

**ИСПОЛЬЗОВАНИЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ ПРИ АНАЛИЗЕ  
СТАТИСТИКИ ДЛЯ КЛАССИФИКАЦИИ РАБОЧЕЙ НАГРУЗКИ  
ИНФОРМАЦИОННЫХ СИСТЕМ ПРЕДПРИЯТИЯ**

**Петунин Сергей Александрович**

К.ф.-м.н., главный специалист, e-mail: petunin@vniia.ru

**Вербов Денис Валерьевич**

Начальник подразделения, e-mail: vedenis@yandex.ru

**Лавринчук Роман Владимирович**

Ведущий специалист по администрированию БД, e-mail: lavrinchuk.roman@gmail.com

ФГУП «ВНИИА им. Н.Л. Духова», Россия, Москва, 101000, Моспочтамт, а/я 918

**Аннотация.** В работе рассматривается применение методов машинного обучения при анализе данных мониторинга рабочей нагрузки баз данных. Предлагается подход к кластеризации запросов, выполняемых в базе данных на основе вводимого коэффициента, отражающего отношение проделанной работы к возвращаемому результату. Приводится пример анализа истории нагрузки с использованием тепловых карт и рассматривается соотношение параметров сложности фактических запросов с введенным коэффициентом.

**Ключевые слова:** базы данных, оптимизация SQL запросов, классификация, машинное обучение, статистика, информационные системы.

**Цитирование:** Петунин С.А., Вербов Д.В., Лавринчук Р.В. Использование методов машинного обучения при анализе статистики для классификации рабочей нагрузки информационных систем предприятия // Информационные и математические технологии в науке и управлении. 2018. № 3 (11). С. 63–72. DOI:10.25729/2413-0133-2018-3-07

**Введение.** В технологиях обработки и анализа данных последних лет устойчивым трендом является массовое применение интеллектуальных статистических методов, в первую очередь, относящихся к группе «машинного обучения». Поэтому использование подобных техник для понимания влияния рабочей нагрузки на информационные базы данных (БД) может помочь повышению эффективности работы этих систем.

В основном работы в этой области ведутся по направлениям: автоматизации анализа и выявлению проблемных мест в дизайне и данных СУБД (в качестве примера можно привести работы [12] и [13]), выбора значений настроек различных параметров СУБД в рамках задачи самодиагностики и корректировки работы СУБД (например, работы [7],[8],[9] и [10]). При этом одной из ключевых задач остается корректное определение типа нагрузки на БД. Для отдельных СУБД ведутся работы по созданию так называемых «автономных» СУБД, достаточно подробный обзор по этой теме представлен в статье [14].

В представляемой работе проведён анализ характеристик sql-запросов на примере 3-х различных баз данных предприятия. В рамках разведочного анализа исследованы показатели трех типов объектов: баз данных, программных модулей и запросов, определены их статистические распределения. На базе методов машинного обучения построены классификационные модели (классификаторы), которые показывают зависимость числовых

метрик, характеризующих рабочую нагрузку СУБД, от различных показателей выполнения запросов (время выполнения, использование ЦПУ, объём транзакций и т.д.). В результате исследования проведена классификация нагрузки на БД с определением влияющих на неё параметров.

**1. Описание рассматриваемых систем.** Рассматриваемые информационные системы (ИС) ФГУП «ВНИИА», ориентированы на задачи автоматизации [2] и используют базы данных (БД) под управлением СУБД Oracle [11]. К наиболее нагруженным БД по количеству выполняемых транзакций можно отнести три из них (в последующем условно называемые db35, erpwork и vorkvniia). Несмотря на логическое и физическое объединение этих баз в единую систему хранения и обработки данных, они принципиально отличаются по структуре, характеру рабочей нагрузки и количеству приложений. Основная характеристика рассматриваемых баз приводится в таблице 1.

Таблица 1

База данных	Объем данных, Гб	Усредненное количество активных приложений	Характер нагрузки
DB35	150	20	Смешанный
ERPWORK	60	2	Смешанный
VORKVNIIA	50	7	Смешанный

Главной задачей данной работы стало определение зависимости между классами ресурсоемкости запросов этих БД и множеством всех ресурсных показателей запросов.

Для этого авторами были проведены следующие исследования статистики работы БД:

- осуществлён анализ исторических данных, полученных из системы мониторинга СУБД Oracle (СМ) [1], ориентированный, в первую очередь, на оценку объёма работы, выполняемой sql-запросами;
- проведена классификация приложений и запросов БД методами кластеризации по характеристикам потреблённых вычислительных ресурсов;
- на базе метода машинного обучения «случайный лес» предложены и валидированы две модели классификаторов рабочей нагрузки.

