УДК 519.256

СОВМЕСТНЫЙ ВЫБОР ОБЪЕКТОВ И ПРИЗНАКОВ В ЗАДАЧАХ МНОГОКЛАССОВОЙ КЛАССИФИКАЦИИ КОЛЛЕКЦИИ ДОКУМЕНТОВ

Адуенко А.А., Стрижов В.В.

Работа посвящена задаче ранжирования поисковой выдачи. Для решения этой задачи предложен алгоритм многоклассовой классификации с совместным отбором объектов и признаков, а также его модификация для сравнения релевантности внутри одного класса. Отбор производится двумя способами: с помощью шаговой регрессии и с помощью генетических алгоритмов. Результаты, полученные разными методами, сравниваются. Алгоритм тестируется на синтетических данных и данных поисковой выдачи Яндекса.

Ключевые слова: многоклассовая классификация, ранжирование поисковой выдачи, логистическая регрессия, выбор признаков, фильтрация объектов, релевантность.

Введение

В работе рассматривается задача многоклассовой классификации документов [1-2]. Документами являются ответы поисковой машины на запросы пользователей. В качестве меток классов используется линейно-упорядоченный набор, отражающий степень релевантности документа запросу. Требуется каждому документу поставить в соответствие число, характеризующее его релевантность запросу. Одними из методов, с помощью которых решают эту задачу являются SVM-регрессия [3] и случайные леса [4]. В данной работе для решения задачи классификации используется многоклассовая логистическая регрессия [5].

Для ранжирования документов по релевантности внутри классов и оценки параметров регрессии предлагается модификация многоклассовой логистической регрессии. В вычислительном эксперименте представлены результаты работы предложенных алгоритмов на данных поисковой выдачи Яндекса [6]. В качестве базового алгоритма, с которым происходит сравнение, используется многоклассовая модификация SVM [7]. Для оценки качества используется Discounted Cumulative Gain [8].

Каждый документ исследуемой коллекции [6] описан 245 признаками. Обучающая выборка содержит почти 105 документов. Поэтому необходимо произвести отбор объектов и признаков. Задача отбора признаков и объектов решается предложенным в работе алгоритмом. В качестве альтернативного решения используется генетический алгоритм [9-10]. Размер и состав наборов признаков, отобранных обоими алгоритмами, сравниваются.

Постановка задачи

Пусть нам заданы выборка $D = \{(\mathbf{x}_i, y_i)\}$, $i \in \mathsf{I} = \{1,\ldots,N\}$, матрица признаков в которой $\mathbf{X} = [\mathbf{x}_1^{\mathrm{T}},\ldots,\mathbf{x}_m^{\mathrm{T}}]^{\mathrm{T}} \in \mathsf{R}^{N\times n}$; N – число записей данных; n – число признаков, и вектор ответов $\mathbf{y} = [y_1,\ldots,y_N]^{\mathrm{T}}$; $y_i \in Y \subseteq \mathsf{N}_0$, где Y – линейно-упорядоченное конечное множество, состоящее более чем из одного элемента.

Для определения принадлежности объектов x классам y используется модель многоклассовой логистической регрессии – параметрическая функция

$$f : (\boldsymbol{\Theta},\mathbf{x}) \to \hat{y} \in Y, \tag{1}$$

отображающая пару «параметры, объект» в метку класса $\hat{y}$ из множества Y. Для оценки адекватности модели задачи используется функция качества $S(\boldsymbol{\Theta},\mathsf{X},\mathsf{A},\mathsf{B})$, где $\boldsymbol{\Theta}$ – набор параметров модели; X – набор индексов некоторого множества объектов; A – набор индексов используемых признаков; B – набор индексов, используемых при обучении объектов.

Поиск оптимального набора параметров $\hat{\boldsymbol{\Theta}}$ осуществляется следующим образом:

$$\hat{\boldsymbol{\Theta}} = \operatorname*{argmin}_{\boldsymbol{\Theta}\in\mathsf{R}^L} Q(\boldsymbol{\Theta},\mathsf{B},\mathsf{A},\mathsf{B}), \tag{2}$$

где L – размерность пространства параметров модели. Задачу поиска оптимальных наборов объектов и признаков $\{\boldsymbol{\chi}_j\}, j \in \mathsf{A}$; $\{\mathbf{x}_i\}, i \in \mathsf{B}$ сформулируем в виде

$$(\mathsf{A},\mathsf{B}) = \operatorname*{argmin}_{\mathsf{A}\subseteq\mathsf{J},\mathsf{B}\subseteq\mathsf{I}} Q(\hat{\boldsymbol{\Theta}},\mathsf{S},\mathsf{B},\mathsf{A}). \tag{3}$$

Требуется по обучающей выборке оценить параметры $\boldsymbol{\Theta}$ модели, чтобы затем классифицировать объекты в предположении, что из исходного множества признаков – столбцов матрицы $\boldsymbol{\chi} = [\boldsymbol{\chi}_1,\ldots,\boldsymbol{\chi}_n]$ и исходного множества объектов $\{\mathbf{x}_i\}, i \in \mathsf{I} = \{1,\ldots,N\}$ – отобраны некоторые подмножества признаков $\{\boldsymbol{\chi}_j\}, j \in \mathsf{A}$ и объек-

тов, оптимальных согласно (3), $|\mathsf{A}| = n^* \le n$, $|\mathsf{B}| = N^* \le N$. Параметр $\boldsymbol{\Theta}$ находится путем максимизации качества модели $Q(\boldsymbol{\Theta},\mathsf{X},\mathsf{A},\mathsf{B})$ на обучающей выборке S.

