

Реляционный подход к извлечению знаний и его применения

Демин А. В.¹, Витяев Е. Е.²

¹Институт систем информатики СО РАН, пр. Лаврентьева, д. 6, г. Новосибирск, 630090, Россия.

²Институт математики СО РАН, пр. ак. Коптюга, д. 4, г. Новосибирск, 630090, Россия.

alexandredemin@yandex.ru, vityaev@math.nsc.ru

Аннотация. В работе рассматривается метод и программная система «Discovery» извлечения знаний из данных. Предложенный метод использует язык логики первого порядка и позволяет обнаруживать на данных произвольные виды закономерностей. Особенностью системы является возможность произвольно задавать класс обнаруживаемых гипотез и находить в данных все закономерности заданного класса. Рассматривается применение разработанной системы для решения ряда актуальных прикладных задач в медицине и биоинформатике.

Ключевые слова: обнаружение закономерностей, извлечение знаний, анализ данных

1 Введение

В настоящий момент разработано достаточно большое количество различных KDD&DM-методов (Knowledge Discovery in Data Bases and Data Mining) и реализующих их программных систем. Данное направление продолжает бурно развиваться и совершенствоваться.

Анализ методов KDD&DM показывает [1], что для любого метода можно выделить типы данных, с которыми работает метод, язык оперирования и интерпретации данных, а также класс гипотез, которые проверяет метод. Это накладывает на KDD&DM-методы ряд ограничений.

1. Информация, содержащаяся в данных, определяется множеством отношений и операций, интерпретируемых в онтологии предметной области. Существующие методы KDD&DM могут работать только с конкретными типами данных и использовать только конкретные виды отношений и операций. Тем самым они, во-первых, не могут использовать всю информацию, содержащуюся в данных, во-вторых, могут получать результаты, не интерпретируемые в онтологии предметной области.
2. Методы обнаруживают в данных только вполне определенные типы гипотез.

Нами был разработан реляционный подход (Relational Data Mining) к методам извлечения знаний и реализующая его программная система «Discovery» [2], снимающие эти ограничения с методов KDD&DM за счет использования языка первого порядка, который практически неограниченно расширяет множество типов используемых данных, а также позволяет описывать разнообразные виды гипотез. Проведены сравнения системы «Discovery» с такими широко распространенными методами, как нейронные сети, решающие деревья, ассоциативные правила, статистические методы, FOIL. В них система «Discovery» показала лучшие результаты [1-3].

Существующие методы не в состоянии поддерживать режим исследования данных, когда обнаруживаемая закономерность заранее неизвестна. Каждый KDD&DM-метод обнаруживает свой специфический класс гипотез. Система «Discovery» способна поддерживать режим исследования данных в интерактивном режиме. Кроме того, она может обнаружить и проверить на данных произвольный класс гипотез, который захочет проверить эксперт.

Система «Discovery» обнаруживает гипотезы, которые сформулированы в заданных экспер-

том (например, врачом) терминах – множестве интерпретируемых отношений и операций, определенных на данных. Интерпретируемость получаемых закономерностей очень важна при принятии ответственных решений в таких областях, как медицина или финансы. К примеру, если речь идет о диагностике заболевания и у нас есть два прогноза, полученные нейронными сетями и системой «Discovery», то доверие будет к тому прогнозу, который понятен и интерпретируем. Невозможно принимать ответственные решения, не понимая, как они получены. Прогнозы, получаемые на основании интерпретируемых правил понятны, и по ним можно принимать решения.

Другой важной задачей, которую решает система «Discovery», является задача наиболее полного извлечения знаний из данных. Полнота извлечения знаний системой «Discovery» обеспечивается двумя путями:

1. использованием теории измерений, позволяющей извлечь практически всю информацию из данных и представить ее множеством отношений и операций, определенных на много-сортной эмпирической системе и интерпретируемых в онтологии предметной области;
2. обнаружением практически любого класса гипотез в терминах выявленных отношений и операций на этой эмпирической системе.

Все эти задачи показывают актуальность разработки «универсальной» версии системы «Discovery». «Универсальность» системы состоит в том, что она снимает ограничения 1, 2 с KDD&DM-методов за счет:

1. предоставления пользователю возможности самому, в диалоге с системой, задавать отношения и операции, которые система будет использовать, и которые интерпретируемы в системе понятий предметной области, что позволяет извлекать из данных всю информацию в соответствии с пунктом 3;
2. возможности задания любого класса гипотез, формулируемого в заданных самим же пользователем множествах отношений и операций, что снимает ограничение 2 с KDD&DM-методов.

