МАШИННЫЕ МЕТОДЫ ОБНАРУЖЕНИЯ ЗАКОНОМЕРНОСТЕЙ
(Вычислительные системы)
1981 год Выпуск 88

УДК 681.3.06:621.391

ИГРОВАЯ ИМИТАЦИОННАЯ МОДЕЛЬ СРАВНЕНИЯ АЛГОРИТМОВ ОБУЧЕНИЯ

А.Н.Манохин, В.Е.Плотникова

I. В в е д е н и е

Обилие описанных в литературе алгоритмов обучения и распознавания делает актуальной задачу их сравнения. Классифицировать и сравнивать алгоритмы можно по многим параметрам: количеству ошибок, трудоемкости, области применения (для бинарных, числовых, разнотипных признаков), теоретической обоснованности и т.д. В настоящей работе сравнение проводится лишь по одной характеристике - количеству ошибок при обучении и распознавании. Экспериментальные результаты сравнения алгоритмов обучения по этому критерию получены либо для модельных примеров, либо для реальных задач. В первом случае строится оценка математического ожидания вероятности ошибки для отдельных распределений (см., например, [1,2]). Во втором - вычисляются различные оценки (скользящая, разбиение на контроль и обучение и т.п.) для конкретных выборок (типичные примеры в [3, 4]). В [5] приведен краткий обзор работ по экспериментальному сравнению алгоритмов. Там же справедливо замечено, что большинство результатов, изложенных в этих работах, невозможно проверить, повторив эксперимент. Кроме того, такие результаты носят разрозненный характер, остается неясным, как подвести итоги многочисленным экспериментам, с какими моделями ставить новые эксперименты.

В данной работе рассматривается игровая статистическая модель задачи распознавания и на ее основе делается попытка дать ответ на поставленные вопросы.

2. Постановка задачи

Напомним основные определения.

Пусть $\mathcal{X} = \{a\}$ - м н о ж е с т в о о б ъ е к т о в с заданной на нем σ-алгеброй и вероятностной мерой ν.

Функции $X_1,\ldots,X_m$ X_{m+1} такие, что $X_i: \mathcal{X} \to R$, где R - множество действительных чисел, называются п р и з н а к а м и. Признак X_{m+1} - целевой и принимает конечное число значений α_1, $\alpha_2,\ldots,\alpha_k$. Пусть P - вероятностная мера, порождаемая в R^{m+1} признаками $X_1,\ldots,X_m,X_{m+1}$.

О б р а з о м Φ_i называется множество $\Phi_i = \{a \in \mathcal{X}$ таких, что $X_{m+1}(a) = \alpha_i\}$. В дальнейшем рассматривается случай $k = 2$, и, не ограничивая общности, будем считать, что $\alpha_1 = 1$, $\alpha_2 = 2$.

Назовем р е ш а ю щ и м п р а в и л о м F отображение, которое ставит в соответствие вектору $\bar{x} \in R^m$ значение $F(x) \in \{1,2\}$. Таким образом, каждому объекту a правило F ставит в соответствие решение о принадлежности его к образу, поскольку объектом a определяется вектор признаков $\bar{x} = (x_1,\ldots,x_m)$, где $x_i = X_i(a)$ при $i = 1,2,\ldots, m$. Пусть $x_{m+1} = X_{m+1}(a)$. Тогда решение $F(\bar{x})$ верно, если $F(\bar{x}) = x_{m+1}$, и ложно, если $F(\bar{x}) \neq x_{m+1}$.

Пусть $O_F = \{\bar{x}: F(\bar{x}) \neq x_{m+1}\}$, тогда в е р о я т н о с т ь о ш и б к и $P_0(F,P) = P(O_F)$ является критерием качества правила F.

Пусть $\bar{y} = (x_1,\ldots,x_m, x_{m+1})$, где $x_i = X_i(a)$ при $i = 1,2,\ldots,m+1$; $S = \{F\}$ - некоторое множество решающих правил. Пусть $\bar{y}_1,\ldots,\bar{y}_n$ - реализации $(m+1)$-мерных случайных величин $\bar{Y}_1,\ldots,\bar{Y}_i$, где $\bar{Y}_i$ - независимые и одинаково распределенные в соответствии с мерой P. Набор $\tilde{X} = \{\bar{y}_1,\ldots,\bar{y}_n\}$ будем называть о б у ч а ю щ е й в ы б о р к о й.

