалий с LOF

По оси X отложена сумма транзакции (Z-оценка), по оси Y – время транзакции (Z-оценка).

Синим цветом показаны транзакции, отнесенные алгоритмом к нормальным, красным – потенциально аномальные. Даже несмотря на внешнюю схожесть с нормальными объектами, вредоносные записи формируют отдельные кластеры или одиночные выбросы, локализованные в областях, отличающихся по многомерной плотности. Учет структуры ковариаций (например, через метрику Махаланобиса) и медианная фильтрация позволяют надежно выявлять такие отклонения даже в условиях присутствия маскирующего шума.

Обобщая результаты, можно отметить, что адаптивный выбор $contamination$, базирующийся на эмпирическом распределении значений LOF, позволяет подстраивать границы аномальности исходя из реальной структуры данных и не требует заранее знать, сколько именно отравляющих примеров было добавлено злоумышленником. Фильтрация шумов посредством медианы, в свою очередь, сокращает ложные срабатывания на «единичные выбросы», которые могут возникать из-за банальных технических ошибок или случайных пиков, но при этом сохраняет общую область действительно подозрительных объектов. Что касается расстояния Махаланобиса, то его включение в расчет LOF особенно актуально в тех случаях, когда злоумышленник намеренно эксплуатирует коррелированные признаки для маскировки вредоносных примеров. Согласно наблюдениям, без учета ковариаций многие из таких объектов могли остаться «незамеченными»; наоборот, при использовании Махаланобиса они приобретают высокое значение LOF, выделяясь из остальной выборки.

Результаты

В ходе проведенного исследования удалось установить, что вмешательство

посредством атак Data Poisoning приводит к заметному ухудшению способности модели машинного обучения корректно классифицировать мошеннические операции. Наибольшие изменения затронули такие показатели, как recall и F1-score, которые прямо характеризуют способность алгоритма улавливать реальные аномалии. Из экспериментов следует (таблица 1), что при обучении на «чистых» данных значение recall для класса мошеннических транзакций (класс 1) достигало 0.71, а F1-score – 0.78. Однако после внесения 20% отравляющих примеров эти метрики снизились до 0.54 и 0.66 соответственно, что свидетельствует о возросшем числе пропусков реальных угроз. В то же время показатель precision практически не изменился, сохраняя относительно высокое значение. Такая динамика говорит о том, что модель не стала чаще ошибаться в определении несуществующих мошеннических операций (то есть не увеличила количество ложных тревог), однако существенно снизила чувствительность к настоящим аномалиям. Это подтверждает предположение, что даже при небольшой доле вредоносных записей в обучающем наборе система теряет способность своевременно выявлять подозрительные транзакции. Следовательно, подтверждается негативное влияние атак Data Poisoning и необходимость в дополнительных механизмах для сохранения высокого уровня обнаружения мошеннических операций.

Метрика	Чистые данные	Отравленные
Precision 0	1.00	1.00
Precision 1	0.85	0.85
Recall 0	1.00	1.00
Recall 1	0.71	0.54
F1-score 1	0.78	0.68
Accuracy	1.00	1.00
Macro avg (recall)	0.86	0.77
Weighted avg	1.00	1.00

Сравнение производительности моделей Таблица 1

Для решения задачи фильтрации вредоносных записей и снижения негативных последствий от их присутствия в данных был предложен адаптированный алгоритм Local Outlier Factor, использующий несколько важных усовершенствований. Во-первых, параметр *contamination* (в классическом LOF – доля точек, которые предполагаются выбросами), традиционно задаваемый вручную, в предлагаемом решении настраивался динамически, исходя из плотностных характеристик распределения. Благодаря этому оказалось возможным автоматически определять долю выбросов, подстраиваясь под конкретный набор данных и не полагаясь на фиксированное априорное предположение. Во-вторых, для снижения уровня ложных тревог использовался медианный фильтр, подавляющий шумовые всплески и тем самым позволяющий

алгоритму сосредоточиться именно на систематических отклонениях, возникающих вследствие атак. В-третьих, учитывалась корреляционная структура признаков путем применения расстояния Махаланобиса, что позволило выявлять аномалии в ситуациях, когда один и тот же объект может казаться «типичным» в евклидовом пространстве, но заметно выбивается при рассмотрении взаимозависимостей между признаками. На соответствующих визуализациях (график распределения LOF-оценок и двумерный график с аномалиями, выделенными красным цветом) хорошо видно, что обновленный алгоритм LOF способен достаточно четко проводить границу между нормальными и подозрительными точками. При пороге в 97% он фиксирует крайние случаи, в которых локальная плотность заметно ниже, чем у основных групп, а при нижней границе в 3% формируется эталонная зона для «консервативно» нормальных записей. На графике, где данные представлены по двум ключевым признакам (времени и сумме транзакции), предварительно стандартизированным через Z-преобразование (преобразование значений в отклонения от среднего, делённые на стандартное отклонение, что устраняет масштабные различия между признаками), точки, признанные аномальными, оказываются отчетливо отделены от основной массы, что не только подтверждает высокую точность работы алгоритма в части отделения настоящих вредоносных объектов, но и свидетельствует о невысоком уровне ложных срабатываний. Следовательно, даже при существенном объеме инъекции отравляющих примеров адаптированный LOF сохраняет способность выявлять отклонения от реальной структуры данных.