Именно такая «универсальность» позволяет решать упомянутые важные задачи исследования данных и проверки экспертных гипотез, которые не решаются другими методами.

На данный момент разработана достаточно «универсальная» версия системы «Discovery», позволяющая пользователю самому задавать класс обнаруживаемых закономерностей, извлекать из данных множество закономерностей заданного класса и использовать найденные закономерности для прогноза и принятия решений. В данной работе описывается метод извлечения знаний, на котором основана работа системы «Discovery», и применения разработанной системы для решения актуальных задач в медицине и биоинформатике.

2 Метод обнаружения закономерностей

2.1 Определение вида гипотез

Будем предполагать, что исходные данные представлены в виде реляционной таблицы D , строки которых соответствуют объектам, а колонки – признакам объектов, т. е. $D = \{D(1), \dots, D(N)\}$, где $D(i)$ – i -я строка таблицы (объект с номером i), $D(i) = \{D(i, 1), \dots, D(i, m)\}$, $D(i, j)$ – значение таблицы на пересечении j -й колонки и i -й строки (значение j -го признака объекта $D(i)$).

Введем иерархию элементов конструирования гипотез, которые мы будем использовать для формализации способа задания видов гипотез.

Переменная по объектам пробегает множества строк (объектов) таблицы данных. В дальнейшем будем отождествлять значения переменных по объектам с номерами строк.

Будем обозначать кортеж переменных по объектам $\langle i_1, i_2, \dots, i_n \rangle$ через $\langle i \rangle$.

Переменная-параметр (в дальнейшем будем называть этот тип переменных просто переменными) может либо принимать значение фиксированной константы $x = \text{const}$, где const – произвольное действительное число, либо принимать значения признаков объектов, в этом случае значение переменной-параметра будет определяться текущим набором значений пере-

менных по объектам $\langle i \rangle = \langle i_1, i_2, \dots, i_n \rangle$: $x = D(g(\langle i \rangle), h(\langle i \rangle))$, где $g(\langle i \rangle)$ и $h(\langle i \rangle)$ – целочисленные функции, задающие номер строки и номер колонки таблицы. В простейшем случае, когда $g(\langle i \rangle) = i_j$, где $i_j \in \langle i \rangle$, $h(\langle i \rangle) = k$, переменная $x\langle i \rangle$ будет просто принимать значения k-го признака объекта i_j : $x\langle i \rangle = D(i_j, k)$. В более сложном случае $g(\langle i \rangle)$ может, к примеру, задавать смещение строки: $g(\langle i \rangle) = g(i_j) + b$, где b – фиксированное смещение относительно строки i_j , $i_j \in \langle i \rangle$, или осуществлять поиск объекта относительно текущих объектов. В дальнейшем, чтобы отразить тот факт, что значение переменной-параметра может определяться текущим набором значений переменных по объектам $\langle i \rangle$, будем обозначать переменную-параметр x через $x(\langle i \rangle)$.

Терм определяется следующим образом: 1) если $x(\langle i \rangle)$ – произвольная переменная-параметр, то $t(\langle i \rangle) = x(\langle i \rangle)$ – терм; 2) если f – n-местная вещественнозначная функция, t_1, \dots, t_n – термы, то $t(\langle i \rangle) = f(t_1(\langle i \rangle), \dots, t_n(\langle i \rangle))$ – терм. Терм может принимать любое вещественное значение.

Предикат определяет отношение на множестве данных. Общий вид предиката: $P(\langle i \rangle) = P(t_1(\langle i \rangle), \dots, t_n(\langle i \rangle))$, где $t_j(\langle i \rangle)$ – терм. Предикат может принимать значение «истина» или «ложь».

Правило служит для представления закономерности. Правило состоит из посылки и заключения. Посылка правила представляет собой конъюнкцию предикатов, заключение – некоторый целевой предикат. Общий вид правила: $\forall \langle i \rangle P_1^e(\langle i \rangle) \& \dots \& P_n^e(\langle i \rangle) \rightarrow P_0^e(\langle i \rangle)$, где P_1, \dots, P_n – предикаты посылки; P_0 – целевой предикат; $\varepsilon \in \{0, 1\}$ – обозначает наличие отрицания, т. е. если $\varepsilon = 0$, то $P^\varepsilon = P$, если $\varepsilon = 1$, то $P^\varepsilon = \neg P$. Каждое правило R характеризуется условной вероятностью $p(R)$, с которой оно предсказывает истинность заключения при условии истинности посылки.

