МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра «Прикладная математика и информатика»
(наименование кафедры)

01.03.02 Прикладная математика и информатика
(код и наименование направления подготовки, специальности)

Системное программирование и компьютерные технологии
(направленность (профиль)/специализация)

БАКАЛАВРСКАЯ РАБОТА

на тему «Алгоритмы EDM для анализа успеваемости в вузе»

Студент	А.Е. Кондалов	
	(И.О. Фамилия)	(личная подпись)
Руководитель	С.В. Баумгертнер	
	(И.О. Фамилия)	(личная подпись)
Консультант	Н.В. Андрюхина	
	(И.О. Фамилия)	(личная подпись)

Допустить к защите

Заведующий кафедрой к.т.н., доцент, А.В. Очеповский _____
(ученая степень, звание, И.О. Фамилия) (личная подпись)

«_____»_____20_____г.

Тольятти 2019

АННОТАЦИЯ

Тема бакалаврской работы – «Алгоритмы EDM для анализа успеваемости в вузе».

Ключевые слова: EDUCATIONAL DATA MINING, АНАЛИЗ УСПЕВАЕМОСТИ В ВУЗЕ, АЛГОРИТМЫ КЛАССИФИКАЦИИ, АЛГОРИТМЫ КЛАСТЕРИЗАЦИИ.

Объект исследования бакалаврской работы – анализ успеваемости студентов вуза.

Предмет исследования бакалаврской работы – алгоритмы EDM для анализа успеваемости в вузе.

Цель бакалаврской работы - исследование и выбор алгоритмов EDM для анализа успеваемости в вузе.

Методы исследования: методы Data mining, теория алгоритмов.

Актуальность бакалаврской работы обусловлена необходимостью применения эффективных алгоритмов EDM для обеспечения качественного анализа успеваемости в вузе.

В первой главе бакалаврской работы рассматриваются методы анализа данных в образовательном процессе вуза. Выполнена постановка задачи анализа успеваемости в вузе.

Вторая глава посвящена исследованию основных алгоритмов анализа успеваемости в вузе.

В третьей главе с помощью программы Weka произведена оценка эффективности выбранных алгоритмов анализа успеваемости в вузе.

В заключении подводятся итоги исследования, формируются окончательные выводы по изучаемой тематике.

Бакалаврская работа состоит из 45 страниц и включает 17 рисунков, 4 таблиц, 28 источников.

ABSTRACT

The title of the bachelor's work is: “EDM algorithms for analysis of academic performance in a high school”

Keywords: EDUCATIONAL DATA MINING, ANALYSIS OF ACADEMIC PERFORMANCE IN A HIGH SCHOOL, CLASSIFICATION ALGORITHMS, CLUSTERING ALGORITHMS.

The object of the bachelor's work is analysis of academic performance in a high school.

The subject of bachelor's work are EDM algorithms for analysis of academic performance in a high school.

The aims of the bachelor's work are research and choice of EDM algorithms for the analysis of academic performance in a high school.

Research methods: Data mining methods, theory of algorithms.

The methods and algorithms for Big data analysis in the educational process of a high school are analyzed.

The choice of the algorithms for analysis of academic performance in a high school is substantiated.

Effectiveness of these algorithms is confirmed.

The bachelor's work consists of an explanatory note on 45 pages including 17 figures, 4 tables, the list of 28 references.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Среди задач управления современным вузом обеспечение высокого качества процесса обучения является одной из ключевых.

Качественное обучение поможет сократить время, затрачиваемое студентом на изучение определенного материала, позволит студентам получить больше полезной информации и определить приоритеты в освоении учебных дисциплин.

Одним из показателей качества образования в вузе является успеваемость студентов.

Однако, как показывает практика, контроль успеваемости в вузе усложняется необходимостью анализа образовательных данных, накопленных за длительный период времени.

В настоящее время инструментом, наиболее полезным для поддержки принятия управленческих решений, направленных на повышение качества обучения в вузе, является технология анализа образовательных данных – Educational Data Mining (EDM).

Вместе с тем, чтобы обеспечить высокое качество анализа массивов образовательных данных необходимо использовать соответствующие аналитические инструменты, в том числе эффективные алгоритмы, способные выявить основные закономерности в созданных массивах [12].

Таким образом**, актуальность** бакалаврской работы обусловлена необходимостью применения эффективных алгоритмов EDM для обеспечения качественного анализа успеваемости в вузе.

Объект исследования: анализ успеваемости студентов вуза.

Предмет исследования: алгоритмы EDM для анализа успеваемости в вузе.

Целью работы является исследование и выбор алгоритмов EDM для анализа успеваемости в вузе.

Для достижения поставленной цели необходимо решить следующие задачи:

− проанализировать необходимую научную и учебно-методическую литературу;

− проанализировать методы анализа образовательных данных;

− проанализировать алгоритмы анализа образовательных данных в вузе;

− выбрать алгоритмы EDM для анализа успеваемости в вузе и подтвердить их эффективность.

Методы исследования: Data mining, теория алгоритмов.

Практическая значимость бакалаврской работы заключается в выработке рекомендаций для выбора эффективных алгоритмов EDM для анализа успеваемости в вузе.

В первой главе бакалаврской работы рассматриваются методы анализа данных в образовательном процессе вуза. Выполнена постановка задачи анализа успеваемости в вузе.

Вторая глава посвящена исследованию основных алгоритмов анализа успеваемости в вузе.

В третьей главе с помощью программы Weka произведена оценка эффективности выбранных алгоритмов анализа успеваемости в вузе.

В заключении подводятся итоги исследования, формируются окончательные выводы по изучаемой тематике.

Бакалаврская работа состоит из 45 страниц и включает 17 рисунков, 4 таблиц, 28 источников.

Глава 1 МЕТОДЫ АНАЛИЗА ДАННЫХ В ОБРАЗОВАТЕЛЬНОМ ПРОЦЕССЕ ВУЗА

1.1 Постановка задачи анализа успеваемости в вузе

Образовательные данные (Educational Data, Student Data) – это данные, которые хранятся в государственном образовательном учреждении, относятся к учащемуся или студенту, обеспечивают улучшенное понимание и принятие решений в образовательных процессах. Данные, которыми располагают подрядчики, выполняющие институциональную услугу или функцию, также являются образовательными данными [21].

Проблемами анализа данных в образовательном процессе вуза занимается специальная научная дисциплина «Анализ образовательных данных» (АОД) - Educational Data Mining, EDM [19].

Анализ образовательных данных является новой развивающейся дисциплиной, связанной с разработкой методов исследования уникальных и данных, которые поступают из учебных заведений, и использованием этих методов для лучшего понимания проблем обучаемых и условий, в которых они учатся [23].

АОД использует методы, инструменты и алгоритмы интеллектуального анализа данных (Data Mining) для исследования данных студентов, преподавателей и административного персонала вузов, сотрудничества между студентами, административных данных и демографических данных.

Как показывает практика, независимо от используемых технологий обучения (очное, заочное или электронное образование), образовательные данные имеют несколько уровней значимой иерархии, которые определяются свойствами самих данных.

Вопросы времени, последовательности и контекста также играют важную роль в анализе образовательных данных.

Следует учесть, что в образовательном процессе вуза существует много проблем, которые необходимо проанализировать.

Выбор конкретной проблемы и объекта анализа зависит от задач, которые решает управленческий аппарат вуза.

Вместе с тем, необходимо отметить, что для образовательного процесса в вузе одними из самых востребованных в АОД являются задачи анализа успеваемости студентов [10].

В общем случае основные задачи анализа успеваемости в вузе – это классификация студентов по успеваемости, выявление факторов, влияющих на общую успеваемость студентов вуза и оценки степени этого влияния.

1.2 Методология анализа образовательных данных

Как отмечено выше, методологической основой АОД является интеллектуальный анализ данных - Data Mining [3].

Data Mining - это подход, использующий различные информационные технологии (ИТ) - системы и инструменты для анализа и извлечения знаний из информации, содержащейся в хранилищах данных организаций.

Наиболее часто используемая структура для понимания жизненного цикла проекта Data Mining - это межотраслевой стандартный процесс интеллектуального анализа данных (CRISP-DM), модель которого представлена на рисунке 1.1 [25].

Рисунок 1.1 – Модель жизненного цикла интеллектуального анализа данных (CRISP-DM)

Основными этапами процесса интеллектуального анализа данных являются:

1. Понимание бизнеса (Understanding the business). На данном этапе аналитик должен понять видение и цели бизнеса и то, как проект Data mining принесет пользу организации. Кроме того, очень важно четко определить требования проекта.

2. Понимание данных (Understanding data). На данном этапе определяются таблицы данных и поля, которые будут затем предметом анализа.

3. Подготовка данных (Preparing data). Этот этап заключается в переносе необходимых данных в набор данных, который будет использоваться в анализе. Процесс очистки данных должен выполняться во время этого преобразования.

4. Создание моделей (Creating models). На данном этапе должна быть запланирована и разработан модель, которая будет использоваться для анализа.

5. Оценка (Evaluation). Это этап эксперимента, для которого необходимо выбрать алгоритмы и инструментарий. Результатом данного этапа являются знания, полученные с помощью существующей модели.

6. Развертывание (Deployment). На данном этапе создается презентация результатов. Если результаты не соответствуют требованиям, необходимо запланировать новую модель.

АОД исследует организационный контекст, и это одна из причин того, что данная методология играет ключевую роль в изучении и улучшении академических показателей, таких, как уровень отсева и уровень выпуска, используемых для управления процессами реструктуризации и организационного управления в вузах.

