TOMSK ТОМСКИЙ
POLYTECHNIC ПОЛИТЕХНИЧЕСКИЙ

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа Инженерная школа информационных технологий и робототехники
Направление подготовки 09.04.04 Программная инженерия
ООП/ОПОП Технологии больших данных
Отделение школы (НОЦ) Информационных технологий

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА МАГИСТРАНТА

Тема работы
Методы и инструменты машинного обучения для исследования факторов, влияющих на качество строительных смесей

УДК 004.415.2:004.421:004.85:691.53

Обучающийся

Группа	ФИО	Подпись	Дата
8ПМ1И	Геращенков Вадим Евгеньевич		10.06.2023 г.

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент ОИТ ИШИТР	Аксёнов С. В.	к.т.н.		10.06.2023 г.

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
профессор ОСГН ШБИП	Профессор Жиронкин С.А.	к.э.н		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
профессор ООД ШБИП	Федорчук Ю. М.	к.т.н.		

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП, должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент ОИТ ИШИТР	Губин Е. И.	к.ф.-м.н.		

Томск – 2023 г.

Томск – 2023

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ООП
по направлению 09.04.04 «Программная инженерия»

Код компетенции	Наименование компетенции
Универсальные компетенции	
УК(У)-1	Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий
УК(У)-2	Способен управлять проектом на всех этапах его жизненного цикла
УК(У)-3	Способен организовывать и руководить работой команды, вырабатывая командную стратегию для достижения поставленной цели
УК(У)-4	Способен применять современные коммуникативные технологии, в том числе на иностранном (-ых) языке (-ах), для академического и профессионального взаимодействия
УК(У)-5	Способен анализировать и учитывать разнообразие культур в процессе межкультурного взаимодействия
УК(У)-6	Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки
Общепрофессиональные компетенции	
ОПК(У)-1	Способен самостоятельно приобретать, развивать и применять математические, естественно-научные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте
ОПК(У)-2	Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач
ОПК(У)-3	Способен анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями
ОПК(У)-4	Способен применять на практике новые научные принципы и методы исследований
ОПК(У)-5	Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем

ОПК(У)-6	Способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности
ОПК(У)-7	Способен применять при решении профессиональных задач методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе, в глобальных компьютерных сетях
ОПК(У)-8	Способен осуществлять эффективное управление разработкой программных средств и проектов
Профессиональные компетенции	
ПК(У)-1	Способен к созданию вариантов архитектуры программного средства
ПК(У)-2	Способен разрабатывать и администрировать системы управления базам данных
ПК(У)-3	Способен управлять процессами и проектами по созданию (модификации) информационных ресурсов
ПК(У)-4	Способен проектировать и организовывать учебный процесс по образовательным программам с использованием современных образовательных технологий
ПК(У)-5	Способен осуществлять руководство разработкой комплексных проектов на всех стадиях и этапах выполнения работ

TOMSK UNIVERSITY ТОМСКИЙ УНИВЕРСИТЕТ

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа Инженерная школа информационных технологий и робототехники
Направление подготовки 09.04.04 Программная инженерия
ООП/ОПОП Технологии больших данных
Отделение школы (НОЦ) Информационных технологий

УТВЕРЖДАЮ:
Руководитель ООП
_____ _____ Губин Е. И.
(подпись) (дата) (Ф.И.О.)

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

Обучающийся:

Группа	ФИО
8ПМ1И	Геращенков Вадим Евгеньевич

Тема работы:

Методы и инструменты машинного обучения для исследования факторов, влияющих на качество строительных смесей	
Утверждена приказом директора (дата, номер)	№ 146-39/с от 26.05.2023 г.

Срок сдачи обучающимся выполненной работы:	15.06.2023 г.

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе *(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.)*	Объектом исследования являются данные строительной смеси, полученные лабораторным путем. Предмет исследования -инструментарий Data Science и методы машинного обучения в анализе данных Для анализа доступна электронная таблица, содержащая сведения о 1030 компонентах строительной смеси с 9ю признакам.
Перечень разделов пояснительной записки, подлежащих исследованию, проектированию и разработке *(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе)*	Аналитический обзор по литературным источникам с целью выяснения достижений мировой науки в рассматриваемой области; анализ данных строительных смесей. Построение классификаторов строительных смесей; определение наиболее важных для классификации признаков; разработка Telergram-бота; обсуждение результатов выполненной работы; заключение работы. Дополнительно должны быть разработаны

	следующие разделы: финансовый менеджмент, ресурсоэффективность и ресурсосбережение; социальная ответственность; раздел на иностранном языке.
Перечень графического материала *(с точным указанием обязательных чертежей)*	1.Схема структуры исходных данных 2.UML-диаграмма вариантов использования. 3.Скриншоты элементов Telegram-бота.

Консультанты по разделам выпускной квалификационной работы
(с указанием разделов)

Раздел	Консультант
Основная часть	доцент ОИТ ИШИТР, к.т.н., доцент Аксёнов С. В.
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	профессор ОСГН ШБИП, д.э.н., профессор Жиронкин С.А.
Социальная ответственность	профессор ООД ШБИП, д.т.н., профессор Федорчук Ю. М.
Раздел на английском языке	Ст. преподаватель. ОИЯ, к.ф.н., ст.пр. Куркан Н. В.

Названия разделов, которые должны быть написаны на иностранном языке:

не менее 20% от ВКР

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	1.03.2023 г.

Задание выдал руководитель ВКР:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент ОИТ ИШИТР	Аксёнов С. В.	к.т.н., доцент		1.03.2023 г.

Задание принял к исполнению обучающийся:

Группа	ФИО	Подпись	Дата
8ПМ1И	Геращенков Вадим Евгеньевич		1.03.2023 г.

TOMSK ТОМСКИЙ POLYTECHNIC UNIVERSITY ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Инженерная школа Информационных технологий и робототехники
Направление подготовки (ООП / ОПОП) 09.04.04 Программная инженерия
Уровень образования магистратура
Отделение школы (НОЦ) Информационных технологий
Период выполнения весенний семестр 2022 /2023 учебного года

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы

Обучающийся:

Группа	ФИО
8ПМ1И	Геращенков Вадим Евгеньевич

Тема работы:

Методы и инструменты машинного обучения для исследования факторов, влияющих на качество строительных смесей

Срок сдачи обучающимся выполненной работы:	15.06.2023 г.

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
10.06.2023	Основная часть	70
10.06.2023	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	10
10.06.2023	Социальная ответственность	10
10.06.2023	Раздел на английском языке	10

СОСТАВИЛ:
руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент ОИТ ИШИТР	Аксёнов С. В.	к.т.н.		

СОГЛАСОВАНО:
руководитель ООП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент ОИТ ИШИТР	Губин Е. И.	к.ф.-м.н.		

Задание принял к исполнению обучающийся:

Группа	ФИО	Подпись	Дата
8ПМ1И	Геращенков Вадим Евгеньевич		1.03.2023 г.

РЕФЕРАТ

Тема магистерской диссертации: «Методы и инструменты машинного обучения для исследования факторов, влияющих на качество строительных смесей».

Цель работы – разработка программного обеспечения и компьютерных моделей для анализа строительных смесей.

Объект исследования - являются данные строительной смеси, полученные лабораторным путем.

Предмет исследования - инструментарий Data Science и методы машинного обучения в анализе данных.

Методы исследования – поиск литературы и источников, анализ информационных материалов, сравнение, консультация со специалистами, методы машинного обучения, методы визуализации.

В процессе исследования проводились аналитический обзор по литературным источникам с целью выяснения достижений мировой науки в рассматриваемой области; анализ строительных смесей; построения классификаторов строительных смесей, определение наиболее важных для классификации признаков; разработка интерактивного бота-Telegram. В работе использованы различные методические материалы и Интернет-ресурсы. Работа будет реализована на языке программирования Python.

В результате исследования будут проведены следующие задачи: анализ строительных смесей; разработаны модели классификации строительных смесей, определены наиболее важные для классификации признаки, разработан интерактивный Telegram-бот.

Результатом выполнения работы являются готовые компьютерные модели машинного обучения компьютерных моделей для анализа строительных смесей, которые можно использовать в будущем в сфере строительства и образования.

Объем работы 133-х страницах, количество рисунков – 53, использованных источников литературы – 49.

Ключевые слова: строительная смесь, набор данных, методы машинного обучения.

Экономическая эффективность – применение программного обеспечения позволит

специалистам и учащимся делать анализ данных бетонных смесей с помощью таблиц и графиков, что значительно сократит время их работы, затрачиваемое на данную процедуру, а также позволит выявить факторы, которые влияют на прочность бетонной смеси.

Оглавление

Введение

Бетон является наиболее используемым материалов в различных областях гражданского строительства . Его глобальные темпы производства растут, чтобы удовлетворить спрос. Механические свойства бетона являются одними из важных параметров при проектировании и оценке его характеристик. За последние несколько десятилетий машинное обучение использовалось для моделирования реальных проблем. Машинное обучение как отрасль искусственного интеллекта набирает популярность во многих научных областях, таких как робототехника, статистика, биоинформатика, информатика и строительные материалы. Машинное обучение имеет много преимуществ по сравнению со статистическими и экспериментальными моделями, такими как оптимальная точность, высокая скорость, быстрота реагирования в сложных средах и экономическая эффективность.

Целью работы является разработка программного обеспечения и компьютерных моделей для анализа строительных смесей.

Объектом исследования являются данные строительной смеси, полученные лабораторным путем.

Предметом исследования является инструментарий Data Science и методы машинного обучения в анализе данных.

Методами исследования является поиск литературы, статей, результатов исследований, анализ информационных материалов, сравнение, консультация со специалистами, методы машинного обучения, методы визуализации.

В ходе магистерского исследования будут решены следующие задачи: провести анализ литературы, посвященной применению методов машинного обучения для решения задачи классификации, провести разведочный анализ и выполнить предобработку данных, применить методы классификации, отбора признаков и заполнения пропущенных значений, исследовать связи и закономерности между различными признаками студентов, построить прогностические модели алгоритмами линейной регрессии, регрессии Лассо, хребта регрессии, случайного леса, дерева решений. Проанализировать полученные результаты, разработать компьютерную модель, произвести сравнение и оценку качества построенных моделей, сделать выводы.

Результатом выполнения работы являются готовые компьютерные модели машинного обучения для анализа строительных смесей, которые можно использовать в будущем в сфере строительства и образования.

Структура дипломной работы – дипломная работа состоит из введения, шести глав, заключения, списка использованной литературы (49 наименований). Работа изложена на 133-х страницах компьютерного текста, 53-х рисунках.

Во введении обосновывается актуальность исследуемой проблемы, сформулированы цель и основные задачи работы, определены предмет, объект, теоретические и методологические основы исследования, раскрыта новизна и практическая значимость работы.

Во второй главе «Методы машинного обучения» рассмотрена сущность понятия, машинное обучение, качество оценки классификатора и методы классификации, используемые в работе.

В третьей главе «Методы и инструменты машинного обучения для исследования факторов, влияющих на качество строительных смесей» были разработаны компьютерные моделей для анализа строительных смесей.

В заключении сформулированы основные выводы и результаты, полученные в работе.

ГЛАВА 1. ОБЗОР ЛИТЕРАТУРЫ

Несколько исследователей работали над различными методами машинного обучения; несколько ключевых результатов были представлены в следующих разделах.

У.Атичи. [3] использовали ИНС и многопараметрический регрессионный анализ для прогнозирования прочности бетонов на основе минеральных добавок, и результаты, полученные с использованием двух методов, сравниваются. В их исследовании множественный регрессионный анализ дал более точные результаты по сравнению с результатами моделей искусственных нейронных сетей при прогнозировании прочности на сжатие с использованием значений неразрушающего контроля. Абобакр и др. [15]обсуждался подход экстремальной обучающей машины (ELM) для прогнозирования прочности бетона на сжатие, модель ELM была создана с использованием лабораторных данных и применена регрессия, данные содержали воду, цемент, мелкий заполнитель, крупный заполнитель и супер пластификатор в качестве входных параметров. Затем ELM сравнили с ANN, что привело к сильному потенциалу ELM для прогнозирования прочности на сжатие высокопрочного бетона.

Туан и др. [16] обсудили прогноз прочности на одноосное сжатие с использованием метода экстремального повышения градиента (XGB), который также сравнивался с моделями SVM и ANN. Они показали, что модели XGB работают лучше и могут давать гораздо более точные результаты по сравнению с другими методами.

Халил Ибрахим [17] обсудил двухуровневые и гибридные ансамбли для бетона с высокими характеристиками с использованием моделей DT для прогнозирования прочности бетонных смесей. В предложенном результате использовались три ансамблевых подхода, и полученный результат показал, что модели DT могут точно предсказывать силы, а также могут генерировать хорошую корреляцию. Исследователь пришел к выводу, что лучшие модели среди одиннадцати моделей были взяты как GB-RS DT (R^2 = 0,9520), GB-GB DT (R^2 = 0,9456) и Bag-bag DT (R^2 = 0,9368).

Цинхуэталь. [18] использовали RF-алгоритм для прогнозирования прочности на сжатие высокоэффективных бетонов, исследование предложило два этапа, упрощение настройки параметров и прогнозирование прочности бетона на сжатие. В результате обсуждалось, что предложенный метод эффективен для оптимизации ввода и дает лучший прогноз, чем без оптимизации переменных, учитывая, что параметры должны быть установлены в разумных пределах. Затем модели сравнили с ранее разработанными моделями, которые выявили сильную способность к обобщению для прогнозирования с помощью алгоритма RF. Бехруз Ахмади-Недушан [19], обсуждалась оценка прочности бетонных смесей на сжатие с использованием оптимизированного алгоритма обучения на основе экземпляров. Были рассмотрены входные

переменные «соотношение воды и связующего», содержание «супер пластификатора», «содержание воды», «содержание летучей золы» и т. д. В качестве инструмента машинного обучения для этого исследования использовался алгоритм K ближайший. Для исследования числа соседей были разработаны пять моделей. Для каждой отдельной модели использовалась модифицированная версия алгоритма эволюции, и был найден и сообщен оптимальный параметр модели. Результат показал, что оптимизированные режимы превзошли по производительности производные стандартные алгоритмы k ближайших, и эта предложенная модель показала лучшую производительность по сравнению с обобщенными моделями нейронных сетей, пошаговой регрессией и моделями модульных нейронных сетей. Ученым Аньяоха [20], предложены мягкие вычисления для оценки прочности на сжатие для высокоэффективного бетона с оценкой состава бетона, был принят метод повышения дерева регрессии с плавным переходом, результат показал, что он был создан с использованием трех наборов нескольких аналитических методов за 28 дней, что повышает доминирование дерева регрессии с плавным переходом в точность предсказания выше, чем у других методов.

Захер Ясинет[21] предложил модели экстремального обучения ML для оценки прочности на сжатие легких пенобетонов с использованием машины экстремального обучения (ELM), которая была подтверждена сравнением многомерного адаптивного регрессионного сплайна (MARS), древовидных моделей M5, а также с CBP. В качестве входных параметров были взяты содержание цемента, плотность в сухом состоянии, водосвязующее отношение и объем вспененного материала. Результат показал, что модели ELM будут работать с большей точностью, чем другие разработанные модели. Ванесса Нильсенет [4], обсудили прогнозирование коэффициента теплового расширения бетона и других свойств с использованием системы ML, где применялись линейная регрессия и машинное обучение RF, результаты показали, что модели RF дадут лучшую точность, чем другие аналоги.

В литературе показано, что для прогнозирования прочности бетона на сжатие использовались несколько типов «алгоритмов машинного обучения», среди которых многие исследователи предпочитали использование методов «ANN и SVM». В частности, Сидик [13] применили метод ANN для прогнозирования прочности на сжатие самоуплотняющегося бетона (SCC), который содержал зольный остаток в качестве одного из основных ингредиентов. Далее, Юсал и Танилдизи [14] использовали аналогичный метод для оценки прочности SCC после воздействия на него высокой температуры. Дантас и др. [15] и Дуан [16] применили метод машинного обучения на основе нейронной сети для бетона, содержащего переработанные заполнители. Чоу и др. [17] , [18]рассмотрели многочисленные «методы машинного обучения» для прогнозирования прочности, которые включали как подход «ANN, так и SVM». Айер и др. [19] запустили сложную версию SVM, т. е. «SVM наименьших квадратов» (LS-SVM). Мотамеди и др. [20] продолжили

систему прогнозирования ML, основанную на SVM, для решения дополнительной сложной проблемы, а именно: «Прочность на сжатие без ограничения» бетонов с кукольной оболочкой «цементно-песчаных» смесей. Фам и др. [21] усовершенствовали LS-SVM с помощью «мета эвристической оптимизации» и использовали модели для прогнозирования прочности «высококачественных бетонов». Омран и др. [22] оценили точность различных «методов интеллектуального анализа данных» для прогнозирования прочности экологически чистого бетона. Читра и др. [23]сообщили об актуальности ИНС для прогнозирования прочности бетонов на основе «нано кремнезема» и медного шлака .

Первым этапом анализа при работе с любого рода данными является ихпредварительная обработка. Существует большое количество методов для обнаружения и устранения аномальных значений в данных. Также принцип обработки данных зависит от их типа. Так, например, категориальные переменные необходимо перевести в воспринимаемый алгоритмами машинного обучения вид, для этого используются методы кодировкиданных. Не менее важным является процесс приведения числовых признаков к одной шкале, для этого применяются методы стандартизации и нормализации.

При отборе признаков, которые в дальнейшем будут участвовать в обучении моделей следует учитывать наличие зависимостей между предикторами и целевой переменной, а также предикторами между собой.

Модели машинного обучения с помощью дополнительных инструментов, таких как SHAP, например, помимо предсказаний могут такжепредоставлять информацию о вкладе каждого отдельного предиктора в результат.

Вывод по разделу

В области инженерно-строительных работ крайне важно иметь точные оценки эффективности строительных материалов. Эти оценки необходимы для разработки правил безопасности, регулирующих использование материалов при строительстве
зданий, мостов и дорог.

Особый интерес представляет оценка прочности бетона. Бетон используется практически при любом строительстве, эксплуатационные характеристики бетона сильно различаются, так как он состоит из огромного количества ингредиентов, которые взаимодействуют комплексно. В результате трудно точно сказать, какова будет прочность готового продукта. Модель, которая бы позволяла определить прочность бетона наверняка, с учетом состава исходных

материалов, могла бы обеспечить более высокий уровень безопасности строительных объектов.

ГЛАВА 2. МЕТОДЫ МАШИННОГО ОБУЧЕНИЯ

2.1. Машинное обучение и анализ данных

В последние годы, с достижениями в области искусственного интеллекта, большое внимание уделяется тенденции использования методов машинного обучения , а также глубокого обучения (ветвь машинного обучения) для прогнозирования механических_свойств бетона. По сравнению с традиционными методами регрессии он имеет специальные алгоритмы, которые могут учиться на данных и отображать более точные результаты в качестве выходных данных. Машинное обучение используется в проектировании конструкций в различных областях : оценка сейсмических характеристик , моделирование прочности на растяжение и прочность на сжатие , идентификация структурной системы и контроль вибрации , и это лишь некоторые из них.

С помощью выборочных исторических данных, известных как обучающие данные, алгоритмы машинного обучения строят математическую модель, которая помогает делать прогнозы или принимать решения без явного программирования. Машинное обучение объединяет информатику и статистику для создания прогностических моделей. Машинное обучение строит или использует алгоритмы, которые учатся на исторических данных. Чем больше мы предоставим информации, тем выше будет производительность

Существуют некоторые варианты определения типов алгоритмов машинного обучения, но обычно их можно разделить на категории в соответствии с их назначением, и основными категориями являются следующие:

- Контролируемое обучение
- Неконтролируемое обучение
- Полуконтролируемое обучение
- Обучение с подкреплением

2.1.1 Контролируемое обучение

В обучении с учителем вы обучаете машину, используя хорошо «помеченные» данные. Это означает, что часть данных уже отмечены правильным ответом. Это также можно сравнимо с обучением, которое может быть в присутствии супервайзера или учителя.

Алгоритм контролируемого обучения учится на размеченных обучающих данных и помогает прогнозировать результаты для непредвиденных данных. Успешное построение, масштабирование и развертывание точной модели машинного обучения с учителем требует больше времени и более глубоких технических знаний от команды специалистов по данным (Рис.2.1). Более того, специалист по данным должен перестраивать модели, чтобы убедиться, что полученные данные остаются верными до тех пор, пока его данные не изменятся.

- Обучение с учителем позволяет вам собирать данные или создавать выходные данные из предыдущего опыта.
- Помогает вам оптимизировать критерии эффективности, используя опыт
- Машинное обучение под наблюдением помогает решать различные типы реальных вычислительных задач.

Например, вы хотите обучить машину, чтобы она помогала вам предсказывать, сколько времени нужно, чтобы доехать домой с вашей работы (Рис. 2) . Здесь вы будете начинать с создания набора помеченных данных. Эти данные будут включать:

- Условия погоды
- Время суток (утро, день, вечер, ночь)
- Каникулы

Все эти детали являются вашим входом. Результатом является количество времени, которое потребовалось, чтобы вернуться домой в этот конкретный день.

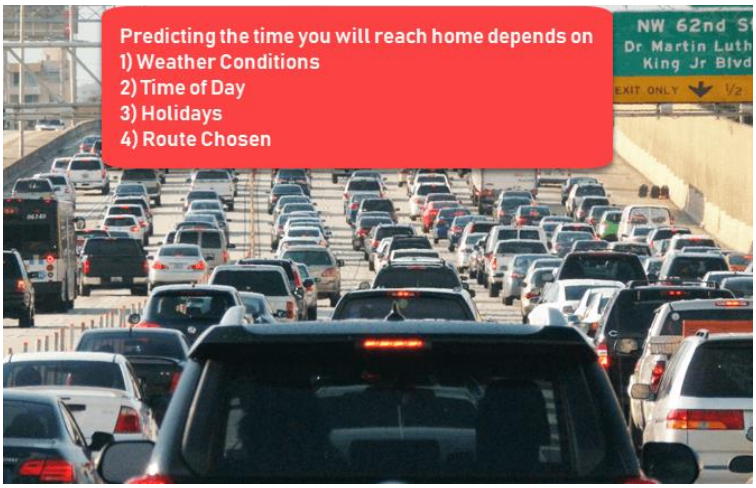

Рис. 2 Контролируемое обучение

Здесь идет понимание того, что если погода дождливая, то вам нужно больше времени, чтобы добраться домой. Но машине нужен набор данных и статистика.

Посмотрим, как вы разработаете модель обучения с учителем примера, которая поможет пользователю определить время в дороге. Изначально, что нужно создать, — это набор обучающих данных. Этот тренировочный набор будет состоять из общего времени в пути и соответствующие факторы, такие как погода, время и т. п. Зная этот тренировочный набора ваш автомобиль может обнаружить прямую зависимость между количеством осадков (дождя) и временем, что потребуется, чтобы доехать до дома.

Таким образом, что мы видим, что чем больше идет дождь на улице, тем дольше вы будете добираться домой. Он также может увидеть связь между временем, когда вы уходите с работы, и временем, когда вы будете в дороге.

Чем ближе вы к 18:00, тем больше времени вам потребуется, чтобы добраться домой. Ваша машина может найти некоторые связи с вашими помеченными данными.