**2. Описание исходных данных.** В качестве исходных данных для проведения анализа сформированы следующие наборы исторических данных СМ: полная статистика за период трёх лет (2015-2017 гг.) и выборки из двух (04.12.2017-04.02.2018) и трех (01.22.2018-23.04.2018) последних месяцев. В связи с требованиями к объёму публикации, в данной работе представлены только результаты, относящиеся к анализу второго и третьего наборов. В табл. 2 представлена простая кортежная модель отобранных данных – перечень переменных, участвующих в анализе. Суффикс «delta» в именах переменных означает суммарную статистику за период, при этом сам период равен 1 часу. Типы данных разбиты на идентификационную и ресурсную группы.

Таблица 2

Имя переменной	Группа	Значение переменной
<i>mame</i>	<i>Идент.</i>	Имя БД
<i>module</i>	<i>Идент.</i>	Имя приложения
<i>sql_id</i>	<i>Идент.</i>	Уникальный идентификатор строки запроса
<i>buffer_gets_delta</i>	<i>Ресурс в/в</i>	Количество чтений 8Кб-блоков данных
<i>disk_reads_delta</i>	<i>Ресурс в/в</i>	Количество операций прямого чтения с диска

<i>direct_writes_delta</i>	Ресурс в/в	Количество операций прямой записи на диск
<i>cpu_time_delta</i>	Ресурс ЦПУ	Время выполнения запроса на процессоре в (мкс.)
<i>elapsed_time_delta</i>	Ресурс ЦПУ	Полное время выполнения запроса в (мкс.).
<i>plsexec_time_delta</i>	Ресурс ЦПУ	Время выполнения процедурного кода (мкс.).
<i>sorts_delta</i>	Ресурс счёт.	Количество операций сортировки
<i>rows_processed_delta</i>	Ресурс счёт.	Количество возвращенных запросом строк
<i>executions_delta</i>	Ресурс счёт.	Количество выполнений запроса

Для анализа работы реляционных СУБД важно определить, к какому типу вычислительной нагрузки относится работающее с данной БД приложение. В самом общем случае разделяют оперативную обработку транзакций (On-Line Transaction Processing – OLTP) и интерактивную аналитическую обработку (On-Line Analytical Processing – OLAP) данных.

При этом аналитический характер запроса может быть обусловлен как целями приложения, так и характером данных, сложностью бизнес-операции или особенностями приложения.

Для классификации нагрузки предлагается ввести оценочную метрику - коэффициент нагрузки ( $K_w$ , WORKLOAD) как отношение условной работы, затраченной СУБД на выполнение запроса, к значению переменной, характеризующей возвращаемый запросом объём информации, в нашем случае – к количеству строк (переменная *rows\_processed\_delta*).

$$K_w = \frac{\text{Работа}}{\text{Количество строк}} \quad (1)$$

Работу СУБД предлагается рассматривать как консолидированную усреднённую оценку на основе значений ресурсных переменных.

### 3. Методики анализа и результаты.

**3.1 Разведочный анализ.** Для начального определения «высокой» и «низкой» нагрузки на БД сформирована статистика по трём категориальным переменным: «БД (name)», «модуль (module)», «запрос (sql\_id)» на базе одного из ресурсных показателей, суммирующего время выполнения запросов (*elapsed\_time\_delta*). Например, рис.1 иллюстрирует упорядоченный список модулей трёх БД за указанный период.