Введем понятие *интерпретации* объекта (переменной, терма или предиката) на множестве исходных данных. Будем обозначать интерпретацию объекта A через $\theta(A)$. Последовательно введем понятие интерпретации для каждого из вышеперечисленных объектов.

Под *интерпретацией переменной* будем понимать присвоение ей значения фиксированной константы либо присвоение значений признаков объектов. Иначе говоря, $\theta(x) = \text{const}$, где const – произвольное действительное число, либо $\theta(x) = D(g\langle i \rangle, h\langle i \rangle)$, $g(\langle i \rangle)$ и $h(\langle i \rangle)$ – целочисленные функции, задающие номер строки и номер столбца таблицы данных. Будем называть переменную проинтерпретированной, если она равна константе, либо ссылается на исходную таблицу данных.

Под *интерпретацией терма* будем понимать присвоение всем переменным, входящим в состав терма, значений фиксированных констант или признаков объектов. Другими словами, если $t = x$, где x – переменная, то $\theta(t) = \theta(x)$, если $t = f(t_1, \dots, t_n)$, где t_1, \dots, t_n – термы, то $\theta(t) = f(\theta(t_1), \dots, \theta(t_n))$.

По аналогии под *интерпретацией предиката* будем понимать присвоение всем его термам значений проинтерпретированных функций, т. е. $\theta(P(t_1, \dots, t_m)) = P(\theta(t_1), \dots, \theta(t_m)) = P(f_1, \dots, f_m)$. Соответственно, будем называть предикат проинтерпретированным, если все его термы проинтерпретированы.

Введем понятия *шаблона термов* и *шаблона предикатов*.

Шаблон термов задает терм и способы его интерпретации на исходных данных. Определим шаблон терма Tf для терма t следующим образом. Если $t = x$, где x – переменная, то $Tf = \langle x, \Theta(x) \rangle$, где $\Theta(x)$ – множество интерпретаций переменной x на исходных данных: $\Theta(x) = \{\theta_1(x), \dots, \theta_m(x)\}$. Если $t = f(t_1, \dots, t_n)$, где t_1, \dots, t_n – термы, то $Tf = \langle f(t_1, \dots, t_n), \Theta(t_1), \dots, \Theta(t_n) \rangle$, где $\Theta(t_i)$ – множество интерпретаций терма t_i на исходных

данных: $\Theta(t_i) = \{\theta_i(t_i), \dots, \theta_m(t_i)\}$. Поскольку способы интерпретации термов задаются шаблонами термов, то мы можем определить множество интерпретаций терма $\Theta(t)$ через другие шаблоны термов, т. е. $\Theta(t) = \{Tf_1, \dots, Tf_k\}$, где Tf_i – некоторые шаблоны термов.

Шаблон предикатов задает предикат и способы его интерпретации на исходных данных. Обозначим $\langle P(t_1, \dots, t_n), \Theta(t_1), \dots, \Theta(t_n) \rangle$ шаблон предикатов, где $\Theta(t_i)$ – множество интерпретаций терма t_i на исходных данных, которое задается шаблонами термов $\Theta(t_i) = \{Tf_1, \dots, Tf_k\}$, где Tf_i – некоторые шаблоны термов. Таким образом, шаблон предикатов, по сути, задает класс предикатов, определяющих один вид отношения, но по-разному проинтерпретированных на исходных данных.

Теперь, используя понятие шаблона предикатов, мы можем определить понятие *класса гипотез* как набор множества шаблонов предикатов и целевого предиката. Обозначим класс гипотез через $\langle \{Tp_1, \dots, Tp_m\}, P_0^\varepsilon \rangle$, где Tp_i – шаблоны предикатов, P_0 – целевой предикат, $\varepsilon \in \{0, 1\}$ обозначает наличие отрицания предиката.

Класс гипотез $\langle \{Tp_1, \dots, Tp_m\}, P_0^\varepsilon \rangle$ определяет класс правил вида $P_1^\varepsilon \& \dots \& P_m^\varepsilon \rightarrow P_0^\varepsilon$, где все предикаты посылки P_1, \dots, P_m должны принадлежать множеству предикатов, определяемому шаблонами Tp_1, \dots, Tp_m .