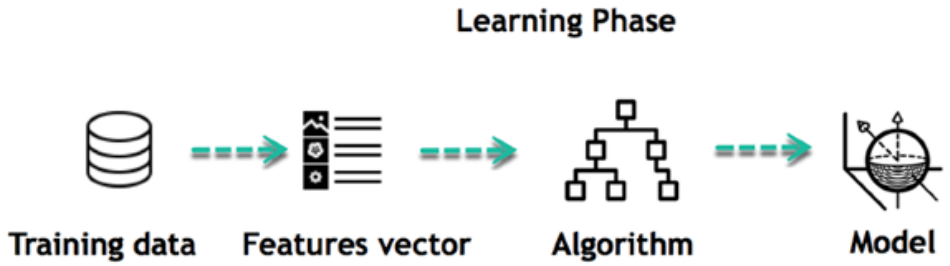

Рис.2.1 Фаза обучения

Это начало вашей модели данных. Это начинает влиять на то, как дождь влияет на то, как люди водят машину. Он также начинает видеть, что больше людей путешествуют в определенное время дня.

На рисунке 2.2 показаны типы контролируемых методов обучения.

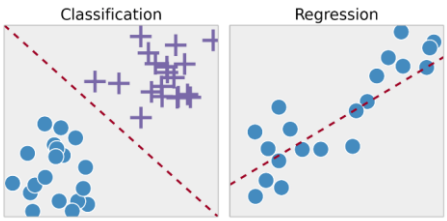

Рис.2.2 Типы контролируемых методов обучения

Регрессия:

Метод регрессии предсказывает одно выходное значение, используя обучающие данные.

Пример. Вы можете использовать регрессию для прогнозирования цены дома на основе обучающих данных. Входными переменными является местность, размер дома и т.д.

Классификация:

Классификация означает группировку вывода внутри класса. Если алгоритм пытается разбить входные данные на два разных класса, он называется бинарной классификацией. Выбор между более чем двумя классами называется многоклассовой классификацией.

Пример: определение того, будет ли кто-то не выполнять обязательства по кредиту.

Сильные стороны: выходные данные всегда имеют вероятностную интерпретацию, а алгоритм можно упорядочить, чтобы избежать переобучения.

Слабые стороны: логистическая регрессия может быть неэффективной при наличии множественных или нелинейных границ решений. Этот метод не является гибким, поэтому он не фиксирует более сложные отношения.

2.1.2. Неконтролируемое обучение

Неконтролируемое обучение — это метод машинного обучения, при котором вам нет надобности контролировать вашу модель. Вместо всего этого вам нужно разрешит модели работать самостоятельно, для обнаружения информацию. В основном это может касатсья неразмеченных данных.

Алгоритмы обучения без учителя позволит вам выполнять задачи более сложные для обработки в сравнении с обучением с учителем. Хотя обучение без учителя может быть более

непредсказуемым по сравнению с другими естественными методами глубокого обучения и обучения с подкреплением.

Вот основные причины для использования неконтролируемого обучения:

- Неконтролируемое машинное обучение находит в данных всевозможные неизвестные закономерности.
- Неконтролируемые методы помогают найти признаки, которые могут быть полезны для категоризации.
- Это происходит в режиме реального времени, поэтому все входные данные должны пройти анализ и должны быть помечены в присутствии учащихся.
- Неразмеченные данные легче получить с компьютера, чем размеченные данные, которые требуют ручного вмешательства.

Рассмотрим работу неконтролируемого обучения и возьмем случай с ребенком и ее семейной собакой (Рис.2.3).

Рис.2.3 Работа неконтролируемого обучения

Оно знает и распознает эту собаку. Через 2-3 недели друг семьи приводит с собой собаку и пытается играть с ребенком (Рис.2.4).

Рис.2.4 Работа неконтролируемого обучения

Малыш не видел эту собаку раньше. Но она признает многие черты (2 уха, глаза, ходьба на 4 лапах) как у ее любимой собаки. Она идентифицирует новое животное как собаку. Это обучение без учителя, когда вас не учат, но вы учитесь на основе данных (в данном случае данных о собаке). Если бы это было обучение с учителем, друг семьи сказал бы ребенку, что это собака.

Типы неконтролируемых методов обучения

Кластеризация

Рис.2.5 Типы неконтролируемых методов обучения

Кластеризация есть важной концепцией, когда разговор идет об обучении без учителя. В основном это касается поиска структуры или шаблона в наборе неклассифицированных данных. Алгоритмы кластеризации обработают ваши данные и найдут естественные кластеры (группы), если они только существуют в наборе ваших данных (Рис.2.5). Вы также можете изменять количество кластеров, которые должны будут идентифицировать ваши алгоритмы. Это позволяет настроить степень детализации этих групп.

Ассоциация

Правила ассоциации позволяют устанавливать ассоциации между объектами данных в больших базах данных. Этот неконтролируемый метод предназначен чтобы обнаружить захватывающие отношения между переменными в больших базах данных. К примеру, люди, покупающие новый дом, чаще будут покупать новую мебель.

Другие примеры:

- Подгруппа больных раком, сгруппированная по измерениям экспрессии их генов.
- Группы покупателей на основе их истории просмотров и покупок
- Группа фильмов по рейтингу кинозрителей

Разница между контролируемым и неконтролируемым обучением (табл 2).

Параметры	**Техника контролируемого машинного обучения**	**Техника неконтролируемого машинного обучения**
Процесс	В модели контролируемого обучения будут заданы входные и выходные переменные.	В модели неконтролируемого обучения будут предоставлены только входные данные.
Входные данные	Алгоритмы обучаются на размеченных данных.	Алгоритмы используются для данных, которые не помечены
Используемые алгоритмы	Машина опорных векторов, нейронная сеть, линейная и логистическая регрессия, случайный лес и деревья классификации.	Неконтролируемые алгоритмы можно разделить на разные категории: например, кластерные алгоритмы, K-средние, иерархическая кластеризация и т. д.
Вычислительная сложность	Обучение с учителем — более простой метод.	Неконтролируемое обучение является вычислительно сложным
Использование данных	Модель контролируемого обучения использует обучающие данные для изучения связи между входными и выходными данными.	Неконтролируемое обучение не использует выходные данные.
Точность результатов	Очень точный и надежный метод.	Менее точный и надежный метод.
Обучение в реальном времени	Метод обучения происходит в автономном режиме.	Метод обучения происходит в режиме реального времени.
Количество классов	Количество классов известно.	Количество классов неизвестно.

Параметры	Техника контролируемого машинного обучения	Техника неконтролируемого машинного обучения
Главный недостаток	Классификация больших данных может стать настоящей проблемой в контролируемом обучении.	Вы не можете получить точную информацию о сортировке данных, а выходные данные, используемые в неконтролируемом обучении, помечены и неизвестны.

Таблица 2. Разница между контролируемым и неконтролируемым обучением.

2.1.3. Полу управляемое обучение

Полу управляемое обучение (обучение с частичным привлечением учителя) предлагает золотую середину между контролируемым и неконтролируемым обучением. Полу управляемое машинное обучение представляет собой комбинацию контролируемых и неконтролируемых методов машинного обучения. При использовании более распространенных методов машинного обучения с учителем алгоритм машинного обучения обучается на размеченном наборе данных (обычно в основном на неразмеченных), в котором каждая запись включает информацию о результатах. Это позволяет алгоритму выводить закономерности и определять отношения между вашей целевой переменной и остальной частью набора данных на основе уже имеющейся информации. Напротив, неконтролируемые алгоритмы машинного обучения учатся на наборе данных без переменной результата. Обучение с полу учителем может решить проблему нехватки помеченных данных (или невозможности пометить достаточно данных) для обучения алгоритма обучения с учителем.

Неразмеченные данные при использовании в сочетании с небольшим количеством размеченных данных могут значительно повысить точность обучения. Для получения размеченных данных для задачи обучения часто требуется квалифицированный человек (например, для расшифровки аудио фрагмента) или физический эксперимент (например, определение трехмерной структуры белка или определение наличия нефти в определенном
месте). Таким образом, стоимость, связанная с процессом маркировки, может сделать большие, полностью маркированные обучающие наборы невозможными, в то время как получение немаркированных данных является относительно недорогим. В таких ситуациях обучение с полу учителем может иметь большую практическую ценность [1, 10, 11, 16].

2.2 Система обучения алгоритмов машинного обучения.

Процесс принятия решения: как правило, алгоритмы машинного обучения используются для прогнозов или классификации данных исторических исследований или наблюдений (к примеру

погода). Основываясь на некоторых входных данных, которые могут быть помечены или не помечены, наш алгоритм сделает оценку закономерности в этих данных.

Функция ошибки: функция ошибки нужна для оценки прогноза созданной модели. Если есть известные примеры, функция ошибок может провести сравнение для оценки точности нашей модели.

Процесс оптимизации модели: если модель может лучше соответствовать точкам данных в обучающем наборе, то веса корректируются, для того чтобы уменьшить несоответствие между известным примером и оценкой модели. Алгоритм будет повторять этот процесс оценки и оптимизации, автономно обновляя веса, до тех пор, пока не будет достигнут предел точности. Методы машинного обучения особенно эффективны в ситуациях, когда необходимо извлечь глубокую и прогностическую информацию из больших, разнообразных и быстро меняющихся наборов данных - больших данных.

Анализ данных включает в себя манипулирование, преобразование и визуализацию данных, чтобы сделать осмысленные выводы из результатов. Отдельные лица, предприятия и даже правительства часто принимают решения на основе этих идей. Аналитики данных могут прогнозировать поведение клиентов, цены на акции или страховые претензии, используя базовую линейную регрессию. Они могут создавать однородные кластеры, используя деревья классификации и регрессии (CART), или они могут получить некоторое представление о влиянии, используя графики для визуализации портфеля компании, занимающейся финансовыми технологиями. До последних десятилетий 20-го века аналитики-люди были незаменимы, когда дело доходило до поиска закономерностей в данных. Сегодня они по-прежнему необходимы, когда дело доходит до подачи нужных 11 данных для алгоритмов обучения и вывода значений из алгоритмического вывода, но машины могут выполнять и выполняют большую часть аналитической работы сами [11].

Машинное обучение представляет собой автоматизацию построения моделей для анализа данных. Когда мы назначаем машинам такие задачи, как классификация, кластеризация и обнаружение аномалий - задачи, лежащие в основе анализа данных, - мы используем машинное обучение. Можно разработать самосовершенствующиеся алгоритмы обучения, которые принимают данные в качестве данных, что входят и предлагают выводы со статистикой. Не полагаясь на жестко закодированное программирование, алгоритмы принимают решения каждый раз, когда обнаруживают какие-либо изменение шаблона. До того, как мы рассмотрим проблемы анализа данных, мы обсудим некоторую терминологию, которая используется для классификации различных типов алгоритмов машинного обучения. Во- первых, большинство алгоритмов можно рассматривать либо как основанные на классификации, когда машины сортируют данные по классам, либо как на основе регрессии, когда машины предсказывают значения. Различать

контролируемые и неконтролируемые алгоритмы не так уж сложно. Алгоритм с учителем обеспечивает целевые значения после достаточного обучения с данными. Напротив, информация, используемая для инструктирования неконтролируемого алгоритма машинного обучения, не требует выходной переменной для управления процессом обучения. Например, контролируемый алгоритм может оценить стоимость дома после просмотра цены (выходной переменной) аналогичных домов, в то время как неконтролируемый алгоритм может искать скрытые закономерности в выставленном на продажу жилье. Какими бы популярными ни были модели машинного обучения, по-прежнему нужны люди, чтобы получить окончательные выводы из анализа данных.

2.3. Предварительная обработка данных

Предварительная обработка данных — это процесс преобразования необработанных данных в понятный формат. Это также важный шаг в интеллектуальном анализе данных, поскольку мы не можем работать с необработанными данными. Качество данных следует проверять перед применением алгоритмов машинного обучения или интеллектуального анализа данных.

Предварительная обработка данных в основном предназначена для проверки качества данных. Качество можно проверить следующим образом:

- **Точность**: чтобы проверить правильность введенных данных или нет.
- **Полнота**: чтобы проверить, доступны ли данные или не записаны.
- **Непротиворечивость:** чтобы проверить, сохраняются ли одни и те же данные во всех местах, которые совпадают или не совпадают.
- **Своевременность**: данные должны обновляться корректно.
- **Правдоподобность**: данные должны быть достоверными.
- **Интерпретируемость**: понятность данных.

В предварительной обработке данных есть 4 основные задачи: очистка данных, интеграция данных, сокращение данных и преобразование данных.

2.4 Очистка данных

Очистка данных — это процесс удаления неверных данных, неполных данных и неточных данных из наборов данных, а также замена отсутствующих значений (Рис.2.6). Вот несколько методов очистки данных:

Рис. 2.6 Предварительная обработка данных

2.3.2 Обработка пропущенных значений

- Стандартные значения, такие как «Недоступно» или «Н/П», могут использоваться для замены отсутствующих значений.
- Отсутствующие значения также можно заполнить вручную, но это не рекомендуется, если набор данных большой.
- Среднее значение атрибута может использоваться для замены отсутствующего значения, когда данные распределены нормально,
  а в случае ненормального распределения может использоваться медианное значение атрибута.
- При использовании алгоритмов регрессии или дерева решений отсутствующее значение может быть заменено наиболее вероятным значением.

2.3.3 Обработка зашумленных данных

Шум обычно означает случайную ошибку или ненужные точки данных. Обработка зашумленных данных — один из самых важных шагов, поскольку он ведет к оптимизации, используемой нами модели. Вот некоторые из методов обработки зашумленных данных.

- **Биннинг:** этот метод предназначен для сглаживания или обработки зашумленных данных. Сначала данные сортируются, затем отсортированные значения разделяются и сохраняются в виде бинов. Существует три метода сглаживания данных в корзине. **Сглаживание методом среднего бина** : в этом методе значения в бине заменяются средним значением бина; **Сглаживание по медиане бина** : в этом методе значения в бине заменяются медианным значением; **Сглаживание**

по границе бина : в этом методе берутся минимальное и максимальное значения значений бина, а ближайшее граничное значение заменяет значения.

- **Регрессия:** используется для сглаживания данных и помогает обрабатывать данные при наличии ненужных данных. Для анализа целевая регрессия помогает определить переменную, подходящую для нашего анализа.
- **Кластеризация:** используется для поиска выбросов, а также для группировки данных. Кластеризация обычно используется в неконтролируемом обучении.

2.3.4 Интеграция данных

Процесс объединения нескольких источников в один набор данных. Процесс интеграции данных является одним из основных компонентов управления данными. При интеграции данных необходимо учитывать некоторые проблемы.

- **Интеграция схемы**: объединяет метаданные (набор данных, описывающих другие данные) из разных источников.
- **Проблема идентификации сущностей:** идентификация сущностей из нескольких баз данных. Например, система или пользователь должны знать *идентификатор учащегося одной базы данных и* имя учащегося другой базы данных, принадлежащей тому же объекту.
- **Обнаружение и разрешение концепций значений данных**: данные, взятые из разных баз данных при слиянии, могут отличаться. Значения атрибутов из одной базы данных могут отличаться от других баз данных. Например, формат даты может отличаться, например, «ММ/ДД/ГГГГ» или «ДД/ММ/ГГГГ».

2.3.5 Сжатие данных

Этот процесс помогает уменьшить объем данных, что упрощает анализ, но дает тот же или почти такой же результат. Это сокращение также помогает уменьшить пространство для хранения. Некоторыми из методов сокращения данных являются уменьшение размерности, уменьшение количества и сжатие данных.

- **Уменьшение размерности:** этот процесс необходим для реальных приложений, поскольку размер данных велик. В этом процессе происходит сокращение случайных переменных или атрибутов, чтобы можно было уменьшить размерность набора данных. Объединение и слияние атрибутов

данных без потери их первоначальных характеристик. Это также помогает сократить пространство для хранения и сократить время вычислений. Когда данные очень многомерны, возникает проблема, называемая «Проклятие размерности».

- **Уменьшение количества**. В этом методе представление данных уменьшается за счет уменьшения объема. В этом сокращении не будет потери данных.
- **Сжатие данных**. Сжатая форма данных называется сжатием данных. Это сжатие может быть без потерь или с потерями. Когда при сжатии не происходит потери информации, это называется сжатием без потерь. В то время как сжатие с потерями уменьшает информацию, но удаляет только ненужную информацию.

2.3.5 Преобразование данных

Изменение формата или структуры данных называется преобразованием данных. Этот шаг может быть простым или сложным в зависимости от требований. Есть несколько методов преобразования данных.

- **Сглаживание:** с помощью алгоритмов мы можем удалить шум из набора данных, что помогает узнать важные особенности набора данных. Сглаживая, мы можем найти даже простое изменение, которое помогает в прогнозировании.
- **Агрегация:** в этом методе данные хранятся и представляются в виде сводки. Набор данных из нескольких источников интегрирован с описанием анализа данных. Это важный шаг, поскольку точность данных зависит от количества и качества данных. Когда качество и количество данных хорошие, результаты более релевантны.
- **Дискретизация:** здесь непрерывные данные разбиты на интервалы. Дискретизация уменьшает размер данных. Например, вместо того, чтобы указывать время занятия, мы можем установить интервал, например, (15:00–17:00 или 18:00–20:00).
- **Нормализация:** это метод масштабирования данных, чтобы их можно было представить в меньшем диапазоне. Пример варьируется от -1,0 до 1,0.

Большая часть машинного обучения - это классификация - мы хотим знать, к какому классу (также известному как группа) относится наблюдение.

Модель классификации — это метод контролируемого обучения, который формирует вывод на основе наблюдаемых значений в виде одного или нескольких категориальных выходных данных. Многие приложения ИИ требуют классификации, но это особенно полезно для приложений электронной коммерции. Алгоритмы классификации , например, могут помочь в прогнозировании того, купит ли покупатель продукт. В этой ситуации две классификации — «да»

и «нет». Алгоритмы классификации не ограничиваются двумя классами и могут использоваться для разделения материалов на множество различных групп. В модели классификации используются различные методы, в том числе логистическая регрессия, многоуровневое восприятие и другие. В этой модели мы классифицируем наши данные по отдельным категориям и присваиваем этим категориям метки. Существует два типа классификаторов:

1. Классификаторы с двумя уникальными классификациями и двумя выходными данными известны как бинарные классификаторы.
2. Классификаторы, имеющие более двух классов, называются многоклассовыми классификаторами.

2.4.1. Дерево решений

Дерево решений — это непараметрический алгоритм обучения с учителем, который используется как для задач классификации, так и для задач регрессии (Рис.2.7). Он имеет иерархическую древовидную структуру, которая состоит из корневого узла, ветвей, внутренних узлов и конечных узлов.

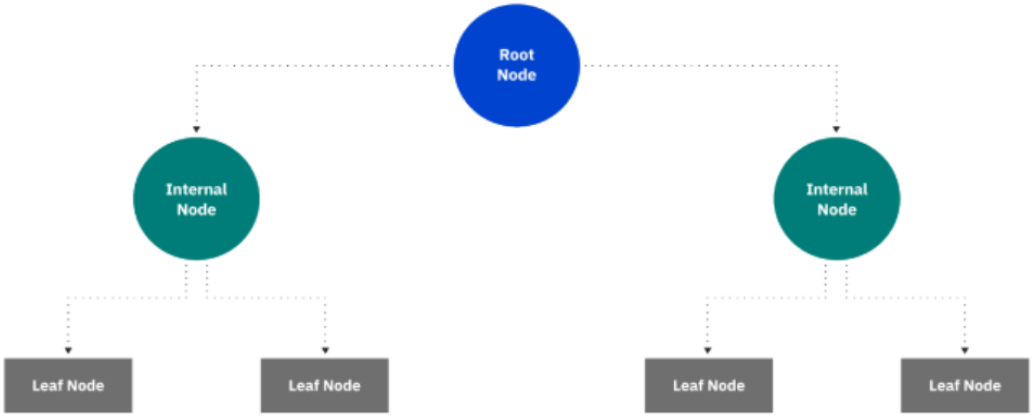

Рис. 2.7 Дерево решений

Как видно из диаграммы выше, дерево решений начинается с корневого узла, который не имеет входящих ветвей (Рис.2.8). Исходящие ветви от корневого узла затем направляются во внутренние узлы, также известные как узлы принятия решений. На основе доступных функций оба типа узлов выполняют оценку для формирования однородных подмножеств, которые обозначаются конечными узлами или конечными узлами. Листовые узлы представляют все возможные результаты в наборе данных. В качестве примера, давайте представим, что вы пытаетесь оценить,

стоит ли вам заниматься серфингом, вы можете использовать следующие правила принятия решений, чтобы сделать выбор:

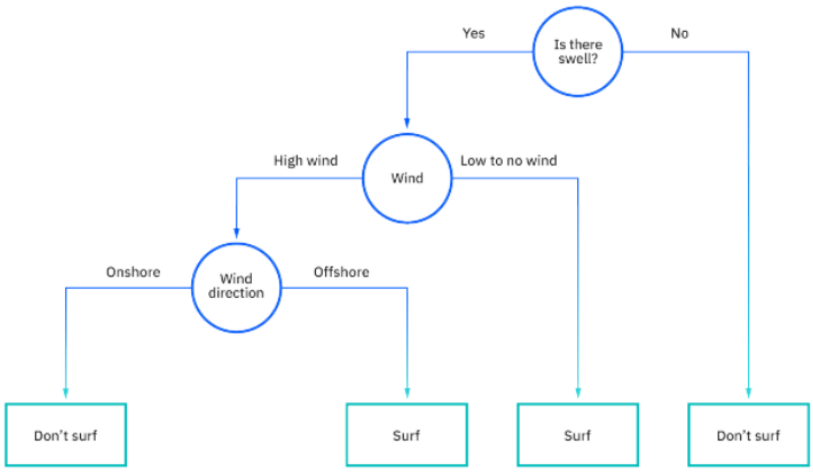

Рис. 2.8 Дерево решений

Обучение дереву решений использует стратегию «разделяй и властвуй», проводя жадный поиск для определения оптимальных точек разделения в дереве. Затем этот процесс разделения повторяется рекурсивно сверху вниз до тех пор, пока все или большинство записей не будут отнесены к определенным меткам классов. Классифицируются ли все точки данных как однородные наборы, во многом зависит от сложности дерева решений. Меньшие деревья легче получают чистые конечные узлы, т. е. точки данных в одном классе. Однако по мере роста дерева становится все труднее поддерживать эту чистоту, и это обычно приводит к тому, что в данное поддерево попадает слишком мало данных. Когда это происходит, это называется фрагментацией данных и часто может приводить к переобучению. В результате деревья решений отдают предпочтение маленьким деревьям, что согласуется с принципом экономии в Бритве Оккама; то есть «сущности не должны умножаться сверх необходимости». Иными словами, деревья решений должны усложнять только в случае необходимости, поскольку самое простое объяснение часто является лучшим. Чтобы уменьшить сложность и предотвратить переоснащение, обычно используется обрезка; это процесс, который удаляет ветви, разделяющиеся на объекты с низкой важностью. Затем соответствие модели можно оценить в процессе перекрестной проверки. Другой способ, с помощью которого деревья решений могут поддерживать свою точность, — это формирование ансамбля с помощью алгоритма случайного леса ; этот классификатор

предсказывает более точные результаты, особенно когда отдельные деревья не коррелируют друг с другом.

Хотя деревья решений можно использовать в различных случаях, другие алгоритмы обычно превосходят алгоритмы деревьев решений. Тем не менее, деревья решений особенно полезны для задач интеллектуального анализа данных и поиска знаний. Давайте рассмотрим основные преимущества и проблемы использования деревьев решений ниже:

Преимущества

- **Простота интерпретации:** логическая логика и визуальное представление деревьев решений упрощают их понимание и использование. Иерархическая природа дерева решений также позволяет легко увидеть, какие атрибуты являются наиболее важными, что не всегда ясно с другими алгоритмами, такими как нейронные сети.

