УДК - 621.391.396

ОБЗОР ОСНОВНЫХ МЕТОДОВ КЛАССИФИКАЦИИ И КЛАСТЕРИЗАЦИИ ДАННЫХ

Д. С.Черезов, Н. А. Тюкачев

Воронежский государственный университет

Поступила в редакцию 10.10.2009

Аннотация. в данной статье рассматриваются алгоритма классификации и кластеризации информации. Описаны достоинства и недостатки алгоритмов и возможные решаемые задачи.

Ключевые слова: Дерево решений, метот Байеса, метот опорных векторов, метод к-средних, иерархическая кластеризация, ЕМ-алгоритм.

Annotation. The paper deals with algorithms of data clasterization and classification. Described are advantages and desadvantages of algorithms and common solved problem. for avery algorithm shown time of work.

The keywords: decision tree, Bayes method, support vector machine, k-means, hierarchial clastering.

ВВЕДЕНИЕ

С ростом информации, обрабатываемой, хранимой и получаемой в ходе работы информационных процессов, возникающих у крупных и средних предприятий, а так же научной деятельности, ее обработка, в полученном виде, становится затруднительна. Возникает необходимость первоначальной обработки информации для ее структурирования, выделения характерных признаков, обобщения, сортировки.

Для этого применяют процессы классификации и кластеризации, позволяющие в полной мере производить требуемую обработку информации, для ее последующего анализа специалистом.

1. КЛАССИФИКАЦИЯ

Классифицирование документов — процесс упорядочения или распределения объектов (наблюдений) по классам с целью отражения отношений между ними[5].

Класс — это множество документов, имеющих определенный общий признак, отличающий эту совокупность от других объектов.

В качестве классификационного деления можно принять различные признаки, зависящие от цели классификации. В основу класса всегда кладут наиболее важный признак документа, отвечающий цели классификации.

Классифицировать объект — значит, указать номер (или наименование) класса, к которому относится данный объект. Классификация объекта — номер или наименование класса, выдаваемый алгоритмом классификации в результате его применения к данному конкретному объекту.

Обучение классификатора — процесс построения алгоритма в случае, когда задано конечное множество объектов, для которых известно, к каким классам они относятся. Это множество называется выборкой. Классовая принадлежность остальных объектов не известна.

2. КЛАСТЕРИЗАЦИЯ

Кластеризация — процесс разбиения заданной выборки объектов (наблюдений) на непересекающиеся подмножества, называемые кластерами, так, чтобы каждый кластер состоял из схожих объектов, а объекты разных кластеров существенно отличались.

Одной из целью кластеризации является понимание данных путём выявления кластерной структуры. Разбиение наблюдений на группы схожих объектов позволяет упростить дальнейшую обработку данных и принятия решений, применяя к каждому кластеру свой метод анализа (стратегия «разделяй и властвуй»).

Кластеризация применяется при необходимости сжатия данных. Если исходная выборка избыточно большая, то можно сократить её, оставив по одному наиболее типичному представителю от каждого кластера.

Так же кластеризация служит для обнаружение новизны. Выделяются нетипичные объекты, которые не удаётся присоединить ни к одному из кластеров.

Для решения задач методами кластерного анализа, необходимо задавать количество кластеров заранее. В первом случае число кластеров стараются сделать поменьше. Во втором случае важнее обеспечить высокую степень сходства объектов внутри каждого кластера, а кластеров может быть сколько угодно. В третьем случае наибольший интерес представляют отдельные объекты, не вписывающиеся ни в один из кластеров[5].

3. УСЛОВНЫЕ ОБОЗНАЧЕНИЯ

Основные условные обозначения, которые приняты в работе: R^d — d-мерное пространство вещественных чисел. $X = (x_0, x_1, .., x_n)$ — множество n точек, где i-я точка является вектором, представленным в виде x_i, j-й компонент которой представляется в виде x_{ij}. Y — n-мерное множество меток классов или кластеров, в зависимости от типа алгоритма, где y_i метке соответствует точка x_i. k — количество кластеров.

4. ДЕРЕВО РЕШЕНИЙ

Описываемый метод служит для решения задач регрессии и классификации. В процессе построения модели, алгоритм, вычисляет степень влияния каждого входного атрибута на значение выходного атрибута и использует атрибут, влияющий на выходной атрибут в наибольшей степени, для разбиения узла дерева решений.

В качестве дерева понимают пару $(\{V\}, \{E\})$, где $\{V\}$ — непустое множество объектов некоторой природы, называемых вершинами, а $\{E\}$ — подмножество двухэлементарных подмножеств множества $\{V\}$ и C — условие перехода от одной вершины к другой[2].