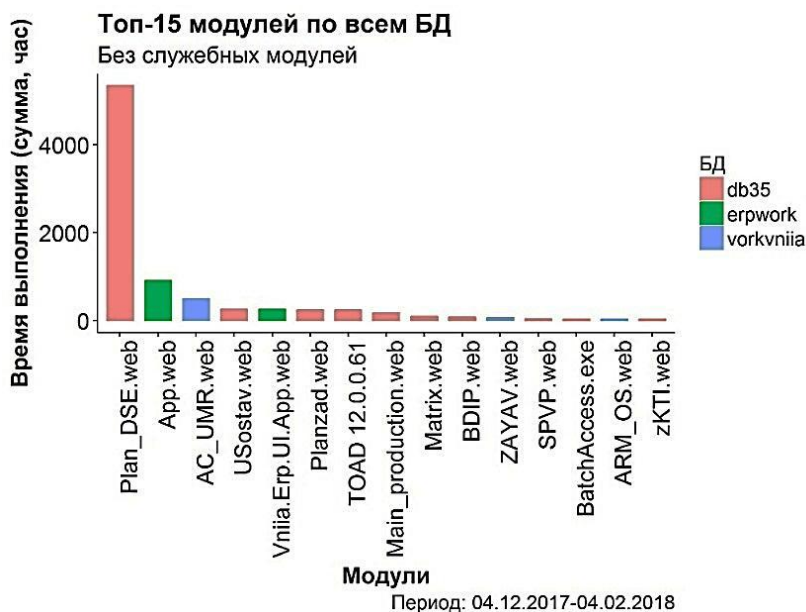


Рис. 1. Список ресурсоёмких модулей по времени выполнения

Применение классических статистик обеспечило проведение разведочного анализа данных, обязательной процедуры, в ходе которой были исключены данные с ошибками, внесённые системой мониторинга и редкие выбросы значений переменных, относящиеся к разово выполненным запросам, которые искажают средние значения статистик, но не влияют на уровень рабочей нагрузки. Для получения консолидированных статистик использовались универсальные методики и программные инструменты, разработанные и проверенные при анализе исторических данных планировщика заданий вычислительных кластеров института [6].

Ниже в таблице 3 приводятся совокупные данные по нагрузке на БД за второй рассматриваемый период.

Таблица 3

Имя БД	Кол-во записей ИД	Кол-во модулей	Кол-во sql_id	Частота запросов sum(executions_delta) (млн.)	Время выполнения sum(elapsed_time_delta) (сут.)
db35	279 460	72	8 157	19 801	131
erpwork	149 589	33	12 274	81	37
vorkvniia	49 451	25	16 306	5 222	26

Основная БД по нагрузке из рассмотренных параметров – db35. Общая длительность выполнения запросов за трехмесячный период для 2-х последних баз составила около 1 месяца, что говорит о низкой нагрузке на серверы БД (с учетом нескольких выделенных процессорных ядер для каждой БД). Еще одним признаком низкой нагрузки БД vorkvniia является малое время ожидания ввода-вывода выполняемых запросов (рис. 1). Такие усредненные характеристики можно объяснить неравномерностью суточной нагрузки на базы данных, когда основная нагрузка приходится на дневное время.

**3.2 Кластеризация.** По мнению многих специалистов-аналитиков самой массовой задачей для применения методов машинного обучения является задача классификации [4]. Методически процедура классификации предполагает разбиение набора данных, характеризующего объект, на различные классы. Цели проведения классификации могут быть различными, но все они связаны либо с задачей прогнозирования, либо с задачей выявления и понимания структуры цифрового объекта. В свою очередь, все методы машинного обучения принято делить на две группы: «обучение с учителем» и «обучение без учителя». Модели первой группы предполагают наличие отклика (функции) между независимыми переменными (предикторами) и зависимой переменной. Основными методами второй группы являются кластеризационные методы, которые классифицируют/разбивают объект согласно различным метрикам «близости» его векторов в n-мерном пространстве параметров модели.

Использование методов кластеризации для нашей задачи позволило выделить группы заданий (или модулей), близко расположенных в пространстве ресурсных переменных (см. табл. 2). В случае кластеризации по модулям берётся среднее значение каждого показателя в группе. В таблице 4 приведен пример разбиения на кластеры для модулей БД db35 в 4-х мерном пространстве усреднённых нормированных оценочных коэффициентов трёх ресурсных групп и усреднённого показателя количества возвращаемых строк. Модули с большими значениями коэффициента нагрузки попали в 6-й кластер.

Таблица 4

Номер кластера	Кол-во строк (среднее)	Коэфф. «ресурса в/в» (среднее)	Коэфф. «ресурса ЦПУ» (среднее)	Коэфф. «сортировки» (среднее)
1	27519	0.5443	0.0102	0.0032
2	738860	0.0041	0.0002	0.000004
3	3485073	0.0001	0.00001	0.00000005
4	309740	0.0006	0.0002	0.00002
5	10939	0.2518	0.0079	0.04392
<b>6</b>	<b>1386</b>	<b>310.9558</b>	<b>11.0909</b>	<b>0.1793</b>

Следующий шаг заключается уже в кластеризации запросов в модулях 6-й группы.