- **Подготовка данных практически не требуется:** деревья решений обладают рядом характеристик, которые делают его более гибким, чем другие классификаторы. Он может обрабатывать различные типы данных, т. е. дискретные или непрерывные значения, а непрерывные значения могут быть преобразованы в категориальные значения с помощью порогов. Кроме того, он также может обрабатывать значения с отсутствующими значениями, что может быть проблематичным для других классификаторов, таких как наивный байесовский метод.

- **Более гибкий:** деревья решений можно использовать как для задач классификации, так и для задач регрессии, что делает его более гибким, чем некоторые другие алгоритмы. Он также нечувствителен к основным отношениям между атрибутами; это означает, что, если две переменные сильно коррелированы, алгоритм выберет только одну из функций для разделения.

Недостатки:

- **Склонность к переоснащению:** сложные деревья решений склонны к переоснащению и плохо обобщают новые данные. Этого сценария можно избежать с помощью процессов предварительной обрезки или последующей обрезки. Предварительная обрезка останавливает рост дерева при недостатке данных, а постобрезка удаляет поддеревья с неадекватными данными после построения дерева.

- **Оценщики с высокой дисперсией:** небольшие отклонения в данных могут привести к совершенно другому дереву решений. Бэггинг или усреднение оценок может быть методом уменьшения дисперсии деревьев решений. Однако этот подход ограничен, поскольку он может привести к сильно коррелированным предикторам.

- **Более дорого:** учитывая, что деревья решений используют жадный подход поиска во время построения, их обучение может быть более дорогим по сравнению с другими алгоритмами.

- **Не полностью поддерживается в scikit-learn:** Scikit-learn — это популярная библиотека машинного обучения, основанная на Python. Хотя в этой библиотеке есть модуль дерева решений (DecisionTreeClassifier, ссылка находится за пределами ibm.com), текущая реализация не поддерживает категориальные переменные.

2.4.2. Случайный лес

Случайный лес — это один из методов машинного обучения, используемый для решения задач регрессии и классификации. Он использует ансамблевое обучение, представляющее метод, объединяющий множество классификаторов чтобы решать сложные задачи.

Алгоритм случайного леса состоит из множества деревьев решений. «Лес», сгенерированный алгоритмом случайного леса, обучается с помощью агрегирования в пакеты или начальной загрузки. Бэггинг — ансамблевый метаалгоритм, повышающий точность алгоритмов машинного обучения.

Алгоритм (случайного леса) устанавливает результат на основе прогнозов деревьев решений. Он прогнозирует, взяв среднее или среднее значение выходных данных различных деревьев. Увеличение количества деревьев повышает точность результата.

Случайный лес устраняет ограничения алгоритма дерева решений. Это уменьшает переоснащение наборов данных и повышает точность. Он генерирует прогнозы, не требуя множества конфигураций в пакетах (например, scikit-learn).

2.4.3 Особенности алгоритма случайного леса

- Это более точно, чем алгоритм дерева решений.
- Это обеспечивает эффективный способ обработки отсутствующих данных.

- Он может дать разумный прогноз без настройки гиперпараметров.
- Это решает проблему переобучения в деревьях решений.
- В каждом дереве случайного леса подмножество признаков выбирается случайным **образом в точке разделения узла.**

2.4.4 Понимание деревьев решений

Деревья решений — это своего рода строительные блоки алгоритма случайного леса. Деревом решений можно назвать также метод поддержки принятия решений, формирующий древовидную структуру (Рис.2.9). Обзор деревьев решений поможет нам узнать, как работают алгоритмы случайного леса.

Дерево решений включает 3 компонента: узлы решений, конечные узлы и корневой узел. Алгоритм дерева решений разбивает обучающий набор данных на ветви, которые впоследствии разделяются на другие ветви. Эта последовательность будет продолжаться до тех пор, пока не будет достигнут листовой узел. Листовой узел в свою очередь не может быть отделен дальше.

Узлы в дереве решений представляют атрибуты, к использующиеся для прогнозирования результата. Узлы решений также обеспечивают связь с листьями. На диаграмме мы видим три типа узлов в дереве решений.

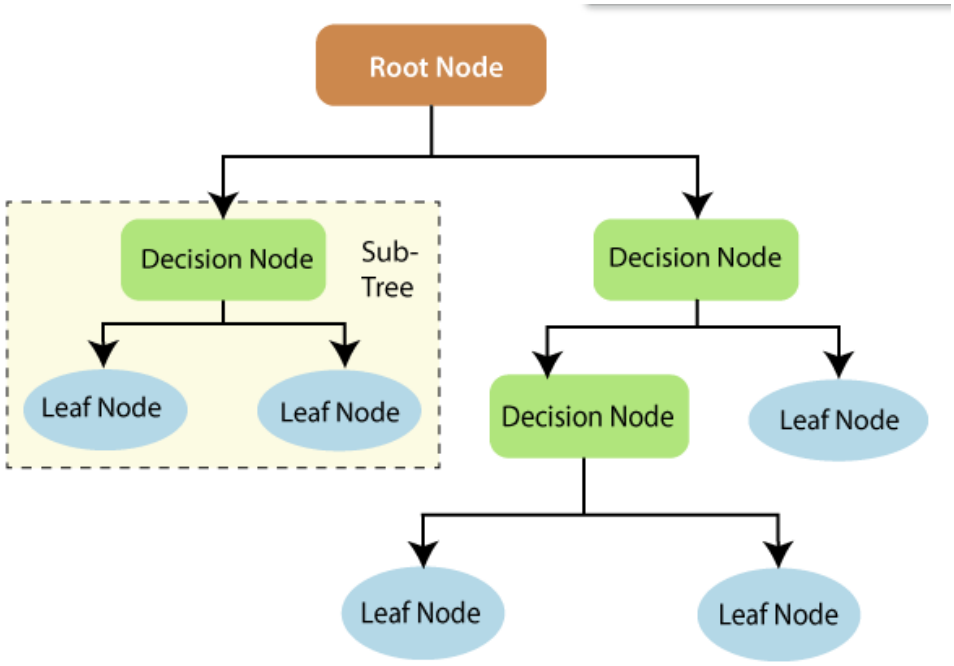

Рис. 2.9 Случайный лес

Теория информации может предоставить больше информации о том, как работают деревья решений. Энтропия и прирост информации являются строительными блоками деревьев решений. Обзор этих фундаментальных концепций улучшит наше понимание того, как строятся деревья решений.

Энтропия — это метрика для расчета неопределенности. Прирост информации является мерой того, как снижается неопределенность в целевой переменной при заданном наборе независимых переменных.

Концепция получения информации включает использование независимых переменных (признаков) для получения информации о целевой переменной (классе). Энтропия целевой переменной (Y) и условная энтропия Y (при заданном X) используются для оценки прироста информации. В этом случае условная энтропия вычитается из энтропии Y.

Получение информации используется при обучении деревьев решений. Это помогает уменьшить неопределенность в этих деревьях. Высокий информационный прирост означает, что высокая степень неопределенности (информационная энтропия) была устранена. Энтропия и прирост информации важны при разделении ветвей, что является важным действием при построении деревьев решений.

Давайте рассмотрим простой пример того, как работает дерево решений. Предположим, мы хотим предсказать, купит ли клиент мобильный телефон или нет. Характеристики телефона легли в основу его решения. Этот анализ может быть представлен в виде диаграммы дерева решений (Рис.2.10).

Корневой узел и узлы решения решения представляют характеристики телефона, упомянутые выше. Листовой узел представляет окончательный результат, либо покупку, либо отказ от покупки. Основные характеристики, определяющие выбор, включают цену, внутреннюю память и оперативную память (ОЗУ). Дерево решений будет выглядеть следующим образом.

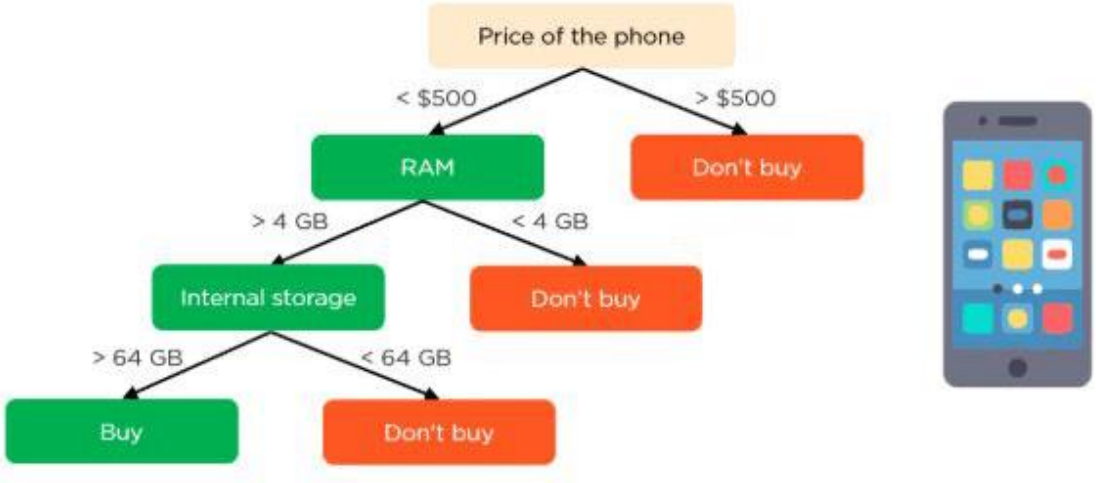

Рис. 2.10 Случайный лес

2.4.5 Применение деревьев решений в случайном лесу

Основное различие между алгоритмом дерева решений и алгоритмом случайного леса состоит в том, что в последнем установление корневых узлов и разделение узлов выполняется случайным образом. Случайный лес использует метод мешков для генерации требуемого прогноза.

Бэггинг предполагает использование разных выборок данных (данные для обучения), а не только одной выборки. Набор обучающих данных содержит наблюдения и функции, которые используются для прогнозирования. Деревья решений дают разные результаты в зависимости от обучающих данных, подаваемых в алгоритм случайного леса. Эти результаты будут ранжированы, и в качестве окончательного результата будет выбран самый высокий результат.

Наш первый пример все еще можно использовать для объяснения того, как работают случайные леса. Вместо одного дерева решений в случайном лесу будет много деревьев решений. Предположим, у нас есть только четыре дерева решений. В этом случае обучающие данные, содержащие наблюдения и функции телефона, будут разделены на четыре корневых узла.

Корневые узлы могут представлять четыре функции, которые могут повлиять на выбор клиента (цена, внутренняя память, камера и оперативная память). Случайный лес разделит узлы, случайным образом выбрав функции. Окончательный прогноз будет выбран на основе результатов четырех деревьев.

Результат, выбранный большинством деревьев решений, будет окончательным выбором. Если три дерева предсказывают покупку, а одно дерево предсказывает отказ от покупки, то окончательным прогнозом будет покупка. В этом случае прогнозируется, что клиент купит телефон.

2.4.6 Классификация в случайных лесах

Классификация в случайных лесах использует ансамблевую методологию для достижения результата. Данные обучения передаются для обучения различных деревьев решений. Этот набор данных состоит из наблюдений и признаков, которые будут выбраны случайным образом во время разделения узлов (Рис.2.11).

Система тропических лесов опирается на различные деревья решений. Каждое дерево решений состоит из узлов решений, конечных узлов и корневого узла. Листовой узел каждого дерева — это конечный результат, полученный этим конкретным деревом решений. Выбор окончательного результата осуществляется по системе мажоритарного голосования. В этом случае выход, выбранный большинством деревьев решений, становится окончательным выходом системы тропических лесов. На диаграмме ниже показан простой классификатор случайного леса.

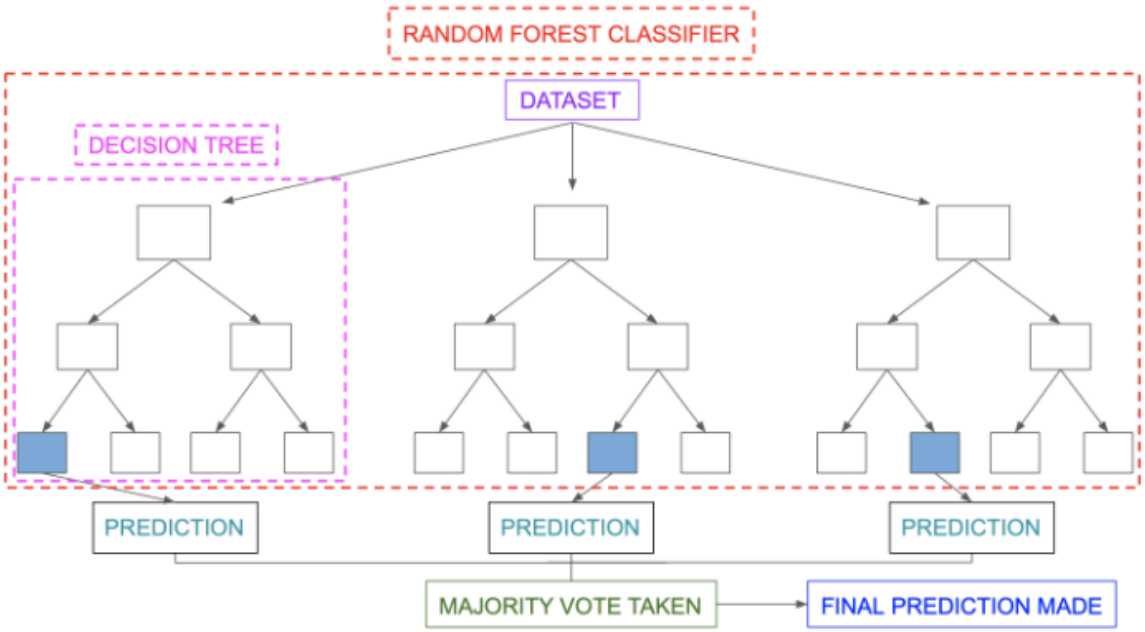

Рис. 2.11 Классификация в случайных лесах

2.4.7 Регрессия в случайных лесах

Регрессия — это еще одна задача, выполняемая алгоритмом случайного леса. Регрессия случайного леса следует концепции простой регрессии. Значения зависимых (признаков) и независимых переменных передаются в модели случайного леса.

Мы можем запускать регрессии случайного леса в различных программах, таких как SAS , R и Python. В регрессии случайного леса каждое дерево дает определенный прогноз. Средний прогноз отдельных деревьев является результатом регрессии. Это противоречит классификации случайных лесов, результат которой определяется режимом класса деревьев решений.

Хотя регрессия случайного леса и линейная регрессия следуют одной и той же концепции, они различаются по функциям. Функция линейной регрессии y=bx + c, где y — зависимая переменная, x — независимая переменная, b — параметр оценки, а c — константа. Функция сложной регрессии случайного леса подобна черному ящику .

Алгоритмы случайного леса не идеальны в следующих ситуациях:

2.4.8 Экстраполяция

Случайная регрессия леса не идеальна для экстраполяции данных. В отличие от линейной регрессии, которая использует существующие наблюдения для оценки значений за пределами диапазона наблюдения. Это объясняет, почему большинство приложений случайного леса связаны с классификацией.

2.4.9 Разреженные данные

Случайный лес не дает хороших результатов, когда данных очень мало. В этом случае подмножество признаков и загрузочная выборка создадут инвариантное пространство. Это приведет к непродуктивным сплитам, что повлияет на результат.

2.4.10 Преимущества случайного леса

- Он может выполнять как задачи регрессии, так и задачи классификации.
- Случайный лес дает хорошие прогнозы, которые легко понять.
- Он может эффективно обрабатывать большие наборы данных.
- Алгоритм случайного леса обеспечивает более высокий уровень точности прогнозирования результатов по сравнению с алгоритмом дерева решений.

2.4.11 Недостатки случайного леса

- При использовании случайного леса для вычислений требуется больше ресурсов.
- Он требует больше времени по сравнению с алгоритмом дерева решений.

Банковское дело

Случайный лес используется в банковской сфере для прогнозирования кредитоспособности заявителя. Это помогает кредитному учреждению принять правильное решение о том, давать ли клиенту кредит или нет. Банки также используют алгоритм случайного леса для обнаружения мошенников.

Здравоохранение

Медицинские работники используют системы случайного леса для диагностики пациентов. Пациенты диагностируются путем оценки их предыдущей истории болезни. Прошлые медицинские записи пересматриваются, чтобы установить правильную дозировку для пациентов.

Фондовый рынок

Финансовые аналитики используют его для определения потенциальных рынков для акций. Это также позволяет им определять поведение акций.

Электронная коммерция

С помощью алгоритмов тропического леса продавцы электронной коммерции могут прогнозировать предпочтения клиентов на основе прошлого поведения потребления.

Заключение

Алгоритм случайного леса — это простой в использовании и гибкий алгоритм машинного обучения. Он использует ансамблевое обучение, которое позволяет организациям решать проблемы регрессии и классификации.

Это идеальный алгоритм для разработчиков, поскольку он решает проблему переобучения наборов данных. Это очень изобретательный инструмент для создания точных прогнозов, необходимых для принятия стратегических решений в организациях.

2.5 Линейная регрессия

Линейная регрессия — это тип алгоритма контролируемого машинного обучения, который вычисляет линейную связь между зависимой переменной и одним или несколькими независимыми функциями. Когда число независимых признаков равно 1, это называется одномерной линейной регрессией, а в случае нескольких признаков — многомерной линейной регрессией. Цель алгоритма — это найти самое лучшее линейное уравнение, которое сможет предсказать величину зависимой переменной на основе независимых переменных. Уравнение выглядит как прямая линия, представляющее отношение между зависимой и независимой переменными (Рис.2.12). Наклон этой линии показывает, как зависимая переменная изменяется на единицу изменения независимой переменной (переменных).

Линейная регрессия используется во многих различных областях, включая финансы, экономику и психологию, для понимания и прогнозирования поведения конкретной

переменной. Например, в финансах линейная регрессия может использоваться для понимания взаимосвязи между ценой акций компании и ее прибылью или для прогнозирования будущей стоимости валюты на основе ее прошлых результатов.

Одной из наиболее важных задач обучения с учителем является регрессия. В регрессионном наборе записей присутствуют значения X и Y, и эти значения используются для изучения функции, поэтому, если вы хотите предсказать Y из неизвестного X, можно использовать эту изученную функцию. В регрессии мы должны найти значение Y, поэтому требуется функция, которая предсказывает непрерывный Y в случае регрессии, учитывая X как независимые функции.

Здесь Y называется зависимой или целевой переменной, а X называется независимой переменной, также известной как предиктор Y. Существует множество типов функций или модулей, которые можно использовать для регрессии. Линейная функция является простейшим типом функции. Здесь X может быть одной функцией или несколькими функциями, представляющими проблему.

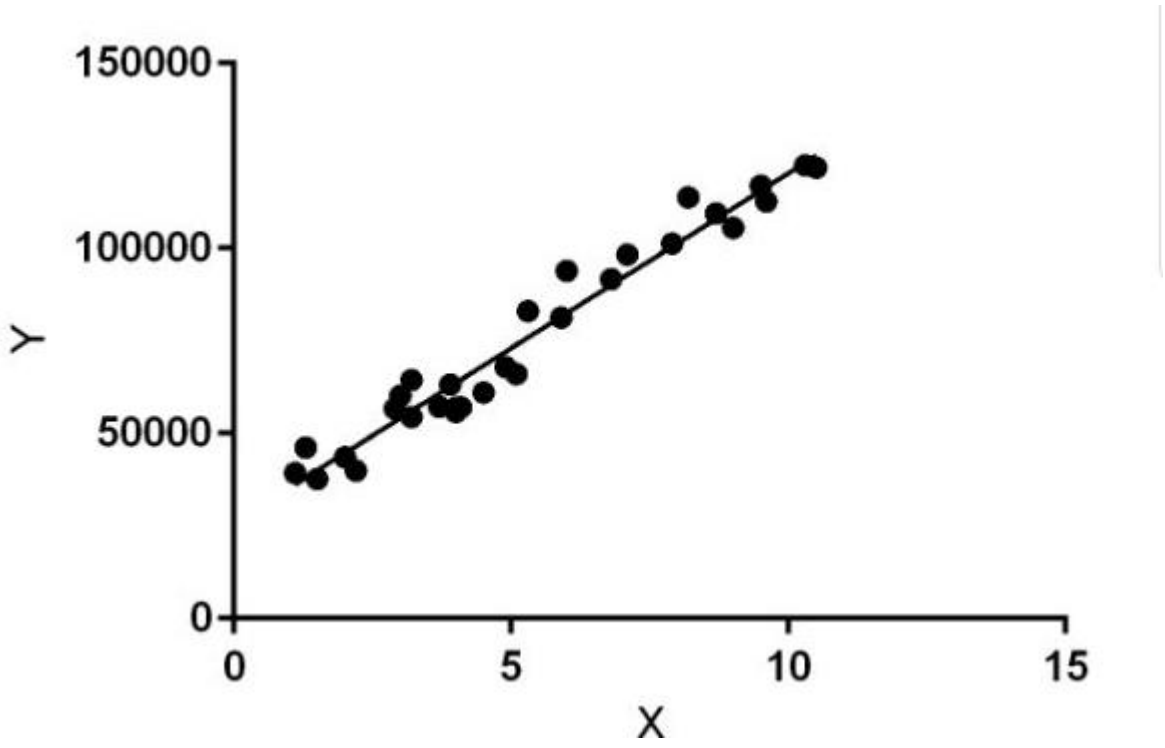

Рис. 2.12 Линейная регрессия

Линейная регрессия выполняет задачу прогнозирования значения зависимой переменной (y) на основе данной независимой переменной (x)). Отсюда и название — линейная регрессия. На

рисунке выше X (вход) — это опыт работы, а Y (выход) — зарплата человека. Линия регрессии лучше всего подходит для нашей модели.

Допущение для модели линейной регрессии

Линейная регрессия является мощным инструментом для понимания и прогнозирования поведения переменной, однако она должна соответствовать нескольким условиям, чтобы быть точными и надежными решениями.

1. **Линейность**: независимые и зависимые переменные имеют линейную связь друг с другом. Это означает, что изменения в зависимой переменной следуют за изменениями в независимой переменной (переменных) линейным образом.
2. **Независимость**: наблюдения в наборе данных не зависят друг от друга. Это означает, что значение зависимой переменной для одного наблюдения не зависит от значения зависимой переменной для другого наблюдения.
3. **Гомоскедастичность**: на всех уровнях независимой переменной (переменных) дисперсия ошибок постоянна. Это указывает на то, что количество независимых переменных не влияет на дисперсию ошибок.
4. **Нормальность**: ошибки в модели нормально распределены.
5. **Нет мультиколлинеарности** : нет высокой корреляции между независимыми переменными. Это указывает на то, что корреляция между независимыми переменными незначительна или отсутствует.

Математическое уравнение, которое оценивает линию простой линейной регрессии:

$$Y=a+bx.$$

x -это независимой переменная или предиктор.

Y – зависимая переменная или же, как ее еще называют, переменная отклика. Это то самое значение, которое ожидается для *y* (в среднем), если знаем величину *x*, т.е. это «предсказанное значение *y*»

- *a* – это свободный член (пересечение) линии оценки; это значение *Y*, когда *x=0* (Рис.1).
- *b* – это угловой коэффициент или градиент оценённой линии; она представляет собой величину, на которую *Y* увеличивается в среднем, если увеличиваем *x* на одну единицу.

- a и b называют коэффициентами регрессии оценённой линии.