2.2 Алгоритм поиска вероятностных закономерностей

Не ограничивая общности, рассмотрим алгоритм поиска закономерностей только для случая, когда задан только один класс гипотез. Для случая, когда задано несколько классов гипотез, поиск закономерностей осуществляется независимо для каждого класса гипотез.

Пусть задан некоторый класс гипотез $Th = \langle \{Tp_1, \dots, Tp_m\}, P_0^\varepsilon \rangle$.

Пусть $\{P_1, \dots, P_m\}$ – множество всех проинтерпретированных предикатов, которые мы можем получить с помощью шаблонов $\{Tp_1, \dots, Tp_m\}$.

$U(Th) = \{A_1, \dots, A_m\}$ – множество всех литер вида $A_i = P_i^\varepsilon$, $P_i \in U(Th)$, $\varepsilon \in \{0, 1\}$ – обозначает наличие отрицания, т. е. если $\varepsilon = 0$, то $P_i^\varepsilon = P_i$, если $\varepsilon = 1$, то $P_i^\varepsilon = \neg P_i$.

$A_0 = P_0^\varepsilon$ – целевое высказывание.

Вероятностной закономерностью [1] будем называть правило $A_1 \& \dots \& A_m \rightarrow A_0$, удовлетворяющее следующим условиям:

- условная вероятность $p(A_0 | A_1 \& \dots \& A_m)$ правила определена, т. е. $p(A_1 \& \dots \& A_m) > 0$;
- условная вероятность $p(A_0 | A_1 \& \dots \& A_m)$ правила строго больше условных вероятностей каждого из его подправил, т.е. для любого правила $A_{i_1} \& \dots \& A_{i_k} \rightarrow A_0$, такого что $\{A_{i_1}, \dots, A_{i_k}\} \subset \{A_1, \dots, A_m\}$, условная вероятность $p(A_0 | A_{i_1} \& \dots \& A_{i_k}) < p(A_0 | A_1 \& \dots \& A_m)$.

Чтобы проверить при помощи обучающего множества D , является ли некоторое правило $A_1 \& \dots \& A_m \rightarrow A_0$ вероятностной закономерностью, необходимо проверить выполнимость вероятностных неравенств а и б, и оценить его статистическую значимость.

Условная вероятность правила $A_1 \& \dots \& A_m \rightarrow A_0$ оценивается на обучающем множестве D следующим образом: $p(A_0 | A_1 \& \dots \& A_m) = N(A_0 \& A_1 \& \dots \& A_m) / N(A_1 \& \dots \& A_m)$, где $N(A_0 \& A_1 \& \dots \& A_m)$ – число событий $A_0 \& A_1 \& \dots \& A_m$ на множестве D , $N(A_1 \& \dots \& A_m)$ – число событий $A_1 \& \dots \& A_m$ на D .

Для проверки статистической значимости правила используется статистический критерий Фишера (точный критерий независимости Фишера для таблиц сопряженности) [4]. Если прави-

ло удовлетворяет этому критерию с некоторым доверительным уровнем α , а также удовлетворяет условиям а и б, то оно будет являться вероятностной закономерностью.

Алгоритм поиска закономерностей основан на семантическом вероятностном выводе [1-2], который позволяет находить все статистически значимые вероятностные закономерности вида $A_1 \& \dots \& A_m \rightarrow A_0$.

Для дальнейшего описания введем несколько определений.

Длиной правила R будем называть величину $len(R)$, равную количеству литералов, входящих в посылку правила.

Правило $A_1 \& \dots \& A_m \& A_{m+1} \rightarrow A_0$ является уточнением правила $A_1 \& \dots \& A_m \rightarrow A_0$, если оно получено добавлением в посылку правила $A_1 \& \dots \& A_m \rightarrow A_0$ произвольной литеры A_{m+1} .

Будем обозначать $Spec(RUL)$ множество уточнений всех правил из RUL , где RUL – произвольное множество правил вида $A_1 \& \dots \& A_m \rightarrow A_0$, $A_i \in U(Th)$.

Опишем алгоритм поиска закономерностей, реализующий семантический вероятностный вывод.