Задача нахождения оптимального набора объектов и признаков решается в работе с помощью предложенного шагового алгоритма и с помощью генетического алгоритма.

Алгоритм многоклассовой логистической регрессии

Сопоставим каждому классу $C_k, k = 1,\dots K$ весовой вектор $\mathbf{w}_k \in \mathsf{R}^n$, где n – число признаков. Тогда для объекта $\mathbf{x}_i$ вероятность попасть в класс C_k в модели логистической регрессии равна

$$P(C_k \mid \mathbf{x}_i) = \frac{\exp \mathbf{w}_k^{\mathrm{T}} \mathbf{x}_i}{\sum_{j=1}^{K} \exp \mathbf{w}_j^{\mathrm{T}} \mathbf{x}_i}, i \in \mathsf{I}. \quad (4)$$

Введем для $P(C_k \mid \mathbf{x}_i)$ обозначение y_{ik}. Для каждого объекта $\mathbf{x}_i, i \in \mathsf{I}$ введем целевой вектор $\mathbf{t}_i$, где $t_{ik} \in [0,1]$ есть принадлежность объекта $\mathbf{x}_i$ классу C_k. В нашем случае на обучающей выборке считаем $t_{ik} = 1$, если объект $\mathbf{x}_i$ лежит в классе C_k, иначе $t_{ik} = 0$. Обозначим целевую матрицу, составленную из t_{ik} $\mathbf{T} = [t_{ik}]$. Запишем функцию правдоподобия выборки, используя (4):

$$p(\mathbf{T} \mid \mathbf{w}_1,\dots,\mathbf{w}_K) = \prod_{i=1}^{N}\prod_{k=1}^{K} P(C_k \mid \mathbf{x}_i)^{t_{ik}} \prod_{i=1}^{N}\prod_{k=1}^{K} y_{ik}^{t_{ik}}. \quad (5)$$

Запишем отрицательный логарифм функции правдоподобия (5) и поставим задачу его минимизации:

$$E(\mathbf{w}_1,\dots,\mathbf{w}_K) = -\sum_{k=1}^{K}\sum_{i=1}^{N} t_{ik} \log y_{ik} \to \min_{\mathbf{w}_1,\dots,\mathbf{w}_K} \quad (6)$$

Для нахождения минимума функции (6) рассчитаем ее градиент и гессиан.

Введем обозначение $a_k^i = \mathbf{w}_k^{\mathrm{T}} \mathbf{x}_i$. Рассчитаем сначала $\frac{\partial a_k^i}{\partial w_j}$ и $\frac{\partial y_{ik}}{\partial a_j^i}$:

$$\frac{\partial a_k^i}{\partial w_j} = \mathbf{x}_i \cdot I_{kj}, \quad (7)$$

где I_{kj} – элемент единичной матрицы. Запишем y_{ik} через $\{a_j^i\}_{j=1}^K$ следующим образом:

$$y_{ik} = \frac{\exp a_k^i}{\sum_{l=1}^{K} \exp a_l^i}, \quad (8)$$

и с учетом (8) получим

$$\frac{\partial y_{ik}}{\partial a_j^i} = \frac{\exp a_k^i}{\sum_{l=1}^{K} \exp a_l^i} \cdot \frac{\partial a_k^i}{\partial a_j^i} - \frac{\exp a_k^i}{\left(\sum_{l=1}^{K} \exp a_l^i\right)^2} \frac{\partial\left(\sum_{l=1}^{K} \exp a_l^i\right)}{\partial a_j^i} =$$

$$= y_{ik} I_{kj} - y_{ik} y_{ij}.$$

Таким образом

$$\frac{\partial y_{ik}}{\partial a_j^i} = y_{ik}(I_{kj} - y_{ij}). \quad (9)$$

Из (7) и (9) получаем

$$\frac{\partial y_{ik}}{\partial \mathbf{w}_j} = \frac{\partial y_{ik}}{\partial a_j^i} \cdot \frac{\partial a_j^i}{\partial \mathbf{w}_j},$$

то есть

$$\frac{\partial y_{ik}}{\partial \mathbf{w}_j} = y_{ik}(I_{kj} - y_{ij})\mathbf{x}_i. \quad (10)$$