2.6. Метрики качества

Измерение качества моделей является одной из обязательных процедур при построении и выборе наилучшей модели машинного обучения. И для каждой задачи классификации, регрессии или кластеризации применяются метрики, придуманные специально под этот тип задач. Качество модели оценивается подсчетом метрики, метрика считается стандартным способом заключается в следующем: перед тем, как начать обучать модель, нужно разделить весь датасет на две выборки: тренировочная и тестовая. На тренировочной выборке обучают модель, а на тестовой замеряют ее качество.

Цель заключается в том, чтобы оценить качество модели на тех данных, которые она еще не видела, т.е. на которых не обучалась. Стандартное разделение пропорций на тренировочные и тестовые данные происходит 70 на 30 или 80 на 20, где большая часть – тренировочная выборка. При этом данные тренировочной и тестовой выборки отбираются случайным образом, чтобы тестовая выборка была репрезентативной и охватила как можно больше различных случаев данных.

Самое простое, что можно посчитать, это сколько процентов значений было угадано, это и есть метрика классификации точности (accuracy) и рассчитывается как отношение количества правильных прогнозов к их общему количеству.

$$Accuracy = \frac{\text{Количество правильных прогнозов}}{\text{Всего прогнозов}} \cdot 100\%$$

Формула 2.1 – Accuracy (точность)

Когда модель обучилась и есть значение метрики, всегда полезно знать, в каких местах модель ошибается.

В реальности, когда в тестовой выборке по тысяче записей, визуально оценить промахи модели очень сложно. На этот случай можно вывести матрицу ошибок (confusion matrix), рассмотреть ее конструкцию так, как она реализована в Python (Рис.2.13).

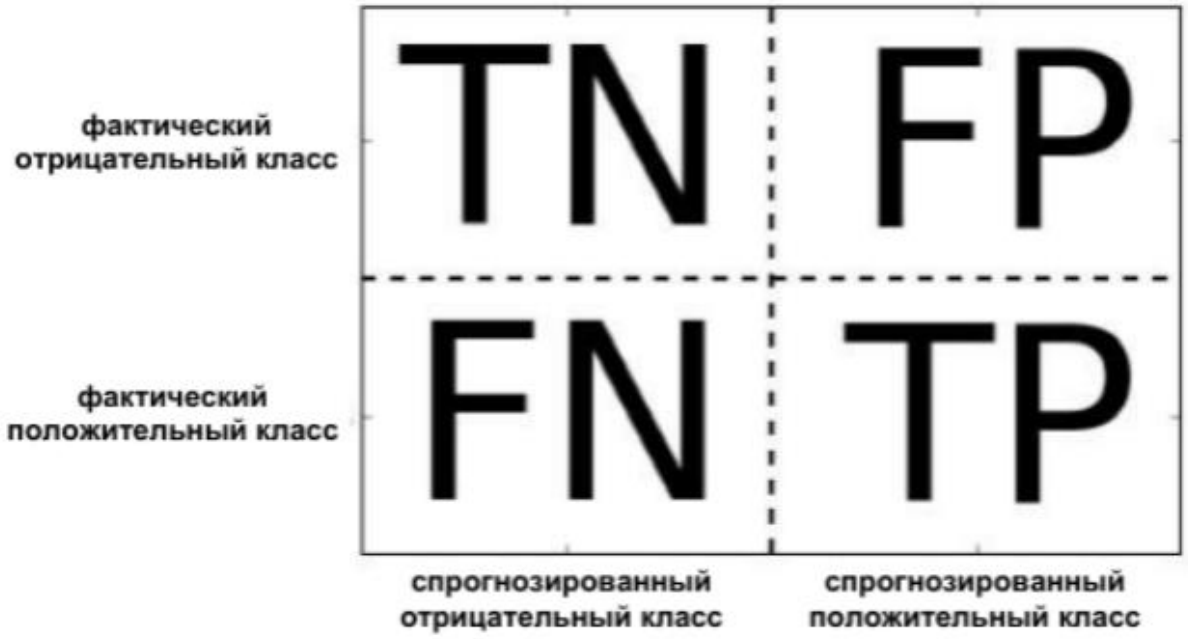

Рис 2.13 – Матрица ошибок (confusion matrix)

По диагонали этой матрицы находится количество элементов, угаданных моделью.

- True Negative – количество элементов, угаданных для отрицательного класса;
- True Positive – количество элементов, которые верно предсказаны для положительного класса;
- False Positive – элементы, которые были предсказаны как положительный класс, но на самом деле относятся к отрицательному классу;
- False Negative – элементы, которые были предсказаны отрицательным классом, но на самом деле относятся к положительному.

У нас еще имеется несколько способов для вывода информации матрицы ошибок, наиболее популярными из них являются точность (precision) и полнота (recall). Точность показывает, сколько из предсказанных положительных экземпляров оказались действительно положительными.

$$Precision = \frac{TP}{TP + FP}$$

Формула 2.2 – Точность (precision)

Точность применяется, как показатель качества модели, когда нужно снизить количество ложно положительных примеров. Например, нам будет важносделать прогноз, будет ли эффективно новое лекарство при лечении болезни [5]. Клинические испытания -дорогие и фармкомпания хочет организовать их тогда, когда полностью твердо будет знать, что медикамент действительно работает. Поэтому очень важно минимизировать количество ложно положительных примеров, т.е. увеличивая точность. Точность также известна как прогностическая ценность положительного результата (Positive Predictive Value – PPV). С другой стороны, полнота (Recall) выводит, сколько об общего числа фактически положительных примеров было предсказано как положительный класс.

$$Recall = \frac{TP}{TP + FN}$$

Формула 2.3 – Полнота (recall)

Полнота используется, как показатель качества модели, когда необходимо узнать все положительные примеры. Пример диагностики рака есть хорошим примером подобной задачи: важно обнаружить всех больных пациентов, при этом, возможно, включив в их число здоровых. Полнота также известна под названием чувствительность (Sensitivity), процент результативных ответов и доля истинно положительных примеров (TruePositive Rate – TPR) [12].

Чтобы полностью оценить эффективность модели, вы должны проверить как точность, так и полноту. К сожалению, точность и полнота часто противоречат друг другу. То есть повышение точности обычно снижает отзыв, и наоборот [12].

Точность и полнота являются важными метриками, это так, но полную картину мы не получим. Одним из главных показателей является F-мера (F-measure), она позволяет одновременно оценить полноту и точность. Здесь используется среднее гармоническое вместо среднего арифметического, сглаживая расчеты за счет исключения экстремальных значений.

Гармоническое среднее - это просто еще один способ вычисления «среднего» значения, обычно описываемый как более подходящий для соотношений (таких как точность и полнота), чем традиционное среднее арифметическое. Для бинарной классификации несбалансированных данных она может быть более лучшей метрикой, чем правильность.

Формула, используемая для оценки F1 в этом случае:

$$\frac{Precision \bullet Recall}{Precision + Recall} \bullet 2$$

Формула 2.4 – F-мера (F1-score)

Еще одним рабочим инструментом для анализа поведения классификаторов при различных пороговых значениях является ROC-кривая (Receiver Operating Characteristic). ROC-кривая позволяет рассмотреть все пороговые значения для классификатора, но вместо точности и полноты она показывает долю ложно положительных примеров (False Positive Rate – FPR) в сравнении с долей истинно положительных примеров (True Positive Rate – TPR).

$$TPR = \frac{TP}{TP + FN}$$

Формула 2.5 – Доля истинно положительных примеров (True Positive Rate)

$$FPR = \frac{FP}{FP + TN}$$

Формула 2.6 – Доля ложно положительных примеров (False Positive Rate)

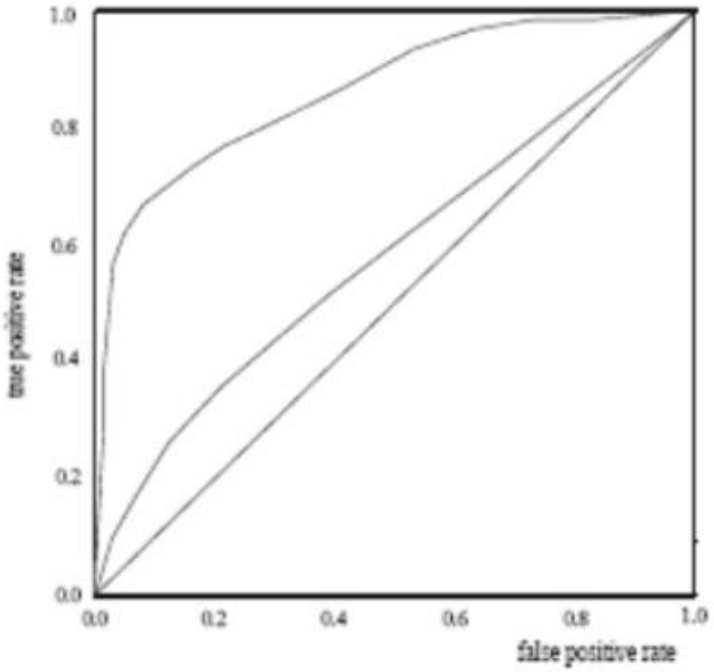

Рис. 2.14 – ROC-кривая

Качество оценивают, как площадь под этой кривой – AUC (Area Underthe Curve) и площадь под ROC-кривой (Рис.2.14) является мерой точности модели. Когда модель ближе к диагонали, она менее точна, и модель с идеальной точностью будет иметь площадь 1,0 [6].

Вывод по разделу:

В этом разделе мы ознакомились с предварительной обработкой данных и с методами и алгоритмами машинного обучения.

ГЛАВА 3. РАЗРАБОТКА КОМПЬЮТЕРНОЙ МОДЕЛИ ДЛЯ ИССЛЕДОВАНИЯ ФАКТОРОВ, ВЛИЯЮЩИХ НА КАЧЕСТВО СТРОИТЕЛЬНЫХ СМЕСЕЙ

Целью работы является разработка программного обеспечения и компьютерных моделей для анализа строительных смесей.

Объектом исследования являются данные строительной смеси, полученные лабораторным путем.

Предметом исследования является инструментарий Data Science и методы машинного обучения в анализе данных.

Методами исследования является изучение литературы, статей, научных работ, анализ материалов, сравнение, консультация со специалистами и учеными, методы машинного обучения, методы визуализации.

Набор данных состоит из 1030 экземпляров с 9 атрибутами и не содержит пропущенных значений. Имеется 8 входных переменных и 1 выходная переменная. Семь входных переменных представляют количество сырья (измеряется в кг/м³), а одна представляет возраст (в днях). Целевой переменной является предел прочности бетона на сжатие, измеряемая в (МПа — МегаПаскаль). Мы изучим данные, чтобы увидеть, как входные характеристики влияют на прочность на сжатие.

3.1 Изучение набора данных

При загрузке этого набора данных, мы видим, что на качество бетона влияют несколько функций. Итак, мы кратко обсудим каждую характеристику:

Цемент (cement): представляет собой мелкоизмельченный минеральный порошок, обычно серого цвета. Важнейшим сырьем для производства цемента являются известняк, глина и мергель. Смешанный с водой цемент служит клеем для связывания песка, гравия и твердых пород в бетоне. Цемент затвердевает как на воздухе, так и под водой и остается в затвердевшем состоянии после достижения.

Шлак (slag): твёрдый остаток после выплавки металла из руды, а также от сжигания угля.

Летучая зола (Flyash): улучшает удобоукладываемость бетона, прокачиваемость, когезию, отделку, предельную прочность и долговечность, а также решает многие проблемы, с которыми сталкивается бетон сегодня, и все это при меньших затратах.

Вода (Water): является важным компонентом при производстве бетона. Влага, которую

обеспечивает вода, также придает бетону прочность в процессе отверждения. Хотя вода является одним из наиболее важных компонентов бетона, она также может быть наиболее разрушительной в чрезмерных количествах.

Суперпластификатор (Superplasticizer): традиционно известны своими водоредуцирующими свойствами, позволяющими улучшить текучесть бетона без добавления дополнительной воды и снижения общей прочности смеси. Эти же качества также позволяют сократить количество цементных материалов.
При смешивании с водой частицы цемента естественным образом притягиваются друг к другу и имеют тенденцию образовывать комки. Это означает, что только часть частиц цемента может должным образом завершить процесс гидратации, что снижает прочность готового продукта. Высокопрочные смеси требуют большего количества цемента, чтобы увеличить процент цемента, который связывается и гидратируется с молекулами воды. Когда в смеси присутствует суперпластификатор, он связывается с частицами цемента и нейтрализует силу, стягивающую их вместе. Это удерживает их от образования кластеров в смеси и высвобождает больше молекул цемента для завершения процесса гидратации. Таким образом, вы получаете тот же результат с меньшим количеством цемента, чем традиционная высокопрочная смесь.

Крупный заполнитель (Coaseseaggregate): крупный заполнитель для бетона — это гравий с округлыми зернами, имеющие гладкую поверхность, а также щебень с угловатыми зернами, имеющие шероховатую поверхность. Щебень и гравий получают путем дробления крупных горных пород.

Мелкий заполнитель (fineaggregate): песок, величина зерен которого не больше 5мм.

Возраст (age): проектный возраст бетона, т. е. возраст, в котором бетон должен приобрести все нормируемые для него показатели качества, назначают при проектировании, исходя из возможных реальных сроков загружения конструкций проектными нагрузками, с учетом способа возведения конструкций и условий твердения бетона.

Предел прочности бетона (Strength) — это важный параметр, который определяет максимальную нагрузку, которую бетонная конструкция может выдержать без разрушения.

Теперь мы импортируем некоторые важные модули (Рис.3.1):

	cement	blastFurnace	flyAsh	water	superplasticizer	courseAggregate	fineaggregate	age	strength
0	540.0	0.0	0.0	162.0	2.5	1040.0	676.0	28	79.99
1	540.0	0.0	0.0	162.0	2.5	1055.0	676.0	28	61.89
2	332.5	142.5	0.0	228.0	0.0	932.0	594.0	270	40.27
3	332.5	142.5	0.0	228.0	0.0	932.0	594.0	365	41.05
4	198.6	132.4	0.0	192.0	0.0	978.4	825.5	360	44.30
...	...	...	...	...	...	...	...	...	...
1025	276.4	116.0	90.3	179.6	8.9	870.1	768.3	28	44.28
1026	322.2	0.0	115.6	196.0	10.4	817.9	813.4	28	31.18
1027	148.5	139.4	108.6	192.7	6.1	892.4	780.0	28	23.70
1028	159.1	186.7	0.0	175.6	11.3	989.6	788.9	28	32.77
1029	260.9	100.5	78.3	200.6	8.6	864.5	761.5	28	32.40

1030 rows × 9 columns

Рис. 3.1 Набор данных

3.2 Набор данных исследования

После чтения набора данных мы должны извлечь информацию из данных, для этого мы используем определенную функцию (Рис.3.2):

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1030 entries, 0 to 1029
Data columns (total 9 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Cement              1030 non-null   float64
 1   Blast Furnace Slag  1030 non-null   float64
 2   Fly Ash             1030 non-null   float64
 3   Water               1030 non-null   float64
 4   Superplasticizer    1030 non-null   float64
 5   Coarse Aggregate    1030 non-null   float64
 6   Fine Aggregate      1030 non-null   float64
 7   Age                 1030 non-null   int64
 8   Strength            1030 non-null   float64
dtypes: float64(8), int64(1)
memory usage: 72.5 KB
```

Рис. 3.2 Извлечение информации из набора данных

Метод df.info () дает описание дата фрейма: сколько в нем строк и столбцов, какие типы данных содержатся, сколько не пустых значений (non-null), сколько памяти занимает.

Метод describe дает статистику по числовым столбцам: среднее, максимум и минимум, квартили,

стандартное отклонение (Рис.3.3).

```
df.describe()
```

	Cement	Blast Furnace Slag	Fly Ash	Water	Superplasticizer	Coarse Aggregate	Fine Aggregate	Age	Strength
count	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000
mean	281.167864	73.895825	54.188350	181.567282	6.204660	972.918932	773.580485	45.662136	35.817961
std	104.506364	86.279342	63.997004	21.354219	5.973841	77.753954	80.175980	63.169912	16.705742
min	102.000000	0.000000	0.000000	121.800000	0.000000	801.000000	594.000000	1.000000	2.330000
25%	192.375000	0.000000	0.000000	164.900000	0.000000	932.000000	730.950000	7.000000	23.710000
50%	272.900000	22.000000	0.000000	185.000000	6.400000	968.000000	779.500000	28.000000	34.445000
75%	350.000000	142.950000	118.300000	192.000000	10.200000	1029.400000	824.000000	56.000000	46.135000
max	540.000000	359.400000	200.100000	247.000000	32.200000	1145.000000	992.600000	365.000000	82.600000

Рис. 3.3 Статистика по числовым столбцам

Теперь мы обрабатываем нулевые значения, присутствующие в наборе данных, для большей точности (Рис.3.4).