Именно благодаря условиям перехода происходит выделение значимых атрибутов. Узел верхнего уровня описывает распределение значений выходного атрибута по всей совокупности данных. Каждый последующий уровень описывается распределением выходного атрибута при соблюдении условий на входные атрибуты, соответствующие этому узлу. Разбиение узлов происходит до тех пор, пока не будет достигнута пороговая вероятность того, что выходное значение будет принимать требуемое значение.

В общем случае условие выполнение условий на i-м уровне можно представить в следующем виде:

$$C_i = (Q_{11} \vee Q_{12} \vee ... \vee Q_{1k}) \wedge \\ \wedge (Q_{21} \vee Q_{22} \vee ... \vee Q_{2k}) \wedge ... \\ \wedge (Q_{i1} \vee Q_{i2} \vee ... \vee Q_{ik}), \quad (1)$$

где Q_{ik} — логическое требуемое условие, i — уровень узла, k — количество условий.

Если конъюнкцию, в формуле 1, заменить на функцию произведения, а дизъюнкцию на функцию max, тогда:

$$C_i = \prod_{h=1}^{i} \max\{Q_{hl}\}_{l=1}^{k}. \quad (2)$$

Условие перехода может принимать различные формы, проверки равенства значений параметров до применения условий, применяемых в других алгоритмах. Это делает этот метод очень гибким и эффективным, но сложным для построения [3].

Достоинства: в общем случае обеспечивает высокую скорость работы. Может применяться как для задач классификации, так и для задач кластеризации.

Недостатки: сложная структура. Для построения дерева решений или для внесения изменений, требуется помощь специалиста в рассматриваемой области.

Время работы алгоритма:

$$T(n) = \Theta\left(n - \frac{n}{p}\right), \quad (3)$$

где n — количество условий, p — среднее количество переходов.

5. МЕТОД БАЙЕСА

Для каждого наблюдения их множества X высчитывается вероятность его возникновения в соответствующем классе по формуле 4:

$$P(x_i \mid y_i) = \frac{count(x_i \mid y_i)}{count(X \mid y_i)}, \quad (4)$$

где $count(x_i \mid y_i)$ — количество элементов x_i в категории y_i. $count(X \mid y_i)$ — количество наблюдений из множества X принадлежащих категории y_i.

Для того что бы учитывать категории с небольшим количеством наблюдений используется нормализация. Тогда вероятность вычисляется по формуле 5:

$$P(x_i \mid y_i) = \frac{\frac{count(x_i \mid y_i)}{count(X \mid y_i)}}{\sum_j^n \frac{count(x_i \mid y_n)}{count(X \mid y_n)}}. \quad (5)$$

При вычислении вероятности требуется так же априорная вероятность возникновения категории y_i равна отношению числа наблюдений в y_i к общему числу наблюдений.

$$P(y_i) = \frac{count(X \mid y_i)}{count(X)}, \quad (6)$$

где $count(X \mid y_i)$ — число наблюдений принадлежащих категории y_i, $count(X)$ — общее число наблюдений.

Классификация набора[4] наблюдений происходит после вычисления вероятности по формуле 7:

$$P(X \mid y_i) = P(y_i)\prod_j^n p(x_j \mid y_i), \quad (7)$$

где $P(y_i)$ — априорная вероятность возникновения y_i, $p(x_j \mid y_i)$ — вероятность возникновения наблюдения x_j в категории y_i.

Достоинства: высокая скорость работы алгоритма как на этапе обучения так и на этапе анализа новых данных

Недостатки: в случае $x` \notin X$ то невозможно классифицировать объект.

Время работы алгоритма:

$$T(n) = \Theta(nk), \quad (8)$$

где n — размер множества X, k — размер множества Y.

6. АЛГОРИТМ SVM

Наиболее эффективным показал себя метод опорных векторов(support vector machine), в основе которого лежит идея разделения облаков данных гиперплоскостями, находящихся на максимальном расстоянии от облаков[6].

Гиперплоскость может быть записана как точки из множества X, удовлетворяющие условие:

$$W \times X - b = 0, \quad (9)$$

где W — вектор, нормаль к гиперплоскости, b — смещение вектора W.

Если $y_i \in \{1, -1\}$ тогда условие можно записать в следующем виде:

$$y_i(W \times X - b) \geq 1, \quad (10)$$

$$y_i(W \times X - b) \leq 1. \quad (11)$$

В случае если $y_i = +1$ то выполняется условие (10), иначе, если $y_i = -1$ то, будет выполняться условие (11). Оба случая можно объединить в одно условие:

$$y_i(x \times w + b) - 1 \geq 0. \quad (12)$$

Неравенство (4) можно записать в виде уравнения Лагранжа:

$$L \equiv \frac{1}{2}\|W\|^2 - \\ -\sum_{i=1}^n \alpha_i y_i(x_i \times w + b) + \sum_{i=1}^n \alpha_i, \quad (13)$$

где α_i — дополнительный коэффициент.