Искомый градиент имеет вид

$$\begin{aligned} &\nabla_{\mathbf{w}_j} E(\mathbf{w}_1,\dots,\mathbf{w}_K) = \\ &= -\sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik} \frac{1}{y_{ik}} \frac{\partial y_{ik}}{\partial \mathbf{w}_j} = \\ &= -\sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik} \frac{1}{y_{ik}} y_{ik}(I_{kj} - y_{ij})\mathbf{x}_i = \\ &= -\sum_{i=1}^{N} t_{ij}\mathbf{x}_i + \sum_{i=1}^{N} y_{ij}\mathbf{x}_i \sum_{k=1}^{K} t_{ik} = \sum_{i=1}^{N}(y_{ij} - t_{ij})\mathbf{x}_i. \end{aligned} \quad (11)$$

Из выражений для градиента (11) и (10) для подматрицы H_{kj} размера $L \times L$ гессиана H получаем

$$\begin{aligned} &\nabla_{\mathbf{w}_k}\nabla_{\mathbf{w}_j} E(\mathbf{w}_1,\dots,\mathbf{w}_K) = \\ &= \nabla_{\mathbf{w}_k}\left(\sum_{i=1}^{N}(y_{ij} - t_{ij})\mathbf{x}_i\right) = \\ &= \sum_{i=1}^{N}\mathbf{x}_i \nabla_{\mathbf{w}_k} y_{ij} \sum_{i=1}^{N} y_{ij}(I_{jk} - y_{ik})\mathbf{x}\mathbf{x}^T. \end{aligned} \quad (12)$$

Гессиан H есть матрица размера $LK \times LK$ вида

$$H = \begin{pmatrix} H_{11} & \cdots & H_{1K} \\ \vdots & \ddots & \vdots \\ H_{K1} & \cdots & H_{KK} \end{pmatrix}.$$

Если бы H была положительно определенной матрицей, то для нахождения оптимального вектора весов можно было бы воспользоваться методом Ньютона-Рафсона:

$$\mathbf{w}^{t+1} = \mathbf{w}^t - \alpha H^{-1} \nabla E, \alpha > 0. \quad (13)$$

Однако из (12) получаем, что в каждой строке матрицы H сумма равна нулю, то есть матрица H вырождена и метод Ньютона-Рафсона не применим. Поэтому в работе используются методы безусловной минимизации первого порядка.

Определив векторы $\mathbf{w}_1,\ldots,\mathbf{w}_K$ по формуле (4), найдем для каждого объекта $\mathbf{x}_i$ вероятности $P(C_k \mid \mathbf{x}_i)$. Класс C_{k^*}, к которому будет отнесен объект $\mathbf{x}_i$, найдем из условия

$$k^* = \underset{k=1,\ldots,K}{\operatorname{argmax}} P(C_k \mid \mathbf{x}_i). \quad (14)$$

Заметим, что алгоритм многоклассовой логистической регрессии, описанный выше, классифицирует объекты и потому не подразумевает сравнения объектов внутри классов (объекты из разных классов сравнимы, так как метки классов линейно-упорядочены). Это ведет к неустойчивой и часто неправильной классификации объектов, у которых несколько классов близки к выполнению (14). Поэтому далее откажемся от требования, что $\hat{y}_i$ принадлежит конечному линейно-упорядоченному множеству Y, заменив его на требование принадлежности значения $\hat{y}_i$ отрезку $[C_1, C_K]$, считая классы линейно-упорядоченными. Тогда рассматриваемая задача перестанет быть задачей классификации, а становится задачей регрессии на отрезок. Для ее решения можно использовать полученное ранее решение задачи классификации. С учетом линейной упорядоченности меток классов в качестве оценки $\hat{y}_i$ рассмотрим

$$\hat{y}_i = \sum_{k=1}^{K} C_k P(C_k \mid \mathbf{x}_i). \quad (15)$$

Алгоритмы отбора объектов и признаков

Рассмотрим пошаговый алгоритм совместного отбора объектов и признаков. Вернемся к задаче отбора признаков (3). Предложим алгоритм, позволяющий отбирать не только признаки, но и объекты. В частном случае этого алгоритма, когда отбор объектов не производится, он переходит в алгоритм отбора признаков. Будем пошагово отбирать объекты и признаки для модели многоклассовой логистической регрессии.

Отбор признаков будем проводить из всего множества признаков. Для отбора объектов введем два множества B_1 и B_2. Из B_1 происходит отбор множества $\widetilde{B}_1 \subseteq B_1$ объектов, которые и будут объектами, по которым будет происходить минимизация (6) и нахождение векторов весов $\mathbf{w}_1,\ldots,\mathbf{w}_K$.

По множеству B_2 происходит контроль, и решение о добавлении или удалении объектов и признаков будет приниматься по изменению значения оптимизируемой функции $E(\mathbf{w}_1,\ldots,\mathbf{w}_K)$ именно на множестве B_2. Различие множеств B_2 и $\widetilde{B}_1$ позволит избежать минимизации $E(\mathbf{w}_1,\ldots,\mathbf{w}_K)$ исключительно за счет уменьшения размеров множества $\widetilde{B}_1$.