**3.3 Прогнозирование.** Мы рассмотрели задачу построения классификаторов для прогнозирования классов коэффициента нагрузки  $K_w$ :

- 1) двух классов, разделённых пороговым значением;
- 2) трёх классов, разделённых двумя пороговыми значениями.

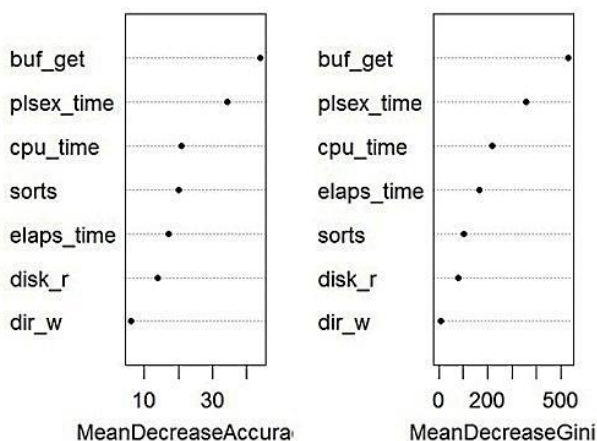
В этих моделях зависимую переменную  $w$  (класс объёма работы) вычислим по значениям коэффициента  $kw$ , который в свою очередь определим как сглаженную усреднённую нормализованную сумму значений ресурсных переменных-предикторов (например, логарифм сумм логарифмов). В первом случае, значение отклика является категориальной переменной, определяя два класса: «Да» и «Нет». Во втором случае отклик принимает три числовых значения: 1,2,3, т.е. модель является регрессионной. Переменные предикторов обозначим через  $x_i$ . Параметры модели следующие:

Формула:  $w \sim \{ x_i \}, i = 1, 7$

Обучающая выборка: train (случайный выбор 70% значений)

Тестовая (контрольная) выборка: test (оставшиеся 30% значений набора данных).

Следующие два графика иллюстрируют результаты применения метода машинного обучения «случайный лес» [3] на наборе данных БД db35.

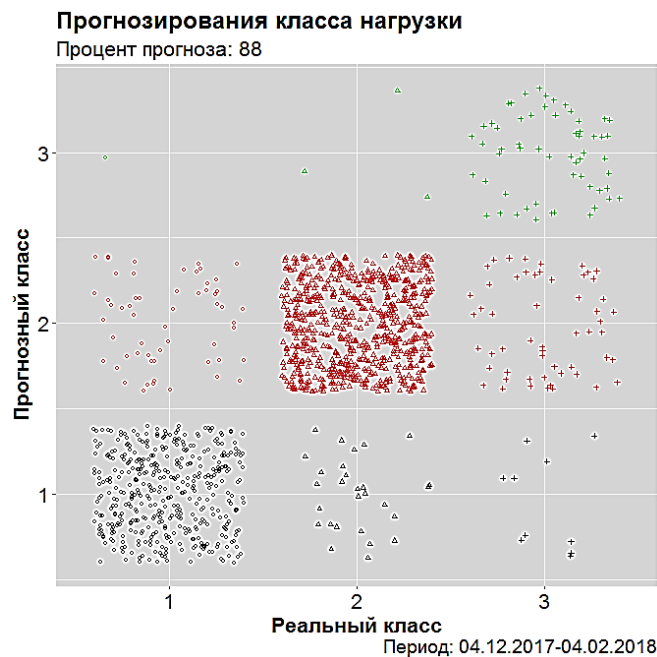


**Рис. 2.** Важность переменных бинарного классификатора

На рисунке 2 показан рейтинг влияния ресурсных переменных-предикторов на определение класса нагрузки, который определяется по двум мерам: усреднённому уменьшению правильности прогноза (MeanDecreaseAccur), и усреднённому уменьшению неоднородности (MeanDecreasyGini, индекс Джини). Точность прогноза модели бинарного