Модель процесса АОД изображена на рисунке 1.2.

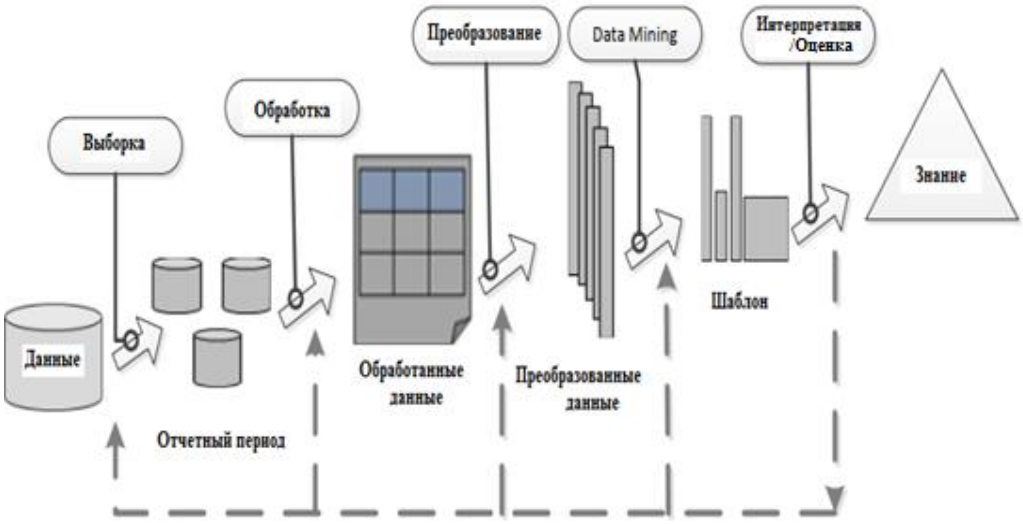

Рисунок 1.2 – Модель процесса АОД

Данный процесс называется Knowledge Discovery in Databases (KDD, обнаружение знаний в базе данных) и состоит из следующих этапов:

1. Выборка исходных данных из оперативных баз данных образовательной среды и миграция в целевое хранилище данных.

2. Предварительная обработка данных - очистка и предварительная обработка данных на основе используемой стратегии принятия решения, размещение данных в правильном формате, удаление дубликатов и обработка пропущенных полей.

3. Преобразование данных - создание наборов данных с необходимыми переменными для упрощения анализа.

4. Интеллектуальный анализ данных на основе методологии Data mining.

5. Интерпретация и оценка - понимание результатов и создание явных знаний посредством визуализации данных в отчетах и информационных панелях.

Методология АОД объединяет методы, алгоритмы и технологии, которые могут быть использованы для проведения различных экспериментов и проектирования моделей управления образовательным процессом.

Следует отметить, что методология АОД опирается на понятие шаблона (паттерна) личностных качеств студентов. Результат реализации созданных моделей позволяет исследователям прогнозировать или получать шаблоны из образовательных данных.

Все методы, используемые в АОД, можно условно разделить на стандартные методы Data mining и специфические методы АОД.

Описание методов интеллектуального анализа данных, используемых в АОД, представлено в таблице 1.1.

Таблица 1.1 - Методы интеллектуального анализа данных (АОД)

Метод	Характер применения в АОД
1	2
Стандартные методы Data mining	
Прогнозирование (Prediction)	Цель прогнозирования - разработать модель, которая позволяет сделать выводы по некоторым аспектам данных, основанные на комбинациях других характеристик данных, например, информация об отсевах студентов может быть собрана, и с помощью анализа этой информации можно сделать прогнозы для принятия корректирующих и предупреждающих действий для новых студентов. В этой группе используются три типа методов: классификация, регрессия и оценка плотности.

1	2
Обнаружение структуры (Structure discovery)	Речь идет о поиске структур данных без предварительного представления о том, что должно быть найдено. Основная задача исследователя - выявить естественную структуру данных. В эту классификацию входят: кластеризация, факторный анализ, анализ социальных сетей и обнаружение доменных структур.
Кластеризация (Clustering)	Открытие новых моделей поведения студентов. Объединение студентов в кластеры с учетом их успеваемости в вузе.
Выявление взаимосвязей (Relationship mining)	Цель метода заключается в обнаружении связей между определенными переменными в наборе данных. В рамках этой категории можно рассматривать ассоциацию, корреляцию, последовательный и причинный анализ данных.
Обнаружение моделей (Model Discovery)	Результаты интеллектуального анализа используются для дальнейшего анализа. Обычно модель создается с помощью методов

	прогнозирования.

1	2
Специфические методы АОД	
Исследование с помощью моделей (Discovery with models)	Обнаружение отношений между поведением студентов и их характеристиками или контекстными переменными. Анализ вопросов исследования по широкому разнообразию контекстов.
Перегонка данных для принятия решений человеком (Distillation of Data for Human Judgment)	Представление данных в виде, удобном для человеческого понимания, и выявление на их основе шаблонов в процессе обучения студента, поведении или сотрудничестве. Разметка данных для использования в дальнейшей разработке модели прогнозирования.

Следует обратить внимание на то, что в представленных описаниях нет четких рекомендаций по применению тех или иных методов АОД для решения конкретных задач анализа показателей образовательного процесса.

Таким образом, для выбора метода анализа успеваемости студентов в вузе разработчику необходимо проанализировать известные подходы к решению данной задачи и обосновать свой выбор.

1.3 Методы анализа успеваемости студентов в вузе

Для АОД и анализа успеваемости студентов в вузе, в частности, используются следующие методы интеллектуального анализа данных: классификация, кластеризация, поиск связывающих правил, логическая регрессия и др.

Наиболее популярными среди них являются методы классификации и кластеризации [6].

Классификация является одним из самых популярных методов, используемая в интеллектуальном анализе образовательных данных.

Это процесс, который состоит из двух этапов: создание модели классификации и предсказание значений зависимой переменной с помощью созданной модели.

Процесс классификации данных включает в себя обучение и собственно классификацию.

В ходе обучения тестовые данные анализируются с помощью выбранного алгоритма классификации.

При классификации тестовые данные используются для оценки точности правил классификации. Если точность приемлема, правила могут быть применены к новым кортежам данных.

Алгоритм обучения классификаторов использует эти предварительно классифицированные примеры для определения набора параметров, требуемых для правильной дискриминации. Затем алгоритм преобразует эти параметры в модель-классификатор.

Как показал анализ источников, классификация интеллектуального анализа данных в рамках метода прогнозирования АОД применяется в основном для решения задачи прогнозирования успеваемости студентов в вузе [22].

Кластеризация может быть определена как обнаружение похожих классов объектов.

Используя методы кластеризации, можно дополнительно определить плотные и редкие области в пространстве объектов и обнаружить общую шаблоны распределения и корреляции между данными атрибутов.

Кластеризацию можно использовать как предварительную обработку для выбора подмножества атрибутов и классификации.

Методы кластеризации используются главным образом для решения задач анализа влияния основные факторов (учебных групп, преподавателей, дисциплин и т.д.) на успеваемость студентов [13].

Для выполнения факторного анализа используется дисперсионный анализ средних. Данный аппарат основывается на гипотезах о нормальности распределений, однородности математических ожиданий и дисперсий, что представляется существенными допущениями для образовательного процесса в вузе.

Выводы к главе 1

1. Проблемами анализа данных в образовательном процессе вуза занимается специальная дисциплина «Анализ образовательных данных» - АОД или EDM. Методологической основой АОД является интеллектуальный анализ данных - Data Mining.

2. Методология АОД объединяет методы, алгоритмы и технологии, которые могут быть использованы для проведения различных экспериментов и проектирования моделей управления образовательным процессом. Данная методология опирается на понятие шаблона (паттерна) личностных качеств студентов.

3. Задача прогнозирования успеваемости студентов вуза является одной из самых востребованных в АОД.

4. Анализ показал, что в специальной литературе нет четких рекомендаций по применению тех или иных методов АОД для решения задач анализа успеваемости в вузе. Вместе с тем, можно выделить две основные

группы методов, которые используются для решения данной задачи – это методы классификации и кластеризации.

Глава 2 ИССЛЕДОВАНИЕ АЛГОРИТМОВ АНАЛИЗА УСПЕВАЕМОСТИ В ВУЗЕ

2.1 Алгоритмы классификации для анализа успеваемости в вузе

По мнению специалистов, наиболее эффективными алгоритмами классификации для решения задач АОД являются деревья решений, наивные байесовские алгоритмы и нейронные сети [18].

Рассмотрим особенности применения каждого их вышеперечисленных для анализа успеваемости студентов вуза.

Как правило, при решении задач анализа успеваемости в вузе используется описанная выше методология KDD.

Процесс начинается со сбора и предварительной обработки данных, за которыми следует построение классификационной модели, и заканчивается оценкой и интерпретацией модели.

2.1.1 Алгоритмы анализа успеваемости на основе деревьев решений

Деревья решений предназначены для построения моделей классификации или регрессии в форме древовидной структуры, используемые для поддержки принятия решений.

Математически задача классификации с помощью дерева решений описывается следующим образом (2.1) [14]:

$$(x, Y) = (x_1, x_2, \ldots, x_k, Y) \qquad (2.1)$$

где Y – целевая переменная, которую необходимо классифицировать и проанализировать;

$\mathbf{x} = (x_1, x_2, \ldots, x_k)$ - вектор переменных, используемых для решения данной задачи.