Коэффициент α_i — возникает в случае представления обучаемой информации в виде функции:

$$x_i \mapsto f(x, \alpha). \quad (14)$$

Таким образом, в качестве обучения понимается минимизация функции (14) по параметрам w, b и так же с условием что $\alpha_i \geq 0$ [6].

Достоинства: за счет применения гиперплоскостей работает при малых объемах обучающей выборки. За счет использования ядра описывающего связь между элементами выборки, можно использовать гиперплоскости разной сложности.

Недостатки: большое время обучение, необходимость ядра для конкретного случая.

Время работы алгоритма:

$$T(n) = \Theta(n^3), \quad (15)$$

где n — размер множества X.

7. МЕТОД К-СРЕДНИХ

Метод К-средних основан на разделении множества наблюдений X на k кластеров, которые локально минимизированы относительно расстояния между информационной точкой и центройдом кластера. Целевая функция алгоритма представлена формулой 13:

$$J = \sum_{h=1}^k \sum_{x_i \in X_h} \|x_i - \mu_h\|^2, \quad (16)$$

где μ_h — значение центройда.

Целевая функция является локально минимизированной для каждого кластера, что означает, что каждая точка из набора данных находиться на минимальном расстоянии от центройда кластера, к которому она относиться[1]. Выбор соответствующего кластера h для заданной точки x_i в процессе работы алгоритма обусловлено неравенством 17:

$$\|x_i - \mu_h\|^2 = \min\{\|x_i - \mu_h\|^2\}_{h=1}^k. \quad (17)$$

Информационной точке будет присвоен тот кластер, центройд которого лежит наиболее близко к рассматриваемой точке [1].

Под обучением алгоритма понимается нахождение центройдов кластеров при помощи формулы 18:

$$\mu_h = \frac{1}{|X_h|} \sum_{x \in X_h} x. \qquad (18)$$

После чего оптимизация центройдов происходит при помощи поочередного вычисления неравенства 17 и формулы 18. Необходимо отметить, что нахождение глобального оптимального решения для алгоритма К-средних является NP-полной проблемой.

Если предположить, что множество центройдов кластеров $\{\mu_h\}_{h=1}^{k}$ принадлежит множеству X, тогда задача кластеризации называется К-усредненной.

Достоинства: можно найти кластер для данных после нахождения центройдов.

Недостатки: результат работы зависит от критериев, выбранных специалистом.

Время работы алгоритма

$$T(n) = \Theta(n^{kd}), \qquad (19)$$

где n — размер множества X, k — количество кластеров, d — размерность x_i.

8. ИЕРАРХИЧЕСКАЯ КЛАСТЕРИЗАЦИЯ

Основная цель алгоритма иерархической кластеризации заключается в построении структуры кластеров. Можно выделить два вида алгоритма:

- Объединяющий.
- Разделяющий.

Расстояние между кластерами принимают за меру, используемую для определения, какие кластеры должны быть объединены, а какие кластеры требуется разделить [2, 3].

В случае объединяющего иерархического алгоритма каждый элемент из множества наблюдений принимается за отдельный кластер и на каждом шаге работы алгоритма наиболее близкие пары элементов объединяются в один кластер. Условие объединения кластеров представлено формулой 20:

$$D = \min(dist(a,b)), \qquad (20)$$

где a, b принадлежит X.

Разделяющий иерархический алгоритм в начале работы объединяет все наблюдения из множества X в один кластер и на каждом шаге работы отделяет максимально отдаленные пары наблюдений. Условие разделения кластера представлено формулой 21:

$$D = \max(dist(a,b)), \qquad (21)$$

где a, b принадлежит X.

В результате работы алгоритма получается структура кластеров, представляющая из себя граф. Работа алгоритма завершается для обоих случаев при достижении требуемого количества кластеров [2].

Достоинства: высокая скорость работы, простота реализации, возможность просмотреть результаты работу на каждом ходу.

Недостатки: нельзя на прямую управлять количеством кластеров. Результат зависит от критериев выбранных специалистом.

Время работы алгоритма

$$T(n) = \Theta(n^2), \qquad (22)$$

где n — размер множества X.

9. EM-КЛАСТЕРИЗАЦИЯ

Среди неиерархических алгоритмов, не основанных на расстоянии, следует выделить EM-алгоритм (Expectation-Maximization). В нем вместо центров кластеров предполагается наличие функции плотности вероятности для каждого кластера с соответствующим значением математического ожидания и дисперсией. Перед стартом алгоритма выдвигается гипотеза о виде распределений, которые оценить в общей совокупности данных сложно [5].

EM алгоритм является основным методом поиска оценки максимального правдоподобия параметра, лежащего в основе распределений из множества заданных данных. Полагается, что все переменные независимы, и все данные имеют k совместных распределений. Основной алгоритм разделен на два шага.

