УДК 519.7

Ю.Ю. Дюличева
Таврический национальный университет им. В.И. Вернадского,
г. Симферополь, Украина
dyulicheva@mail.ru

О задачах фильтрации обучающих данных

В статье приведен краткий обзор современных подходов к выявлению выбросов в обучающих данных; вводятся строгие понятия чистого и мажоритарного выбросов относительно модели алгоритмов обучения; установлено существование моделей алгоритмов обучения и обучающих выборок, относительно которых множество чистых выбросов непусто; доказано необходимое и достаточное условие существования пустого множества чистых выбросов, связанное с ёмкостью модели алгоритмов обучения.

Введение

Разработка фильтров – процедур, позволяющих выделять релевантную информацию в имеющихся начальных данных, является одной из центральных задач теории обучаемых систем (Machine Learning). Открытым остается вопрос о том, какой признак или объект таблицы обучения действительно следует называть релевантным.

Задача построения фильтров – это вызывающе сложная задача. Перечислим наиболее важные вопросы, которые возникают перед исследователем при построении фильтров [1-8]:

- как отделить шум от объектов-исключений, требующих отдельной интерпретации и характеризующих некоторую особенность класса, и как определить объект-исключение в строгом смысле?
- удаление шума может и не привести к улучшению качества обучения, например, в тех случаях, когда один и тот же шум содержится и в обучающей и в контрольной выборке. В этом случае выявление и удаление шума из обучающей выборки приводит к резкому ухудшению обобщающей способности алгоритма обучения на объектах контрольной выборки. Как выявлять и учитывать такой шум?
- как учитывать несогласованность различных моделей обучающих алгоритмов в отнесении объектов к выбросам или шумам? Например, одни и те же обучающие данные одной моделью алгоритмов обучения могут классифицироваться как шум, а другой – как ценная обучающая информация. Как учитывать такую несогласованность алгоритмов обучения в рамках одной и той же модели?
- как выявлять шум, имеющий систематический характер появления в обучающей выборке?
- как разработать фильтры, выявляющие одновременно и релевантные признаки, и релевантные объекты?

– как разработать фильтры, учитывающие природу содержащейся в обучающей выборке информации, например, характер поведения целевой зависимости и объектов в обучающей выборке до построения алгоритма обучения? Решение этого вопроса позволило бы согласовывать выбор модели алгоритмов обучения с характером имеющейся начальной информации, поскольку зачастую выбор модели алгоритмов обучения – это субъективный выбор исследователя, не имеющий никакого обоснования и направленный на привлечение дополнительной (недостающей) информации.
– следует ли удалять выбросы, выявленные фильтром, или лучше снабдить фильтр корректором выбросов (такой вариант целесообразно рассматривать в случаях с обучающими выборками малого размера)?

Целью данной работы является поиск подходов к формализации и строгой постановке задач, направленных на решение перечисленных выше вопросов, связанных с фильтрацией обучающих данных.

Первый раздел статьи содержит краткий обзор современных подходов к построению фильтров, выявляющих выбросы в обучающей выборке. Во втором разделе вводятся строгие понятия чистого и мажоритарного выбросов относительно модели алгоритмов обучения; устанавливается существование моделей алгоритмов обучения и обучающих выборок, относительно которых множество чистых выбросов непусто; доказывается необходимое и достаточное условие существования пустого множества чистых выбросов, связанное с ёмкостью модели алгоритмов обучения; на основании ранее проведенных автором исследований [9], предлагается модификация алгоритма синтеза эмпирического решающего леса для выявления чистых выбросов. В заключении подводятся итоги и указываются основные направления дальнейших исследований.

Современные подходы к выявлению выбросов в обучающих данных

В статистике и регрессионном анализе «выброс» определяется [10] как «объект, который не удовлетворяет той же самой модели, что и все остальные объекты». К сожалению, такое интуитивное определение справедливо и для зашумленных данных и для данных-исключений, характеризующих особенности исследуемого класса. В задачах классификации с учетом гипотезы компактности обучающий объект 1-го класса относят к выбросам, если он находится внутри «плотной» области (кластера) объектов 2-го класса. Обнаружение выбросов в выборках малых размеров, содержащих номинальные, неупорядоченные признаки, не является задачей статистики и регрессионного анализа, однако представляет интерес в задачах обучения распознаванию.