Требуется построить алгоритм дерева решений, способный классифицировать зависимую целевую переменную Y.

Алгоритм дерева решений разбивает набор данных на все меньшие и меньшие подмножества, что приводит к постепенному развитию связанного дерева решений.

Конечный результат выполнения алгоритм дерева решений - дерево с узлами решения и конечными узлами.

Узел принятия решения имеет две или более ветвей. Конечный узел представляет собой классификацию или решение.

Самый верхний узел решений в дереве, который соответствует лучшему предиктору, называемому корневым узлом [17].

Деревья решений могут обрабатывать как категориальные, так и числовые данные.

К преимуществам данного метода относятся:

− простота понимания и интерпретации;

− не требуется большая точность данных;

− помогает определить худшие, лучшие и ожидаемые значения для разных сценариев;

− может приняться с другими методами принятия решений.

Недостатки деревьев решений:

− нестабильность, так ка небольшое изменение в данных может привести к значительному изменению структуры дерева оптимальных решений;

− невысокая точность;

− для данных, включающих в себя категориальные переменные с различным количеством уровней, прирост информации в деревьях решений смещается в пользу атрибутов с большим количеством уровней;

− алгоритм построения может быть очень сложным, особенно если многие значения являются неопределенными и / или если многие результаты связаны между собой.

Деревья решений обычно используются в исследовании операций, особенно в анализе решений, чтобы помочь определить стратегию, которая,

скорее всего, достигнет цели, но также являются популярным инструментом в машинном обучении.

Следует отметить, что алгоритмы на основе деревьев решений довольно широко применяются в системах анализа данных для образовательной сферы, в том числе для анализа успеваемости студентов вуза (рисунок 2.1) [4].

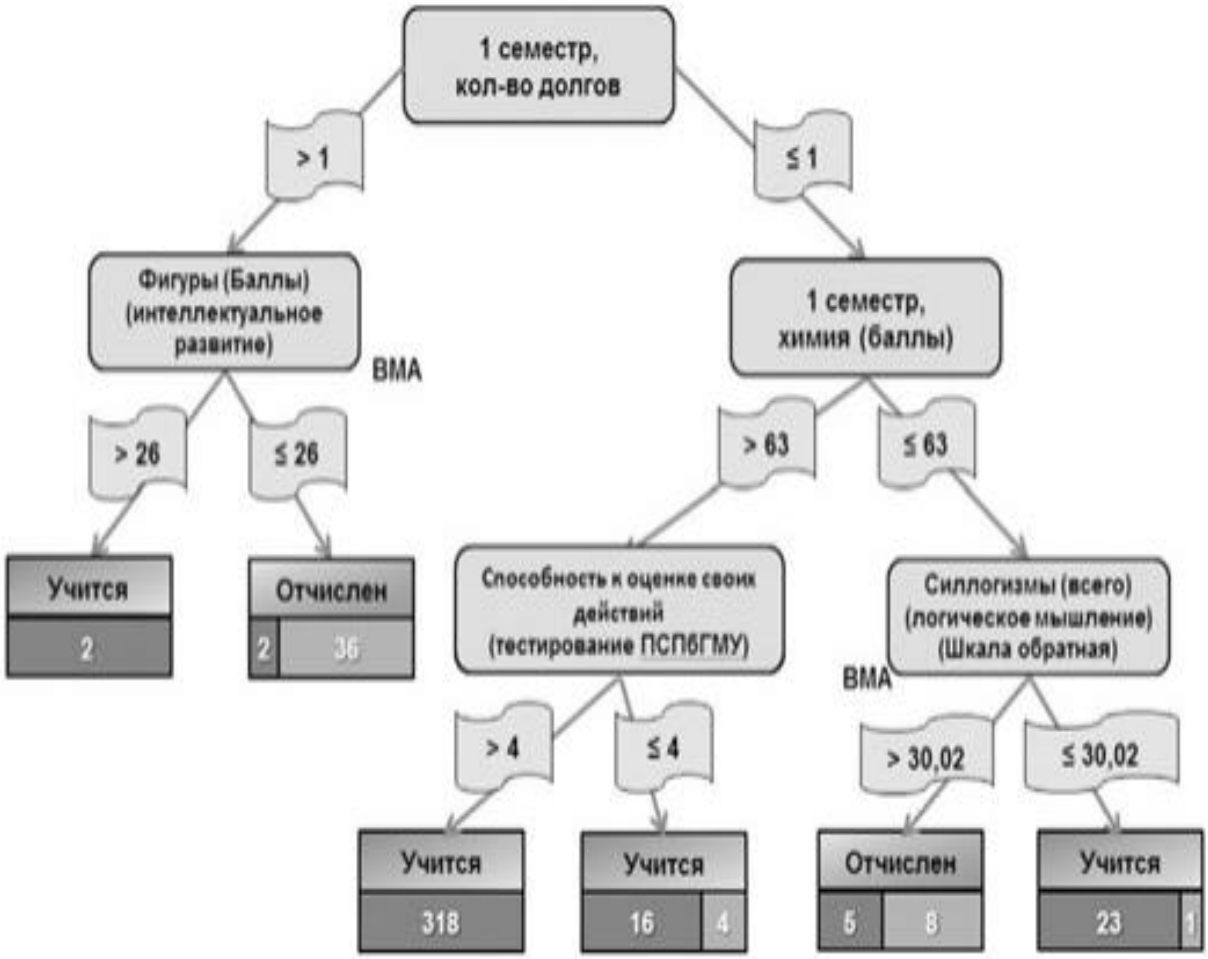

Рисунок 2.1 – Пример дерева решений, построенного по данным успеваемости студентов

Как показал анализ источников наиболее востребованным в АОД алгоритмом классификации на основе деревьев решений является алгоритм C4.5 [8].

В каждом узле дерева C4.5 выбирает атрибут данных, который наиболее эффективно разбивает его набор выборок на подмножества, обогащенные одним или другим классом.

Критерием разбиения в C4.5 является нормированный прирост информации (разница в энтропии), определяемый коэффициентом прироста Gain Ratio.

Атрибут с наибольшим нормированным выигрышем информации выбирается для принятия решения.

Пример алгоритма генерации дерева решений состоит из следующих шагов:

Шаг 1. Начало.

Шаг 2. Забор данных, введенных пользователем.

$I_n=\{I_1,\ldots,I_n\}$

Шаг 3. Подготовка набора данных.

$Dn=\{\{I_1,\ldots,I_n\}D\}$

Шаг 4. Очистка набора данных.

$D_I=\{S_1,\ldots S_n, C_1,\ldots..C_n, I_1,\ldots..I_n, a_1,\ldots..a_n\}$

Шаг 5. Обработка.

Пока ($D_n!=0$)

{

Если ($a_n==I_n$)

Проверить C_n, S_n;

}

Шаг 6. Генерация результатов.

$R=\{S_c, S_n , C_n\}$,

где:

I_n – данные, введенные пользователем;

D_n – набор данных;

D – база данных;

D_I – контент набора данных.

S_c- баллы;

$a_1,\ldots a_n$ – ответы;

$S_1,\ldots S_n$ – предмет;

C_1,….C_n – категории.

Код алгоритма C4.5 на языке Python представлен на рисунке 2.2.

```
import math
import utils
def freq(table, col, v):
    """ Возвращает число варинтов _v_
        в столбце _col_ таблицы _table_.
    """
    return table[col].count(v)
def info(table, res_col):
    """ Вычисляет энтропию таблицы _table_
        где res_col столбец = _res_col_.
    """
    s = 0 # sum
    for v in utils.deldup(table[res_col]):
        p = freq(table, res_col, v) / float(len(table[res_col]))
        s += p * math.log(p, 2)
    return -s
def infox(table, col, res_col):
    """ Вычисляет энтропию таблицы _table_
        После разбиения на две подтаблицы с column _col_.
    """
    s = 0 # sum
    for subt in utils.get_subtables(table, col):
        s += (float(len(subt[col])) / len(table[col])) * info(subt, res_col)
    return s
def gain(table, x, res_col):
    """ Критерий для выбора атрибутов для разбиения.
    """
    return info(table, res_col) - infox(table, x, res_col)
```

Рисунок 2.2 - Код алгоритма C4.5 на языке Python

Необходимо отметить, что C4.5 относится к жадным алгоритмам и восприимчив к шумам.

В настоящее время используется более эффективная модификация данного алгоритма - алгоритм J48.

2.1.2 Алгоритмы анализа успеваемости на основе наивного байесовского классификатора

Наивный байесовский классификатор (алгоритм) - это метод классификации, основанный на теореме Байеса с предположением независимости среди предикторов.

Проще говоря, наивный байесовский классификатор предполагает, что наличие определенной функции в классе не связано с наличием любой другой функции.

Абстрактно наивный байесовский алгоритм (НБА) - это модель условной вероятности. Другими словами, байесовская классификация - это алгоритм, основанный на байесовском правиле условной вероятности.

Байесовское правило (теорема Байеса) - это метод оценки вероятности свойства с учетом набора данных в качестве доказательства или ввода.

Математически байесовское правило описывается следующим образом (2.2):

$$p(C \mid F_1,...,F_n) = \frac{p(C)\,p(F_1,...,F_n \mid C)}{p(F_1,...,F_n)} \qquad (2.2)$$

где C - переменная класса, зависящая от переменных $F_1,\ldots, F_n$.