E-алгоритм (шаг ожидания):

$$z_{ij} = \frac{p(x_j \mid y_i)p(y_i)}{p(x_j)}, \qquad (23)$$

где $p(x_j \mid y_i)$ определяет вероятность возникновения x_i в кластере y_i, $p(y_i)$ определяет вероятность возникновения y_i, $p(x_j)$ определяет вероятность возникновения x_i.

M-алгоритм (шаг максимизации):

$$u_i = \frac{\sum_{j}^{n} z_{ij} x_i}{\sum_{j}^{n} z_{ij}}, \qquad (24)$$

$$\sigma_i^2 = \frac{\sum_j^n z_{ij}(x_j - u_i)^2}{\sum_j^n z_{ij}}, \qquad (25)$$

где u_i среднее распределения I. σ_i^2 дисперсия распределения i. z_{ij} вычисленный вес (вероятность) наблюдения j, принадлежащего кластеру i.

Если оцененное сверху значение правдоподобия меньше указанного порогового значения или число итераций равно максимальному числу, то работа алгоритма прекращается и получается окончательная кластеризация [4].

Значение правдоподобия для данного алгоритма выражается функцией 26:

$$L = \sum_j^n \log \sum_i^k p(x_j \mid c_i) p(c_i). \qquad (26)$$

Достоинства: не требует выделения специалистом метрик, есть возможность использовать совместно с другими алгоритмами обработки данных. Может работать на малых объемах данных.

Недостатки: результат работы зависит от первоначального вида распределения.

Время работы алгоритма

$$T(n) = \Theta(n^2), \qquad (27)$$

где n — размер множества.

ЗАКЛЮЧЕНИЕ

Были представлены различные методы машинного обучения, позволяющие классифицировать информацию, как на основании имеющихся прецедентов, так и при помощи специалистов. Также рассмотрены алгоритмы кластеризации, основанные на структурном, метрическом и вероятностных подхода.

Так, в виду сложности построения, алгоритм классификации дерево решений, применяется в основном в экономических предприятиях, например для определения наиболее предпочтительного варианта, выдача кредитов, и при определении семантического значения слова.

Метод Байеса применяется при наличии большого числа объектов в обучающей выборке, для вычисления вероятности возникновения объектов наиболее точно.

Метод опорных векторов применяется в экономике для вычисления регрессии разных значений и дальнейшего прогнозирования. В качестве классификатора используется в информационно-поисковых системах.

Метод К-средних используется для выделения групп объектов в экономике, при анализе данных, а так же в информационно-поисковых системах.

Метод иерархической кластеризации используется при сборе статистических данных и реализован в статистических пакетах. Так же используется при кластеризации текстовых документов.

ЕМ-алгоритм применятся в информационно-поисковых системах для кластеризации большого объема данных.

СПИСОК ЛИТЕРАТУРЫ

1. *Sugato B.* Semi-supervised Clustering: Probabilistic Models, Algorithms and Experiments http://www.cs.utexas.edu/users/sugato/papers/sugato-phdthesis.pdf

2. *Касьянов В.Н.* Графы в программировании: обработка, визуализация и применение/ В.Н. Касьянов, В.А. Евстигнеев — СПб.: БХВ-Петербург, 2003. — 1104 с.

3. *Tan P.J.* MML inference of decision graph with milti-way and dynamic attributes / P. J. Tan, D. L. Dowe http://www.csse.monash.edu.au/~dld/Publications/2003/Tan+Dowe2003_MMLDecisionGraphs.pdf.

4. *Eibe F.* Data mining. Practical machine learning tools and techniques. F.Eibe, I.Witter. — 2005. — 525 с.

5. *Sholom M.W.* Text minig. Predictive methods of analyzing unstructured information. M. W. Sholom, N.Indurkhya, T.Zhang, F.J.Damarau. — 2004. — 236 с.

6. *Burges C.* A tutorial on support vector machines for pattern recognition. http://research.microsoft.com/en-us/ um/people/cburges/papers/SVMTutorial.pdf.

Черезов Дмитрий Сергеевич — аспирант кафедры Программирования и информационных технологий, Воронежский государственный университет. Тел. 208-470. E-mail: Dmitry.cherezov@gmail.com

Тюкачев Николай Аркадиевич — к.ф.-м.н., доц. кафедры Программирования и информационных технологий, Воронежский государственный университет. Тел. 208-470. E-mail: nik@cs.vsu.ru

Cherezov D.S. — Post-Graduate Student, the dept. of Programming and Informational Technologies, Voronezh State University. Tel.: (4732) 208-470, E-mail: Dmitry.cherezov@gmail.com

Tukachev N.A. — candidat of physics-math. Science, Associate Proffessor, the dept. of the Programming and Information Technologies, Voronezh State University. Tel. (4732) 208-470. E-mail: nik@cs.vsu.ru