В зарубежной литературе, как правило, *фильтром* (filter) называют алгоритм предварительной обработки обучающих данных, которые содержат шум или случайные ошибки в метках принадлежности обучающих объектов к классам.

Несомненно, центральной проблемой теории обучаемых систем является проблема разработки методов предотвращения переподгонки на обучающих данных. Трудность заключается в том, что не существует универсальных методов предотвращения переподгонки также, как и не существует универсальных моделей алгоритмов и методов обучения, применимых для решения любых задач классификации [11]. Одни методы направлены на предотвращение переподгонки в

процессе обучения, другие – на фильтрацию обучающих данных *до обучения* и выполнение обучения на редуцированной обучающей выборке, полученной из исходной после удаления выбросов.

Современные подходы к построению фильтров можно разделить на два класса:

- подходы, которые используют один и тот же обучающий алгоритм и для фильтрации обучающих данных, и для обучения;
- подходы, использующие совокупность обучающих алгоритмов для фильтрации обучающих данных и один алгоритм, не участвовавший в фильтрации, для обучения.

Подходы, основанные на использовании одного обучающего алгоритма, состоят из двух этапов. На первом этапе обучающий алгоритм используется для выявления и удаления выбросов из обучающей выборки; на втором этапе этот же самый алгоритм используется для построения классификатора по редуцированной обучающей выборке. В качестве примера, использующего такой подход, можно отметить работу Джона [4], в которой строятся устойчивые (робастные) решающие деревья (robust decision tree). Процесс редукции решающих деревьев (РД) направлен на предотвращение настройки решающего дерева на выбросы, т.е. на предотвращение переподгонки на обучающих данных (overfitting avoidance). Джон предложил RobustC4.5 алгоритм, направленный на удаление тех объектов из обучающей выборки, которые неверно классифицируются редуцированным РД (построенным с использованием алгоритма C4.5), после чего происходит перестройка РД по редуцированной обучающей выборке. По сути дела, при таком подходе принимается гипотеза о том, что объекты, отнесенные к выбросам на некотором «узком» подмножестве обучающего множества, являются выбросами и для всей обучающей выборки. Алгоритм RobustC4.5 будет продолжать редукцию и перестройку решающего дерева до тех пор, пока не будет построено корректное на редуцированной обучающей выборке РД. Такой подход устойчивой (робастной) настройки (robust or resistant fitting) является общепринятым в регрессионном анализе. Поскольку до построения алгоритма обучения нельзя сказать, содержит ли обучающая выборка шум, то, очевидно, алгоритм RobustC4.5 будет плохо обрабатывать сложно устроенную начальную информацию, не содержащую шумов.

Известны подходы, использующие два обучающих алгоритма один из которых работает как фильтр, выявляя и удаляя выбросы; другой – как обычный алгоритм обучения [2]. Такой подход предшествовал появлению подходов, в которых фильтры основаны на построении совокупности алгоритмов обучения.

Подходы, основанные на голосовании совокупности обучающих алгоритмов, также состоят из двух этапов. На первом этапе по обучающей выборке (например, с помощью скользящего контроля) строятся m алгоритмов обучения. Для фильтрации обучающих данных используются $m-1$ алгоритмов обучения. Каждый объект из обучающего множества получает $m-1$ меток классов, «вычисленных» с помощью $m-1$ алгоритмов обучения. Для каждого обучающего объекта фильтр сравнивает $m-1$ меток классов (например, с помощью процедуры голосования) и выявляет выбросы; алгоритм, не участвовавший в фильтрации обучающих данных, используется как обычный алгоритм обучения.