На первом шаге генерируем множество RUL_1 всех правил единичной длины, имеющих вид $R = A_i \rightarrow A_0$, $A_i \in U(Th)$, $len(R) = 1$. Все правила из RUL_1 проходят проверку на выполнение условий для вероятностных закономерностей. Правила, прошедшие проверку, будут являться вероятностными закономерностями. Обозначим REG_1 множество всех вероятностных закономерностей, обнаруженных на первом шаге, т. е. $REG_1 = \{R_i\}$, где $i \in I_1$, $R_i = A_j \rightarrow A_0$, $A_j \in U(Th)$, $len(R_i) = 1$, R_i – вероятностная закономерность.

На шаге $k \leq d$ генерируется множество RUL_k всех уточнений правил, сгенерированных на предыдущем шаге, $RUL_k = Spec(RUL_{k-1})$. Все правила из RUL_k проходят проверку на выполнение условий для вероятностных закономерностей. Обозначим REG_k множество всех вероятностных закономерностей, обнаруженных на данном шаге, т. е. $REG_k = \{R_i\}$, где $i \in I_k$, $R_i = A_1 \& \dots \& A_k \rightarrow A_0$, $A_j \in U(Th)$, $len(R_i) = k$, R_i – вероятностная закономерность.

На шаге $k > d$ генерируется множество RUL_k всех уточнений всех вероятностных закономерностей, обнаруженных на предыдущем шаге, $RUL_k = Spec(REG_{k-1})$. Все правила из RUL_k проходят проверку на выполнение условий для вероятностных закономерностей. Обозначим RUL_k множество всех вероятностных закономерностей, обнаруженных на данном шаге, т. е. $REG_k = \{R_i\}$, где $i \in I_k$, $R_i = A_1 \& \dots \& A_k \rightarrow A_0$, $A_j \in U(Th)$, $len(R_i) = k$, R_i – вероятностная закономерность.

Алгоритм останавливается, когда невозможно далее уточнить ни одно правило, т. е. когда $RUL_k = Spec(REG_{k-1}) = REG_{k-1} = \emptyset$. Результирующее множество всех закономерностей REG будет равно объединению всех REG_i : $REG = \bigcup_i REG_i$.

Шаги алгоритма $k \leq d$ называются базовым перебором, а шаги $k > d$ – дополнительным перебором. Величина d называется глубиной базового перебора и является параметром алгоритма.

Проверка правил на выполнение условий для вероятностных закономерностей осуществляется путем проверки описанных выше статистических критериев с некоторым доверительным уровнем α .

2.3 Формирование прогноза и принятие решения

Будем предполагать, что исходная задача может быть представлена как задача выбора одного варианта исхода из заранее известного набора исходов. Под прогнозом будем понимать высказывание о варианте исхода с некоторой оценкой его точности, которую мы будем называть оценкой точности прогноза. В качестве оценки точности прогноза наиболее естественно ис-

пользовать оценку его вероятности, однако в некоторых задачах могут быть использованы и другие способы оценки точности прогноза, больше соответствующие специфике задачи. Опишем способ получения прогноза и механизм принятия решения, основанный на множестве правил, предсказывающих различные варианты исходов [5].

Пусть нам известно множество вариантов исхода $\{\text{исход}_1, \dots, \text{исход}_n\}$. Каждый вариант исхода исход_i можно представить некоторым целевым предикатом TP_i . Таким образом, будем считать, что нам задан некоторый набор целевых предикатов $\{TP_1, \dots, TP_n\}$, представляющих различные варианты исхода.

Пусть PR – множество правил, предсказывающих один и тот же целевой предикат (вариант исхода). Будем называть это множество правил *предиктором*. Таким образом, для каждого варианта исхода исход_i мы будем иметь предиктор PR_i , состоящий из множества правил, предсказывающих данный вариант исхода (целевой предикат TP_i , соответствующий данному варианту исхода).

Определим способ формирования прогноза предиктора на основе множества прогнозов отдельных правил, входящих в его состав. Для этого необходимо определить способ формирования оценки точности прогноза предиктора. Под оценкой точности прогноза предиктора PR для объекта с номером i будем понимать величину $pr_{PR}(i) \in [0, 1]$, где pr_{PR} – отображение, определяющее способ формирования итогового прогноза. Отображение pr_{PR} ставит в соответствие множеству прогнозов отдельных правил значение из интервала $[0, 1]$, т. е. $pr_{PR}(i) : \{pr_R(i) : R \in PR\} \rightarrow [0, 1]$, где $pr_R(i)$ – прогноз правила R для объекта с номером i : $pr_R(i) = p(R)$, если правило R применимо к объекту с номером i , $pr_R(i) = 0$ в противном случае.