НБА определяет класс переменной C как наиболее вероятный из всех возможных классов с помощью оценки (2.3) апостериорного максимума (MAP):

$$\mathrm{C}_{\text{мар}} = \arg\max_{c \in C} p(C=c) \prod_{i=1}^{n} p(F_i = f_i \mid C = c) \qquad (2.3)$$

Рассмотрим пример системы анализа успеваемости студентов, использующей НБА (рисунок 2.3) [28].

Рисунок 2.3 - Структурная схема системы анализа успеваемости, использующей НБА

Наивная байесовская модель проста в интерпретации и особенно полезна для очень больших наборов данных. Известно, что наряду с простотой, НБА по своим возможностям превосходит даже самые сложные методы классификации.

НБА обычно применяются для прогнозирования данных на основе исторических результатов.

Однако у данного метода есть недостатки.

Если у категориальной переменной есть категория (в наборе тестовых данных), которая не наблюдалась в наборе обучающих данных, то модель назначит нулевую вероятность и не сможет сделать прогноз.

Это часто называют нулевой частотой. Чтобы решить эту проблему, можно использовать метод сглаживания.

Одним из самых простых методов сглаживания является метод Лапласа.

2.1.3 Алгоритмы анализа успеваемости на основе нейронных сетей

Нейронные сети - это алгоритмы, которые имитируют работу человеческого мозга. Они состоят из массива взаимосвязанных узлов, которые обмениваются информацией друг с другом, сравнимо с тем, как нейроны мозга, связанные дендритами и аксонами, обмениваются информацией [11].

Нейронные сети относятся к обучаемым алгоритмам. Они учатся итеративно с течением времени, наблюдая за различными примерами, подобно тому, как человек может учиться посредством наблюдения. Однако, в отличие от людей, нейронным сетям часто требуются большее количество наблюдений для достижения достаточной прогностической способности.

Нейронные сети отличаются от других алгоритмов классификации тем, что внутренне информация обрабатывается параллельно. Это отличается их от последовательной обработки, которую используют многие другие алгоритмы, например, классификаторы дерева решений.

Нейронные сети активно применяются для прогнозирования успеваемости студентов.

Рассмотрим пример математической модели прогнозирования успеваемости студентов на основе радиальной базисной нейронной сети имеет вид [5].

Простейшая нейронная сеть радиального типа функционирует по принципу многомерной интерполяции, состоящей в отображении p различных входных векторов x_i, где i=1,2,…,p, из входного N-мерного пространства во множество из p чисел d_i, где i=1,2,…,p.

Для реализации данного процесса нужно использовать p скрытых нейронов радиального типа и задать такую функцию отображения $F(x)$, для которой выполняется условие интерполяции вида (2.4):

$$F(x)=d_i. \qquad (2.4)$$

Использование p скрытых нейронов, соединяемых связями с весами с выходными линейными нейронами, означает формирование выходных

сигналов сети путем суммирования взвешенных значений соответствующих базисных функций.

На рисунке 2.4 изображена структурная схема радиальной нейронной сети, позволяющая спрогнозировать успеваемость студента по конкретной дисциплине.

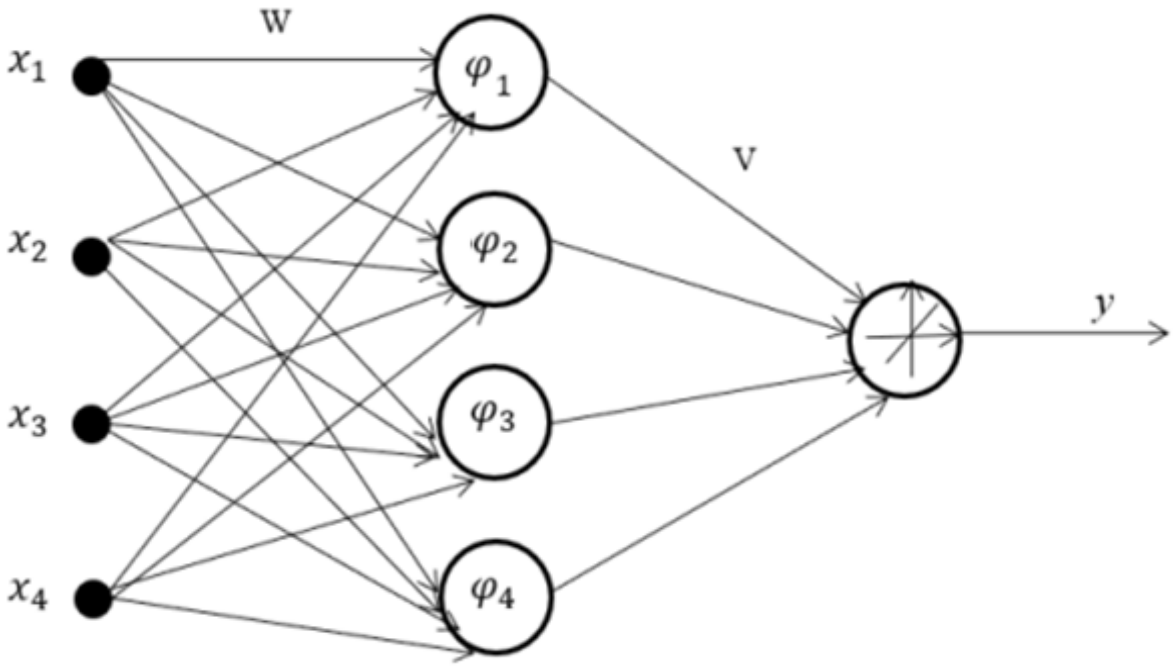

Рисунок 2.4 – Структурная схема радиальной базисной нейронной сети для прогнозирования успеваемости студента

На рисунке 2.4 изображены:

*x*1 — средняя оценка студента по предшествующим смежным дисциплинам, освоение которых в полном объеме является необходимым условием;

*x*2 — свободное посещение;

*x*3 — оценка за тестирование остаточных знаний, проводимое перед началом изучения данной дисциплины;

*x*4 — наличие задолженностей по другим дисциплинам.

$\varphi_1,..,\varphi_4$ — радиальные функции, на вход которых подаём исходные данные с весовыми коэффициентами W;

V - линейный нейрон, который выполняет суммирование выходных сигналов от скрытых нейронов с заданными весовыми коэффициентами;

y — прогнозируемая оценка по дисциплине, получаемая на выходе сети.

Наиболее часто для решения задач классификации образовательных данных используются алгоритмы, основанные на персептроне, например, алгоритм WINNOW.

Персептрон поддерживает двухклассовую классификацию.

Иными словами, он классифицирует новый экземпляр объекта x в класс 2, если выполняется условие (2.5) или в класс 1 в противном случае (2.5):

$$\sum_i x_i w_i > \theta. \qquad (2.5)$$

Он принимает экземпляры по одному и обновляет веса w_i при необходимости.

Персептрон инициализирует свои веса w_i и θ, а затем принимает новый экземпляр (x, y), применяя правило порога для вычисления предсказанного класса y′.

Если предсказанный класс верен (y′ = y), персептрон бездействует.

Однако, если прогнозируемый класс неверен, персептрон обновляет свои веса.

Наиболее распространенный способ использования алгоритма персептрона для обучения из серии обучающих экземпляров - это многократное выполнение алгоритма через обучающий набор до тех пор, пока он не найдет вектор предсказания, который является правильным для всего обучающего набора.

Это правило прогнозирования затем используется для прогнозирования меток в наборе тестов.

Достоинствами нейронных сетей являются:

− хранение информации по всей сети;

− умение работать с неполными знаниями;

− отказоустойчивость;

− распределенная память;

− возможность машинного обучения;

− возможность параллельной обработки.

Недостатками нейронных сетей являются:

− аппаратная зависимость;

− приблизительность результата;

− многоэтапность принятия решения;

− трудность обнаружения внутренних проблем.

Следует также отметить, что в отличие от классификаторов на основе деревьев решений, нейронная сеть не хранит какого-либо явного представления о том, как она достигла своего результата, что существенно снижает ее функциональные возможности.

В работе [24] был выполнен сравнительный анализ алгоритмов классификации для решения задач анализа успеваемости студентов вуза.

Атрибуты образовательных данных, используемых для анализа, приведены в таблице 2.1.

Таблица 2.1 – Атрибуты образовательных данных и их значения (используется 10 бальная шкала оценок)

Атрибут	Значения
1-й опрос	Отсутствует, Присутствует
1-е письменное задание	Оценка<3, 3≤Оценка≤6, Оценка>6
2-й опрос	Отсутствует, Присутствует
2-е письменное задание	Оценка<3, 3≤Оценка≤6, Оценка>6
3-й опрос	Отсутствует, Присутствует
3-е письменное задание	Оценка<3, 3≤Оценка≤6, Оценка>6
4-й опрос	Отсутствует, Присутствует
4-е письменное задание	Оценка<3, 3≤Оценка≤6, Оценка>6

Результаты сравнения алгоритмов сведены в таблицу 2.2.

Таблица 2.2 – Сравнение характеристик алгоритмов классификации для решения задач анализа успеваемости студентов вуза

Характеристика/ Алгоритм (макс. балл-3)	J48	НБА	WINNOW
Эффективность обработки больших данных	3	3	2
Точность результата	3	2	1
Простота интерпретации	3	2	2
Итого	9	7	5

Как следует из таблицы, наиболее высокие показатели у алгоритма J48, что делает его предпочтительным вариантом для классификации в задачах анализа успеваемости в вузе на основе больших данных.

2.2 Алгоритмы кластеризации для анализа успеваемости в вузе

Как показал анализ источников, для анализа успеваемости в вузе используются следующие алгоритмы кластеризации:

- иерархическая кластеризация;
- k-means.