Вместе с тем необходимо учитывать некоторые ограничения полученных результатов. Одно из ключевых замечаний касается зависимости итогового качества детектирования выбросов от исходного распределения признаков и свойств медианного фильтра, выбранного для подавления шума. Хотя автоматическая настройка параметра contamination эффективно адаптируется к имеющимся данным, ее корректность в определенной мере опирается на форму и статистические характеристики выборки. Кроме того, эксперименты проводились на статическом наборе финансовых транзакций, тогда как во многих реальных приложениях данные поступают потоком, и поведение системы может меняться с течением времени. Для подтверждения универсальности полученных выводов требуется более масштабное тестирование на других реальных наборах данных, включая случаи с более изощренными или специально адаптивными атаками. Тем не менее даже в рамках описанных ограничений результирующая методика демонстрирует значительный прирост устойчивости к Data Poisoning и подтверждает свою практическую ценность.

Заключение

В ходе проведенной работы удалось показать, что атаки Data Poisoning серьезно подрывают способность моделей машинного обучения выявлять

мошеннические транзакции и иные аномалии. Сравнение ключевых метрик на «чистых» и «отравленных» данных наглядно продемонстрировало, как резко может упасть показатель recall и F1-score, что в условиях финансовой сферы или кибербезопасности оборачивается дополнительными рисками. Разработанный и апробированный в рамках данного исследования улучшенный алгоритм LOF (с динамической настройкой параметра contamination, медианным фильтром и метрикой Махаланобиса) показал высокую эффективность в фильтрации вредоносных точек, сохранив точность и чувствительность к аномалиям даже при существенной доле отравляющих примеров. Практическая ценность предложенного метода подтверждается и тем, что внесенные дополнения не требуют значительных вычислительных ресурсов, поскольку могут быть интегрированы в стандартные процедуры предобработки (такие как нормализация и PCA) и несложные модификации расчета LOF.

Дальнейшие перспективы исследований включают в себя адаптацию предложенного подхода к потоковым данным, где статистика признаков и их взаимосвязи могут меняться во времени, а также тестирование описанного решения на более разнообразных наборах данных, в том числе отражающих различные варианты реальных атак. Не менее важно оценить эффективность подобных мер в смежных областях, например, при анализе сетевого трафика, медицинских данных или систем идентификации пользователей, где надежность обнаружения аномалий играет критическую роль. Предварительные результаты позволяют заключить, что комплексный учет структурных особенностей данных, адаптивное определение уровня выбросов и фильтрация шумов способны обеспечить алгоритмам машинного обучения дополнительную защиту от злонамеренных вмешательств и, тем самым, повысить общий уровень безопасности критически важных приложений.

Литература

1. Намиот Д. Е. Введение в атаки отравлением на модели машинного обучения // International Journal of Open Information Technologies. 2023. №3. URL: https://cyberleninka.ru/article/n/vvedenie-v-ataki-otravleniem-na-modeli-mashinnogo-obucheniya (дата обращения: 30.01.2025).
2. Сивков Д. И., Федосенко М. Ю. Атаки и методы защиты при использовании методов машинного обучения в контексте стегоанализа цифрового контента // Экономика и качество систем связи. 2024. №3 (33). URL: https://cyberleninka.ru/article/n/ataki-i-metody-zaschity-pri-ispolzovanii-metodov-mashinnogo-obucheniya-v-kontekste-stegoanaliza-tsifrovogo-kontenta (дата обращения: 30.01.2025).
3. Гололобов Н. В. Защита от атаки отравления данных на основе нейронной сети с долгой краткосрочной памятью: выпускная квалификационная работа магистра / Н. В. Гололобов; науч. рук. Е. Ю. Павленко; Санкт-Петербургский политехнический университет Петра Великого. – Санкт-Петербург, 2023.

URL: https://elib.spbstu.ru/dl/3/2023/vr/vr23-2911.pdf/en/info (дата обращения: 30.01.2025).
4. He P., Xu H., Ren J., Cui Y., Liu H., Aggarwal C. C., Tang J. Sharpness-Aware Data Poisoning Attack // ArXiv, abs/2305.14851, 2023. URL: https://arxiv.org/abs/2305.14851 (дата обращения: 30.01.2025).
5. Alghushairy O., Alsini R., Soule T., Ma X. A Review of Local Outlier Factor Algorithms for Outlier Detection in Big Data Streams // Big Data Cogn. Comput., 5, 1, 2020. URL: https://www.semanticscholar.org/paper/A-Review-of-Local-Outlier-Factor-Algorithms-for-in-Alghushairy-Alsini/e449b9b3fe04fe260731a3c74d2123bf6eaadf5b (дата обращения: 30.01.2025).
6. Попова И. А. Обнаружение аномалий в наборе данных с помощью алгоритмов машинного обучения без учителя Isolation Forest и Local Outlier Factor // StudNet. 2020. №12. URL: https://cyberleninka.ru/article/n/obnaruzhenie-anomaliy-v-nabore-dannyh-s-pomoschyu-algoritmov-mashinnogo-obucheniya-bez-uchitelya-isolation-forest-i-local-outlier (дата обращения: 30.01.2025).
7. Xu Z., Kakde D., Chaudhuri A. Automatic Hyperparameter Tuning Method for Local Outlier Factor, with Applications to Anomaly Detection // IEEE International Conference on Big Data, 2019. С. 4201-4207. DOI: 10.1109/BigData47090.2019.9006151. URL: https://www.researchgate.net/publication/339482662_Automatic_Hyperparameter_Tuning_Method_for_Local_Outlier_Factor_with_Applications_to_Anomaly_Detection (дата обращения: 30.01.2025).
8. Babaei K., Chen Z., Maul T. H. Detecting Point Outliers Using Prune-based Outlier Factor (PLOF) // ArXiv, abs/1911.01654, 2019. URL: https://www.semanticscholar.org/paper/Detecting-Point-Outliers-Using-Prune-based-Outlier-Babaei-Chen/2d2c30fe39362d0e94a09906f921cbf92f34b2e9 (дата обращения: 30.01.2025).