Приведем описание алгоритма. Алгоритм на входе имеет шесть параметров: l_1; u_1; l_2; u_2; D_1 и D_2, смысл которых разъясняется ниже. На начальном этапе задаем множество объектов модели $Y \subseteq X_1$ и множество признаков модели $\mathsf{A} = \{1\}$, где 1 соответствует постоянному признаку. Пусть на очередном шаге значение функции потерь на множестве X_2 равно E. Затем в имеющуюся модель по очереди добавляем признаки $f_i \in \mathsf{J}$. Для полученных моделей $Y_j = Y, \mathsf{A}_j = \mathsf{A} \cup \{j\}$ считаем значение функции потерь E_j на множестве X_2. Находим

$$j^* = \arg\min_j e_j. \quad (16)$$

Вычислим значение критерия $r = (E - E_{j^*})/e$. Если $r > u_1$, добавляем признак f_{j^*} в модель, в противном случае фиксируем признак f_{j^*} и пробуем в новую модель добавить еще один признак – если общее число добавляемых признаков не превышает D_1. Критерий добавления остается прежним: $r > u_1$, но r рассчитывается уже при добавлении в модель полученного на очередном шаге набора признаков.

Затем пытаемся добавить новый объект в модель аналогично добавлению признака, с той разницей, что для принятия решения о добавлении объекта используется параметр u_2, а число добавляемых объектов не превышает D_2.

После этого осуществляем поочередно удаление признаков и удаление объектов из модели по

аналогичному правилу. С той разницей, что при удалении признаков r рассчитывается по формуле $r = (E - E_{i^*})/E^{j^*}$, где j^* определяется из условия (16) при $\mathsf{A}_j = \mathsf{A} \setminus \{j\}$. Максимальное число удаляемых признаков также равно D_1. Удаление происходит, если $r < l_1$.

По аналогии с удалением признаков происходит удаление объектов – с той лишь разницей, что удаление происходит, если $r < l_2$, а максимальное число удаляемых объектов равно D_2. Чтобы алгоритм останавливался, требуется наложить условия $l_1 < r_1$ и $l_2 < r_2$ на значения параметров. Если на очередной итерации не происходит добавление или удаление признаков или объектов, то алгоритм останавливается.

Опишем генетический алгоритм отбора признаков. Рассмотрим обучающую выборку S и поставим задачу отбора множества признаков $\{\boldsymbol{\chi}_j\}, j \in \mathsf{A}$, используемых в логистической регрессии:

$$[\mathsf{A}, \{\mathbf{w}_1, \ldots, \mathbf{w}_K\}] = \\ = \underset{\{\mathbf{w}_1,\ldots,\mathbf{w}_K\} \in \mathsf{R}^{|\mathsf{A}|}, \mathsf{A} \subseteq \mathsf{J}}{\operatorname{argmin}} E(\mathbf{w}_1, \ldots, \mathbf{w}_K). \qquad (17)$$

Опишем итеративный алгоритм, который применялся для решения задачи отбора признаков (17). Будем характеризовать набор индексов использующихся признаков A вектором $\mathbf{b}$ из 0 и 1 размерности $|\mathsf{J}| = n$. Пусть перед r-ой итерацией алгоритма есть некоторый набор векторов $V = \{\mathbf{b}_1, \ldots, \mathbf{b}_v\}$, где $v = |V|$. Каждому вектору $\mathbf{b}_i$ из V сопоставим число

$$e_i = \min_{\mathsf{A}=\mathsf{A}(\mathbf{b}_i), \{\mathbf{w}_1,\ldots,\mathbf{w}_K\} \in \mathsf{R}^{|\mathsf{A}|}} E(\mathbf{w}_1, \ldots, \mathbf{w}_K).$$

Исключим из набора V долю α векторов с наибольшими значениями e_i и заменим их дубликатами векторов с долей α с наименьшими значениями e_i. Затем разобьем векторы на пары $\{(\mathbf{b}_{i_k}, \mathbf{b}_{j_k})\}_{k=1}^{\lfloor \frac{v}{2} \rfloor}$ (если v нечетно, то один вектор может остаться без пары). Внутри каждой пары проведем операцию скрещивания, которая заменяет пару векторов на некоторую другую пару векторов. Правила этой замены будут описаны ниже. Затем с каждым из получившихся векторов с вероятностью p происходит мутация, то есть случайный бит b меняется на противоположный.

Опишем операцию скрещивания. Рассмотрим пару векторов: $\mathbf{b}_i = (b_1^i, \ldots b_n^i)^{\mathrm{T}}$; $\mathbf{b}_j = (b_1^j, \ldots b_n^j)^{\mathrm{T}}$. Сгенерируем случайное натуральное число $z \in 1, \ldots n$. Тогда результатом операции скрещивания, примененной к векторам $\mathbf{b}_i$ и $\mathbf{b}_j$, будут новые векторы $\mathbf{b}_i' = (b_1^i, \ldots b_{z-1}^i b_z^j \ldots b_n^j)^{\mathrm{T}}$ и $\mathbf{b}_j' = (b_1^j, \ldots, b_{z-1}^j b_z^i, \ldots, b_n^i)^{\mathrm{T}}$.