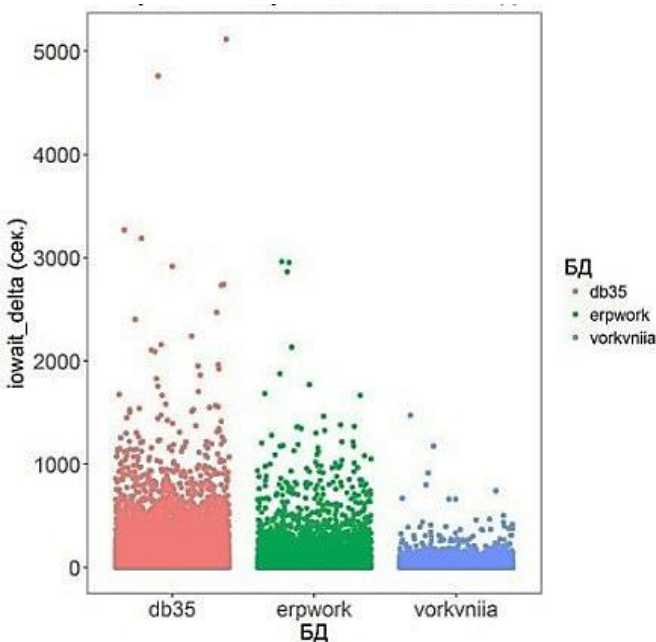
классификатора на тестовой выборке составила 91.42%. Валидация модели была проведена с помощью оценки площади ROC-кривой, которая составила 0.98.

Точность прогноза второй модели регрессионного типа на тех же данных составила 88% (рис. 3).



**Рис. 3.** Карта распределения попадания запросов тестовой выборки в оценочный класс эффективности рабочей нагрузки

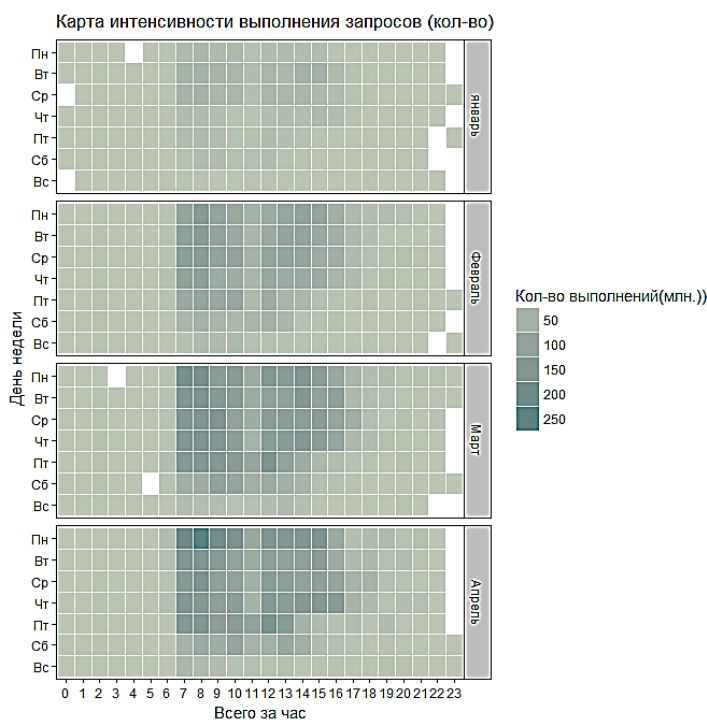
**3.4 Оценка нагрузки на основе времен ожидания и тепловых карт.** Мы рассмотрели нагрузку на БД по второму набору данных в разрезе времен ожидания БД. Рассматриваются в первую очередь ожидания по вводу-выводу, как наиболее характерные для баз данных (рис. 4).



**Рис. 4.** Распределение времени ожидания по БД

Как видно, максимальная нагрузка по вводу-выводу соответствует второй и третьей базам, однако во время работы могут быть пиковые события ожидания. Кроме того, любопытно оценить какие преимущественно операции выполняются в базе данных.

Также крайне полезным является анализ нагрузки с течением времени. Далее будем рассматривать базу DB35, как имеющую максимальную нагрузку. Для анализа интенсивности нагрузки, рассмотрим тепловую карту за последние календарные месяцы.



**Рис. 5.** Тепловая карта интенсивности выполнения запросов

На тепловой карте (рис. 5) отражены интенсивности выполнения запросов. Опуская январь, где причиной снижения интенсивности запросов традиционно являются зимние каникулы и отпуска, можно проследить возрастание интенсивности в рабочие часы, как в целом, так и по дням недели. Из этих данных как минимум следует неравномерность нагрузки в рабочее время, что позволяет предположить наличие скрытых резервов оптимизации работы базы данных за счет перераспределения нагрузки по времени.

Если рассматривать запросы в контексте выполняемых операций, то для анализируемой базы наибольший вес, как по количеству запросов, так и по коэффициенту нагрузки принимает

оператор выборки. Следующими по коэффициенту нагрузки идут блоки процедурного кода и оператор MERGE.