Отображение D, ставящее в соответствие произвольной обучающей выборке $\{\bar{y}_1,\ldots,\bar{y}_n\}$ решающее правило $F \in S$, называется а л г о р и т м о м о б у ч е н и я.

Качество алгоритма определяется как

$$M_0(D,P) = \int P_0(D(\tilde{X}), P)P^n(d\tilde{X}).$$

При фиксированной мере P будем считать, что алгоритм D_1 н е х у ж е алгоритма D_2, если верно соотношение $M_0(D_1,P) \leq M_0(D_2,P)$. Алгоритм D_1 л у ч ш е алгоритма D_2, если неравенство строгое.

Очевидно, что для одних P лучшим может оказаться алгоритм D_1, а для других D_2. В этом и состоит основная трудность при сравнении алгоритмов.

Рассмотрим теоретико-игровую постановку задачи распознавания [6]. Стратегией игрока будем считать алгоритмы обучения D, стратегией природы – вероятностную меру P в R^{n+1}. Функцией риска будет $M_0(D, P)$. Тогда результат стратегии игрока тем лучше, чем меньше $M_0(D,P)$. Полагаем, что $D \in \mathcal{L}$, где $\mathcal{L}$ – множество алгоритмов обучения (т.е. стратегий игрока), а $P \in \pi$, где π – множество вероятностных мер в R^{n+1} (т.е. стратегий природы).

Рассмотрим классические принципы определения оптимальной стратегии игрока. В дальнейшем символ $\succcurlyeq$ означает отношение "не хуже", а $\succ$ – "лучше". На множестве $\mathcal{L}$ определена функция $M(D)$, где $D \in \mathcal{L}$, тогда отношение между двумя алгоритмами D_1 и D_2 задается следующим образом: $D_1 \succcurlyeq D_2$, если $M(D_1) \le M(D_2)$.

Назовем о п т и м а л ь н ы м такой алгоритм D_0, на котором достигается $M(D_0) = \inf\limits_{D \in \mathcal{L}} M(D)$.

Первый принцип – принцип минимакса.

В этом случае $M(D) = \sup\limits_{P \in \pi} M_0(D,P)$.

Назовем б е з р а з л и ч н ы м такой алгоритм $D_Б$, который любой обучающей выборке $\tilde{X}$ ставит в соответствие рандомизированное решающее правило:

$$F_Б = \begin{cases} 1 \text{ с вероятностью } 1/2, \\ 2 \text{ с вероятностью } 1/2. \end{cases}$$

Пусть $P_Н$ – вероятностная мера на $\mathcal{X}$ такая, что для нее верно равенство $P(\bar{x}/1) = P(\bar{x}/2)$, где $P(\bar{x}/i)$ – условное распределение в пространстве R^n, при $X_{n+1}(a) = i$. Такая стратегия $P_Н$ не дает никакой информации для распознавания.

УТВЕРЖДЕНИЕ. Е с л и м н о ж е с т в о $\mathcal{L}$ с о д е р ж и т б е з р а з л и ч н ы й а л г о р и т м $D_Б$, а м н о ж е с т в о π с о д е р ж и т с т р а т е г и ю $P_Н$, т о $D_Б$ я в л я е т с я о п т и м а л ь н ы м п о к р и т е р и ю м и н и м а к с а.

ДОКАЗАТЕЛЬСТВО следует из того, что для любого $D \in \mathcal{L}$ верно $M_0(D,P_Н) = \frac{1}{2}$ и для любой $P \in \pi$ верно $M_0(D_Б, P) = \frac{1}{2}$. Как видим, принцип минимакса является неудовлетворительным, так как он приводит к тому, что при очень слабых ограничениях правило "подбрасывания монеты" является оптимальным алгоритмом.

Второй принцип байесовский.

На множестве стратегий природы π вводится вероятностная мера μ и $M(D) = \int M_0(D,P)\mu(dP)$.

Применение этого принципа встречает следующие ограничения:

а) на достаточно широком классе стратегий π трудно определить меру μ (например, на множестве всех стратегий P, у которых соответствующие распределения $P(\bar{x}/X_{m+1}(a) = i)$ абсолютно непрерывны, либо на множестве всевозможных P);

б) даже когда мера μ задана (например, когда каждый признак имеет конечное число градаций), выделение одной конкретной μ не оправдано, так как по-прежнему возможен целый класс мер μ.