Функционал качества многоклассовой классификации

Рассмотрим функционал качества Q_2 оценки качества решения задачи. Рассмотрим произвольный запрос $q_j \in Q$, где Q – множество всех запросов, и соответствующие ему документы и оценки их релевантности $\Omega_j = \{\mathbf{x}_i, \hat{y}_i\}, i \in \mathsf{I}_j$. Здесь I_j задает набор индексов документов, соответствующих запросу q_j. Для каждого q_j отсортируем документы внутри Ω_j по убыванию их оценок релевантности $\mathbf{y}$, получим множество Ω_j^*. При этом документы $\mathbf{x}_i, \mathbf{x}_j$ с одинаковыми оценками релевантности $\hat{y}_i = \hat{y}_j$ располагаются в порядке убывания их реальных релевантностей y_i и y_j. Обозначим $\mathrm{ind}(\mathbf{x}_i)$ – номер документа $\mathbf{x}_i$ в I.

В качестве функционала качества будем использовать DCG, усредненный по запросам:

$$Q_2(\hat{y}) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} DCG_j, \qquad (18)$$

где

$$DCG_j = \sum_{i=1}^{|\Omega_j|} \frac{y_{\mathrm{ind}(\mathbf{x}_i)}}{\log_2 i + 1}. \qquad (19)$$

Последняя сумма берется по элементам $\{\mathbf{x}_i, \hat{y}_i\}, i \in \mathsf{I}_j$. Проанализируем функционал качества DCG.Рассмотрим произвольный запрос q и два документа $\mathbf{x}_1$ и $\mathbf{x}_2$, относящихся к этому запросу. Предположим, что $K = 5$ и для $C_1, \ldots C_5$ вычислены следующие вероятности $P(C_k | \mathbf{x}_1)$, $P(C_k | \mathbf{x}_2)$; $k = 1, \ldots 5$ по формуле (4) – см. таблицу 1.Оба вектора $\mathbf{x}_1$ и $\mathbf{x}_2$ в соответствии с (14) будут отнесены к классу C_2. Однако вектор $\mathbf{x}_2$ следует ранжировать выше, чем $\mathbf{x}_1$, так как для $\mathbf{x}_2$ вероятности попасть во все классы выше C_2 не ниже, чем для $\mathbf{x}_1$, а вероятность попасть в класс C_4 выше.

Это находит отражение и в функционале (19), так как если $y_2 > y_1$, а $\hat{y}_2 = \hat{y}_1$, то значение DCG_q для запроса q будет ниже, чем если бы было выполнено $\hat{y}_2 > \hat{y}_1$. С этой трудностью справляется предложенная модификация алгоритма многоклассовой логистической регрессии. Например, для указанных векторов $\mathbf{x}_1$ и $\mathbf{x}_2$ в соответствии с (15) $1{,}8 = \hat{y}_2 > \hat{y}_1 = 1{,}05$. Предположение о том, что модификация логистической регрессии будет лучше в терминах функционала

качества Q_2 (19), чем исходный алгоритм логистической регрессии, подтверждается экспериментальным путем.

Таблица 1. Вероятности классов (пример)

Класс	$\mathbf{x}_1$	$\mathbf{x}_2$
$C_1 = 0$	0,3	0,05
$C_2 = 1$	0,.5	0,5
$C_3 = 2$	0,.1	0,1
$C_4 = 3$	0,05	0,3
$C_5 = 4$	0,05	0,05

Вычислительный эксперимент

Цель вычислительного эксперимента сравнить работу базового алгоритма SVM и предложенного в работе алгоритма. Алгоритмы сравнивались на реальных данных Яндекса [6], а также на синтетической выборке объектов трех классов, обладающей свойством линейной разделимости.

В качестве синтетической выборки была взята линейно разделимая выборка объектов, принадлежащих трем классам. В выборке было 2700 объектов, по 900 объектов каждого класса (см. рис. 1).

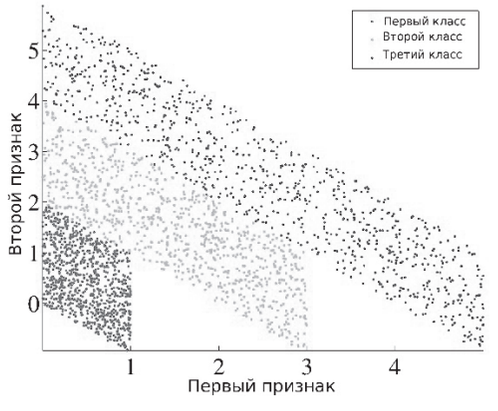
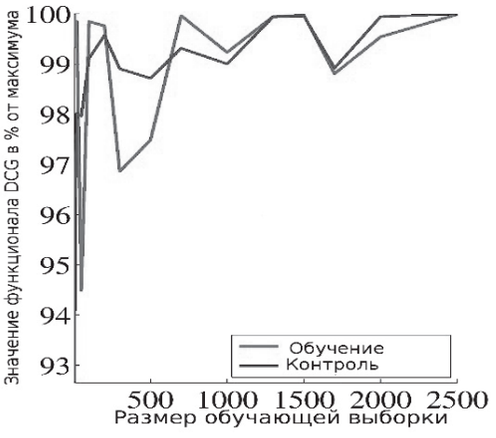