Далее была рассмотрена выборка из порядка 20 запросов, имеющих наибольшее значение коэффициента нагрузки или число выполнений. Анализировался план выполнения запроса и специфические особенности, такие как обращение к хранимым функциям или выборка данных из внешнего источника. Целью данного анализа являлась проверка эффективности обнаружения «тяжелых» запросов с применением метода кластеризации на основе коэффициента нагрузки.

Основными причинами попадания в этот список были такие особенности запросов, как множественные вложенные соединения, просмотр таблиц без использования индекса, использование в запросе вызовов функций и процедур на PL коде. В отдельных случаях попадание в группу обусловлено высокой частотой вызовов, что может быть объяснено, в том числе, недостаточной поддержкой кэширования данных на уровне приложения.

Ниже, в таблице 5 приводится пример распределения запросов по группам, где поля:  $pe$  – суммарное количество выполнений в тыс. штук,  $s\_elap$  – суммарное время выполнения в часах,  $av\_elap$  – среднее время выполнения в минутах,  $s\_cpu$  – суммарное потребление ресурсов ЦПУ в часах,  $av\_cpu$  – среднее время потребления ЦПУ в минутах,  $kcru$  – отношение  $s\_cpu$  к  $s\_elap$ ,  $s\_kw$  – суммарный коэффициент РН,  $s\_kwe$  – суммарный коэффициент РН без учета кол-ва выполнений и кол-ва строк парсинга запроса.

Таблица 5

sql_id	n	ne	s_elap	av_elap	s_cpu	av_cpu	kcpu	s_kw	s_kwe	s_wait	av_wait	kwait	s_len	тип запросов
Ozpb4j4pnpksf	1236	180	241	11,7	147	7,1	0,6	2	77605	46,2	2,2	0,2	6908004	Вложенные JOIN
3twray3uusdn1	1136	114	174	9,2	138	7,3	0,8	52	66079	9,9	0,5	0,1	6947776	FULL SCAN
g72f2qgavzst	702	80	160	13,7	150	12,8	0,9	419	41564	1,5	0,1	0,0	2161458	FULL SCAN
d1xd527z294r3	15659	16	144	0,6	57	0,2	0,4	13995	302757	76,3	0,3	0,5	162101968	Исп. PL кода
f3vs0aj2rgp1k	22111	22	127	0,3	60	0,2	0,5	12384	339388	58,8	0,2	0,5	156103660	Исп. PL кода
9w1xhgvk1z41s	1084	14081114	95	5,3	90	5,0	1,0	0	27956	0,0	0,0	0,0	94308	Кол-во вызовов
3d2rzvyr4fx7z	1239	37	83	4,0	47	2,3	0,6	309	52956	31,0	1,5	0,4	5202561	FULL SCAN
110mgxa8pbrnj	249	31777	75	18,1	75	18,1	1,0	2978	11467	0,0	0,0	0,0	23157	Прочее
1zkw63rg1hp35	13060	13	61	0,3	36	0,2	0,6	9368	272601	16,7	0,1	0,3	91380820	Исп. PL кода
17s6wjuju6n5w	890	2141960	58	3,9	53	3,6	0,9	1843	40401	0,8	0,1	0,0	277680	Кол-во вызовов
c1xkw6c56nvwpp	25917	26	58	0,1	48	0,1	0,8	21880	442243	6,4	0,0	0,1	173462481	Исп. PL кода
8rxqa0nfmk67d	17057	17	57	0,2	39	0,1	0,7	6651	310905	6,8	0,0	0,1	175499473	Исп. PL кода
Otbatn71xsy3h	24293	24	51	0,1	38	0,1	0,7	10122	391414	8,7	0,0	0,2	242565605	Исп. PL кода
fkfxdrr2790b	1122	1	50	2,7	40	2,1	0,8	5257	48480	1,9	0,1	0,0	14553462	Вложенные JOIN
7qwnqcadjmj6p	670	40	45	4,0	37	3,3	0,8	19	37413	2,1	0,2	0,0	4074270	FULL SCAN
cb5xpmx24tcmz	377	4	45	7,1	14	2,2	0,3	19206	22826	32,1	5,1	0,7	32422	Исп. PL кода
dx4gjf0vnxq1u	11277	17	44	0,2	17	0,1	0,4	4145	210478	7,5	0,0	0,2	70345926	FULL SCAN
f23sa5sqz59xq	391	40	42	6,5	36	5,5	0,9	17	22532	0,4	0,1	0,0	4358086	Вложенные JOIN
cuz1ud6nb0awh	357	4	41	6,9	13	2,1	0,3	9931	19522	29,4	4,9	0,7	561204	FULL SCAN
8qnnwrsz3r6a	732	2033588	39	3,2	30	2,5	0,8	831	34870	6,1	0,5	0,2	63684	Кол-во вызовов