Рассмотрим и сравним данные алгоритмы.

2.2.1 Алгоритмы иерархической кластеризации

Методы иерархической кластеризации являются подразделяются на агломеративные и дивизивные иерархические методы построения дендрограмм – деревьев, созданных на основе матрицы близости (рисунок 2.5) [27].

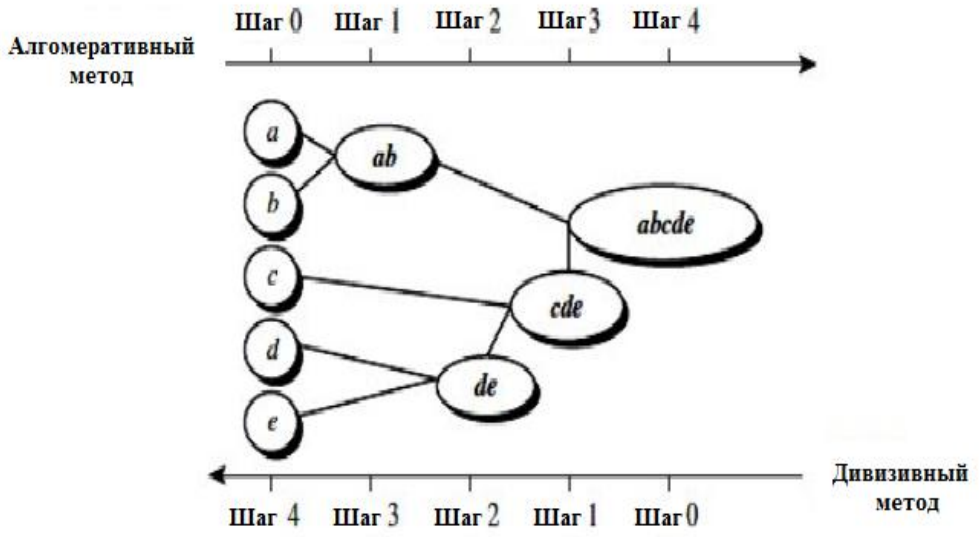

Рисунок 2.5 – Процесс иерархической кластеризации

В агломеративных алгоритмах новые кластеры создаются путем объединения более мелких кластеров, т.е. кластеры последовательно объединяются, пока не останется только один кластер (принцип «от листьев к стволу»).

Дивизивные иерархические алгоритмы основаны на делении большого кластера на мелкие.

Группы непрерывно разделяются, пока не будет столько кластеров, сколько объектов (принцип «от ствола к листьям»).

К алгоритмам иерархической кластеризации относятся [15]:

1. Метод ближайшего соседа.

Расстояние между двумя кластерами принимается равным минимальному расстоянию между двумя элементами из разных кластеров (2.6):

$$\min \{d(a,b) : a \in A, b \in B\}, \qquad (2.6)$$

где:

$d(a,b)$ – расстояние между элементами a и b, принадлежащими кластерам A и B, соответственно.

2. Метод дальнего соседа.

Расстояние между двумя кластерами принимается равным максимальному расстоянию между двумя элементами из разных кластеров (2.7):

$$\max \{d(a,b) : a \in A, b \in B\}, \qquad (2.7)$$

3. Метод Уорда.

В данном методе для оценки расстояний между кластерами используются дисперсионный анализ.

За расстояние между кластерами принимается приращение суммы квадратов расстояний объектов до центра кластера, получаемого в результате их объединения (2.8):

$$\Delta = \sum_i (x_i - \bar{x})^2 - \sum_{x_i \in A} (x_i - \bar{a})^2 - \sum_{x_i \in B} (x_i - \bar{b})^2. \qquad (2.8)$$

На каждом шаге алгоритма объединяются такие два кластера, которые приводят к минимальному увеличению дисперсии.

Данный метод применяется для задач с близко расположенными кластерами.

К преимуществам иерархической кластеризации относятся простота понимания и реализации.

Недостатки иерархической кластеризации состоят в том, что она редко обеспечивает лучшее решение, включает в себя множество произвольных решений, не работает с неполными данными, плохо работает со смешанными типами данных и плохо работает с большими массивами данных, что особенно критично для АОД.

2.2.2 Алгоритм k-means

Алгоритм кластеризации k-means относится к алгоритмам неиерархической кластеризации.

Данный алгоритм основан на генерации определенного количества непересекающихся, плоских (неиерархических) кластеров.

Математически алгоритм k-means описывается с помощью следующей формулы (2.6):

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2, \qquad (2.6)$$

где:

k-число кластеров;

S_i – полученные кластеры;

μ_i – центры масс (центроиды) векторов $x_j \in S_i$. (i =1,2,…,k).

Блок-схема алгоритма k-means изображена на рисунке 2.6.

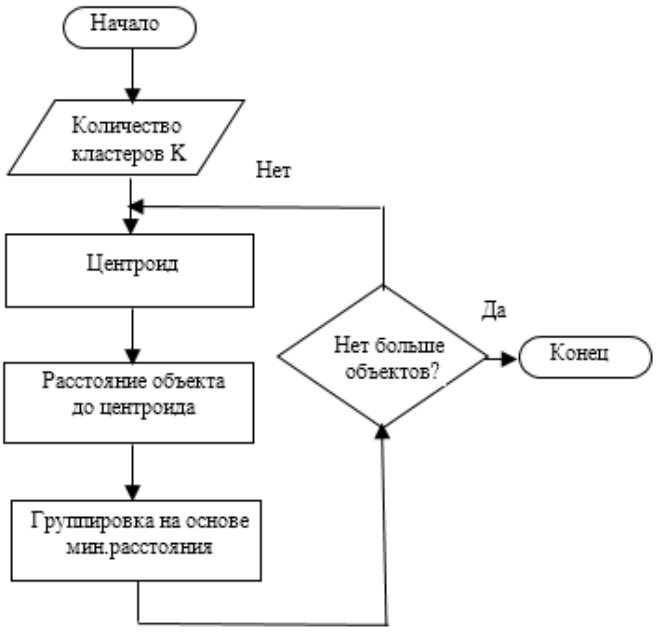

Рисунок 2.6 – Блок-схема алгоритма k-means

Достоинства алгоритма:

− для больших данных и малых k данный алгоритм в большинстве случаев работает быстрее, чем иерархическая кластеризация;

− k-means создает более плотные кластеры, чем иерархические кластеры;

− экземпляр может перейти на другой кластер, когда пересчитываются центроиды.

Код алгоритма k-means на языке Python + OpenCV представлен на рисунке 2.7.

```
if name == "main":

color_tab = [CV_RGB(255,0,0),CV_RGB(0,255,0),CV_RGB(100,100,255),
CV_RGB(255,0,255),CV_RGB(255,255,0)]
img = cvCreateImage(cvSize(500, 500), 8, 3)
rng = cvRNG(-1)
cvNamedWindow( "clusters", 1 )
while True:
   cluster_count = cvRandInt(rng)%(MAX_CLUSTERS-1) + 2
   sample_count = cvRandInt(rng)%999 + 1
   points = cvCreateMat(sample_count, 1, CV_32FC2)
   clusters = cvCreateMat(sample_count, 1, CV_32SC1)
   for k in range(cluster_count):
      first = k*sample_count/cluster_count
      last = (k+1)*sample_count/cluster_count if k != cluster_count else
sample_count
      if first < last:
         cvRandArr(rng, cvGetRows(points, None, first, last),
CV_RAND_NORMAL,
            cvScalar(cvRandInt(rng)%img.width,cvRandInt(rng)%img.height),
cvScalar(img.width*0.1,img.height*0.1))
   cvRandShuffle( points, rng )
   # K Means Clustering
   cvKMeans2(points, cluster_count, clusters,
cvTermCriteria(CV_TERMCRIT_EPS+CV_TERMCRIT_ITER, 10, 1.0))
   cvZero( img )
   for i in range(sample_count):
      pt = points[i,0]
      cvCircle(img, cvPoint(cvRound(pt[0]), cvRound(pt[1])), 2,
color_tab[clusters[i,0]], CV_FILLED, CV_AA, 0)
  cvShowImage( "clusters", img )
  if '%c' % (cvWaitKey(0) & 255) in ['\x1b','q','Q']: # 'ESC'
     break
```

Рисунок 2.7 - Код алгоритма k-means на языке Python + OpenCV

Недостатки алгоритма k-means:

− сложно предсказать значение k;

− плохо работает с глобальными кластерами;

− различные начальные разделы могут привести к различным конечным кластерам;

− не работает с кластерами (в исходных данных) разного размера и разной плотности.

Существует разновидность алгоритма k-means - алгоритм «дальнего первого» (Farthest first clustering) [20], в котором каждый центр кластера размещается по очереди в точке, наиболее удаленной от существующих центров кластера. Эта точка должна находиться в области данных.

Такой подход значительно ускоряет кластеризацию в большинстве случаев, так как требует меньше переназначений и настроек.

С учетом результатов известных исследований [9,27] был выполнен сравнительный анализ алгоритмов кластеризации для решения задач анализа успеваемости студентов вуза.

Сравнение алгоритмов произведено с помощью программы Weka, характеристики которой описаны в следующей главе.

Результаты анализа алгоритмов кластеризации сведены в таблицу 2.3.