Наиболее естественным способом определения $pr_{PR}(i)$ является его задание равным оценке точности прогноза правила, имеющего максимальную оценку точности прогноза, т. е. $pr_{PR}(i) = \max_R \{pr_R(i) : R \in PR\}$. Однако в зависимости от выбранного способа оценки точности прогноза возможны и другие способы задания $pr_{PR}(i)$.

Таким образом, прогнозом исхода исход_j будем считать прогноз предиктора PR_j , содержащего множество правил, предсказывающих данный исход.

Для осуществления принятия решения необходимо определить решающее правило $DecRule(i)$, которое должно на основе множества прогнозов отдельных предикторов выбирать конкретный вариант исхода из множества возможных исходов, т. е. $DecRule(i) : \{pr_{PR}(i)\} \rightarrow \{\text{исход}_1, \dots, \text{исход}_n\}$, где $\{\text{исход}_1, \dots, \text{исход}_n\}$ – множество возможных исходов.

Для осуществления принятия решений на основе прогнозов предикторов предлагается использовать следующий механизм определения решающего правила. Пусть имеется набор предикторов $\{PR_j\}$, $j = 1, \dots, n$. Каждый предиктор PR_j соответствует некоторому варианту исхода исход_j . Обозначим через $pr_{PR}^j(i)$ оценку точности прогноза j -го предиктора для объекта с номером i . Выбор варианта исхода $DecRule(i)$ для i -го объекта осуществляется следующим образом. Для каждого предиктора рассчитывается показатель согласованности его прогноза по формуле $Ctrl_j(i) = pr_{PR}^j(i) - \max_{k \neq j} \{pr_{PR}^k(i)\}$, т. е. как разность между оценкой точности прогноза данного предиктора и максимальной оценкой точности прогнозов остальных предикторов. В качестве варианта исхода для i -го объекта выбирается исход, соответствующий предиктору, показатель согласованности которого строго больше заданного порога $\delta > 0$, т. е. $DecRule(i) = \text{исход}_k$, где $k = \arg \max_{j=1, \dots, n} \{Ctrl_j(i) : Ctrl_j(i) > \delta\}$. Порог δ будем называть *порогом согласованности*. В случае если не существует прогноза, показатель согласованности которого выше указанного порога, то решение о выборе варианта исхода не принимается. Величина порога δ зависит от специфики решаемой задачи и должна устанавливаться исследователем.

3 Применение в задачах медицинской диагностики

Мы использовали разработанную нами систему для решения задачи дифференциальной диагностики фолликулярного рака и фолликулярной аденомы щитовидной железы.

В настоящее время в вопросах дооперационной диагностики заболеваний щитовидной железы наибольшие трудности вызывает диагностика фолликулярной опухоли при попытке отличить аденому от рака. Как показывает опыт, в настоящее время совпадение цитологических и окончательных гистологических диагнозов не превышает 56 % [6]. Таким образом, встает вопрос о совершенствовании дооперационной морфологической диагностики. Помимо повышения точности цитологической диагностики, особый интерес также вызывает предварительная диагностика заболевания по данным УЗИ, поскольку, как правило, именно при УЗИ впервые обнаруживается узел в щитовидной железе. Однако пока еще в медицине не существует методов, позволяющих диагностировать фолликулярную опухоль только по данным УЗИ [7].

Совместно с медиками Дорожной клинической больницы Новосибирска мы провели два исследования возможности диагностики фолликулярной опухоли, основанные на данных цитологического анализа и данных УЗИ.

Исходными данными для первого исследования послужили цитологические препараты 197 больных с уже известными диагнозами (86 случаев рака и 111 случаев аденомы). Все препараты были проанализированы по 30 цитологическим признакам и представлены в виде таблицы данных, которая использовалась системой «Discovery» для извлечения диагностических правил.

Всего системой было извлечено 88 диагностических правил с условной вероятностью не меньшей чем 0,9, из них 43 – с условной вероятностью равной 1. Все найденные правила были статистически значимы при уровне критерия Фишера 0,001.

Для наиболее объективной оценки прогностических возможностей системы было проведено тестирование с использованием метода скользящего контроля: из общей выборки циклически исключался один пример, система обучалась на оставшихся примерах, а затем тестировалась на исключенном примере, после чего он возвращался назад. Данный процесс продолжался до тех пор, пока система не прошла через всю выборку данных. В Таблице 1 представлены результаты тестирования системы при использовании правил, имеющих условную вероятность выше заданного порога (0,9, 0,95 и 1).