**Заключение.** Применение современных статистических методов для анализа рабочей нагрузки информационных систем предоставляет в сравнении с системами мониторинга дополнительные возможности. Так, при решении задачи оптимизации работы БД необходимо выявить sql-запросы для последующего анализа и оптимизации. Предложенный в данной работе подход на основе введенного коэффициента с использованием механизмов кластеризации и прогнозирования для выявления аномальных по нагрузке модулей и запросов за выбранный календарный период позволяет локализовать запросы, требующие оптимизации, и сократить время работы аналитика на первичный анализ работы БД.

Обработка данных и визуализация результатов проведенного исследования сделана в рамках инструментальных возможностей статистической экосистемы на языке программирования R [5].

#### СПИСОК ЛИТЕРАТУРЫ

1. Вербов Д.В. Анализ факторов, влияющих на выбор методов интеграции, с позиций производительности и надежности информационных систем // Тр. 5-й Международной научной конференции «Информационные Технологии и Системы» (ИТиС – 2016). Банное, Россия, Челябинск, ЧелГУ. С. 217–218.
2. Вербов Д.В., Лавринчук Р.В. Автоматизация статического и динамического анализа информационных систем в условиях интеграции // Тр. 4-й Международной научной конференции «Информационные Технологии и Системы» (ИТиС – 2015). Банное, Россия, Челябинск, ЧелГУ. С. 107–109.
3. Груздев А.В. Прогнозное моделирование в IBM SPSS Statistics, R и Python. Метод деревьев решений и случайный лес. М.: ДМК Пресс. 2018.
4. Джеймс Г., Уиттон Д., Хасты Т., Тибширани Р. Введение в статистическое обучение с примерами на языке R. Москва. ДМК Пресс. 2016. 456 с.
5. Официальный сайт R-проекта. Режим доступа: <http://www.r-project.org>.



6. Петунин С.А. Методика начального анализа рабочей нагрузки вычислительных кластеров // Тр. 4-й Международной научной конференции «Информационные Технологии и Системы» (ИТиС). Челябинск. изд. Челябинского гос. университета. 2015. С. 137–138.
7. Benoit D.G. Automatic diagnosis of performance problems in database management systems // Proceedings of Second International Conference on Autonomic Computing (ICAC'05). IEEE. 2005. Pp. 326–327.
8. Dageville B., Dias K. Oracle's Self-Tuning Architecture and Solutions // IEEE Data Eng. Bull. 2006. T. 29. №. 3. Pp. 24–31.
9. Elnaffar S. S., Martin P. Towards workload-aware DBMSS: identifying workload type and predicting its change. Queen's University Kingston. Ontario. Canada. 2004. 170 p.
10. Ogeer N. Multiple Buffer Pools and Dynamic Resizing of Buffer Pools in PostgreSQL. Queen's University Kingston. Ontario. Canada. 2004. 93 p.
11. Oracle9i Database Performance Guide and Reference, Release 1(9.0.1), Part# A87503-02, Oracle Corp. (2001).
12. Ranganwamy M.S., Shobha G. SQL Query Dissembler—A Self Optimizing Autonomic System // International journal of advanced research in computer science. 2011. T. 2. №. 2. Pp. 219–223.
13. Sanders G.L., Shin S. Denormalization effects on performance of RDBMS // Proceedings of the 34th Hawaii International Conference on System Sciences. IEEE. 2001. Pp. 1–9.
14. Van Aken D. et al. Automatic database management system tuning through large-scale machine learning // Proceedings of the 2017 ACM International Conference on Management of Data. ACM. 2017. Pp. 1009–1024.