```
df.isnull().sum()
```

```
Cement                0
Blast Furnace Slag    0
Fly Ash               0
Water                 0
Superplasticizer      0
Coarse Aggregate      0
Fine Aggregate        0
Age                   0
Strength              0
dtype: int64
```

S

В данном наборе данных отсутствуют нулевые значения.

3.3 Исследовательский анализ данных

Первым шагом в проекте Data Science является понимание данных и получение информации из данных, прежде чем приступать к моделированию. Это включает в себя проверку любых пропущенных значений, построение характеристик по отношению к целевой переменной,

наблюдение за распределением всех функций и так далее. Давайте импортируем данные и начнем анализ.

Далее проверим корреляции между входными функциями, это даст представление о том, как каждая переменная влияет на все другие переменные. Один из способов количественной оценки взаимосвязи между двумя переменными — использовать коэффициент корреляции Пирсона (Рис.3.5), который является мерой линейной связи между двумя переменными

Принимает значение от -1 до 1, где:

- -1 указывает на совершенно отрицательную линейную корреляцию.
- 0 указывает на отсутствие линейной корреляции.
- 1 указывает на абсолютно положительную линейную корреляцию.

ЗНАЧЕНИЕ (по модулю)	ИНТЕРПРЕТАЦИЯ
до 0,2	очень слабая корреляция
до 0,5	слабая корреляция
до 0,7	средняя корреляция
до 0,9	высокая корреляция
свыше 0,9	очень высокая корреляция

Рис. 3.5 Оценка качества корреляции набора данных

Чем дальше коэффициент корреляции находится от 0, тем более сильная связь между 2я переменными.

Но бывают также случаи, когда мы хотим понять корреляцию между более чем одной парой переменных. В этих случаях мы можем создать матрица корреляции (Рис.3.6), которая представляет

собой квадратную таблицу, показывающая коэффициенты корреляции между несколькими попарными комбинациями переменных.

	cement	blastFurnace	flyAsh	water	superplasticizer	courseAggregate	fineaggregate	age	strength
cement	1.00	-0.28	-0.40	-0.08	0.09	-0.11	-0.22	0.08	0.50
blastFurnace	-0.28	1.00	-0.32	0.11	0.04	-0.28	-0.28	-0.04	0.13
flyAsh	-0.40	-0.32	1.00	-0.26	0.38	-0.01	0.08	-0.15	-0.11
water	-0.08	0.11	-0.26	1.00	-0.66	-0.18	-0.45	0.28	-0.29
superplasticizer	0.09	0.04	0.38	-0.66	1.00	-0.27	0.22	-0.19	0.37
courseAggregate	-0.11	-0.28	-0.01	-0.18	-0.27	1.00	-0.18	-0.00	-0.16
fineaggregate	-0.22	-0.28	0.08	-0.45	0.22	-0.18	1.00	-0.16	-0.17
age	0.08	-0.04	-0.15	0.28	-0.19	-0.00	-0.16	1.00	0.33
strength	0.50	0.13	-0.11	-0.29	0.37	-0.16	-0.17	0.33	1.00

Рис. 3.6 Матрица корреляции

Визуализируем матрицу корреляции (Рис.3.7) с помощью параметры стиля доступны в pandas:

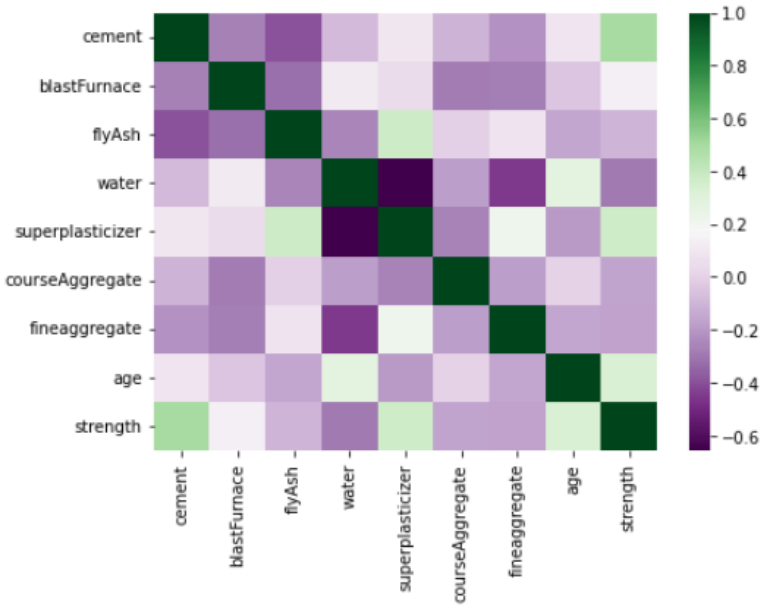

Рис. 3.7 Визуализация матрицы корреляции

Мы можем наблюдать высокую положительную корреляцию между **пределом прочности на сжатие** (CC_Strength) и **цементом**. Это верно, потому что прочность бетона действительно

увеличивается с увеличением количества цемента, используемого при его приготовлении. Кроме того, **возраст** и **суперпластификатор** являются двумя другими факторами, влияющими на прочность на сжатие.

Существуют и другие сильные корреляции между признаками:

- Сильная отрицательная корреляция между **суперпластификатором** и **водой**.
- положительная корреляция между **суперпластификатором** и **летучей золой**, **мелким заполнителем**.

Эти корреляции полезны для детального понимания данных, поскольку они дают представление о том, как одна переменная влияет на другую. Далее мы можем использовать **парный график** в seaborn, чтобы построить попарные отношения между всеми признаками и распределениями признаков по диагонали.

Мы можем построить графики рассеяния между **Strength** и другими функциями, чтобы увидеть более сложные отношения.

Диаграмма рассеяния – один из инструментов статистического контроля, анализа. С ее помощью выявляется зависимость и характер связи между двумя разными параметрами экономического явления, производственного процесса. Диаграмма разброса показывает вид и тесноту взаимосвязи между парами данных. К примеру, между:

1. качеством продукта и влияющим фактором;
2. двумя разными характеристиками качества;
3. двумя обстоятельствами, влияющими на качество, и т.п.

Диаграммы рассеяния применяются для обнаружения корреляции между данными. Если корреляционная зависимость присутствует, то установить контроль над наблюдаемым явлением значительно проще.

Диаграмма разброса (Рис.3.8) представляет наблюдаемое явление в пространстве двух измерений. Если одну величину рассматривать как «причину», влияющую на другую величину, то ей будет соответствовать ось X (горизонтальная ось). Реагирующей на это влияние величине соответствует ось Y (вертикальная ось). Когда четко классифицировать переменные невозможно, распределение производится пользователем.

Предел прочности против (цемент, возраст, вода)

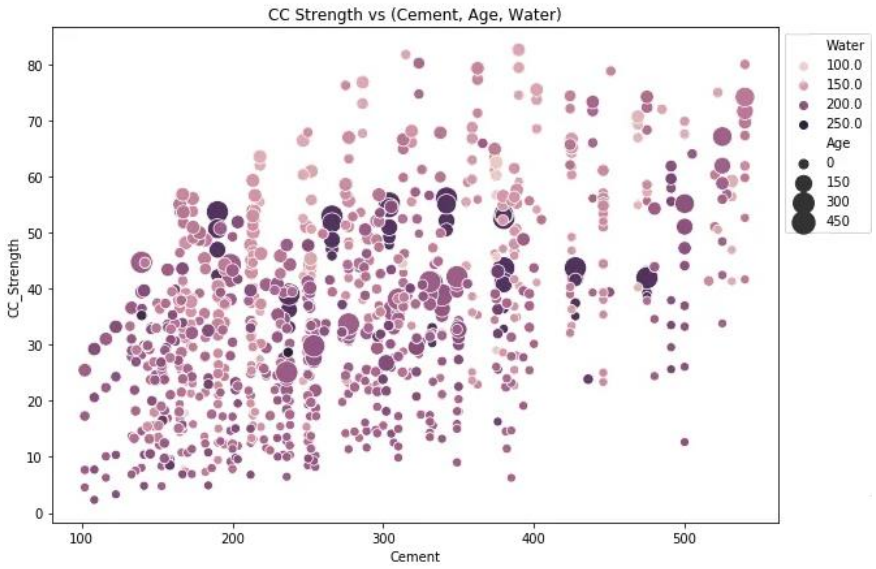

Рис. 3.8 Диаграмма разброса **предела прочности в сравнении с цементом, возрастом, водой**

Наблюдения, которые мы можем сделать из этого графика:

- **Прочность на сжатие увеличивается по мере увеличения количества цемента**, поскольку точки перемещаются вверх, когда мы движемся вправо по оси X.
- **Прочность на сжатие увеличивается с возрастом** (поскольку размер точек представляет возраст), это не всегда так, но может быть до некоторой степени.
- **Цемент с меньшим возрастом требует больше цемента для более высокой прочности**, так как меньшие точки движутся вверх, когда мы движемся вправо по оси X.

- **Чем старше цемент, тем больше воды ему требуется**, что можно подтвердить, наблюдая за цветом точек. Более крупные точки темного цвета указывают на большой возраст и большее количество воды.

- **Прочность бетона увеличивается, когда при его приготовлении используется меньше воды,** поскольку точки на нижней стороне (ось Y) темнее, а точки на верхнем конце (ось Y) ярче.

Предел прочности бетона по сравнению с (мелкий заполнитель, супер пластификатор, летучая зола) (Рис.3.9)

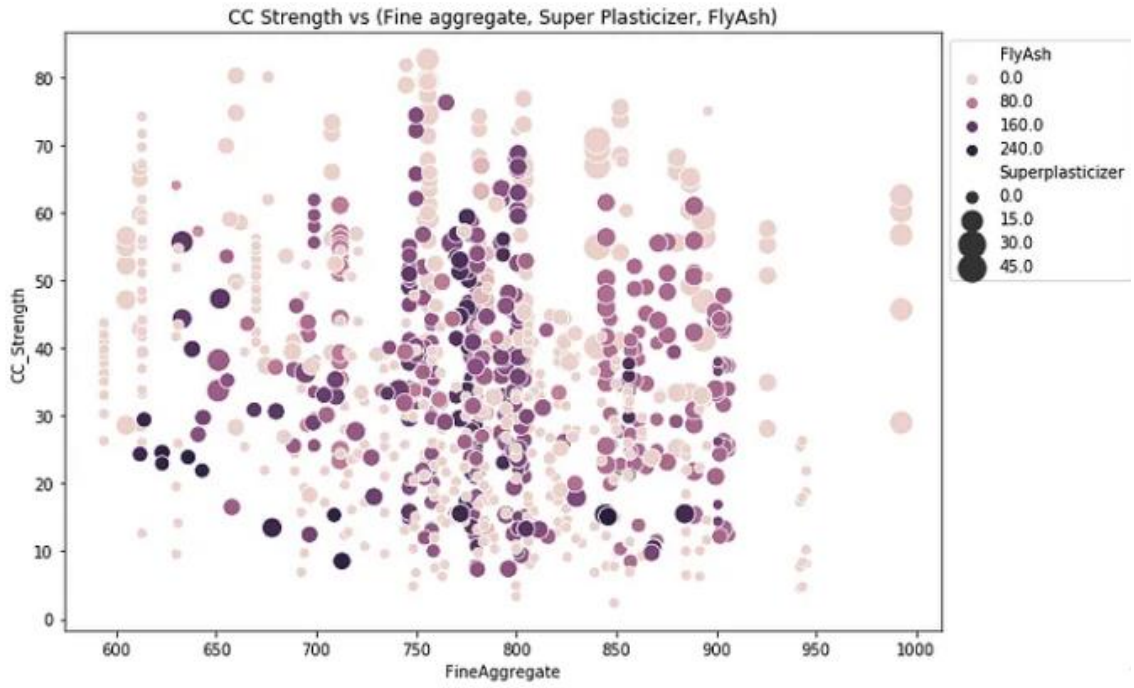

Рис. 3.9 Предел прочности бетона по сравнению с мелким заполнителем, супер пластификатором, летучей золой.

Исходя из этого графика, мы видим, что:

- **Снижается предел прочности на сжатие если количество летучей золы увеличивается**, поскольку более темные точки концентрируются в области, представляющей низкую прочность на сжатие.
- **Прочность на сжатие увеличивается с добавлением в смесь супер пластификатора,** поскольку чем больше точка, тем выше они находятся на графике.

Мы можем визуально понимать графики 2D, 3D и максимум до 4D (функции, представленные цветом и размером), как показано выше, мы можем дополнительно использовать функции построения графиков по строкам и столбцам Seaborn для дальнейшего анализа, но, тем не менее, нам не хватает возможность отслеживать все эти корреляции самостоятельно. По этой причине мы

можем обратиться к машинному обучению, чтобы зафиксировать эти отношения и лучше понять проблему.

3.4 Разделение зависимых и независимых переменных

Перед началом построения модели мы должны разделить набор данных на две части:

1. **Независимые** переменные содержат список тех переменных, от которых зависит конкретное качество.
2. **Зависимая** переменная — это та переменная, которая зависит от значений других переменных.

Мы не используем полные данные для создания модели. Некоторые данные выбираются случайным образом и сохраняются для проверки качества модели. Это известно, как данные тестирования, а остальные данные называются данными обучения, на которых построена модель. Обычно 70% данных используются в качестве данных для обучения, а остальные 30% используются в качестве данных для тестирования.

3.5 Построение модели

После подготовки данных мы можем подогнать различные модели к обучающим данным и сравнить их производительность, чтобы выбрать алгоритм с хорошей производительностью. Поскольку это проблема регрессии, мы можем использовать RMSE (среднеквадратическую ошибку) и оценку R^2 в качестве показателей оценки.

3.6 Линейная регрессия

Мы начнем с линейной регрессии, поскольку это алгоритм перехода к любой проблеме регрессии. Алгоритм пытается сформировать линейную зависимость между входными функциями и целевой переменной, т. е. он соответствует прямой линии, заданной формулой:

$$y = W * X + b = \sum_{i=1}^{n} w_i * x_i + b$$

Формула 3.1 – Линейная регрессия

Где w_i соответствует коэффициенту признака x_i.

Величину этих коэффициентов можно дополнительно контролировать, используя условия регуляризации для функций стоимости. Добавление суммы величин коэффициентов приведет к тому, что коэффициенты будут близки к нулю, этот вариант линейной регрессии называется регрессией **Лассо.** Добавление суммы квадратов коэффициентов к функции стоимости приведет к тому, что коэффициенты будут находиться в одном диапазоне, и этот вариант называется регрессией **хребта.** Оба эти варианта помогают снизить сложность модели и, следовательно, снизить вероятность переобучения данных.

В результате обработки данных мы получаем следующие величины (Рис.3.10):

Model	RMSE	R2
LinearRegression	10.29	0.57
LassoRegression	10.68	0.54
RidgeRegression	10.29	0.57

Рис. 3.10 Результаты обработки данных линейной регрессии

Между производительностью этих трех алгоритмов нет большой разницы, мы можем построить коэффициенты, назначенные тремя алгоритмами для функций (Рис.3.11).

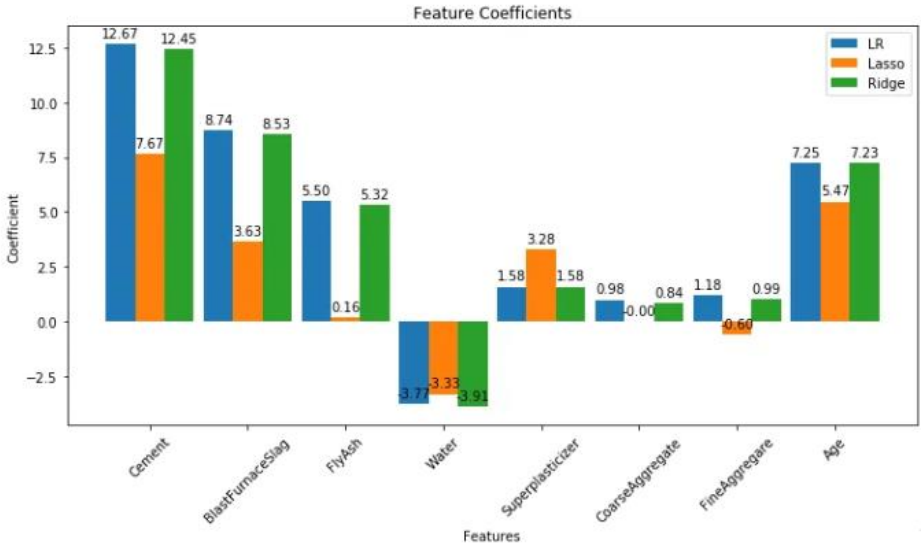

Рис. 3.11 Важность признаков алгоритмов линейной регрессии

Как видно на рисунке, регрессия Лассо подталкивает коэффициенты к нулю, а коэффициенты обычной линейной регрессии и гребневой регрессии почти одинаковы.

Мы можем дополнительно увидеть, каковы прогнозы, нанеся на график истинные значения и прогнозируемые значения (Рис.3.12).

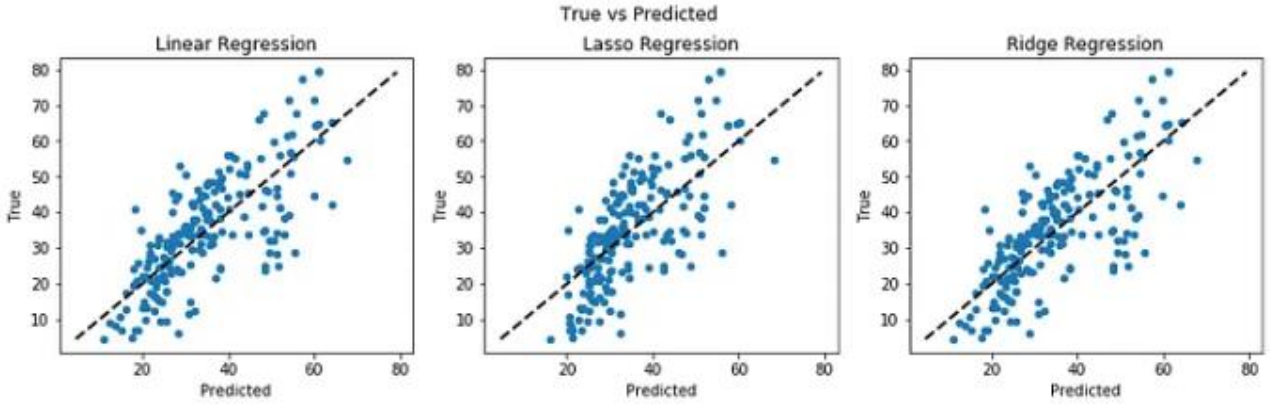

Рис. 3.13 График рассеяния для проверки значений прогноза регрессий

Если прогнозируемые значения и целевые значения равны, то точки на точечной диаграмме будут лежать на прямой линии. Как мы видим здесь, ни одна из моделей не предсказывает прочность на сжатие правильно.

3.7 Деревья решений

Алгоритм дерева решений представляет данные в виде древовидной структуры, где каждый узел представляет собой решение, принятое в отношении функции. В этом случае этот алгоритм даст лучшую производительность, поскольку у нас много нулей в некоторых входных функциях, как видно из их распределения на парном графике выше (Рис.3.13). Это поможет деревьям решений строить деревья на основе некоторых условий функций, которые могут еще больше повысить производительность.

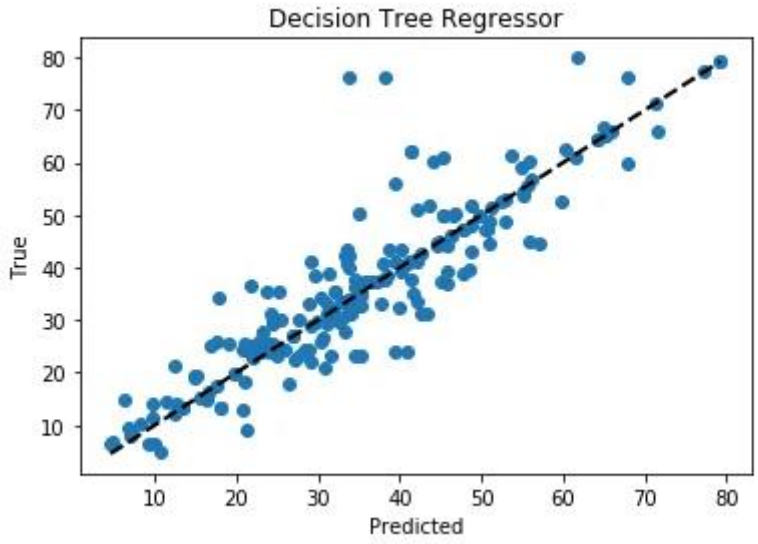

Рис. 3.13 График рассеяния для проверки значений прогноза дерева решений

Model	RMSE	R2
Decision Tree Regressor	7.31	0.78

Рис. 3.14 Результаты обработки данных дерева решений

Среднеквадратическая ошибка (RMSE) снизилась с 10,29 до 7,31, поэтому регрессор дерева решений значительно улучшил производительность. Это можно наблюдать на графике, так как больше точек ближе к линии (Рис.3.14).

3.8 Случайные леса

Использование регрессора дерева решений улучшило нашу производительность, мы можем еще больше повысить производительность, объединив больше деревьев. Random Forest Regressor тренирует случайно инициализированные деревья со случайными подмножествами данных, выбранных из обучающих данных, это сделает нашу модель более надежной (Рис.3.15).

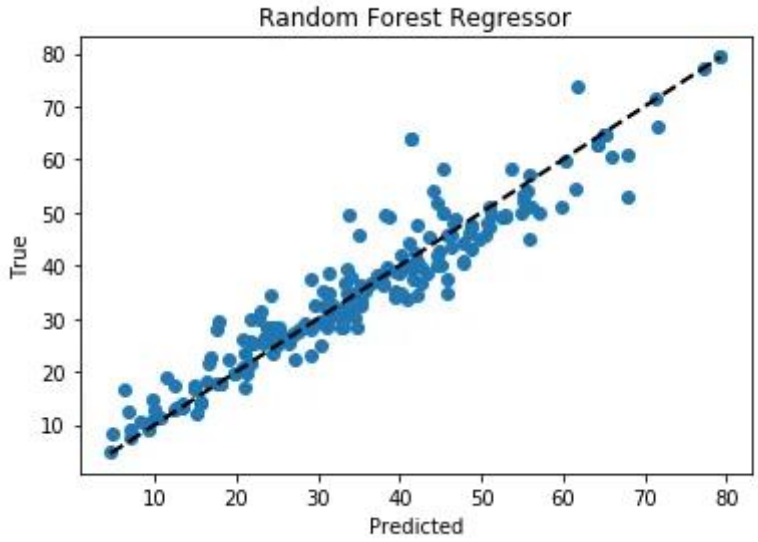

Рис. 3.15 График рассеяния для проверки значений прогноза случайных лесов

Model	RMSE	R2
Random Forest Regressor	5.08	0.89

Рис. 3.16 Результаты обработки данных случайных лесов

RMSE еще больше сократился за счет объединения нескольких деревьев (Рис.3.16). Мы можем построить график важности признаков для древовидных моделей. Важность функции показывает, насколько важна функция для модели при прогнозировании.

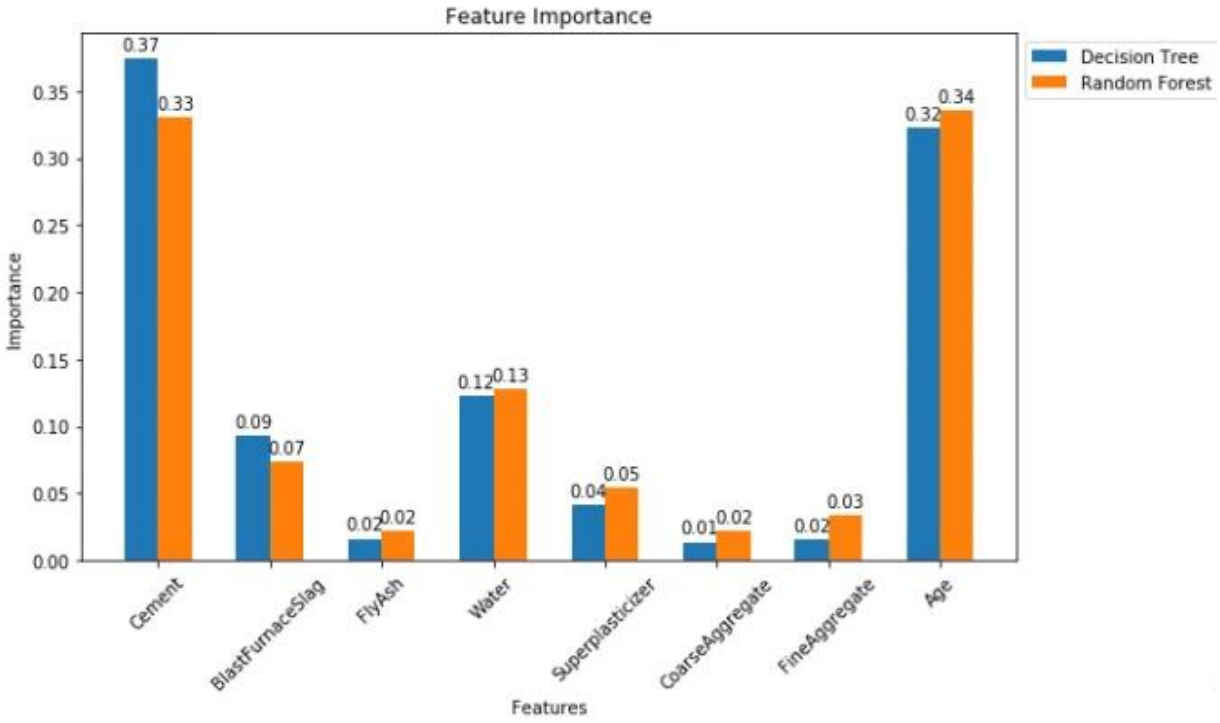

Рис. 3.17 Важность признаков алгоритмов дерева решений и случайных лесов

Цемент и возраст считаются наиболее важными характеристиками древесных моделей. Летучая зола, крупные и мелкие заполнители являются наименее важными факторами при прогнозировании прочности бетона (Рис.3.17).

3.9 Сравнение:

Наконец, давайте сравним результаты всех алгоритмов (Рис.3.18).

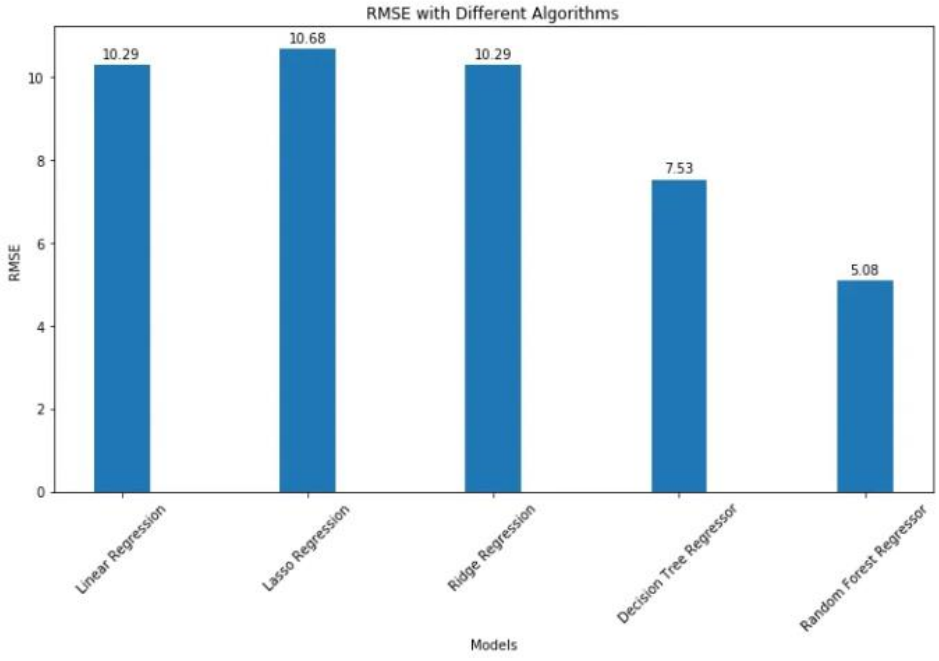

Рис. 3.18 Сравнение результатов всех алгоритмов (RMSE - среднеквадратическое отклонение)

3.10 Заключение

Мы проанализировали данные о прочности на сжатие и использовали машинное обучение для прогнозирования прочности бетона на сжатие. Мы использовали линейную регрессию и ее варианты, деревья решений и случайные леса, чтобы делать прогнозы и сравнивать их эффективность. Random Forest Regressor имеет самый низкий RMSE и является хорошим выбором для этой проблемы

ГЛАВА 4. РАЗРАБОТКА TELEGRAM-БОТА

4.1. Общая информация по работе

Цель работы – получить навыки создания телеграмм-бота, работающего с моделью классификатором.

Задачи:

1. Установка библиотеки работы с Telegram API

2. Загрузка модели машинного обучения

3. Создание бота

4. Задание функций взаимодействия с ботом для обработки входящих сообщений и нажатия кнопок в чате

5. Запуск Telegram-бота

Ход работы:

4.2 Установка библиотеки работы с Telegram API

Инсталляция пакетов pytelegrambotapi и emoji (последний - для вывода эмодзи)

```
%%bash
pip install pytelegrambotapi

pip install emoji
```

4.3 Загрузка модели машинного обучения

Отключение предупреждений в проекте

```
import warnings
warnings.filterwarnings('ignore')
```

4.4 Подключение к Google Drive. Подключение библиотек pickle и numpy

```
import pickle
import numpy as np
from google.colab import drive
drive.mount('/content/gdrive')
```

4.5 Загрузка модели из файла Concrete_strength_4features.pkl в model

```
modelPath = '/content/gdrive/My Drive/model.pkl'
model = pickle.load(open(modelPath, 'rb'))
```

4.6 Подключение библиотеки emoji. Получение эмодзи, использующихся в проекте.