Таблица 1. Результаты тестирования на цитологических данных.

<i>Порог условной вероятности правил</i>	<i>Количество правильных прогнозов</i>	<i>Процент отказов от принятия решения</i>
90 %	93 %	8 %
100 %	96 %	13 %

Во втором исследовании нами были использованы данные УЗИ 170 больных с уже известными диагнозами (70 случаев рака и 100 случаев аденомы). Результаты УЗИ каждого больного были проанализированы по 9 признакам и представлены в виде таблицы.

Всего системой было обнаружено 105 диагностических правил с условной вероятностью не меньше 0,8, из них 53 с условной вероятностью не меньше 0,9 и 28 с условной вероятностью 1. Все найденные правила были статистически значимы при уровне критерия Фишера 0,001.

В Таблице 2 представлены результаты тестирования системы методом скользящего контроля. Как видно из таблицы, обнаруженные системой диагностические правила позволяют с достаточно высокой степенью точности (до 96 %) диагностировать фолликулярный рак и фолликулярную аденому, основываясь только на данных УЗИ.

Таблица 2. Результаты тестирования на УЗИ данных.

<i>Порог условной вероятности правил</i>	<i>Количество правильных прогнозов</i>	<i>Процент отказов от принятия решения</i>
80 %	83 %	4 %

90 %	87 %	19 %
100 %	96 %	33 %

Мы провели сравнение точности прогнозов системы «Discovery» с прогнозами, полученными при помощи нейронной сети. Тестирование нейронной сети также проводилось методом скользящего контроля. На цитологических данных нейронная сеть показало точность, равную 91 %, а на данных УЗИ – 86 %. Таким образом, проведенное сравнение показало, что система «Discovery» имеет более высокую точность прогнозов, чем нейронные сети.

Проведенные нами исследования позволяют сделать вывод о возможности успешного применения системы «Discovery» для решения сложных задач диагностики в медицине. Полученные нами диагностические правила позволяют с достаточно высокой степенью точности (до 96 %) диагностировать фолликулярный рак и фолликулярную аденому. Кроме того, при помощи системы «Discovery» нами были получены интерпретируемые правила, которые дают не только вероятностный прогноз, но и его объяснение.

4 Применение метода в биоинформатике

Рассмотрим применение системы «Discovery» в биоинформатике для распознавания сайтов связывания транскрипционных факторов (ССТФ).

Задача обнаружения ССТФ очень важна для понимания механизмов регуляции транскрипции генов. Несмотря на то что в настоящее время разработано несколько подходов к распознаванию ССТФ, данная задача пока еще не может считаться окончательно решенной.

Традиционным методом предсказания ССТФ является позиционная весовая матрица (PWM). Данный метод основан на предположении о независимости нуклеотидных позиций. Однако многие авторы указывают на то, что предположение о независимом вкладе каждой позиции в энергию связывания фактора с сайтом не соответствует сущности биологического процесса [8]. Точность предсказания может быть увеличена за счет учета окружающего контекста, в котором встретился сайт, и учета зависимости между нуклеотидами. В отличие от PWM система «Discovery» способна обнаружить взаимосвязь между нуклеотидами, которые, в общем случае, могут находиться на значительном удалении друг от друга.

При помощи системы «Discovery» мы проанализировали ДНК мишени трех семейств транскрипционных факторов: SREBP, EGR1 и HNF4. Обучающие выборки последовательностей (позитивные выборки) были извлечены из базы данных TRRD [9]. Выборки контрастных последовательностей (негативные выборки) были сгенерированы случайным образом с сохранением нуклеотидных частот, как в анализируемых выборках сайтов.

Мы использовали систему «Discovery» для поиска вероятностных закономерностей следующего вида:

$$(Pos_1(s) = N_1)^{\varepsilon_1} \& (Pos_2(s) = N_2)^{\varepsilon_2} \& \dots \& (Pos_k(s) = N_k)^{\varepsilon_k} \rightarrow (TFBS(s) = I),$$

где $(Pos_i(s) = N)^{\varepsilon}$ – предикат, означающий, что в позиции i последовательности нуклеотидов s находится (при $\varepsilon = 0$) или не находится (при $\varepsilon = 1$) символ $N \in \{A, C, G, T\}$; $(TFBS(s) = I)$ – целевой предикат, означающий, что последовательность нуклеотидов s является сайтом связывания.