Таблица 2.3 – Сравнение характеристик алгоритмов кластеризации для решения задач анализа успеваемости студентов вуза

Характеристика/ Алгоритм	k-means	Иерархические алгоритмы
1	2	3
Эффективность обработки больших данных	+	-

1	2	3
Способность выделять кластеры разной структуры	+	-
Вычислительная сложность	+	-
Итого	3	0

Как следует из таблицы, наиболее высокие показатели у алгоритма k-means, что обосновывает его широкое применение для кластеризации в задачах анализа успеваемости в вузе на основе больших данных.

Выводы по второй главе

1. Среди алгоритмов классификации наиболее эффективными для решения задач АОД являются алгоритмы на основе деревьев решения, в частности алгоритм J48.

2. Наиболее эффективным алгоритмом кластеризации для решения задач анализа данных является алгоритм неиерархической кластеризации k-means.

Глава 3 ОЦЕНКА ЭФФЕКТИВНОСТИ АЛГОРИТМОВ ДЛЯ АНАЛИЗА УСПЕВАЕМОСТИ В ВУЗЕ

Для анализа и оценки эффективности алгоритмов J48 и k-means использована программа интеллектуального анализа данных Weka.

Программа Weka (Waikato Environment for Knowledge Analysis) — свободное программное обеспечение для анализа данных и машинного обучения, написанное на Java в Университете Уаикато (Новая Зеландия), распространяющееся по лицензии GNU GPL (рисунок 3.1) [16].

Рисунок 3.1 – Главное окно программы Weka

Функционально Weka - это набор алгоритмов машинного обучения для задач интеллектуального анализа данных. Она содержит инструменты для подготовки, классификации, регрессии, кластеризации данных, анализа правил сопоставления и визуализации.

Преимущества Weka:

− свободное распространение;

− переносимость, поскольку она полностью реализована на языке программирования Java и, следовательно, работает практически на любой современной вычислительной платформе;

− обширная коллекция методов предварительной обработки данных и моделирования;

− простота использования благодаря графическому интерфейсу пользователя.

Weka поддерживает несколько стандартных задач интеллектуального анализа данных, в частности, предварительную обработку данных, кластеризацию, классификацию, регрессию, визуализацию и выбор функций.

Все методы программы основаны на предположении, что данные доступны в виде одного плоского файла или отношения, где каждая точка данных описывается фиксированным количеством атрибутов (обычно это числовые или номинальные атрибуты, но также поддерживаются некоторые другие типы атрибутов).

Фрагмент структурной схемы платформы Weka представлен на рисунке 3.2.

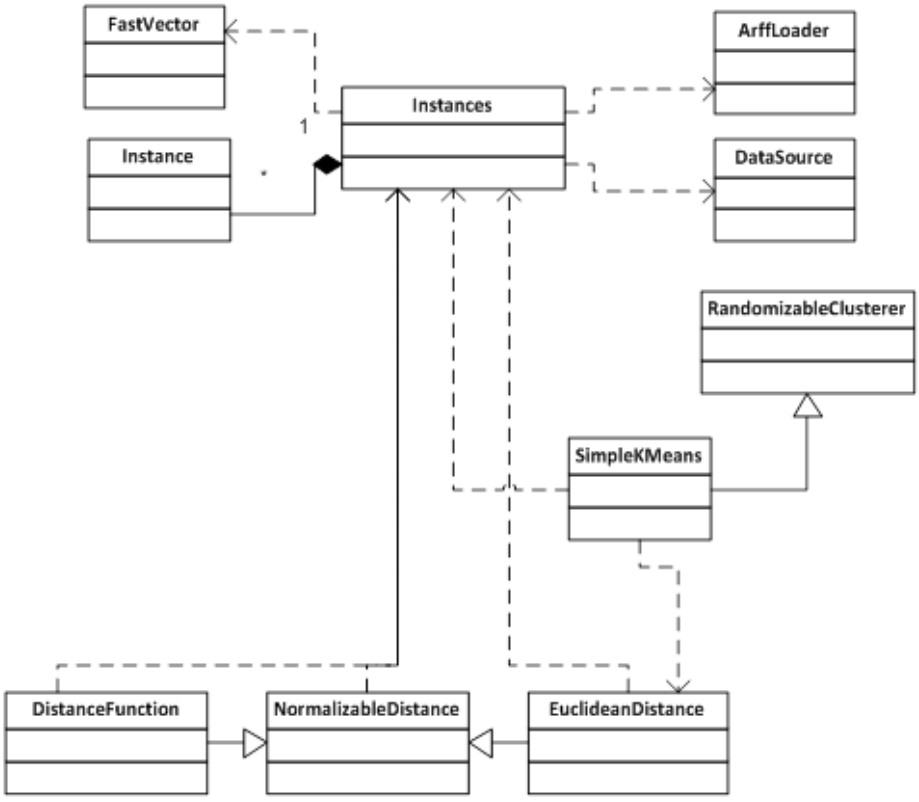

Рисунок 3.2 - Фрагмент структурной схемы платформы Weka

Программа обеспечивает доступ к базам данных SQL с помощью Java Database Connectivity и может обрабатывать результат, возвращаемый запросом к базе данных.

Weka предоставляет доступ к глубокому обучению на основе библиотеки Deeplearning4j.

Следует отметить, что программа не поддерживает мультиреляционный Data mining.

Другой важной областью, которая в настоящее время не охватывается алгоритмами, включенными в библиотеку программы, является моделирование последовательности.

3.1 Оценка эффективности алгоритма классификации J48

Предварительно необходимо создать тестовый файл успеваемости студентов.

Создаем таблицу успеваемости в книге Excel (рисунок 3.3).

	A	B	C
1	Math	Physic	IT
2	3	4	4
3	3	4	4
4	5	4	4
5	3	4	4
6	5	4	4
7	3	3	3
8	3	4	4
9	3	4	4
10	5	4	4
11	5	4	4
12	3	3	3
13	4	4	4
14	5	5	5
15	4	4	4
16	3	4	4
17	4	3	4
18	4	3	4
19	4	4	4
20	3	3	3
21	4	4	4

Рисунок 3.3 – Таблица успеваемости студентов по 3-м дисциплинам

Выполняем экспорт файла в книги Excel в формат CSV для загрузки в программу Weka (рисунок 3.4).

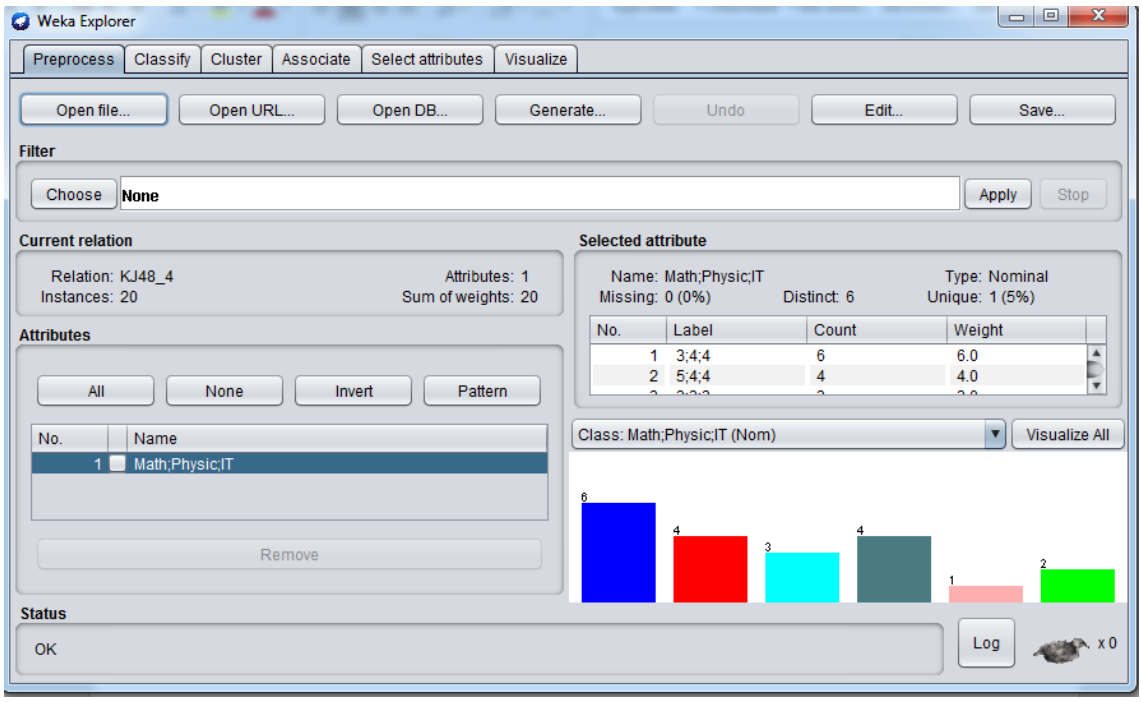

Рисунок 3.4 – Экран формы подготовки данных для анализа

Выбираем алгоритм J48 и запускаем процедуру классификации (рисунок 3.5).