```
import emoji
#https://unicode.org/emoji/charts/emoji-list.html
pen_Emoji = emoji.emojize(':pen:')
smiling_faceEmoji = emoji.emojize(':grinning_face:')
check_mark_Emoji = emoji.emojize(':check_mark:')
cross_mark_Emoji = emoji.emojize(':cross_mark:')
unhappyFace_Emoji = emoji.emojize(':slightly_frowning_face:')
```

4.7 Проверка работы модели машинного обучения

```
Прочность цемента: 63.71
```

4.8 Обучение классификатора

Перед созданием Telegram-бота необходимо получить TELEGRAM_API_TOKEN – ключ из символов, с помощью которого Вы управляете своим ботом. (Вы его получаете после создания нового бота в Telegram с помощью бота BotFather. Подробнее можно посмотреть здесь: https://botcreators.ru/blog/kak-sozdat-svoego-bota-v-botfather/ или https://habr.com/ru/post/350648/

4.9 Создание бота.

Нужно заменить 'AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA' на Ваш ключ.

```
TELEGRAM_API_TOKEN = 'AAAA22746:AAEzTV0rWdEu5uYzRfT1MmbOFKd2yUEYPTY'

import telebot
from telebot import types

bot = telebot.TeleBot(TELEGRAM_API_TOKEN)
```

4.10 Задание функций взаимодействия с ботом для обработки входящих сообщений и нажатия кнопок в чате

Функция *get_text_message* обрабатывает текстовый ввод, когда Вы первоначально вводите сообщение в чат Вашего бота. Если Вы ввели команду /start, то бот выводит сообщения, связанные с запросом ввода переменной для модели и регистрирует новую функцию обрабатывающую ввод текстовой информации в чат. Т.е. бот после ввода значения для получения доли цемента, запрашивает данные для доли воды, и т.д. После получения всех необходимых переменных выводятся все их значения и две кнопки подтверждающие или не подтверждающие правильность ввода. Результат нажатия той или иной кнопки - callback_data.

```
@bot.message_handler(content_types=['text'])
def get_text_message(message):
  if message.text == '/start':
    bot.send_message(message.from_user.id,'Сейчас мы рассчитаем прочность
бетона!')
    bot.send_message(message.from_user.id,'Введите количество цемента:')
    bot.register_next_step_handler(message, get_cement)
  else:
    bot.send_message(message.from_user.id,'Ввведите команду /start')


def get_cement(message):
  global cement
  cement = message.text
  cement = float(cement)
  bot.send_message(message.from_user.id,'Введите количество
суперпластификатора:')
  bot.register_next_step_handler(message, get_superplasticizer)
```

```
def get_superplasticizer(message):
  global superplasticizer
  superplasticizer = message.text
  superplasticizer = float(superplasticizer)
  bot.send_message(message.from_user.id,'Введите срок от начала заливки
бетона:')
  bot.register_next_step_handler(message, get_age)

def get_age(message):
  global age
  age = message.text
  age = float(age)
  bot.send_message(message.from_user.id,f'Цемент = {cement}, суперпласт. =
{superplasticizer}, срок = {age}')
  keyboard = types.InlineKeyboardMarkup()
  key_yes = types.InlineKeyboardButton(text = f'{check_mark_Emoji} ДА!!!',
callback_data='yes')
  keyboard.add(key_yes)
  key_no = types.InlineKeyboardButton(text = f'{cross_mark_Emoji} Ошибка в
данных', callback_data='no')
  keyboard.add(key_no)
  bot.send_message(message.from_user.id, text = 'Правильно ли введены
данные?', reply_markup=keyboard)
```

```
def get_age(message):
  global age
  age = message.text
  age = float(age)
  bot.send_message(message.from_user.id,f'Цемент = {cement}, вода = {water}, суперпласт. =
{superplasticizer}, срок = {age}')
  keyboard = types.InlineKeyboardMarkup()
  key_yes = types.InlineKeyboardButton(text = f'{check_mark_Emoji} ДА!!!', callback_data='yes')
  keyboard.add(key_yes)
  key_no = types.InlineKeyboardButton(text = f'{cross_mark_Emoji} Ошибка в данных',
callback_data='no')
  keyboard.add(key_no)
  bot.send_message(message.from_user.id, text = 'Правильно ли введены данные?',
reply_markup=keyboard)
```

Следующий блок кода обрабатывает нажатия на кнопки, основываясь на значениях callback_data. Если пользователь подтвердил правильность ввода (‘yes’), то данные загружаются в модель регрессор и рассчитывается результат. Если не подтвердил правильность (‘no’), то данные запрашиваются заново. Бот также предлагает перезапустить расчет и после успеха ввода.

```
@bot.callback_query_handler(func = lambda call: True)
def callback_worker(call):
  if call.data == 'yes':
    inputData = np.array([cement, superplasticizer, age]).reshape(1,-1)
    strength = round(model.predict(inputData)[0],2)
    bot.send_message(call.message.chat.id,f' Прочность бетона : {strength}
МПА')
    keyboard = types.InlineKeyboardMarkup()
    key_yes = types.InlineKeyboardButton(text = 'Да, пожалуйста',
callback_data='call_again')
    keyboard.add(key_yes)
    key_no = types.InlineKeyboardButton(text = 'Нет, спасибо',
callback_data='no_thanks')
    keyboard.add(key_no)
    bot.send_message(call.message.chat.id, text = 'Хотите выполнить ещё один
расчет?', reply_markup=keyboard)
  elif call.data == 'no':
    bot.send_message(call.message.chat.id,f' Жалко... {unhappyFace_Emoji}')
    bot.send_message(call.message.chat.id,'Давайте введем данные заново')
    bot.send_message(call.message.chat.id,'Введите количество цемента:')
    bot.register_next_step_handler(call.message, get_cement)
  elif call.data == 'call_again':
    bot.send_message(call.message.chat.id,'Новый расчет')
    bot.send_message(call.message.chat.id,'Введите количество цемента:')
    bot.register_next_step_handler(call.message, get_cement)
  elif call.data == 'no_thanks':
    bot.send_message(call.message.chat.id,f'Всегда Вам рады
{smiling_faceEmoji}')
```

4.11 Запуск Telegram-бота

Открываем в Telegram нашего бота и производим вычисления (Рис.4.1). Пока будет активна следующая ячейка бот будет работать по заданному алгоритму.