В рамках этого подхода известны многочисленные эвристики, отличающиеся по числу алгоритмов, входящих в фильтр, по способу построения классификаторов, а также по методу отнесения обучающих объектов к выбросам. Например, в работе [2]

на части обучающей выборки строится совокупность классификаторов, а на оставшейся части обучающей выборки с помощью процедур голосования (мажоритарный фильтр (majority filter) или фильтр согласия (consensus filter)) принимается решение об отнесении объектов к выбросам. В работе [3] были проведены исследования, направленные на построение фильтров на основе совокупности алгоритмов обучения, и предложены следующие виды фильтрации: фильтрация, основанная на обычном (невзвешенном) голосовании алгоритмов обучения, построенных на различных подмножествах обучающей выборки с помощью либо скользящего контроля (*фильтр скользящего контроля*), либо баггинга (*фильтр баггинга*); фильтрация, основанная на удалении тех объектов обучающей выборки, которые получают наибольший вес в процессе бустинга (*фильтр бустинга*). Кроме того, в работе [12] экспериментально показано, что баггинг устойчив к зашумленным обучающим данным, а бустинг плохо обрабатывает зашумленные данные, поскольку присваивает неверно помеченным объектам большой вес. Это наблюдение послужило мотивацией использования процедуры бустинга для фильтрации обучающих данных. Известны также различные комбинации перечисленных фильтров [2], [3], [12]: фильтр согласия + фильтр скользящего контроля, мажоритарный фильтр + фильтр скользящего контроля, фильтр согласия + фильтр баггинга, мажоритарный фильтр + фильтр баггинга.

Понятие чистого и мажоритарного выброса относительно модели алгоритмов обучения

Пусть X^λ – обучающая выборка, $y^*: X \to Y$ – неизвестная целевая зависимость, X – множество всех допустимых объектов, Y – множество ответов – меток классов. Известны значения целевой зависимости y^* на обучающей выборке, т.е. $y_i = y^*(x_i)$ для любого $i = 1,2,\text{K}\,,\lambda$.

Определение 1 [11]. ***Моделью A алгоритмов обучения*** будем называть параметрическое семейство отображений $A = \{a(\tilde{x},\gamma) \mid a: X \times \Gamma \to Y\}$, из которого выбирается искомый алгоритм обучения (Γ – множество допустимых значений параметра γ, определяющего вид модели).

Определение 2. Объект $\tilde{x} \in X^\lambda$ называется ***чистым выбросом относительно модели A алгоритмов обучения***, если $L\left(a(\tilde{x}), y^*(\tilde{x})\right) = 1$ для любого алгоритма $a \in A$, где $L\left(a(\tilde{x}), y^*(\tilde{x})\right) = \left[a(\tilde{x}) \neq y^*(\tilde{x})\right]$ – функция потерь от ошибки при классификации объекта $\tilde{x} \in X^\lambda$ алгоритмом $a \in A$.

Обозначим $CLO\left(A, X^\lambda\right)$ – множество чистых выбросов относительно модели A алгоритмов обучения и заданной обучающей выборки X^λ. Очевидно следующее:

Утверждение 1. *Существует модель A алгоритмов обучения и обучающая выборка X^λ, $\lambda \geq 2$, относительно которых множество чистых выбросов $CLO\left(A, X^\lambda\right)$ непусто.*

Теорема. *Множество чистых выбросов* $CLO\left(A, X^{\lambda}\right)$ *относительно модели* A *алгоритмов обучения и обучающей выборки* X^{λ}, *пусто тогда и только тогда, когда* $VCD(A) = \infty$.

Доказательство. *Достаточность*. Пусть $VCD(A) = \infty$; предположим, что при этом множество чистых выбросов $CLO\left(A, X^{\lambda}\right) \neq \varnothing$, тогда найдется хотя бы один объект $\tilde{x}$ из заданной обучающей выборки X^{λ} такой, что $a(\tilde{x}) \neq y^{*}(\tilde{x})$ для любого алгоритма обучения $a \in A$, т.е. не все объекты из обучающей выборки X^{λ} (для любого λ) с помощью алгоритмов обучения из модели A могут быть разбиты на два класса всеми возможными способами. Следовательно, $VCD(A) < \infty$. Пришли к противоречию с условием.