Рис. 1. Синтетическая выборка, три класса описаны точками в пространстве двух признаков

В вычислительном эксперименте строилась зависимость функционала качества Q_2 (19) от числа элементов в обучающей выборке. Разбиение на обучающую и тестовую выборки осуществлялось случайно. Для каждого размера обучающей выборки проводилось 10 экспериментов. Полученные значения Q_2 усреднялись. На рис. 2 приведена зависимость функционала качества Q_2 на обучении и тесте в зависимости от размеров обучающей выборки. В терминах функционала Q_2 предложенный алгоритм оказывается более предпочтительным, поскольку в отличие от SVM при размере выборки в 700 элементов значение Q_2 и на обучении, и на контроле равно максимально возможному. Более того, для нахождения весов признаков с помощью алгоритма SVM необходимо решить задачу минимизации для двойственных переменных, количество которых равно числу объектов в обучении, что при большой обучающей выборке весьма затратно.



а)

б)

Рис. 2. Зависимость функционала Q_2 от размера обучающей выборки: а) для базового алгоритма; б) для предложенного алгоритма

Оптимизация же функции $E(\mathbf{w}_1,\ldots,\mathbf{w}_K)$ в логистической регрессии происходит в пространстве заметно меньшей размерности $d = Km$. Для демонстрации работы пошагового алгоритма отбора объектов и признаков к рассматриваемой синтетической выборке было добавлено 10 шумовых признаков. В качестве множества B_1 был взят случайный набор из 150 объектов, по 50 объектов каждого класса. В качестве множества B_2 был взят случайный набор из 1000 объектов выборки.

Использовались следующие параметры алгоритма: $D_1 = D_2 = 2$; $l_1 = l_2 = 0$; $r_1 = r_2 = 0{,}04$. Сравнивались два алгоритма: отбор только объектов и отбор объектов и признаков. В обоих случаях все шумовые признаки были отфильтрованы. При отборе и объектов, и признаков было отобрано 97 из 150 объектов. Число ошибок классификации на всей выборке в случае с отбором только объектов составило 32, для алгоритма отбора и объектов, и признаков – 13. Полученные результаты говорят о применимости приведенного алгоритма совместного отбора объектов и признаков для задач, где число объектов не слишком велико.

Реальные данные представляют собой выборку объемом 97290 объектов, которые относятся к пяти линейно-упорядоченным классам {0; 1; 2; 3; 4}. Объекты представляют собой выдачи Яндекса на поисковые запросы. Все признаки нормированы на отрезок [0; 1], номера классов соответствуют релевантности полученной выдачи соответствующему запросу. Общее число признаков равно 245.

Таблица 2. Сравнение логистической регрессии и базового алгоритма SVM

Алгоритм	Значение Q_2
SVM	3,520
Логистическая регрессия	3,639

Подготовка данных. Особенностью представленной выборки является малое число объектов классов 3 и 4, менее 3% объектов каждого из классов и наличие почти постоянных признаков. Для устранения мультиколлинеарности воспользуемся методом главных компонент [11]. В данной работе были взяты 63 главные компоненты из условия $\lambda / \lambda_{max} > \beta = 3 \cdot 10^{-3}$, где λ_{max} – максимальное собственное число матрицы $\mathbf{X}^{\mathrm{T}}\mathbf{X}$; λ – собственное число, соответствующее рассматриваемой главной компоненте. Кроме того, с учетом малого числа объектов классов 3 и 4 обучающая выборка была дополнительно сбалансирована и содержала примерно одинаковое количество объектов из каждого класса.

Сравнение логистической регрессии и SVM. Приведем значения функционала Q_2 при классификации объектов с помощью многоклассовой логистической регрессии и с помощью базового алгоритма SVM (см. таблицу 2).

Алгоритм логистической регрессии оказывается более предпочтительным в терминах функционала Q_2 (18).