```
bot.polling(non_stop=True, interval=0)
```

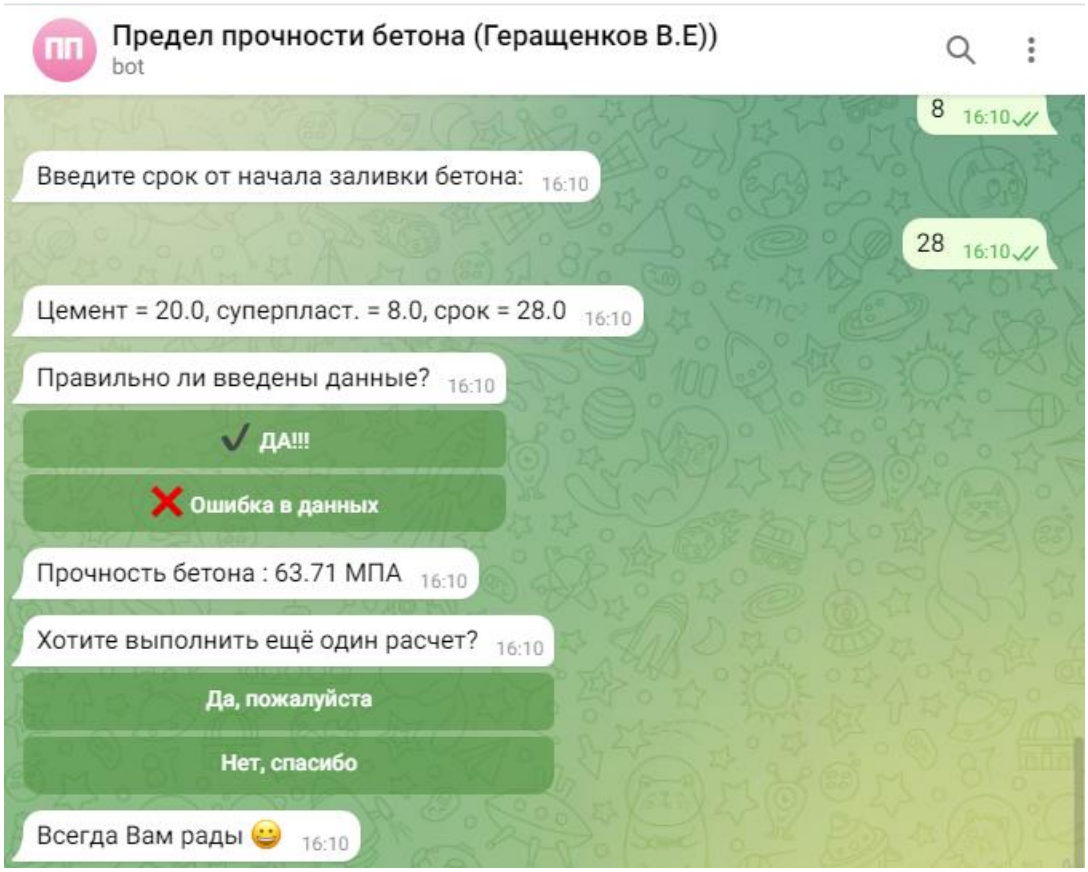

Рис. 4.1 Разработка Telegram-бота

Вывод по разделу:

В этом разделе реализован Telegram-бот, позволяющий осуществлять прогноз предела прочности на сжатие бетона за указанным пользователем срок по концентрации связующих компонентов в нем.

ЗАДАНИЕ К РАЗДЕЛУ
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ»

Обучающемуся:

Группа	ФИО
8ПМ1И	Геращенков Вадим Евгеньевич

Школа	Инженерная школа информационных технологий и робототехники	**Отделение школы (НОЦ)**	Отделение информационных технологий
Уровень образования	магистратура	**Направление/ООП/ОПОП**	09.04.04 «Программная инженерия»

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» :	
1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих	*Бюджет – 550 378 руб.* *Затраты на заработную плату – 238 277 руб.*
1. Нормы и нормативы расходования ресурсов	*Тариф на электроэнергию 5,8 кВт/ч*
1. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	*Налог во внебюджетные фонды 27,1Районный коэффициент – 1,3* *Накладные расходы – 70%*
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Расчет инновационного потенциала НТИ	SWOT-анализ; технология QuaD; Оценка научного уровня исследования.
1. Расчет сметы затрат на выполнение проекта	Расчет материальных затрат. Расчет основной и дополнительной заработной платы. Расчет отчислений во внебюджетные фонды. Расчет бюджета проекта.
Перечень графического материала:	
1. Матрица SWOT ***2. График разработки*** ***3. Бюджет НТИ***	

4. Реестр рисков НТИ

Дата выдачи задания к разделу в соответствии с календарным учебным графиком	01.03.2023 г.

Задание выдал консультант по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
профессор ОСГН ШБИП	Жиронкин Сергей Александрович	к.э.н.		

Задание принял к исполнению обучающийся:

Группа	ФИО	Подпись	Дата
8ПМ1И	Геращенков Вадим Евгеньевич		

Глава 5. ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

Введение

Развитие строительной отрасли требует применения новейших технологий таких как строительные машины, инструменты, современные и инновационные строительные материалы не только в самом процессе строительства, но также требует применения современных информационных технологий в процессах проектирования и эксплуатации строительных объектов. Применение таких технологий как искусственный интеллект является не просто данью современной моде, но и необходимостью.

Компании, применяющие искусственный интеллект в своих проектах, способны генерировать на 50% больше прибыли.

Применение ИИ не подразумевает использование роботов для выкладки кирпичей или вождения грузовиков. Это касается использование алгоритмов, способных решать самые сложные задачи и повышать эффективность и производительность.

С помощью ИИ можно проводить тесты на жизнеспособность решений, а также на эффективность материалов. Например, есть программное обеспечение, которое использует подключенные данные и машинное обучение для прогнозирования и приоритизации проблем высокого риска и дает представление об основных проблемах, с которыми сталкиваются руководители строительства.

Таким образом, IT играют важную роль в строительстве, где от качества используемой информации зависит эффективность использования строительных материалов.

5.1 Организация и планирование работ

При организации процесса был определен полный перечень необходимых работ, а также их исполнители и рациональная продолжительность. В качестве структуры, показывающей необходимые данные, был использован линейный график работ, представленный в таблице 1.

Таблица 1 – Перечень работ и продолжительность их выполнения

Этапы работы	**Исполнители**	**Загрузка исполнителей**
Постановка целей и задач	НР	НР – 100%
Составление и утверждение ТЗ	НР, И	НР – 100% И – 10%
Разработка календарного плана	НР, И	НР – 100% И – 10%
Анализ исследуемой области	И	И – 100%
Проектирование архитектуры ПО	НР, И	НР – 70% И – 100%
Проектирование базы данных	НР, И	НР – 70% И – 70%
Выбор языка программирования ифреймворка	И	И – 100%
Разработка ПО	И	И – 100%
Тестирование ПО	И	И – 100%
Оценка эффективности полученныхрезультатов	НР, И	НР – 50% И – 100%
Оформление пояснительной записки	И	И – 100%
Примечание: НР – научный руководитель, И – инженер		

5.1.2 Продолжительность этапов работ

Для расчета продолжительности этапов работ был выбран экспертный опытно-статистический метод. Определение вероятных (ожидаемых) значений продолжительности работ было выполнено по формуле (1):

$$t_{ож} = \frac{3 \cdot t_{min} + 2 \cdot t_{max}}{5} \quad (1)$$

где: tmin – минимальная продолжительность работы, дн.;

tmax – максимальная продолжительность работы, дн.

Для построения линейного графика необходимо рассчитать длительность этапов в рабочих днях, а затем перевести ее в календарные дни.Расчет продолжительности этапа в рабочих днях был рассчитан по формуле (2):

$$T_{\text{РД}} = \frac{t_{\text{ож}}}{K_{\text{ВН}}} \times \text{К}_{\text{Д}} \quad (2)$$

где: КВН – коэффициент выполнения работ, учитывающий влияние внешних факторов на соблюдение предварительно определенных;

КД - коэффициент, учитывающий дополнительное время на компенсацию непредвиденных задержек и согласование работ. Примем КД = 1,1.

Формула расчета продолжительности этапа в календарных днях (3):

$T_{КД} = T_{РД} \cdot T_{К}$, (3)

где ТК – коэффициент календарности, позволяющий перейти от длительности работ в рабочих днях к их аналогам в календарных днях, и рассчитываемый по формуле (4):

$$T_K = \frac{T_{\text{КАЛ}}}{T_{\text{КАЛ}} - T_{\text{ВД}} - T_{\text{ПД}}} \quad (4)$$

где:

$T_{\text{КАЛ}}$ – календарные дни, дн.;

$T_{\text{ВД}}$ – выходные дни, дн.;

$T_{\text{ПД}}$ – праздничные дни, дн.

При шестидневной рабочей неделе 2022 году коэффициент календарности равен: 365

$$T_K = \frac{365}{365-118} = 1.48$$

Полученные результаты трудозатрат на выполнение проекта отображены в

таблице 2, а линейный график работ – на рисунке 32.

Таблица 2 – Трудозатраты на выполнение проекта

Этап	Исполнители	Продолжительность работ, дни			Трудоемкость работ по исполнителям, чел-дн.			
					$T_{РД}$		$T_{КД}$	
		tmin	*tmax*	*to ж*	НР	И	НР	И
Постановка целей и задач	НР	1	2	1,4	1,54	–	1,88	–
Составление и утверждение ТЗ	НР, И	4	6	4,8	5,28	0,53	6,44	0,64
Разработка календарного плана	НР, И	1	2	1,4	1,54	0,15	1,88	0,19
Анализ исследуемой области	И	6	8	6,8	–	7,48	–	9,13
Проектирование архитектуры ПО	НР, И	12	18	14,4	11,09	15,84	13,53	19,32
Проектирование базы данных	НР, И	12	18	14,4	11,09	11,09	13,53	13,53
Выбор языка программирования и фреймворка	И	2	4	2,8	–	3,08	–	3,76
Разработка ПО	И	24	30	26,4	–	29,04	–	35,43

Тестирование ПО	И	4	6	4,8	–	5,28	–	6,44
Оценка эффективности полученных результатов	НР, И	1	2	1,4	0,77	1,54	0,94	1,88
Оформление пояснительной записки	И	12	18	14,4	–	15,84	–	19,32
Итого:				93	31,31	89,87	38,2	109,64

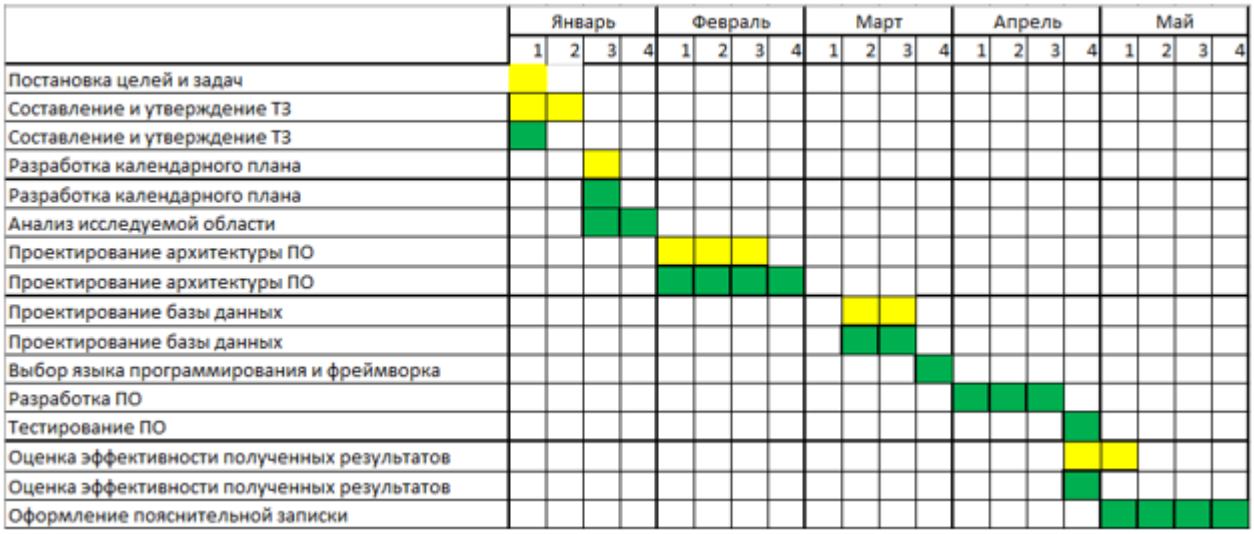

Рис. 5 – Линейный график работ

Желтым цветом выделены работы научного руководителя, зеленым – инженера.

5.2 Расчет сметы затрат на выполнение проекта

В состав затрат на создание проекта включается величина всех расходов,

необходимых для реализации комплекса работ, составляющих содержание данной разработки. Расчет сметной стоимости ее выполнения производился последующим статьям затрат:

1. материалы и покупные изделия;
2. заработная плата;
3. социальный налог;
4. расходы на электроэнергию (без освещения);
5. амортизационные отчисления;
6. прочие (накладные расходы) расходы.

5.2.3 Расчет затрат на материалы

Данная статья включает стоимость материалов, используемых при разработке проекта. При разработке настоящей работы затраты на материалыотсутствовали.

5.2.4 Расчет заработной платы

Стоимость спецоборудования для научных работ приведена в таблице 3.

Таблица 3 – Стоимость спецоборудования для научных работ

№ п / п	Наименование оборудования	Кол-во единиц оборудования	Цена единицы оборудования, тыс.руб.	Общая стоимость оборудования, тыс.руб.
1.	Ноутбук Acer Aspire 7745G	1	50	50

Баланс рабочего времени сотрудников приведен в таблице 4.

Таблица 4 – Баланс рабочего времени сотрудников

Показатели рабочего времени	**Руководитель Аксенов С.В.**	**Исполнитель Геращенков В.Е.**

Календарное число дней	365	365
Количество нерабочих дней - выходные дни - праздничные дни	5 2 1 4	5 2 1 4
Потери рабочего времени - отпуск - невыходы по болезни	48	48
Действительный годовой фонд рабочего времени	251	251

Размер заработной платы рассчитывается по следующей формуле:

$$З_м = З_б \times (k_{пр} + k_д) \times k_р \quad (5)$$

Где: Зб – базовый оклад, руб.;

kпр – премиальный коэффициент;

kд – коэффициент доплат и надбавок;

kр – районный коэффициент.

Расчет заработной платы представлен в таблице 5.

Таблица 5 – Расчёт заработной платы

Исполнители	$З_б$, руб.	$k_{пр}$	$k_д$	$k_р$	$З_м$, руб	$З_{дн}$, руб.	$Т_р$, раб. дн.	$З_{осн}$, руб.
Руководитель	37700	-	-	1,3	49010.0	2030,7	15	30460
Исполнитель	19200	-	-	1,3	24960,0	1034,2	180	186156

Итоговая заработная плата исполнителей приведена в таблице 6.

Таблица 6 – Заработная плата исполнителей

Заработная плата	**Руководитель**	**Исполнитель**
Основная зарплата	30460	186156
Дополнительная зарплата	3046	18615
Зарплата исполнителя	33506	204711
Итого по статье $C_{зп}$	238277	

Отчисление во внебюджетные вычисляются как 27.1% от общей заработной платы и составляют 64573 рублей.

Расходы на командировки составляют 30317 рублей. Накладные расходы составляют 166793,9 рубля.

Затрат на электроэнергию составляют 5,8*72=417.6 рублей Сводная таблица затрат на проект представлена в таблице 7.

Таблица 7 – Группировка затрат по статьям

Статьи									
Сырье, материалы (за вычетом возвратных отходов), покупные изделия и полуфабрикаты	Специальное оборудование длянаучных (экспериментальных) работ	Основная заработная плата	Дополнительная заработная плата	Отчисления на социальные нужды	Научные и производственные командировки	Оплата работ, выполняемых сторонними организациями и предприятиями	Прочие прямые расходы	Накладные расходы	Итого плановая себестоимость
0	50000	216616	21661	64573	30317	0	417	166793,9	550 377,9

5.3.1 Реестр рисков проекта

Реестр рисков приведён в таблице 8.

Таблица 8 – Реестр рисков

№	Риск	Потенциальное воздействие	Вероятность наступления (1-5)	Влияние риска (1-5)	Уровень риска*	Способы смягчения риска
1	Болезнь исполнителя	Увеличение срока выполнения проекта	2	3	Средний	Найм нескольких исполнителей
2	Выход компьютераиз строя	Дополнительные финансовые затраты, увеличение срока выполнения проекта	1	5	Низкий	Покупка более надежного компьютера, покупка нескольких компьютеров
3	Ограничение доступа к международой научной литературе	Снижение качества проработанности проекта	3	4	Высокий	Настройка VPN на компьютере, переезд за границу

Итого выявлено 3 риска.

5.3 Определение ресурсной, финансовой, бюджетной, социальной и экономической эффективности исследования

Определение эффективности происходит на основе расчета интегрального показателя эффективности научного исследования. Его нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности

Интегральный финансовый показатель разработки определяется как:

$$I_{\text{финр}}^{исп.i} = \frac{\Phi_{\text{р}i}}{\Phi_{\text{max}}},$$

где $I_{\text{финр}}^{\text{исп.i}}$ – интегральный финансовый показатель разработки;

$\Phi_{\text{р}i}$ – стоимость *i*-го варианта исполнения;

Φ_{max} – максимальная стоимость исполнения научно-исследовательского проекта (в т.ч.

аналоги).

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное увеличение бюджета затрат разработки в разах (значение больше единицы), либо соответствующее численное удешевление стоимости разработки в разах (значение меньше единицы, но больше нуля).

Для определения эффективности были рассмотрены следующие аналоги:

- **Аналог 1** – За аналог 1 принят продукт компании Lenovo ThinkPad X1 Carbon Gen.10 (Intel). – компьютер для разработки программного обеспечения и компьютерных моделей для анализа строительных смесей
- **Аналог 2** – За аналог 2 принят компьютер Apple MacBook Pro 2023 – компьютер для разработки программного обеспечения и компьютерных моделей для анализа строительных смесей.

Смета бюджетов для рассмотренных аналогов приведена в Таблице 15.

Таблица 15 – Смета бюджетов для рассмотренных аналогов

№	**Проектируемая АСУ ТП**	**Аналог 1**	**Аналог 2**
Бюджет затрат, руб.	550 378	753 183	679 992

Рассчитаем интегральный финансовый показатель для трех систем:

$$I_{\text{финРП}}^{\text{исп.}i} = \frac{\Phi_{Pi}}{\Phi_{MAX}} = \frac{550\ 378}{753\ 183} = 0{,}73;$$

$$I_{\text{финА1}}^{\text{исп.}i} = \frac{\Phi_{Pi}}{\Phi_{MAX}} = \frac{753\ 183}{753\ 183} = 1;$$

$$I_{\text{финА2}}^{\text{исп.}i} = \frac{\Phi_{Pi}}{\Phi_{MAX}} = \frac{679\ 992}{753\ 183} = 0{,}90;$$

Далее определим интегральный показатель ресурсоэффективности вариантов исполнения проекта по формуле:

$$I_{pi} = \sum a_i \cdot b_i,$$

где a_i – весовой коэффициент i-го варианта исполнения; b_i – бальная оценка i-го варианта исполнения, устанавливается экспертным путем.

Сравнительная оценка характеристик вариантов исполнения проекта приведена в Таблица 16 – Сравнительная оценка характеристик вариантов исполнения

Критерии	**Весовой коэффициент параметра**	**Реализованный проект**	**Аналог №1**	**Аналог №2**
Безопасность	0,25	4	5	4
Надежность	0,20	5	4	5
Экономичность	0,15	5	4	4
Удобство в эксплуатации	0,15	4	4	5
Повышение производительности	0,25	5	4	4
Итого	**1**	**4,6**	**4,2**	**4,35**

Интегральный показатель эффективности вариантов исполнения разработки определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{\text{ИСП1}} = \frac{I_{\text{РП}}}{I_{\text{финРП}}^{\text{исп.}i}} = \frac{4,6}{0,73} = 6,30;$$

$$I_{\text{ИСП2}} = \frac{I_{\text{А1}}}{I_{\text{финА1}}^{\text{исп.}i}} = \frac{4,2}{1} = 4,2;$$

$$I_{\text{ИСП3}} = \frac{I_{\text{А2}}}{I_{\text{финА2}}^{\text{исп.}i}} = \frac{4,35}{0,90} = 4,83.$$

Сравнение интегрального показателя эффективности вариантов исполнения разработки позволит определить сравнительную эффективность проекта и выбрать наиболее целесообразный вариант из предложенных. Результат сравнительной эффективности проекта и сравнительная эффективность анализа (Таблица 17) получены с помощью формулы:

$$Э_{СР} = \frac{I_{ИСП1}}{I_{ИСП2}}.$$

Таблица 17 – Сравнительная эффективность разработки

№	Показатели	Разработка	Аналог 1	Аналог 2
1	Интегральный финансовый показатель	0,74	1	0,87
2	Интегральный показатель ресурсоэффективности	4,6	4,2	4,35
3	Интегральный показатель эффективности	6,30	4,2	4,83
4	Сравнительная эффективность вариантов исполнения	1,50	1	1,31

Таким образом, основываясь на определении ресурсосберегающей, финансовой эффективности исследования, проведя необходимый сравнительный анализ, можно сделать вывод о превосходстве выполненной разработки над аналогами как по финансовой эффективности, так и по ресурсной эффективности.

3.3 Выводы по разделу

Таким образом, в ходе выполнения раздела были применены различные аналитические инструменты и расчеты, с помощью которых были решены следующие задачи:

– оценка коммерческого потенциала и перспективности проведения научных исследований, определение потенциальных потребителей и выявление конкурентных преимущества разработки, а именно: повышение надежности, безопасности и производительности, а также более низкая цена;

– составление SWOT-анализа, в котором были определены стратегии по использованию возможностей и нивелированию угроз и слабых сторон;

– проведено планирование научно-исследовательских работ, расчет трудозатрат и составлен календарный план-график проекта, определена ресурсная, финансовая, бюджетная, социальная и экономическая эффективности исследования.

С учетом решеных задач можно сделать вывод о том, что проект является конкурентоспособным и более ресурсоэффективным по сравнению с имеющимися аналогами на рынке.

ЗАДАНИЕ К РАЗДЕЛУ
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Обучающемуся:

Группа	ФИО
8ПМ1И	Геращенков Вадим Евгеньевич

Школа	Инженерная школа информационных технологий и робототехники	**Отделение школы (НОЦ)**	Отделение информационных технологий
Уровень образования	магистратура	**Направление/ООП/ОПОП**	09.04.04 «Программная инженерия»

Исходные данные к разделу «Социальная ответственность» :	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Объект исследования – результаты лабораторных испытаний. Область применения – научные исследования, медицинские учреждения и лаборатории. Рабочая зона оборудована 4 местами, каждое из которых включает в себя: стул, компьютер с периферийными устройствами, расположенном на столе. Технологический процесс представляет собой работу с языком программирования Python, в редакторе Google Colab.
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Правовые и организационные вопросы обеспечения безопасности: • специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; • организационные мероприятия при компоновке рабочей зоны.	- ГОСТ 12.2.032-78 «ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования». - ГОСТ Р 50923-96 «Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения».
2. Производственная безопасность: 2.1. Анализ выявленных вредных и опасных факторов. 2.2. Обоснование мероприятий по снижению воздействия.	Вредные производственные факторы: - Отклонение показателей микроклимата. - Недостаточная освещенность рабочей зоны. - Превышение уровня шума. - Повышенный уровень электромагнитных излучений. Опасные производственные факторы: - Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека.

3. Экологическая безопасность:	– Анализ воздействия объекта на литосферу: утилизация отходов, связанные с выходом из строя ПК, люминесцентных ламп и др.
4. Безопасность в чрезвычайных ситуациях:	Типичная ЧС – пожар. – разработка превентивных мер по предупреждению ЧС; – разработка действий в результате возникшей ЧС и мер по ликвидации её последствий.

Дата выдачи задания к разделу в соответствии с календарным учебным графиком	01.03.2023 г.

Задание выдал консультант по разделу «Социальная ответственность»:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
профессор ООД ШБИП	Федорчук Юрий Митрофанович	к.т.н.		

Задание принял к исполнению обучающийся:

Группа	ФИО	Подпись	Дата
8ПМ1И	Геращенков Вадим Евгеньевич		

6.СОЦИАЛЬНАЯ ОТВЕТСВЕННОСТЬ

6.1 Введение

Социальная ответственность - ответственность отдельного ученого и научного сообщества перед обществом. Первостепенное значение при этом имеет безопасность применения технологий, которые создаются на основе достижений науки, предотвращение или минимизация возможных негативных последствий их применения, обеспечение безопасного как для испытуемых, как и для окружающей среды проведения исследований.

Анализ результатов лабораторных испытаний, обработка и разработка модели на основе полученных данных, а также соответствующих программных средств осуществлялась на ПЭВМ.

В данном разделе рассматриваются опасные и вредные факторы, оказывающие влияние на производственную деятельность разработчика, воздействие объекта исследования на окружающую среду, правовые и организационные вопросы и мероприятия в чрезвычайных ситуациях.

6.2 Производственная безопасность

6.2.1 Вредные факторы

6.2.1.1 Отклонение показателей микроклимата в помещении

Проанализируем микроклимат в помещении, где находится рабочее место. Микроклимат производственных помещений определяют следующие параметры: температура, относительная влажность, скорость движения воздуха. Эти факторы влияют на организм человека, определяя его самочувствие.

Оптимальные и допустимые значения параметров микроклимата приведены в таблице 1 и 2.

Таблица 1. Оптимальные нормы микроклимата

Период года	Температура воздуха, С°	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	19-23	40-60	0.1
Теплый	23-25		0.2

Таблица 2. Допустимые нормы микроклимата

Период года	Температура воздуха, С°		Относительная влажность воздуха, %	Скорость движения воздуха, м/с
	Нижняя допустимая граница	Верхняя допустимая граница		
Холодный	15	24	20-80	<0.5
Теплый	22	28	20-80	<0.5

Общая площадь рабочего помещения составляет 42м^2, объем составляет 147м^3. По СанПиН 2.2.2/2.4.1340-03 санитарные нормы составляют 6,5 м^2 и 20 м^{3}объема на одного человека. Исходя из приведенных выше данных, можно сказать, что количество рабочих мест соответствует размерам помещения по санитарным нормам.

После анализа габаритных размеров рассмотрим микроклимат в этой комнате. В качестве параметров микроклимата рассмотрим температуру, влажность воздуха, скорость ветра.

В помещении осуществляется естественная вентиляция посредством наличия легко открываемого оконного проема (форточки), а также дверного проема. По зоне действия такая вентиляция является общеобменной. Основной недостаток - приточный воздух поступает в помещение без предварительной очистки и нагревания. Согласно нормам, СанПиН 2.2.2/2.4.1340-03 объем воздуха необходимый на одного человека в помещении без дополнительной вентиляции должен быть более 40м^3[1]. В нашем случае объем воздуха на одного человека составляет 42 м^3, из этого следует, что дополнительная вентиляция не требуется. Параметры микроклимата поддерживаются в холодное время года за счет систем водяного отопления с нагревом воды до 100°С, а в теплое время года – за счет кондиционирования, с параметрами согласно [2]. Нормируемые параметры микроклимата,

ионного состава воздуха, содержания вредных веществ должны соответствовать требованиям [3].

6.2.2.2 Превышение уровней шума

Одним из наиболее распространенных в производстве вредных факторов является шум. Он создается вентиляционным и рабочим оборудованием, преобразователями напряжения, рабочими лампами дневного света, а также проникает снаружи. Шум вызывает головную боль, усталость, бессонницу или сонливость, ослабляет внимание, память ухудшается, реакция уменьшается.

Основным источником шума в комнате являются компьютерные охлаждающие вентиляторы и. Уровень шума варьируется от 35 до 42 дБА. Согласно СанПиН 2.2.2 / 2.4.1340-03, при выполнении основных работ на ПЭВМ уровень шума на рабочем месте не должен превышать 80 дБА [4].

При значениях выше допустимого уровня необходимо предусмотреть средства индивидуальной защиты (СИЗ) и средства коллективной защиты (СКЗ) от шума.

Средства коллективной защиты:

1. Устранение причин шума или существенное его ослабление в источнике образования;
2. Изоляция источников шума от окружающей среды (применение глушителей, экранов, звукопоглощающих строительных материалов, например, любой пористый материал – шамотный кирпич, микропористая резина, поролон и др.);
3. Применение средств, снижающих шум и вибрацию на пути их распространения;

Средства индивидуальной защиты;

1. Применение спецодежды и защитных средств органов слуха: наушники, беруши, антифоны.

6.2.1.3 Повышенный уровень электромагнитных излучений

Источником электромагнитных излучений в нашем случае являются дисплеи ПЭВМ. Монитор компьютера включает в себя излучения рентгеновской, ультрафиолетовой и инфракрасной области, а также широкий диапазон электромагнитных волн других частот.

Согласно СанПиН 2.2.2/2.4.1340-03 напряженность электромагнитного поля по электрической составляющей на расстоянии 50 см вокруг ВДТ не должна превышать 25В/м в диапазоне от 5Гц до 2кГц, 2,5В/м в диапазоне от 2 до 400кГц [1]. Плотность магнитного потока не должна превышать в диапазоне от 5 Гц до 2 кГц 250нТл, и 25нТл в диапазоне от 2 до 400кГц. Поверхностный электростатический потенциал не должен превышать 500В [1]. В ходе работы использовалась ПЭВМ типа Acer VN7-791 со следующими характеристиками: напряженность электромагнитного поля 2,5В/м; поверхностный потенциал составляет 450 В (основы противопожарной защиты предприятий ГОСТ 12.1.004 и ГОСТ 12.1.010 – 76.) [5].

При длительном постоянном воздействии электромагнитного поля (ЭМП) радиочастотного диапазона при работе на ПЭВМ у человеческого организма возникают сердечно-сосудистые, респираторные и нервные расстройства, головные боли, усталость, ухудшение состояния здоровья, гипотония, изменения сердечной мышцы проводимости. Тепловой эффект ЭМП характеризуется увеличением температуры тела, локальным селективным нагревом тканей, органов, клеток за счет перехода ЭМП на теплую энергию.

Предельно допустимые уровни (ПДУ) облучения (по *ОСТ 54 30013-83)*:

a. до 10 мкВт/см2, время работы (8 часов);

b. от 10 до 100 мкВт/см2, время работы не более 2 часов;

c. от 100 до 1000 мкВт/см2, время работы не более 20 мин. при условии пользования защитными очками;

d. для населения в целом ППМ не должен превышать 1 мкВт/см2.

Защита человека от опасного воздействия электромагнитного излучения осуществляется следующими способами:

СКЗ:

1. защита временем;
2. защита расстоянием;
3. снижение интенсивности излучения непосредственно в самом источнике излучения;
4. заземление экрана вокруг источника;
5. защита рабочего места от излучения.

СИЗ:

1. Очки и специальная одежда, выполненная из металлизированной ткани (кольчуга). При этом следует отметить, что использование СИЗ возможно при кратковременных работах и является мерой аварийного характера. Ежедневная защита обслуживающего персонала должна обеспечиваться другими средствами;
2. Вместо обычных стекол используют стекла, покрытые тонким слоем золота или диоксида олова (SnO_2).

6.2.1.4 Недостаточная освещенность

Для обеспечения требуемой освещенности необходимо использовать совмещенное освещение, создаваемое сочетанием естественного и искусственного освещения. При данном этапе развития осветительной техники целесообразно использовать люминесцентные лампы, которые по сравнению с лампами накаливания имеют большую светоотдачу на ватт потребляемой мощности и более естественный спектр.

Минимальный уровень средней освещенности на рабочих местах с постоянным пребыванием людей должен быть не менее 200 лк.

В расчётном задании должны быть решены следующие вопросы:

- выбор системы освещения;
- выбор источников света;
- выбор светильников и их размещение;
- выбор нормируемой освещённости;
- расчёт освещения методом светового потока.

В данном расчётном задании для всех помещений рассчитывается общее равномерное освещение.

Таблица 3. Габариты помещения.

Параметр	Обозначение	Значение, м
Длина	A	12
Ширина	B	10
Высота помещения	H	3,5

Расчёт общего равномерного искусственного освещения горизонтальной рабочей поверхности выполняется методом коэффициента светового потока, учитывающим световой поток, отражённый от потолка и стен.

Световой поток лампы определяется по формуле:

$$Ф_{рас} = E_н \cdot S \cdot K_з \cdot Z / (N \cdot \square) \quad (1)$$

Где $E_н$ – нормируемая минимальная освещённость по СНиП 23-05-95, лк; S – площадь освещаемого помещения, м^2; $K_з$ – коэффициент запаса, учитывающий загрязнение светильника (источника света, светотехнической арматуры, стен и пр., т. е. отражающих поверхностей), наличие в атмосфере цеха дыма, пыли (табл. 4.9); Z – коэффициент неравномерности освещения, отношение $E_{ср} / E_{min}$. Для люминесцентных ламп при расчётах берётся равным 1,1; N – число ламп в помещении; □– коэффициент использования светового потока.

Коэффициент использования светового потока показывает, какая часть светового потока ламп попадает на рабочую поверхность. Он зависит от индекса помещения i, типа светильника, высоты светильников над рабочей поверхностью h и коэффициентов отражения стен □$_с$ и потолка □$_п$.

Индекс помещения определяется по формуле:

$$i = S / h \cdot (A + B) \quad (2)$$

Проведем расчет индекса помещения:

Площадь помещения:

$$S = A * B = 12 * 10 = 120 \text{ м}^2$$

Индекс:

$$i = \frac{S}{h * (A + B)} = \frac{120}{2.35 * (12 + 10)} = 2.32$$

Согласно этим данным коэффициент использования светового потока будет равен 56 % или в долях = 0,56.

Коэффициенты отражения оцениваются субъективно (табл. 4.10) [БЖД Практикум 2009-2020].

Согласно указанной методике выбираем тип источника света.

Наиболее подходящим вариантом является 40 ваттная лампа ЛБ, у которой Ф=2800 лм. Для выбранного типа лампы подходит светильник ОД-2-40 с размерами: длина = 1230 мм, ширина = 266 мм.

Из уравнения 1.5.1 находим количество ламп для помещения

N= $E_н$ *S* $K_з$ * Z/Ф*□□= 200 • 120 • 1,3 • 1,1 / 2800 • 0,56 = 21,875;

Принимаем N=24 лампы или 12 светильников.

Размещаем светильники в 3 ряда по 4 светильника в ряду с соблюдением условий:

L – расстояние между соседними светильниками или рядами (если по длине (А) и ширине (В) помещения расстояния различны, то они обозначаются L_A и L_B),

L – расстояние между соседними светильниками или рядами (если по длине (А) и ширине (В) помещения расстояния различны, то они обозначаются L_A и L_B),

l – расстояние от крайних светильников или рядов до стены.

Оптимальное расстояние *l* от крайнего ряда светильников до стены рекомендуется принимать равным *L*/3.

Сначала определим световой поток расчетный.

Ф = $E_н$ • S • $K_з$ • Z / (N • □□□□=200 • 120 • 1,3 • 1,1 / (24 • 0,56) = 2554 лм;

Проведем проверку выполнения условия соответствия:

- 10% ≤ ($Ф_{расч}$ – $Ф_{станд}$) / $Ф_{расч}$ • 100% ≤ + 20%

Подставляя численные значения получаем:

- 10% ≤ (2800 – 2554) / 2554 • 100 % ≤ + 20%

- 10% ≤ +9,6% ≤ + 20%

Результат расчета укладывается в допустимые пределы.

Определим мощность осветительной установки:

P = N • P_i = 24 • 40 Вт = 960 Вт.

Теперь определим расстояния между светильниками по длине и ширине помещения.

12000 = 3 • L_A + 4 • 1230 + 2 / 3 • L_A;

L_A= (12000 - 4920) • 3 / 11 = 1930 мм;

L_A / 3 = 644 мм;

10000 = 2 • L_B + 3 • 266 + 2 / 3 • L_B;

L_B= (10000 - 798) • 3 / 8 = 3450 мм;

L_B / 3 = 1150 мм.

Рисуем схему размещения светильников на потолке для обеспечения общего равномерного освещения.

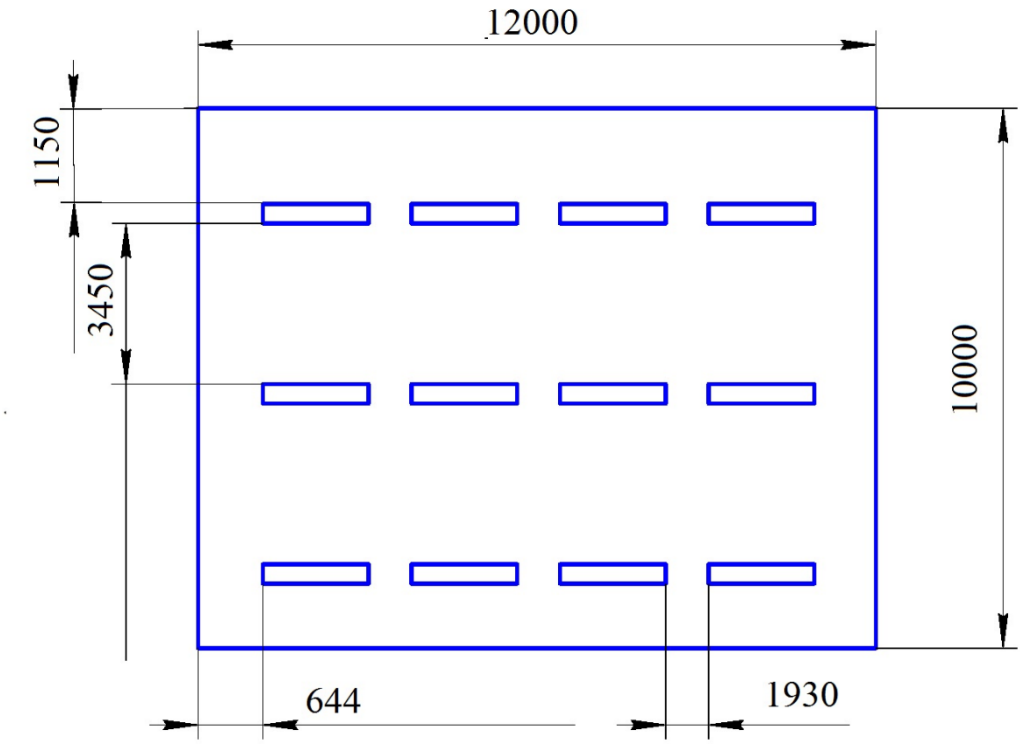

Рис.6 План размещения светильников на потолке

6.2.2 Опасные факторы

6.2.2.1 Электроопасность; класс электроопасности помещения, безопасные номиналы I, U, $R_{заземления}$, СКЗ, СИЗ; Поражение электрическим током

К опасным факторам можно отнести наличие в помещении большого количества аппаратуры, использующей однофазный электрический ток напряжением 220 В и частотой 50Гц. По опасности электропоражения комната относится к помещениям без повышенной опасности, так как отсутствует повышенная влажность, высокая температура, токопроводящая пыль и возможность одновременного соприкосновения токоведущих элементов с заземленными металлическими корпусами оборудования [6].

Лаборатория относится к помещению без повышенной опасности поражения электрическим током. Безопасными номиналами являются: $I < 0{,}1$ А; $U < (2\text{-}36)$ В; $R_{зазем} < 4$ Ом.

Для защиты от поражения электрическим током используют СИЗ и СКЗ.

Средства коллективной защиты:

1. защитное заземление, зануление;
2. малое напряжение;

3. электрическое разделение сетей;

4. защитное отключение;

5. изоляция токоведущих частей;

6. оградительные устройства.

7. Использование щитов, барьеров, клеток, ширм, а также заземляющих и шунтирующих штанг, специальных знаков и плакатов.

Средства индивидуальной защиты:

1. Использование диэлектрических перчаток, изолирующих клещей и штанг, слесарных инструментов с изолированными рукоятками, указатели величины напряжения, калоши, боты, подставки и коврики.

2.

6.2.2.2 Пожароопасность, категория пожароопасности помещения, марки огнетушителей, их назначение и ограничение применения; Приведена схема эвакуации.

По взрывопожарной и пожарной опасности помещения подразделяются на категории А, Б, В1-В4, Г и Д.

Согласно НПБ 105-03 лаборатория относится к категории В – горючие и трудно горючие жидкости, твердые горючие и трудно горючие вещества и материалы, вещества и материалы, способные при взаимодействии с водой, кислородом воздуха или друг с другом только гореть, при условии, что помещения, в которых находится, не относятся к категории наиболее опасных А или Б.

По степени огнестойкости данное помещение относится к 1-й степени огнестойкости по СНиП 2.01.02-85 (выполнено из кирпича, которое относится к трудносгораемым материалам).

Возникновение пожара при работе с электронной аппаратурой может быть по причинам как электрического, так и неэлектрического характера.

Причины возникновения пожара неэлектрического характера:

а) халатное неосторожное обращение с огнем (курение, оставленные без присмотра нагревательные приборы, использование открытого огня);

Причины возникновения пожара электрического характера: короткое замыкание, перегрузки по току, искрение и электрические дуги, статическое электричество и т. п.

Для локализации или ликвидации загорания на начальной стадии используются первичные средства пожаротушения. Первичные средства пожаротушения обычно применяют до прибытия пожарной команды.

Огнетушители водо-пенные (ОХВП-10) используют для тушения очагов пожара без наличия электроэнергии. Углекислотные (ОУ-2) и порошковые огнетушители предназначены для тушения электроустановок, находящихся под напряжением до 1000В. Для тушения токоведущих частей и электроустановок применяется переносной порошковый огнетушитель, например, ОП-5.

В общественных зданиях и сооружениях на каждом этаже должно размещаться не менее двух переносных огнетушителей. Огнетушители следует располагать на видных местах вблизи от выходов из помещений на высоте не более 1,35 м. Размещение первичных средств пожаротушения в коридорах, переходах не должно препятствовать безопасной эвакуации людей.

Для предупреждения пожара и взрыва необходимо предусмотреть:

1. Специальные изолированные помещения для хранения и разлива легковоспламеняющихся жидкостей (ЛВЖ), оборудованные приточно-вытяжной вентиляцией во взрывобезопасном исполнении - соответствии с ГОСТ 12.4.021-75 и СНиП 2.04.05-86;

2. Специальные помещения (для хранения в таре пылеобразной канифоли), изолированные от нагревательных приборов и нагретых частей оборудования;

3. Первичные средства пожаротушения на производственных участках (передвижные углекислые огнетушители ГОСТ 9230-77, пенные огнетушители ТУ 22-4720-80, ящики с песком, войлок, кошма или асбестовое полотно);

4. Автоматические сигнализаторы (типа СВК-3 М 1) для сигнализации о присутствии в воздухе помещений предвзрывных концентраций горючих паров растворителей и их смесей.

Лаборатория полностью соответствует требованиям пожарной безопасности, а именно, наличие охранно-пожарной сигнализации, плана эвакуации, изображенного на рисунке 1, порошковых огнетушителей с поверенным клеймом, табличек с указанием направления к запасному (эвакуационному) выходу.

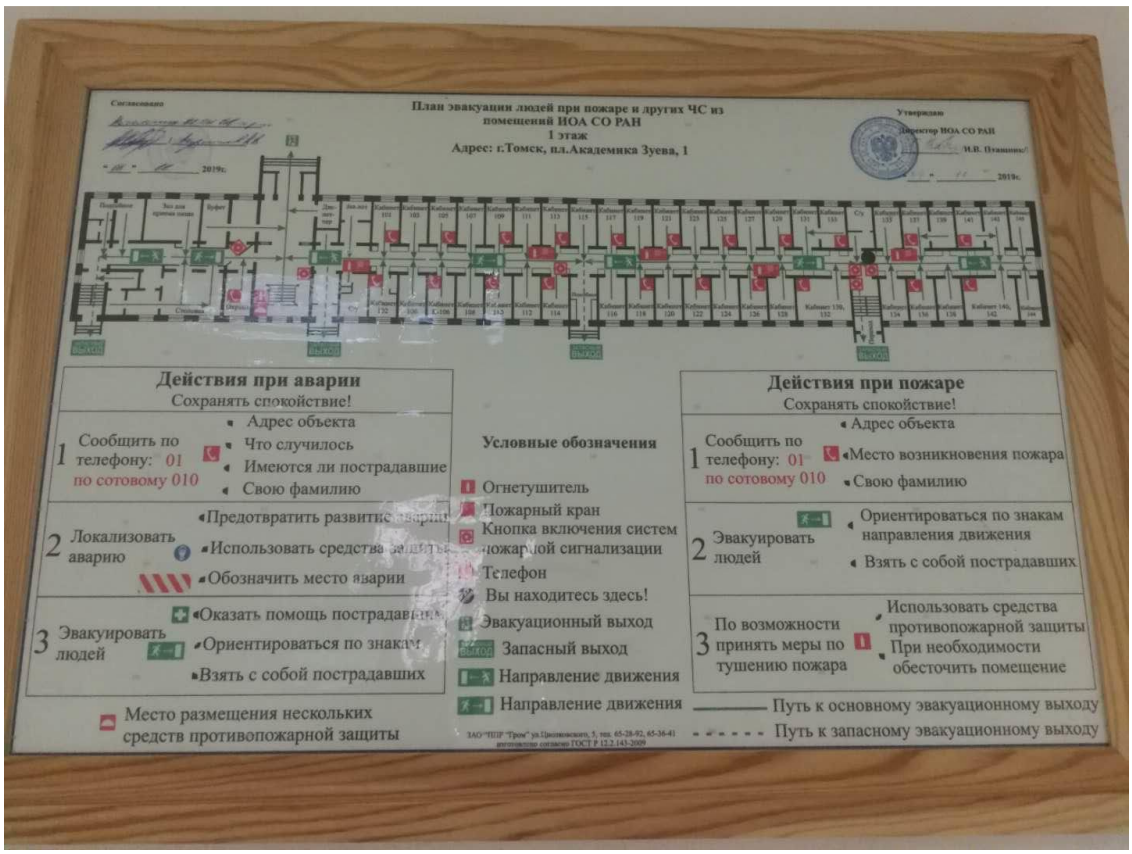

Рис 6.1. План эвакуации

6.3 Экологическая безопасность

Во время выполнения выпускной квалификационной работы вынуждены использовать черновики (предварительная запись информации) на бумажном носителе. Записи несут в себе конфиденциальную, а иногда даже секретную информацию. Чтобы повторно использовать бумагу для записей необходимо бумагу с записями шредировать с помощью шредера, спрессовать для уменьшения объема, упаковать в герметичную упаковку и хранить на складе до накопления объема для 1 транспортной единицы, после чего отправить на утилизацию макулатуры в ближайший ее пункт приема.

В компьютерах огромное количество компонентов, которые содержат токсичные вещества и представляют угрозу, как для человека, так и для окружающей среды.

К таким веществам относятся:

- Свинец (накапливается в организме, поражая почки, нервную систему);
- Никель и цинк (могут вызывать дерматит);

- Щелочи (прожигают слизистые оболочки и кожу).

Поэтому компьютер требует специальных комплексных методов утилизации.

Утилизацию компьютера можно провести следующим образом:

- Отделить металлические детали от неметаллов;
- Разделить углеродистые металлы от цветмета;
- Пластмассовые изделия (крупногабаритные) измельчить для уменьшения объема;
- Копир-порошок упаковать в отдельную упаковку, точно также, как и все проклассифицированные и измельченные компоненты оргтехники, и после накопления на складе транспортных количеств отправить предприятиям и фирмам, специализирующимся по переработке отдельных видов материалов.
- Люминесцентные лампы утилизируют следующим образом. Не работающие лампы немедленно после удаления из светильника должны быть упакованы в картонную коробку, бумагу или тонкий мягкий картон, предохраняющий лампы от взаимного соприкосновения и случайного механического повреждения. После накопления ламп объемом в 1 транспортную единицу их сдают на переработку на соответствующее предприятие. Недопустимо выбрасывать отработанные энергосберегающие лампы вместе с обычным мусором, превращая его в ртутьсодержащие отходы, которые загрязняют ртутными парами

6.4 Безопасность в чрезвычайных ситуациях

Природная чрезвычайная ситуация – обстановка на определенной территории или акватории, сложившейся в результате возникновения источника природной чрезвычайной ситуации, который может повлечь или повлек за собой человеческие жертвы, ущерб здоровью людей и (или) окружающей природной среде, значительные материальные потери и нарушение условий жизнедеятельности людей.

Производство находится в городе Томске с континентально-циклоническим климатом. Природные явления (землетрясения, наводнения, засухи, ураганы и т. д.), в данном городе отсутствуют.

Возможными ЧС на объекте в данном случае, могут быть сильные морозы и диверсия.

Для Сибири в зимнее время года характерны морозы. Достижение критически низких температур приводит к авариям систем тепло- и водоснабжения, сантехнических коммуникаций и электроснабжения, приостановке работы. В этом случае при подготовке к зиме следует предусмотреть, а) газобаллонные калориферы (запасные обогреватели), б) дизель или бензоэлектрогенераторы; в) запасы питьевой и технической воды на складе (не менее 30 л на 1 человека); г) теплый транспорт для доставки работников на работу и с работы домой в случае отказа муниципального транспорта. Их количества и мощности должно хватать для того, чтобы работа на производстве не прекратилась.

В производственном помещении, где выполнялось научно-техническое исследование, наиболее вероятно возникновение чрезвычайных ситуаций (ЧС) техногенного характера.

Для предупреждения вероятности осуществления диверсии предприятие необходимо оборудовать системой видеонаблюдения, круглосуточной охраной, пропускной системой, надежной системой связи, а также исключения распространения информации о системе охраны объекта, расположении помещений и оборудования в помещениях, системах охраны, сигнализаторах, их местах установки и количестве. Должностные лица раз в полгода проводят тренировки по отработке действий на случай экстренной эвакуации.

6.5 Вывод

В рамках раздела «Социальная ответственность» процесс выполнения и результаты дипломной работы были рассмотрены с точки зрения социальной ответственности за моральные, общественные, экономические, экологические последствия и ущерб здоровью человека. Дополнительно был выполнен анализ на предмет выявления основных опасных и вредных факторов и оценена степень их воздействия на человека, общество и природную среду. Были предложены методы для защиты и минимизации воздействий выявленных факторов, а также методы предотвращения и устранения возможных чрезвычайных ситуаций.

Перечень НТД

1. ГОСТ 54 30013-83. Электромагнитные излучения СВЧ. Предельно допустимые уровни облучения. Требования безопасности;
2. ГОСТ 12.4.154-85. «ССБТ. Устройства, экранирующие для защиты от электрических полей промышленной частоты»;
2. ГН 2.2.5.1313-03. Предельно допустимые концентрации (ПДК) вредных веществ в воздухе рабочей зоны;
3. СанПиН 2.2.4/2.1.8.055-96. «Электромагнитные излучения радиочастотного диапазона (ЭМИ РЧ)».;
4. СанПиН 2.2.4.548-96. Гигиенические требования к микроклимату производственных помещений.;
5. СН 2.2.4/2.1.8.562-96. Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки.;
6. ГОСТ 12.4.123-83. Средства коллективной защиты от инфракрасных излучений. Общие технические требования.;
7. ГОСТ Р 12.1.019-2009. Электробезопасность. Общие требования и номенклатура видов защиты.;
8. ГОСТ 12.1.030-81. Электробезопасность. Защитное заземление. Зануление.;
9. ГОСТ 12.1.004-91. Пожарная безопасность. Общие требования.;
10. ГОСТ 12.2.037-78. Техника пожарная. Требования безопасности;
11. СанПиН 2.1.6.1032-01. Гигиенические требования к качеству атмосферного воздуха;
12. ГОСТ 30775-2001. Ресурсосбережение. Обращение с отходами. Классификация, идентификация и кодирование отходов.;
13. СНиП 21-01-97. Противопожарные нормы.;
14. ГОСТ 12.4.154. Система стандартов безопасности труда. Устройства, экранирующие для защиты от электрических полей промышленной частоты. Общие технические требования, основные параметры и размеры.

Приложение I (справочное)

DEVELOPMENT OF A COMPUTER MODEL FOR THE STUDY OF FACTORS AFFECTING THE QUALITY OF CONSTRUCTION MIXTURES.

Студент

Группа	ФИО	Подпись	Дата
8ПМ1И	Геращенков Вадим		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент	Аксенов Сергей Владимирович	к.т.н		

Консультант-лингвист отделения иностранных языков ШБИП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ст.преподаватель	Куркан Наталья Викторовна	к.ф.н		

CHAPTER 7 . DEVELOPMENT OF A COMPUTER MODEL FOR THE STUDY OF FACTORS AFFECTING THE QUALITY OF CONSTRUCTION MIXTURES.

The aim of the work is to develop software and computer models for the analysis of building mixtures.

The object of the study is the data of the construction mixture obtained in the laboratory.

The subject of the research is Data Science tools and machine learning methods in data analysis.

The research methods are the study of literature, articles, scientific papers, analysis of materials, comparison, consultation with specialists and scientists, machine learning methods, visualization methods.

The dataset consists of 1030 instances with 9 attributes and no missing values. There are 8 input variables and 1 output variable. Seven input variables represent the amount of raw material (measured in kg/m³) and one represents age (in days). The target variable is the compressive strength of the concrete, measured in (MPa - MegaPascal). We will examine the data to see how input characteristics affect compressive strength.

7.1 Exploring the dataset

When loading this dataset, we see that several features affect the quality of the concrete. So, we will briefly discuss each characteristic:

Cement (cement): is a finely ground mineral powder, usually gray in color. The most important raw materials for cement production are limestone, clay and marl. Cement mixed with water serves as an adhesive to bind sand, gravel and hard rock into concrete. Cement hardens both in air and under water and remains in a hardened state once reached.

Slag (slag): solid residue after smelting metal from ore, as well as from burning coal.

Fly Ash: Improves concrete workability, pumpability, cohesion, finish, ultimate strength and durability, and solves many of the problems concrete faces today, all at a lower cost.

Water (Water): is an important component in the production of concrete. The moisture that water provides also gives concrete strength during the curing process. Although water is one of the most

important components of concrete, it can also be the most destructive in excessive amounts.
Superplasticizer: Traditionally known for their water reducing properties, they improve the flow of concrete without adding additional water and reducing the overall strength of the mixture. These same qualities also make it possible to reduce the amount of cement materials.
When mixed with water, cement particles naturally attract each other and tend to form lumps. This means that only a portion of the cement particles can properly complete the hydration process, which reduces the strength of the finished product. High strength mixes require more cement to increase the percentage of cement that binds and hydrates with water molecules. When a superplasticizer is present in the mix, it binds to the cement particles and neutralizes the force that pulls them together. This keeps them from clustering in the mixture and releases more cement molecules to complete the hydration process. So you get the same result with less cement than a traditional high strength mix.

Coarse aggregate (Coaseseaggregate): Coarse aggregate for concrete is gravel with rounded grains having a smooth surface, as well as crushed stone with angular grains having a rough surface. Crushed stone and gravel are obtained by crushing large rocks.
Fine aggregate (fineaggregate): sand, the grain size of which is not more than 5mm.

Age (age): the design age of concrete, i.e. the age at which concrete must acquire all the quality indicators normalized for it, is assigned during design, based on the possible real terms of loading structures with design loads, taking into account the method of construction of structures and concrete hardening conditions.

The tensile strength of concrete (Strength) is an important parameter that determines the maximum load that a concrete structure can withstand without failure.

Now we import some important modules (Figure 3.1):

	cement	blastFurnace	flyAsh	water	superplasticizer	courseAggregate	fineaggregate	age	strength
0	540.0	0.0	0.0	162.0	2.5	1040.0	676.0	28	79.99
1	540.0	0.0	0.0	162.0	2.5	1055.0	676.0	28	61.89
2	332.5	142.5	0.0	228.0	0.0	932.0	594.0	270	40.27
3	332.5	142.5	0.0	228.0	0.0	932.0	594.0	365	41.05
4	198.6	132.4	0.0	192.0	0.0	978.4	825.5	360	44.30
...	...	...	...	...	...	...	...	...	...
1025	276.4	116.0	90.3	179.6	8.9	870.1	768.3	28	44.28
1026	322.2	0.0	115.6	196.0	10.4	817.9	813.4	28	31.18
1027	148.5	139.4	108.6	192.7	6.1	892.4	780.0	28	23.70
1028	159.1	186.7	0.0	175.6	11.3	989.6	788.9	28	32.77
1029	260.9	100.5	78.3	200.6	8.6	864.5	761.5	28	32.40

1030 rows × 9 columns

Figure. 3.1 Data set

7.2 Study dataset

After reading the data set, we must extract information from the data, for this we use a specific function (Fig.3.2):

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1030 entries, 0 to 1029
Data columns (total 9 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Cement              1030 non-null   float64
 1   Blast Furnace Slag  1030 non-null   float64
 2   Fly Ash             1030 non-null   float64
 3   Water               1030 non-null   float64
 4   Superplasticizer    1030 non-null   float64
 5   Coarse Aggregate    1030 non-null   float64
 6   Fine Aggregate      1030 non-null   float64
 7   Age                 1030 non-null   int64
 8   Strength            1030 non-null   float64
dtypes: float64(8), int64(1)
memory usage: 72.5 KB
```