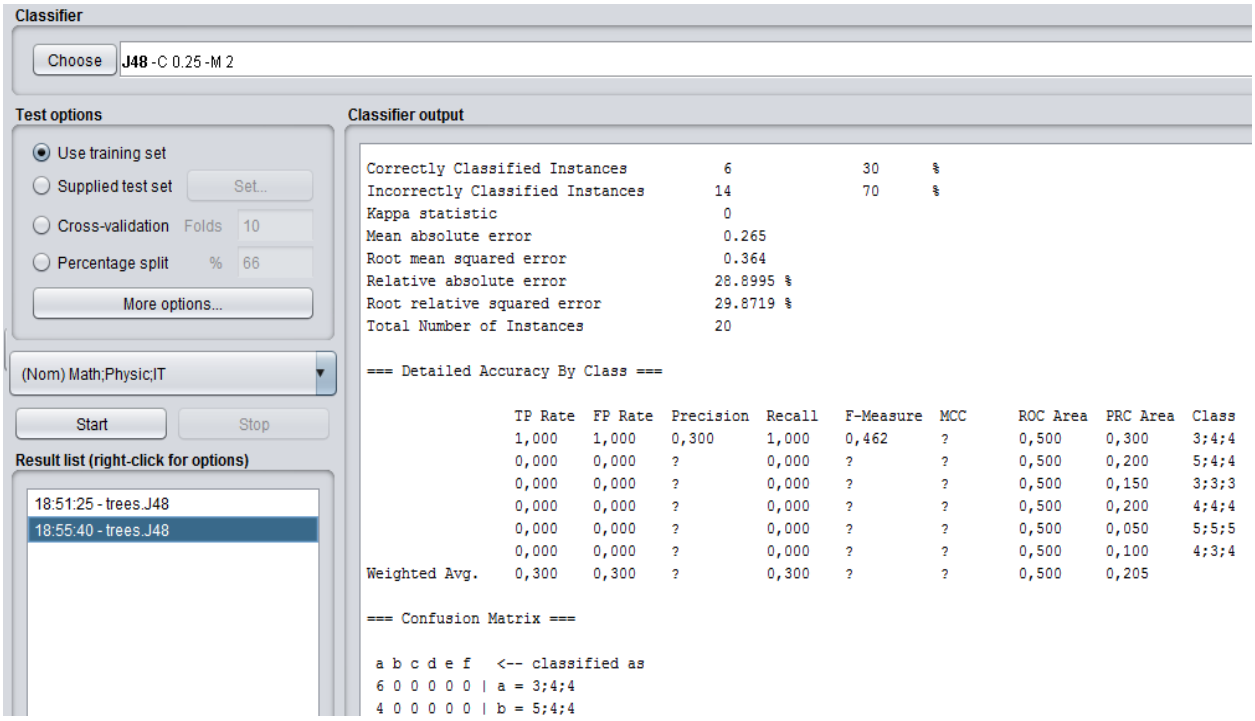

Рисунок 3.5 – Экран результатов классификации

На рисунке 3.6 изображена кривая прироста классификации.

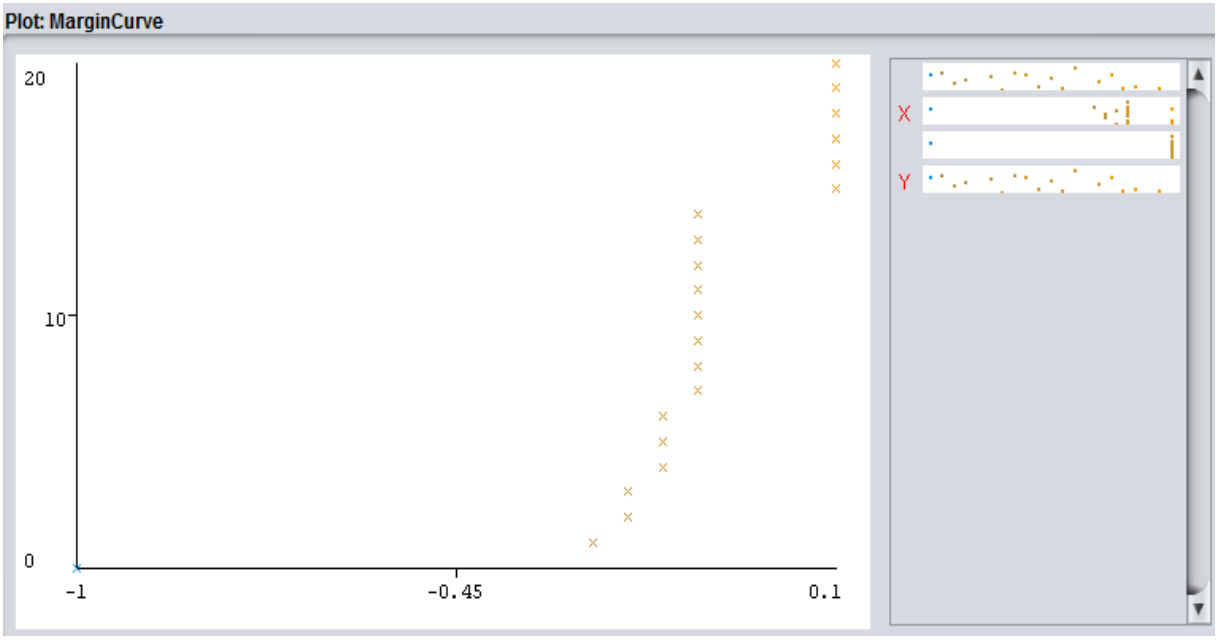

Рисунок 3.6 - Кривая прироста классификации

Как следует из отчета анализа средняя ошибка классификации не превышает 0.265 и среднеквадратическая ошибка равна 0.364, что позволяет сделать вывод о высокой точности алгоритма J48.

3.2 Оценка эффективности алгоритма кластеризации k-means

Выбираем алгоритм SimpleKMeans и запускаем процедуру кластеризации (рисунок 3.7).

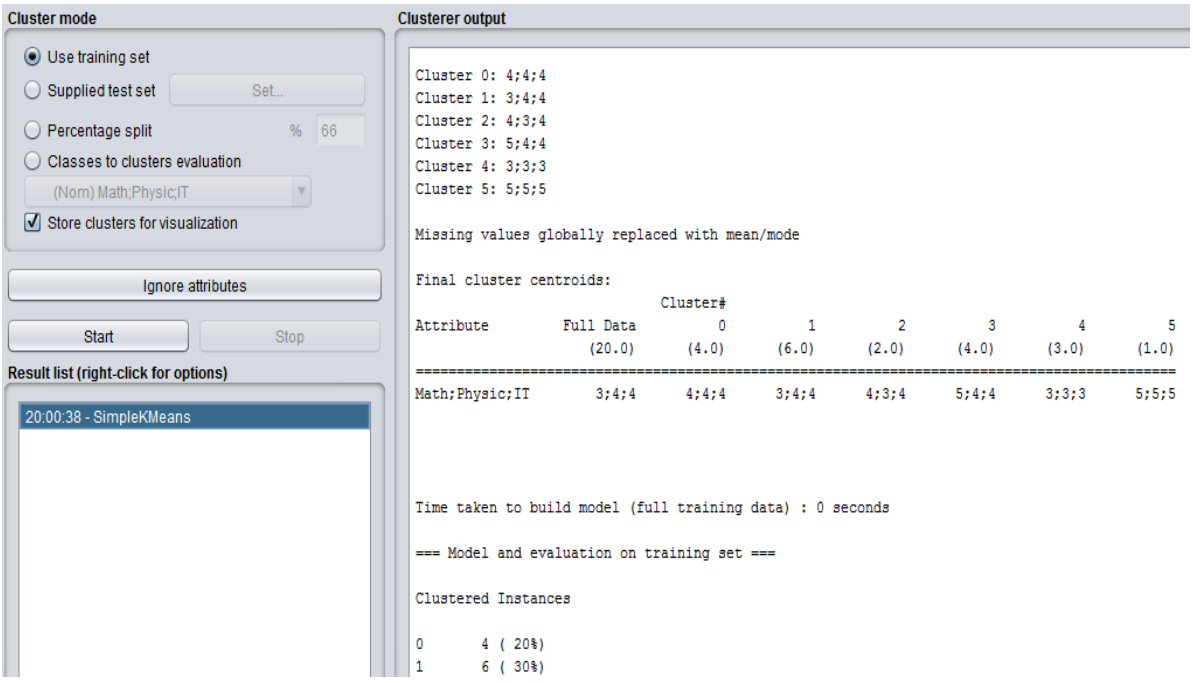

Рисунок 3.7 – Экран результатов кластеризации

На рисунке 3.8 представлен график визуализации кластеров.