Необходимость. Пусть множество чистых выбросов $CLO\left(A, X^{\lambda}\right) = \varnothing$ относительно модели A алгоритмов обучения и обучающей выборки X^{λ}; предположим, что при этом $VCD(A) < \infty$. Условие $VCD(A) < \infty$ означает [9], что для любого λ найдутся такие h объектов из обучающей выборки, которые с помощью алгоритмов обучения из модели A можно разбить на два класса всеми возможными способами, но никакие $h+1$ объектов из обучающей выборки X^{λ} нельзя разбить на два класса всеми возможными способами с помощью алгоритмов из A. Следовательно, в модели A алгоритмов обучения обязательно найдется алгоритм a, не способный правильно отделить $h+1$ объектов из обучающей выборки X^{λ} для любого λ. Это означает, что $CLO\left(A, X^{\lambda}\right) \neq \varnothing$. Установлено противоречие с условием.

Определение 3. Объект $\tilde{x} \in X^{\lambda}$ называется ***абсолютным чистым выбросом относительно обучающей выборки*** X^{λ}, если $L\left(a(\tilde{x}), y^{*}(\tilde{x})\right) = 1$ для любого алгоритма $a \in A$ и любой модели A алгоритмов обучения, где $L\left(a(\tilde{x}), y^{*}(\tilde{x})\right) = \left[a(\tilde{x}) \neq y^{*}(\tilde{x})\right]$ – функция потерь от ошибки при классификации объекта $\tilde{x} \in X^{\lambda}$ алгоритмом $a \in A$.

Обозначим $ACLO\left(X^{\lambda}\right)$ – множество абсолютных чистых выбросов относительно заданной обучающей выборки X^{λ}.

Утверждение 2. *Множество абсолютных чистых выбросов* $ACLO\left(X^{\lambda}\right)$, $\lambda \geq 2$, *относительно любой обучающей выборки* X^{λ} *пусто*.

Это утверждение следует из того, что для любой обучающей выборки X^{λ} можно построить такую модель A, в которой найдется хотя бы один алгоритм обучения, корректный относительно обучающей выборки X^{λ}, т.е. поточечно «настроенный» на обучающую выборку.

Определение 4. Объект $\tilde{x} \in X^{\lambda}$ называется ***мажоритарным выбросом относительно модели*** A ***алгоритмов обучения***, если существуют такие $m > \left[|A|/2\right]$

алгоритмов $a \in A$, что для любого из них $L\left(a(\tilde{x}), y^*(\tilde{x})\right) = 1$, где $L\left(a(\tilde{x}), y^*(\tilde{x})\right) = \left[a(\tilde{x}) \neq y^*(\tilde{x})\right]$ – функция потерь от ошибки при классификации объекта $\tilde{x} \in X^\lambda$ алгоритмом $a \in A$.

Очевиден следующий алгоритм выявления чистых выбросов в обучающей выборке X^λ. Под областью компетентности $Comp\left(a, X^\lambda\right)$ алгоритма $a \in A$ относительно обучающей выборки X^λ будем понимать множество всех обучающих объектов, которые правильно классифицируются алгоритмом α, т.е. $Comp\left(a, X^\lambda\right) = \left\{\tilde{x} \in X^\lambda \mid a(\tilde{x}) = y^*(\tilde{x})\right\}$. Если $Comp\left(a, X^\lambda\right)$ – область компетентности алгоритма $a \in A$ относительно выборки X^λ, тогда $InComp\left(a, X^\lambda\right) = X^\lambda \setminus Comp\left(a, X^\lambda\right)$ – область некомпетентности алгоритма $a \in A$ относительно обучающей выборки X^λ.

Для каждого алгоритма $a \in A$ построим область некомпетентности $InComp\left(a, X^\lambda\right)$, тогда область некомпетентности модели A алгоритмов обучения относительно обучающей выборки X^λ определяется

$$InComp\left(A, X^\lambda\right) = \operatorname{I}_{a \in A} InComp\left(a, X^\lambda\right).$$

Непустое множество $InComp\left(A, X^\lambda\right)$ образует множество чистых выбросов относительно выбранной модели A алгоритмов обучения и обучающей выборки X^λ.