Наконец, рассмотрим третий принцип - принцип минимакса потерь [6, гл. IУ].

Вводим $M(D) = \sup_{P\in\pi} (M_0(D,P) - \inf_{D'\in\mathcal{L}} M_0(D', P))$. Хотя этот принцип менее распространен, однако, по нашему мнению, именно он наиболее адекватно отражает логику выбора решения в задачах распознавания.

Содержательная интерпретация принципа минимакса потерь следующая. При фиксированной стратегии P наилучшее возможное значение равно $\inf_{D'\in\mathcal{L}} M_0(D',P)$. Вообще говоря, inf может и не достигаться в $\mathcal{L}$, но мы можем сколь угодно близко приблизиться к этому значению. Для упрощения изложения в дальнейшем везде будем предполагать, что inf и sup достигаются. Тогда значение $M_0(D,P) - \inf_{D'\in\mathcal{L}} M_0(D',P)$ есть потери, обусловленные тем, что мы применяем фиксированный алгоритм D, а не оптимальный для данной стратегии природы P. Далее, для каждого фиксированного D ищем стратегию природы, для которой эти потери наибольшие, эта величина и будет $M(D)$. Оптимальный алгоритм "минимизирует" "максимальные" потери, которые обусловлены применением одного и того же алгоритма для всех P, а не своего "лучшего" для каждой P.

3. Сравнительные характеристики для конечного класса $\mathcal{L}$ и выбор лучшего алгоритма

Далее мы будем рассматривать ситуацию, более близкую к реальной. Имеется конечный класс алгоритмов $\mathcal{L} = \{D_1,\ldots,D_L\}$. Информация о природе задается множеством стратегий $\pi = \{ P \}$. При-

ведем несколько примеров π, основанных на распространенных в распознавании подходах:

1) π - такой класс, что $P(\bar{x}/1)$ и $P(\bar{x}/2)$ - многомерные нормальные функции распределения;

2) π - класс, в котором распределения $P(\bar{x}/1)$ и $P(\bar{x}/2)$ абсолютно непрерывны;

3) π - класс всевозможных мер на $\mathcal{X}$.

Исследователю необходимо, зная класс π, выбрать один алгоритм из $\mathcal{L}$ для решения задачи. Подчеркиваем, что ему предлагается сразу указать алгоритм и не разрешается, например, пытаться получить решение несколькими алгоритмами, чтобы затем, сравнив решение по каким-то критериям, выбрать лучшее. Эта ситуация рассматривается в последнем разделе.

Итак, для конечного набора $\mathcal{L}$ нам необходимо дать сравнительные характеристики алгоритмов и выделить лучший для фиксированного множества π.

Введем матрицу наибольших потерь

$$C = \{c_{ij}\}_{i,j=1}^{L},$$

где $c_{ij} = \sup\limits_{P\in\pi} (M_0(D_j, P) - M_0(D_i, P))$ дает "наибольшие" потери, которые можно понести, если использовать D_j, отказавшись от D_i. Пусть матрица

$$R = \{P_{ij}\}_{i,j=1}^{L},$$

где P_{ij} - стратегия, на которой достигается $c_{ij} = \sup\limits_{P\in\pi}(M_0(D_j, P) - M_0(D_i, P))$. Если sup не достигается, то в матрице R в качестве элемента на i,j-м месте берем некоторую стратегию P_{ij}, на которой значение c_{ij} "почти" достигается, т.е.

$$M_0(D_j, P_{ij}) - M_0(D_i, P_{ij}) \geq c_{ij} - \varepsilon,$$

где $\varepsilon > 0$ - заданное, сколь угодно малое число.