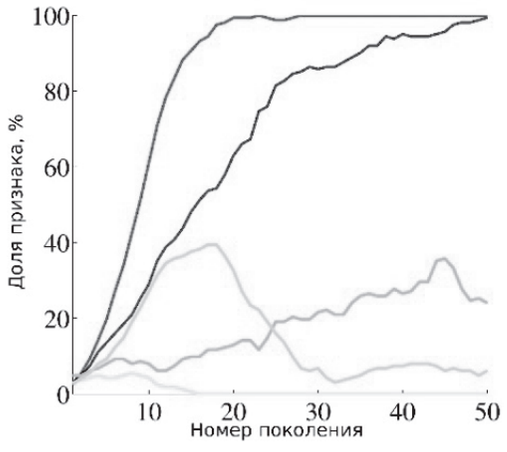


Рис. 3. Зависимость доли объектов, обладающих рассматриваемым признаком, от номера итерации

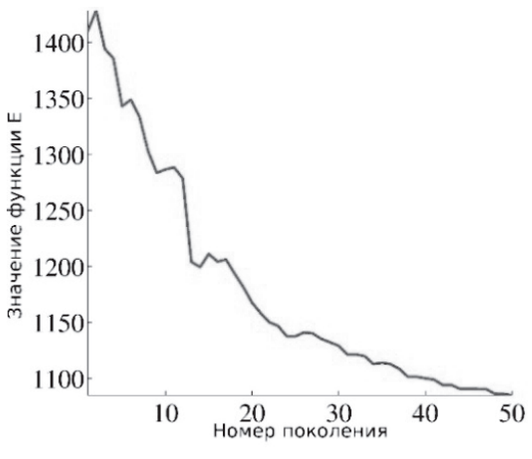


Рис. 4. Зависимость минимального значения функции потерь от номера итерации

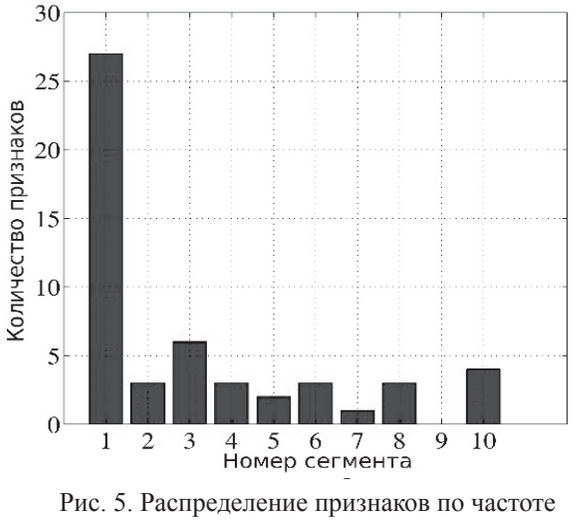


Рис. 5. Распределение признаков по частоте встречаемости в наборе

Отбор объектов и признаков. В рассматриваемой задаче отбор объектов в силу их большого числа затратен, а потому не проводился. Проводился лишь отбор признаков. Наборы признаков, отбираемые двумя предложенными в работе алгоритмами, заметно пересекаются, однако при рассматривавшихся параметрах алгоритма шагового отбора он отбирает меньше признаков. Число отобранных признаков и качество классификации в терминах Q_2 (19) приведено в таблице 3.

Для генетического алгоритма графики зависимости доли объектов с наличием признака в множестве V приведены на рис. 3. Кривая зависимости минимального на V значения функции потерь E (6) от номера итерации показана на рис. 4. Распределение признаков по доли объектов, им обладающих, после 50 итераций генетического алгоритма иллюстрирует рис. 5.

Далее по отобранным наборам признаков решалась задача нахождения векторов весов $\mathbf{w}_1,\ldots,\mathbf{w}_K$ в соответствии с (6).Затем применялась предложенная модификация алгоритма логистической регрессии.

Результаты. Таблица 2 демонстрирует полученные результаты в терминах Q_2 (18). Достигнутое значение $Q_2 = 4{,}058$ превосходит результат базового уровня, полученный Яндексом [6]. Это позволяет говорить о применимости предложенного алгоритма для ранжирования документов. Для сравнения качества с существующими алгоритмами был использован пакет SVM^{light} в режиме построения регрессии. Полученное значение функционала качества DCG равно 4,234 при обучении по всей обучающей выборке. Это на 4,5% выше, чем получено предложенным алгоритмом, потому предложенный алгоритм можно, по-видимому, улучшить еще.

Таблица 3. Сравнение качества Q_2 для двух алгоритмов отбора признаков

Алгоритм отбора	Число признаков	Q_2 для лог. регрессии	Q_2 для модиф. лог. регрессии
Пошаговый	12	3,612	4,028
Генетический	18	3,639	4,058

Заключение

В данной работе рассматривалась задача ранжирования коллекции документов поисковой выдачи. Для ее решения предложен алгоритм многоклассовой логистической регрессии. Представлена его модификация, которая по результатам вычислительного эксперимента значительно улучшает качество ранжирования в терминах функционала *DCG*. Также рассмотрен алгоритм пошагового отбора объектов и признаков. По качеству ранжирования на отобранных признаках он не уступает рассматривавшемуся в работе генетическому алгоритму.