---

**UDK 004.658.2**

**THE USE OF MACHINE LEARNING METHODS IN THE ANALYSIS OF STATISTICS  
FOR THE CLASSIFICATION OF THE WORKLOAD OF ENTERPRISE INFORMATION  
SYSTEMS**

**Sergey A. Petunin**

PhD., Lead Specialist, e-mail: petunin@vniia.ru

**Denis V. Verbov**

Department Manager, e-mail: vedenis@yandex.ru

**Roman V. Lavrinchuk**

Senior database administrator, e-mail: lavrinchuk.roman@gmail.com

Dukhov Automatics Research Institute, (VNIIA),

Moscow Post Office, P.O. Box 918, Moscow 101000, Russia

**Abstract.** The paper considers the application of machine learning methods in the analysis of database workload monitoring data. An approach is proposed for clustering queries performed in the database based on the input coefficient, which reflects the relation of the work done to the returned result. An example of analysis of load history using heat maps is given and the ratio of complexity parameters of actual queries to the introduced coefficient is considered.

**Keywords:** database, optimization of SQL query, classification, machine learning, statistics, information systems.

## References

1. Verbov D.V. Analiz faktorov, vliyayushhikh na vybor metodov integratsii, s pozitsij proizvoditelnosti i nadezhnosti informatsionnykh sistem [Analysis of the factors influencing the choice of integration methods from the standpoint of performance and reliability of information systems ] // Tr. 5-y Mezhdunarodnoy nauchnoy konferentsii «Informatsionnyye Tekhnologii i Sistemy» (ITiS – 2016) = Proceedings of the 4th International Scientific Conference "Information Technologies and Systems" (ITiS – 2016). Chelyabinsk, Chelyabinsk State University. Pp. 217–218. (in Russian).
2. Verbov D.V., Lavrinchuk R.V. Avtomatizatsiya staticheskogo i dinamicheskogo analiza informatsionnykh sistem v usloviyakh integratsii [Automation of static and dynamic analysis of information systems in terms of integration] // Tr. 4-y Mezhdunarodnoy nauchnoy konferentsii «Informatsionnyye Tekhnologii i Sistemy» (ITiS – 2015). = Proceedings of the 4th International Scientific Conference "Information Technologies and Systems" (ITiS – 2015). Chelyabinsk, Chelyabinsk State University. Pp. 107–109. (in Russian).
3. Gruzdev A.V. Prognoznoe modelirovanie v IBM SPSS Statistics R i Python metod derevev reshenij i sluchajnyj les [Predictive modeling in IBM SPSS Statistics, R and Python. A method of decision trees and a random forest]. Moscow. DMK-Press. 2018 (in Russian)
4. James G., Wheatton D., Hasti T., Tibshirani R. An Introduction to Statistical Learning: with Applications in R. Moscow, DMK-Press, 2016. 456 p. (in Russian)
5. The R Project for Statistical Computing. Available at: URL <http://www.R-project.org/>.
6. Petunin S.A. Metodika nachalnogo analiza rabochej nagruzki vychislitelnykh klasterov [Technique initial workload analysis computing clusters] // Tr. 4-y Mezhdunarodnoy nauchnoy konferentsii «Informatsionnyye Tekhnologii i Sistemy» (ITiS – 2015). = Proceedings of the 4th International Scientific Conference "Information Technologies and Systems" (ITiS – 2015). Chelyabinsk. Chelyabinsk State University. Pp. 137–138. (in Russian)
7. Benoit D.G. Automatic diagnosis of performance problems in database management systems // Proceedings of Second International Conference on Autonomic Computing (ICAC'05). IEEE. 2005. Pp. 326–327.
8. Dageville B., Dias K. Oracle's Self-Tuning Architecture and Solutions // IEEE Data Eng. Bull. 2006. T. 29. № 3. Pp. 24–31.
9. Elnaffar S. S., Martin P. Towards workload-aware DBMSS: identifying workload type and predicting its change. Queen's University Kingston. Ontario. Canada. 2004. 170 p.
10. Ogeer N. Multiple Buffer Pools and Dynamic Resizing of Buffer Pools in PostgreSQL. Queen's University Kingston. Ontario. Canada. 2004. 93 p.
11. Oracle9i Database Performance Guide and Reference, Release 1(9.0.1), Part# A87503-02, Oracle Corp. (2001).
12. Rangaswamy M.S., Shobha G. SQL Query Dissembler—A Self Optimizing Autonomic System // International journal of advanced research in computer science. 2011. T. 2. № 2. Pp. 219–223.
13. Sanders G.L., Shin S. Denormalization effects on performance of RDBMS // Proceedings of the 34th Hawaii International Conference on System Sciences. IEEE. 2001. Pp. 1–9.
14. Van Aken D. et al. Automatic database management system tuning through large-scale machine learning // Proceedings of the 2017 ACM International Conference on Management of Data. ACM. 2017. Pp. 1009–1024.