Пусть $M(D_i)$ - функция, определяющая предпочтение для принципа минимакса потерь. Тогда справедливо

УТВЕРЖДЕНИЕ.

$$\min_{i=1,2,\ldots,L} M(D_i) = \min_{i=1,2,\ldots,L} \max_{j=1,\ldots,L} c_{ij}.$$

Из этого утверждения сразу следует, что знание матрицы C позволяет просто определить алгоритм, оптимальный в смысле принципа минимакса потерь. Для этого достаточно в матрице C найти значение $l \in [1, L]$ такое, что

$$\max_{j=1,\ldots,L} c_{jl} = \min_{i=1,\ldots,L} \cdot \max_{j=1,\ldots,L} c_{ji}.$$

Аналитические решения для задач определения матриц C,R и оптимального в смысле минимакса потерь алгоритма D_i найти трудно. Предлагается строить для этих задач приближения на основе имитационных моделей. Имитация здесь понимается в том смысле, что мы будем имитировать поведение второго игрока (т.е. природы) с помощью псевдослучайных датчиков. При построении матрицы C специалист на основе теоретических знаний, опыта, интуиции пытается сформулировать для пары D_i, D_j стратегию природы $P_{ij}^{(1)}$ так, чтобы получить как можно большее значение величины $M_0(D_j,P) - M_0(D_i,P)$. Полученное значение $M_0(D_j,P_{ij}^{(1)}) - M_0(D_i,P_{ij}^{(1)})$ будем считать первым приближением $\tilde{c}_{ij}$ для c_{ij}, а приближение $\tilde{P}_{ij}$ полагаем равным $P_{ij}^{(1)}$. Далее можно проверять новую гипотетическую стратегию $P_{ij}^{(2)}$ и т.д. На k-м шаге проверяем для пары D_i, D_j гипотетическую стратегию $P_{ij}^{(k)}$. Если $M_0(D_j,P_{ij}^{(k)}) - M_0(D_i,P_{ij}^{(k)}) > \tilde{c}_{ij}$, то полагаем приближение $\tilde{c}_{ij} = M_0(D_j,P_{ij}^{(k)}) - M_0(D_i,P_{ij}^{(k)})$, а $\tilde{P}_{ij} = P_{ij}^{(k)}$. В противном случае приближения остаются без изменения. Такие приближения строятся для каждой пары D_i, D_j, следовательно, получаем приближенные матрицы $\tilde{C}$ и $\tilde{R}$.

Приведем еще одну рекомендацию построения последовательности $P_{ij}^{(k)}$. Автор (авторы) алгоритма D_i пытаются предложить контрпримеры, которые дают максимальные потери в математическом ожидании вероятности ошибки, если отказаться от его алгоритма D_i в пользу D_j. Естественно, автор D_j проделывает то же самое относительно своего алгоритма.

Определение $M_0(D,P)$ для фиксированной пары D,P также является сложной проблемой. Среди широко распространенных алгоритмов обучения и распознавания лишь для линейной дискриминантной функции в случае, когда $P(\bar{x}/1)$ и $P(\bar{x}/2)$ - нормальные распределения с общей матрицей ковариаций, предложено точное выражение для

$M_0(D,P)$ (см.[7]),на основе которого Пикялис и Раудис [8] построили вычислительный алгоритм и табулировали значения $M_0(D,P)$.Трудности, которые встретились при рассмотрении этой задачи в классических условиях (нормальность, линейность), не позволяют надеяться на быстрое и успешное ее решение в других ситуациях. Поэтому предлагается приближать $M_0(D,P)$ на основе моделирования, техника которого широко известна. Моделируем обучающую выборку $\tilde{X}$ объема n и контрольную выборку $\tilde{Y}$ объема n_1. По выборке X алгоритм D строит решающее правило, которое дает некоторые решения на $\tilde{Y}$. Считаем частоту ошибок на Y, повторяем эксперимент NK раз и усредняем частоту ошибок по экспериментам. Полученная величина и будет приближением для $M_0(D,P)$. Назовем приближенно оптимальным в смысле минимакса потерь алгоритм, определяемый на основе матрицы $\tilde{C}$.

4. Классы оптимальных стратегий

В предыдущем разделе мы пытались на основе игровой постановки задачи определить единственный алгоритм обучения. Но принципы определения последнего могут быть спорными, поскольку нет очевидных общепринятых принципов. В теории игр считается, что если мы не можем указать лучший алгоритм, то следует сократить множество $\mathcal{Z}$, выбросив явно "ненужные" алгоритмы. Приведем некоторые определения теории игр [6, гл.4], необходимые в рассматриваемой задаче.