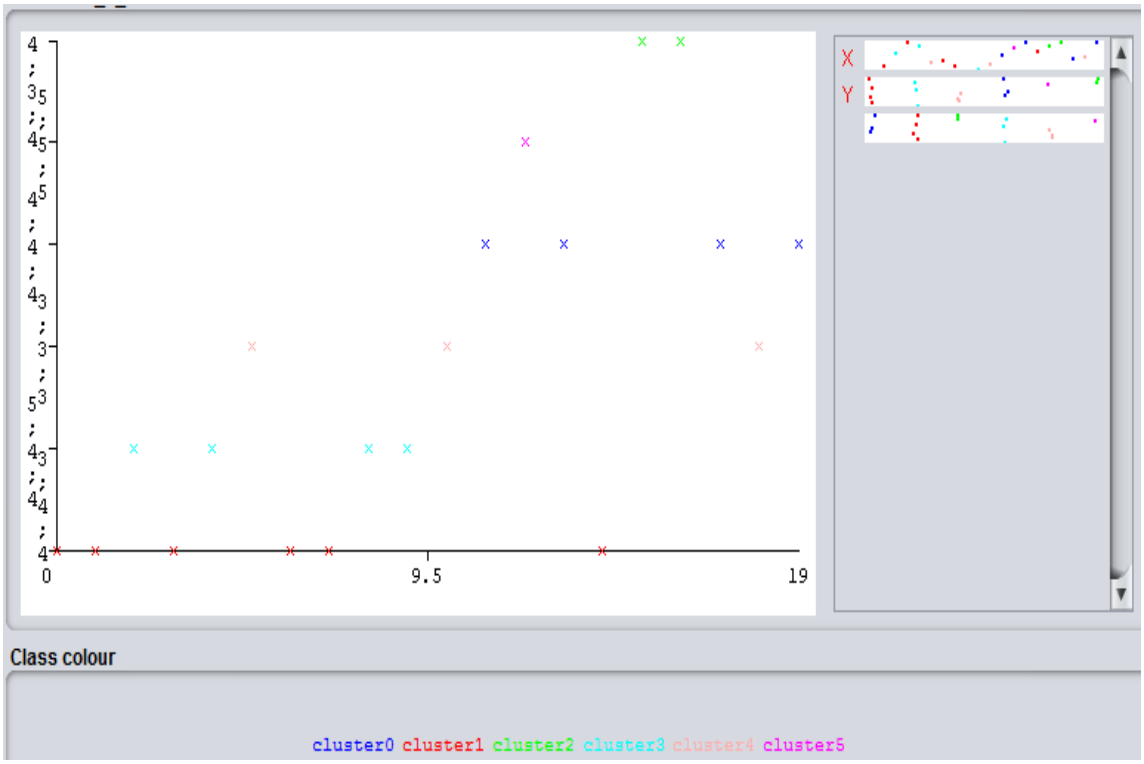

Рисунок 3.8 – График визуализации кластеров

Как следует из отчета анализа внутри кластера сумма квадратов ошибок равна 0, что позволяет сделать вывод о высокой точности алгоритма k-means.

Полученные результаты позволяют сделать вывод, что выбранные алгоритмы J48 и k-means могут использоваться для анализа успеваемости студентов в вузе.

Выводы по главе 3

1. Программа Weka поддерживает такие стандартные задачи интеллектуального анализа данных как кластеризация и классификация, что позволяет ее использовать для оценки эффективности алгоритмов J48 и k-means.

2. Как показал анализ тестовых данных, алгоритмы J48 и k-means обеспечивают высокую точность классификации и кластеризации, что

позволяет сделать вывод о высокой эффективности указанных алгоритмов и возможности их использования для анализа успеваемости студентов в вузе.

ЗАКЛЮЧЕНИЕ

Представленная бакалаврская работа посвящена актуальной проблеме применения алгоритмов EDM для анализа успеваемости в вузе.

В ходе выполнения бакалаврской работы достигнуты следующие результаты:

1. Проанализирована научная и учебно-методическая литература по исследуемой проблеме.

2. Проанализированы методы интеллектуального анализа массивов данных в образовательном процессе вуза. Как показал анализ, задачи анализа успеваемости в вузе следует рассматривать как задачи классификации и кластеризации интеллектуального анализа образовательных данных.

3. Проанализированы известные алгоритмы EDM для анализа успеваемости в вузе. Сравнительный анализ алгоритмов классификации и кластеризации показал, что наиболее высокие показатели у алгоритмов J48 и k-means, соответственно.

4. В программе Weka выполнен анализ тестовых данных успеваемости студентов, который подтвердил высокую эффективность выбранных алгоритмов.

Результаты бакалаврской работы могут быть рекомендованы для решения задач анализа успеваемости студентов вуза на основе методов интеллектуального анализа образовательных данных.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

Нормативно-правовые акты

1. ГОСТ 19.701-90 ЕСПД. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения.

Научная и методическая литература

2. Андреев И.М. Описание алгоритма CART / И.М. Андреев // Математика в приложениях. -2004. - №3-4. –С.48-53.

3. Барсегян А.А. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP / А.А. Барсегян, М.С. Куприянов, В.В. Степаненко, И.И. Холод. - СПб.: БХВ-Петербург, 2007. - 384 с.

4. Исаева Е. Р. и [др.] Поиск прогностических критериев академической успеваемости / Е. Р. Исаева и [др.] // Университетское управление: практика и анализ. – 2017. –Т. 21. - №2. –С. 163-172.

5. Прошкина Е.Н. Анализ и прогнозирование успеваемости студентов на основе радиальной базисной нейронной сети / Е.Н. Прошкина, Балашова И.Ю. // Материалы III Междунар. науч. конф. (г. Самара, март 2018 г.). – Казань : Молодой ученый, 2018. - С.24-27.

6. Солтан Г.Ж. и [др.] Интелектуальный анализ данных в задачах управления качеством образовательного процесса / Г.Ж. Солтан и др. // Инженерное образование. -2013. -№ 13. – С. 36-43.

7. Шевченко В.А. Прогнозирование успеваемости студентов на основе методов кластерного анализа / В.А. Шевчено // Вестник ХНАДУ. – 2015. – Вып. 68. – С. 15-18.

Электронные ресурсы

8. Алгоритм C4.5 [Электронный ресурс]. — Режим доступа: http://datascientist.one/algorithm-c4-5/ (дата обращения 02.04.2019).

9. Беликова М.Ю. и [др.] Экспериментальное сравнение алгоритмов кластеризации в задаче группировки данных о грозовых разрядах

[Электронный ресурс]. — Режим доступа: https://e-notabene.ru/kp/article_25261.html (дата обращения 02.04.2019).

10. Белоножко П.П. Анализ образовательных данных: направления и перспективы применения [Электронный ресурс] / П.П. Белоножко, А.П. Карпенко, Д.А. Храмов // Интернет-журнал «НАУКОВЕДЕНИЕ». - 2017. – Т. 9. - №4. — Режим доступа: http://naukovedenie.ru/PDF/15TVN417.pdf (дата обращения 02.04.2019).

11. Барский А. Б. Введение в нейронные сети [Электронный ресурс] / А. Б. Барский. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 358 c. — Режим доступа: http://www.iprbookshop.ru/52144.html (дата обращения 02.04.2019).

12. Билл Фрэнкс Революция в аналитике [Электронный ресурс] : как в эпоху Big Data улучшить ваш бизнес с помощью операционной аналитики / Фрэнкс Билл. — М. : Альпина Паблишер, 2017. — Режим доступа: http://www.iprbookshop.ru/58563.html (дата обращения 02.04.2019).

13. Гранков М.В. Анализ и кластеризация основных факторов, влияющих на успеваемость учебных групп вуза [Электронный ресурс] / М.В. Гранков, В.М. Аль-Габри, М.Ю. Горлова // Инженерный вестник Дона. – 2016. -№4. — Режим доступа: ivdon.ru/ru/magazine/archive/n4y2016/3775 (дата обращения 02.04.2019).

14. Дерево решений [Электронный ресурс]. — Режим доступа: https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D1%80%D0%B5%D0%B2%D0%BE_%D1%80%D0%B5%D1%88%D0%B5%D0%BD%D0%B8%D0%B9 (дата обращения 02.04.2019).

15. Иерархическая кластеризация [Электронный ресурс]. — Режим доступа: https://ru.wikipedia.org/wiki/%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BA%D0%BB%D0%B0%D1%81%D1%82%D0%B5%D1%80%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F (дата обращения 02.04.2019).

16. Программа интеллектуального анализа данных WEKA [Электронный ресурс]. — Режим доступа:https://www.cs.waikato.ac.nz/ml/index.html (дата обращения 02.04.2019).

17. Чубукова И. А. Data Mining [Электронный ресурс] / И. А. Чубукова. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 470 с. — Режим доступа: http://www.iprbookshop.ru/56315.html (дата обращения 02.04.2019).

18. Data Mining and Big Data Analytics Technical Committee [Электронный ресурс]. — Режим доступа: https://cis.ieee.org/technical-committees/data-mining-and-big-data-analytics-technical-committee (дата обращения 02.04.2019).

19. Educational Data Mining [Электронный ресурс]. — Режим доступа: http://educationaldatamining.org/ (дата обращения 02.04.2019).

20. Farthest First Algorithm [Электронный ресурс]. — Режим доступа: https://www.coursehero.com/file/p59ksrcb/3-FARTHEST-FIRST-ALGORITHM-Farthest-first-is-a-Variant-of-K-means-that-places/ (дата обращения 02.04.2019).

21. What Is Student Data? [Электронный ресурс]. — Режим доступа: https://dataqualitycampaign.org/resource/what-is-student-data/ (дата обращения 02.04.2019).

Литература на иностранном языке

22. Ahmed A.B., Elaraby I.S. Data Mining: A prediction for Student's Performance Using Classification Method, World Journal of Computer Application and Technology 2(2): 43-47, 2014.

23. Baker R. S. Educational data mining: An advance for intelligent systems in education. IEEE Intelligent Systems, 2014, 29 (3), pp. 78–82.

24. Kotsiantis S., C. Pierrakeas and P. Pintelas. Efficiency of Machine Learning Techniques in Predicting Students’ Performance in Distance Learning

Systems, TR-02-03, Department of Mathematics, University of Patras, Hellas, 2002, pp. 297-304.

25. Moscoso-Zea O., Vizcaino M.,ши Mora S. L. Evaluation of Methods and Algorithms of Educational Data Mining, 7th Research in Engineering Education Symposium, 2017, pp. 972-980.

26. Narwal S., Mintwal K. Comparison the Various Clustering and Classification Algorithms of WEKA Tools, International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3,2013, pp.866-878.

27. Pallavi, Godara S. A Comparative Performance Analysis of Clustering Algorithms, International Journal of Engineering Research and Applications, vol. 1(3), pp.441-445.

28. Ramya P., Mahesh Kumar M. Student Performance Analysis Using Educational Data Mining, Special Issue International Journal of Computer Science and Information Security, vol. 14, 2016.