Качество распознавания сайтов связывания факторов SREBP, EGR1 и HNF4 оценивалось в сравнении с качеством распознавания метода оптимальной весовой матрицы (PWM) путем осуществления специальной процедуры скользящего контроля. При каждой итерации скользящего контроля методы обучались на выборках ССТФ, оставляя ровно одну последовательность для контроля качества распознавания. Далее обученные методы применялись к контрольной последовательности, и оценивался уровень перепредсказания (ошибка второго рода) для контрольных негативных объектов (100 000 последовательностей, сгенерированных случайным образом с сохранением нуклеотидного состава). Количество итераций проводимой процедуры соответствовало объему обучающей выборки ССТФ.

Далее, по окончании скользящего контроля мы упорядочили контрольные сайты по уровням перепредсказания. Полученные результаты для всех трех семейств транскрипционных факторов показывают, что ошибка перепредсказания, соответствующая системе «Discovery», меньше

таковой для PWM при каждом уровне ошибки недопредсказания. Таким образом, система «Discovery» имеет более высокую точность распознавания. В Таблице 3 приведены ошибки второго рода для обоих методов, вычисленные при фиксированном пороге ошибки первого рода, равном 50 %.

Таблица 3. Сравнение точности распознавания «Discovery» и PWM.

ССТФ	Количество последовательностей	Длина последовательностей	Ошибка второго рода	
			PWM	Discovery
SREBP	38	18	4.70E-04	3.90E-04
EGR1	22	10	4.06E-03	2.39E-03
HNF4	30	13	2.14E-04	7.00E-05

Таким образом, предложенный подход позволил снизить ошибки распознавания ССТФ по сравнению с традиционно используемым методом весовых матриц. Учет нуклеотидных позиций не по отдельности, как это происходит в методе весовых матриц, а во взаимосвязи позволил системе «Discovery» увеличить точность распознавания ССТФ.

5 Заключение

Результаты экспериментов, представленные в данной работе, показывают преимущества реляционного подхода на примере решения реальных задач.

Благодарности

Работа выполнена при финансовой поддержке интеграционных проектов СО РАН № 3, 87, 136, интеграционного проекта РАН №15/10 и грантов РФФИ №№ 11-07-00560-а, 11-07-0388-а.

Литература

- [1] Витяев Е.Е. Извлечение знаний из данных. Компьютерное познание. Модели когнитивных процессов. – Новосибирск: НГУ, 2006. – 293 с.
- [2] Демин А.В., Витяев Е.Е. Разработка универсальной системы извлечения знаний «Discovery» и ее применение // Вестник НГУ, серия: Информационные технологии. – 2009. – Т. 7. – Вып. 1. – С. 73-83.
- [3] Kovalerchuk B., Vityaev E. Data Mining in Finance: Advances in Relational and Hybrid methods. – Kluwer Academic Publishers, 2000. – p.308.
- [4] Кендал М., Стюарт А. Статистические выводы и связи. М.: Наука, 1973. 899 с.
- [5] Демин А.В., Витяев Е.Е. Метод предсказания в языке первого порядка // Информационные технологии работы со знаниями: обнаружение, поиск, управление. – Новосибирск, 2008. – Вып. 175: Вычислительные системы. – С. 57-88.
- [6] Пупышева Т. Л. Морфометрия клеток фолликулярных пролифератов щитовидной железы в тонкоигольных аспиратах // Новости клинической цитологии России. – 2002. – Т.6. – № 1-2. – С.24-26.
- [7] Богин Ю. Н., Бондаренко В. О., Шапиро Н. А., Орлов В. М. Комплексная экспресс-диагностика заболеваний щитовидной железы // Метод. рекомендации. – Москва, 1992. – 175 с.
- [8] Benos P. V., Bulyk M. L., Stormo G. D. Additivity in protein-DNA interactions: how good an approximation is it? // Nucleic Acids Res. 2002. Vol. 30. P. 4442–4451.
- [9] Kolchanov N. A., Ignatieva E. V., Ananko E. A., Podkolodnaya O. A., Stepanenko I. L., Merkulova T. I., Pozdnyakov M. A., Podkolodny N. L., Naumochkin A. N., Romashchenko A. G. Transcription Regulatory Regions Database (TRRD): its status in 2002 // Nucleic Acid Res. 2002. Vol. 30. P. 312–317.