Алгоритмы D_i, D_j будем считать *эквивалентными*, если $M_0(D_i,P) = M_0(D_j,P)$ для всех стратегий $P \in \pi$.Очевидно, что из набора эквивалентных алгоритмов следует оставить лишь один.Поэтому полагаем, что в исходном наборе нет эквивалентных алгоритмов.

Алгоритм $D_i \in \mathcal{Z}$ называется *допустимым*, если не существует такого $D_j \in \mathcal{Z}$, что $M_0(D_j,P) \le M_0(D_i,P)$ для всех стратегий $P \in \pi$. Очевидно, что множество $\mathcal{Z}$ может быть без потерь сокращено до множества допустимых алгоритмов. Для того чтобы утверждать, что D_i не является допустимым, достаточно установить, что в матрице C существует j такое, что $c_{ij} = 0$.

Множество алгоритмов $\mathcal{Z}_0$ будем называть π-полным, если для любого $P \in \pi$ и для каждого $D_i \in \mathcal{Z}_0$ существует $D_j \in \mathcal{Z}_0$ такой, что $M_0(D_j,P) \le M_0(D_i,P)$.

Любое множество алгоритмов $\mathcal{Z}$ можно сократить до его π-полного подмножества $\mathcal{Z}_0$ в том смысле, что

$$\min_{D_i \in \mathcal{L}} M_0(D_i, P) = \min_{D_j \in \mathcal{L}_0} M_0(D_j, P) \text{ для любого } P \in \pi.$$

Поэтому можно ставить вопрос о минимальных по мощности (в данном случае по количеству алгоритмов) π-полных множествах алгоритмов.

УТВЕРЖДЕНИЕ. Если для некоторого $D_i \in \mathcal{L}$ существует $P \in \pi$ такая, что $M_0(D_i, P) < M_0(D_j, P)$ для всех $D_j \in \mathcal{L}$, где $j \neq i$, то D_i входит в любое π-полное множество. Если для любой $P \in \pi$: $M_0(D_i, P) > \min_{j \neq i, j \in \mathcal{L}} M_0(D_j, P)$, то алгоритм D_i не входит ни в одно минимальное π-полное множество для множества $\mathcal{L}$.

Таким образом, для того чтобы показать, что алгоритм D_i не может быть выброшен без потерь из $\mathcal{L}$, необходимо найти $P \in \pi$, для которой $M_0(D_i, P) < M_0(D_j, P)$ для любого $D_j \in \mathcal{L}$, $j \neq i$. Именно такой пример должен стремиться отыскать специалист (например, автор) для алгоритма D_i, чтобы показать, что его нельзя исключить из пакета на основе принципа π-полноты.

Сформулируем процедуру построения π-полного множества алгоритмов. Каждая стратегия $P \in \pi$ порождает подмножество $\mathcal{L}_P = \{D_j \in \mathcal{L}$ таких, что $\min_{D_i \in \mathcal{L}} M_0(D_i, P) = M_0(D_j, P)\}$. Множество всех $P \in \pi$ порождает некоторое множество G различных подмножеств множества $\mathcal{L}$. Необходимо построить такое подмножество $\mathcal{L}_0$, что для любого $\mathcal{L}' \in G$ пересечение $\mathcal{L}_0 \cap \mathcal{L}'$ непусто и $\mathcal{L}_0$ содержат минимальное количество элементов.

Для этого выделим все одноэлементные подмножества, входящие в G. Соответствующие им алгоритмы образуют множество $\mathcal{L}_0^1$. Далее строим новое множество G^1, вычеркивая из G те множества, которые имеют непустое пересечение с $\mathcal{L}_0^1$. Для G^1 решаем полным перебором ту же задачу построения множества $\mathcal{L}_0^2$ такого, чтобы для любого $\mathcal{L}' \in G^1$ пересечения $\mathcal{L}_0^2 \cap \mathcal{L}'$ было непусто и $\mathcal{L}_0^2$ - минимально. Полагаем $\mathcal{L}_0 = \mathcal{L}_0^1 \cup \mathcal{L}_0^2$.

Предлагается по конечному набору $\pi^K = \{P_1, \ldots, P_K\}$, $\pi^K \subset \pi$, строить приближенный класс допустимых алгоритмов и приближенный π-полный класс алгоритмов.