Литература

1. Bishop C. M. Pattern recognition and machine learning. New-York: Springer, 2006, 738 p.
2. Bishop C. M., Nasrabadi N. M. Pattern recognition and machine learning // Journal of electronic imaging, 2007. Vol. 16. No. 4. Pp.1-2.
3. Joachims T. Optimizing search engines using clickthrough data. // Proceedings of the eighth ACM SIGKDD international conference on knowledge discovery and data mining. New York: ACM, 2002. Pp. 133-142.
4. Hastie T., Tibshirani R., Friedman J. The elements of statistical learning: data mining, inference and prediction. // Berlin: Springer, 2001, 745 p.
5. Friedman J., Hastie T., Tibshirani R. Additive logistic regression: a statistical view of boosting. // The annals of statistics, 2000. Vol. 28. No. 2. Pp. 337-374.
6. Интернет-математика 2009: задача и данные. URL: http://imat2009.yandex.ru. Дата обращения: 11.03.2009.
7. Kumar M. A., Gopal M. Fast multiclass SVM classification using decision tree based on one-against-all method. // Neural processing letters, 2010. Vol. 32. No. 3. Pp. 311-323.
8. Jarvelin K., Kekalainen J. IR evaluation methods for retrieving highly relevant documents. // In proceedings of the 23rd annual international ACM SIGIR conference and development in information retrieval, 2000. Pp. 41-48.
9. Oh I. S., Lee J. S., Moon B. R. Hybrid genetic algorithms for feature selection. // IEEE transactions on pattern analysis and machine intelligence, 2004. Vol. 26. No. 11. Pp. 1424–1437.
10. Leardi R., Boggia R., Terrile M. Genetic algorithms as a strategy for feature selection. // Journal of chemometrics, 1992. Vol. 6. No. 5. Pp. 267–281.
11. Jolliffe I. T. Principle Component Analysis. // New York: Springer, 2002, 487 p.

JOINT FEATURE AND OBJECT SELECTION IN MULTICLASS CLASSIFICATION OF DOCUMENTS´ COLLECTION

Aduenko A.A., Strijov V.V.

The article is dedicated to the problem of search engine results ranking. The algorithm of multiclass classification with joint selection of features and objects is proposed. It is modified for interclass relevance comparison. Features and objects selection is performed with stepwise regression and with genetic algorithm. Results obtained using both algorithms are compared. Proposed multiclass classification algorithm is tested on synthetic data and on data of Yandex search engine results.

Keywords: *multiclass classification, search engine results ranking, logistic regression, feature selection, object filtering, relevance.*

Адуенко Александр Александрович, студент Московского физико-технического института. Тел. (8-499) 135-41-63. E-mail: aduenko1@gmail.com

Стрижов Вадим Викторович, к.ф.-м.н., доцент, научный сотрудник Вычислительного центра РАН. Тел. (8-499) 135-41-63. E-mail: strijov@ccas.ru

НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 681.518: 339.13

СРАВНИТЕЛЬНАЯ ЭФФЕКТИВНОСТЬ МЕТОДОВ И СРЕДСТВ ИНФОРМАЦИОННОЙ ПОДДЕРЖКИ УПРАВЛЕНЧЕСКИХ РЕШЕНИЙ

Ануфриев Д.П., Димов Э.М., Маслов О.Н., Халимов Р.Р.

В статье анализируется эффективность разных способов реализации метода статистического имитационного моделирования (СИМ) в интересах управления сложными системами (СС).

Ключевые слова: сложные организационно-технические системы, эффективность управления; поддержка управленческих решений; метод статистического имитационного моделирования, варианты реализации.

Введение

Повышенный интерес к информационной технологии статистического имитационного моделирования (СИМ) [1-2] в настоящее время связан с преодолением главных препятствий на пути ее практического применения: длительности, трудоемкости и высокой стоимости процесса разработки СИМ-моделей. До сих пор основной проблемой считалось создание компьютерной программы, реализующей СИМ-модель, а не разработка и обеспечение эффективности применения (точности и адекватности) собственно модели. Резкому сокращению времени создания СИМ-моделей способствуют рост вычислительных возможностей современных ЭВМ и появление высокоэффективных специализированных программных продуктов (GPSS World; AnyLogic и др.). Соответственно обновилась идеология СИМ – от методов управления представлением информации при помощи иерархии абстракций наблюдается переход к интеллектуализации интерфейса пользователя с визуализацией данных путем анимации и динамической графики. В результате сегодня «не нужно быть ни профессиональным программистом, ни профессиональным математиком, чтобы разрабатывать имитационные модели» [3], и центр тяжести моделирования переходит с задач программирования на проблему создания СИМ-модели как таковой.

Разработка СИМ-моделей в интересах управления иерархическими и многокритериальными сложными системами (СС) организационно-технического типа (социально-экономическими, экологическими, военными и т.п.), неотъемлемыми компонентами которых являются лица, принимающие решения (далее ЛПР), до настоящего времени остается актуальной проблемой, которая имеет важное практическое значение. Анализ и синтез систем управления (СУ) здесь встречает ряд принципиальных трудностей: сложно, например, оптимизировать режим работы СС, компоненты которой (подсистемы и элементы) имеют возможность самостоятельно максимизировать свои функционалы. Для решения такого