Могут быть предложены различные способы определения и построения областей некомпетентности алгоритма $a \in A$. В частности, в работе [9] область некомпетентности или область отказа $InComp\left(d, X^\lambda\right)$ решающего дерева d – это интервал, соответствующий ветви, ранг которой превышает некоторое пороговое значение. Множество чистых выбросов $InComp\left(A, X^\lambda\right)$ относительно индуктивной модели A (эмпирического решающего леса) определяется как пересечение областей некомпетентности по всем решающим деревьям, входящим в лес $InComp\left(A, X^\lambda\right) = \operatorname{I}_{d \in A} InComp\left(d, X^\lambda\right)$. Если в результате построения эмпирического решающего леса $InComp\left(A, X^\lambda\right) \neq \varnothing$, то объекты из обучающей выборки, принадлежащие множеству $InComp\left(A, X^\lambda\right)$, образуют чистые выбросы относительно выбранной модели A алгоритмов обучения и удаляются из обучающей выборки.

Заключение

В работе введены строгие понятия чистого и мажоритарного выбросов; установлено существование модели алгоритмов обучения и обучающей выборки, относительно которых множество чистых выбросов непусто; доказано необходимое и достаточное условие существования пустого множества чистых выбросов, связанное с ёмкостью модели алгоритмов обучения.

В дальнейшем представляется перспективной разработка теоретически обоснованных критериев выявления чистых выбросов в обучающей выборке и методов построения фильтров на их основе.

Автор благодарит профессора В.И. Донского за внимание к работе и ценные замечания.

Литература

1. Quinlan J.R. Induction of Decision Trees // Machine Learning. – 1986. – 1(1). – P. 81-106.
2. Brodley C.E., Friedl M.A. Identifying Mislabeled Training Data // Journal of Artificial Intelligence Research. – 1999. – № 11. – P. 131-167.
3. Verbaeten S. Anneleen Van Assche Ensemble Methods for Noise Elimination in Classification Problems // Technical report, Department of Computer Science, K.U. Leuven, Belgium, 2003.
4. John G.H. Robust Decision Trees: Removing Outliers from Databases // Knowledge Discovery and Data Mining. – 1995. – P. 174-179.
5. Kubica J., Moore A. Probabilistic Noise Identification and Data Cleaning // Technical Report CMU-RI-TR-02-26, CMU, 2002.
6. Gamberger D., Lavrac N. Conditions for Occam's Razor Applicability and Noise Elimination // European Conference on Machine Learning. – 1997. – P. 108-123.
7. Schwarm S., Wolfman S. Cleaning Data with Bayesian Methods // Final project report for CSE574. – University of Washington, Winter.
8. Muggleton S., Srinivasan A., Bain M. Compression, Significance and Accuracy // Proceedings of the 9th International Workshop on Machine Learning. – 1992. – P. 338-347.
9. Weisberg S. Applied Linear Regression: John Wiley&Sons. – 1985. – 324 p.
10. Dietterich T.G. An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization // Machine Learning. – 2000. – 40(2). – P. 139-157.
11. Воронцов К.В. Комбинаторный подход к оценке качества обучаемых алгоритмов // Математические вопросы кибернетики / Под ред. О.Б. Лупанова. – М.: Физматлит, 2004. – Т. 13. – С. 5-36.
12. Вапник В.Н. Восстановление зависимостей по эмпирическим данным. – М.: Наука, 1979. – 448 с.
13. Донской В.И., Дюличева Ю.Ю. Индуктивная модель *r*-корректного эмпирического леса // Труды Международной конференции по индуктивному моделированию. – Львов. – 2002. – С. 54-58.

Ю.Ю. Дюлічева

Про задачі фільтрації навчальних даних

У статті наведено стислий огляд сучасних підходів до виявлення викидів у навчальних даних; введено строгі поняття чистого і мажоритарного викидів відносно моделі алгоритмів навчання; встановлено існування моделей алгоритмів навчання і навчальних вибірок, щодо яких множина чистих викидів не порожня; доведено необхідну і достатню умову існування порожньої множини чистих викидів, пов'язану з ємністю моделі алгоритмів навчання.

Yu.Yu. Dyulicheva

About Filtering Problems of Training Sample

A brief modern approaches review of outliers detection in training sample is considered; the accurate concepts of clear and majority outliers with respect to learning algorithms model are introduced; necessary and sufficient conditions for existence of empty set of clear outliers concerned with VCD of learning algorithms model are proved in the paper.

Статья поступила в редакцию 26.04.2006.