Алгоритм D_i будет допустимым, если $\tilde{c}_{ij} > 0$ для всех $j = 1, 2, \ldots, L$, $j \neq i$. Если для некоторых $i, j (i \neq j)$ $\tilde{c}_{ij} = 0$, где $\tilde{c}_{ij}$ построено на основе $\pi^K = \{P_1, \ldots, P_K\}$, то алгоритм D_i является

приближенно недопустимым. Исследователь стремится выдвинуть стратегию P_{K+1} так, чтобы, как указывалось выше, увеличить $\tilde{c}_{ij}$, и тем самым опровергнуть тезис о недопустимости D_i.

Множество $\mathcal{L}_0$ назовем приближенно π-полным, если оно является π^K-полным. Исследователь стремится построить P_{K+1} так, чтобы для алгоритма D_i, если он не входит в $\mathcal{L}_0$, выполнялось соотношение $M_0(D_i, P_{K+1}) < M_0(D_j, P_{K+1})$ для всех $D_j \in \mathcal{L}$. Тогда тезис о полноте $\mathcal{L}_0$ опровергнут и необходимо его расширить.

5. Задача сравнения решающих правил

Выше мы рассматривали задачу в постановке, когда необходимо сделать выбор одного или нескольких алгоритмов обучения. Теперь рассмотрим вторую постановку. Предположим, что исследователь, используя алгоритмы $D_1, \ldots, D_L$, получит для обучающей матрицы $\tilde{X}$ решающие правила $F_1, \ldots, F_L$, где $F_i = D_i(\tilde{X})$. Перед ним встает проблема выбора одного конкретного F_i в качестве конечного результата. Обычный, хотя может быть прямо и не формулируемый, подход заключается в следующем. Кроме алгоритма D_i, сопоставляющего обучающей матрице $\tilde{X}$ правило $F_i = D_i(\tilde{X})$, задают оценку O_i, которая сопоставляет $\tilde{X}$ действительное число. Это оценка для вероятности ошибки $P_0(F, \tilde{X})$. В качестве конечного выбирается F_q, для которого

$$O_q(\tilde{X}) = \min_{i=1,\ldots,L} O_i(\tilde{X}).$$

Нетрудно понять, что такая стратегия определяет некоторый новый алгоритм обучения D_0, который задает свое

$$\varepsilon_0 = \sup_{P \in \pi} \left(M_0(D_0, P) - \min_{i=1,\ldots,L} M_0(D_i, P) \right).$$

Выбранная система оценок тем лучше, чем меньше ε_0. Применение D_0 оправдано тогда, когда $\varepsilon_0 < \min_{i=1,\ldots,L} M(D_i)$, где $M(D_i)$ определено в п.3. Характеристика ε_0 аналогично C и R может приближаться по модельным распределениям.

Литература

1. NESS J.W. van, SIMPSON C. On the effects of dimension in discriminant analysis.- Technometrics, 1976, v.18, N 2, May, p.175-187.

2. GESSAMAN M.P., GESSAMAN P.H. A comparison of some multivariate discrimination procedures. - J.A.S.A.,1972,v.63, N 338, p.468-472.

3. DAY N.E. Linear and guadratic discrimination in pattern recognition.-IEEE Trans.on Inf.Theory, 1969, v.IT-15, N 3, p.419-420.

4. PATRICK E.A., CHEN L.V.L. Interactive use of problem knowledge for clustering and decision making. - IEEE Trans. on Comp., 1971,v.C-20, N 2,p.216-222.

5. РАУДИС Ш., ПИКЯЛИС В., ЮШКЯВИЧЮС К. Экспериментальное сравнение тринадцати алгоритмов классификации. -В кн.: Статистические проблемы управления. Вып. II, Вильнюс, 1975, с.53-69.

6. БЛЕКУЭЛЛ Д., ГИРШИК М.А. Теория игр и статистических решений.-М: ИЛ, 1958. - 374 с.

7. SITGREAVES R. Some results on the distribution of the classification statistics. - In: Studies in item analysis and prediction. Stanford University Press,1961,p.381-394.

8. РАУДИС Ш., ПИКЯЛИС В. Табулирование зависимости ожидаемой ошибки классификации линейной дискриминантной функции от объема обучающей выборки. -В кн.: Статистические проблемы управления.Вып. II, Вильнюс, 1975, с.81-96.

Поступила в ред.-изд.отд.
25 марта 1981 года