Figure. 3.2 Extracting information from a dataset

The df.info() method gives a description of the date frame: how many rows and columns it contains, what data types it contains, how many non-empty values (non-null), how much memory it takes.

The describe method gives statistics on numerical columns: mean, maximum and minimum, quartiles, standard deviation (Fig.3.3).

```
df.describe()
```

	Cement	Blast Furnace Slag	Fly Ash	Water	Superplasticizer	Coarse Aggregate	Fine Aggregate	Age	Strength
count	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000
mean	281.167864	73.895825	54.188350	181.567282	6.204660	972.918932	773.580485	45.662136	35.817961
std	104.506364	86.279342	63.997004	21.354219	5.973841	77.753954	80.175980	63.169912	16.705742
min	102.000000	0.000000	0.000000	121.800000	0.000000	801.000000	594.000000	1.000000	2.330000
25%	192.375000	0.000000	0.000000	164.900000	0.000000	932.000000	730.950000	7.000000	23.710000
50%	272.900000	22.000000	0.000000	185.000000	6.400000	968.000000	779.500000	28.000000	34.445000
75%	350.000000	142.950000	118.300000	192.000000	10.200000	1029.400000	824.000000	56.000000	46.135000
max	540.000000	359.400000	200.100000	247.000000	32.200000	1145.000000	992.600000	365.000000	82.600000

Fig. 3.3 Numeric column statistics

Now we process null values present in the data set for greater accuracy (Figure 3.4).

```
df.isnull().sum()
```

```
Cement                0
Blast Furnace Slag    0
Fly Ash               0
Water                 0
Superplasticizer      0
Coarse Aggregate      0
Fine Aggregate        0
Age                   0
Strength              0
dtype: int64
```

Fig. 3.4 Handling null dataset values

7.3 Exploratory data analysis

The first step in a Data Science project is to understand the data and gain insights from the data before proceeding with modeling. This includes checking for any missing values, plotting against the target variable, observing the distribution of all features, and so on. Let's import the data and start the analysis.

Next, we will check the correlations between the input features, this will give an idea of how each variable affects all other variables. One way to quantify the relationship between two variables is to use the Pearson correlation coefficient (Figure 3.5), which is a measure of the linear relationship between two variables.

It takes a value from -1 to 1, where:

• -1 indicates a completely negative linear correlation.

• 0 indicates no linear correlation.

• 1 indicates a perfectly positive linear correlation.

7.3 Exploratory data analysis

The first step in a Data Science project is to understand the data and gain insights from the data before proceeding with modeling. This includes checking for any missing values, plotting against the target variable, observing the distribution of all features, and so on. Let's import the data and start the analysis.

Next, we will check the correlations between the input features, this will give an idea of how each variable affects all other variables. One way to quantify the relationship between two variables is to use the Pearson correlation coefficient (Figure 3.5), which is a measure of the linear relationship between two variables.

It takes a value from -1 to 1, where:

• -1 indicates a completely negative linear correlation.

• 0 indicates no linear correlation.

• 1 indicates a perfectly positive linear correlation.

The farther the correlation coefficient is from 0, the stronger the relationship between the 2 variables.

But there are also times when we want to understand the correlation between more than one

pair of variables. In these cases, we can create a correlation matrix (Figure 3.6), which is a square table showing the correlation coefficients between several pairwise combinations of variables.

	cement	blastFurnace	flyAsh	water	superplasticizer	courseAggregate	fineaggregate	age	strength
cement	1.00	-0.28	-0.40	-0.08	0.09	-0.11	-0.22	0.08	0.50
blastFurnace	-0.28	1.00	-0.32	0.11	0.04	-0.28	-0.28	-0.04	0.13
flyAsh	-0.40	-0.32	1.00	-0.26	0.38	-0.01	0.08	-0.15	-0.11
water	-0.08	0.11	-0.26	1.00	-0.66	-0.18	-0.45	0.28	-0.29
superplasticizer	0.09	0.04	0.38	-0.66	1.00	-0.27	0.22	-0.19	0.37
courseAggregate	-0.11	-0.28	-0.01	-0.18	-0.27	1.00	-0.18	-0.00	-0.16
fineaggregate	-0.22	-0.28	0.08	-0.45	0.22	-0.18	1.00	-0.16	-0.17
age	0.08	-0.04	-0.15	0.28	-0.19	-0.00	-0.16	1.00	0.33
strength	0.50	0.13	-0.11	-0.29	0.37	-0.16	-0.17	0.33	1.00

Fig. 3.6 Correlation matrix

We visualize the correlation matrix (Figure 3.7) using the style options available in pandas:

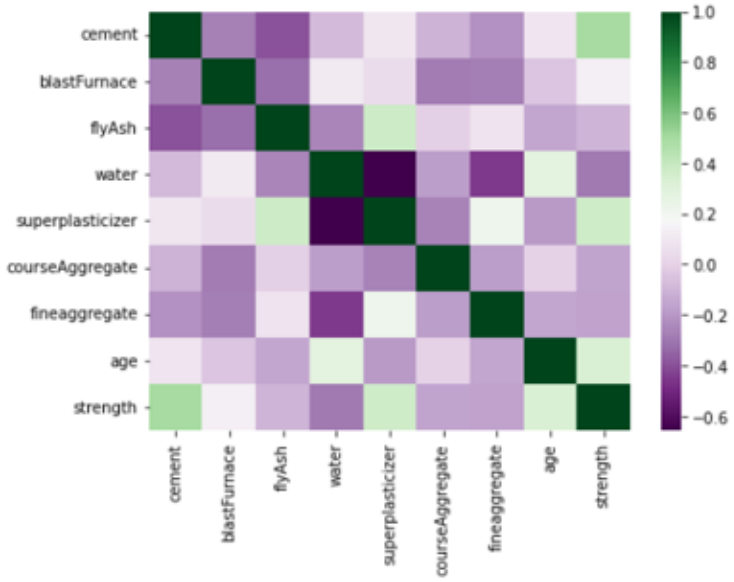

Fig. 3.7 Visualization of the correlation matrix

We can observe a high positive correlation between compressive strength (CC_Strength) and cement. This is true because the strength of concrete does increase with the amount of cement used in its preparation. In addition, age and superplasticizer are two other factors that affect compressive strength.

There are other strong correlations between traits:

• Strong negative correlation between superplasticizer and water.

• positive correlation between superplasticizer and fly ash, fine aggregate.

These correlations are useful for understanding the data in detail because they provide insight into how one variable affects another. Next, we can use the paired plot in seaborn to plot the pairwise relationships between all the features and the diagonal distributions of the features.

We can plot scatter plots between Strength and other features to see more complex relationships.

Scatterplot is one of the tools of statistical control, analysis. With its help, the dependence and nature of the relationship between two different parameters of an economic phenomenon, a production process, is revealed. A scatterplot shows the type and tightness of the relationship between data pairs.

1. product quality and influencing factor;
2. two different quality characteristics;
3. two circumstances affecting the quality, etc.

Scatterplots are used to find correlations between data. If the correlation dependence is present, then it is much easier to establish control over the observed phenomenon.

The scatterplot (Fig.3.8) represents the observed phenomenon in the space of two dimensions. If one value is considered as a “cause” affecting another value, then the X-axis (horizontal axis) will correspond to it. The value reacting to this influence corresponds to the Y-axis (vertical axis). When it is not possible to clearly classify the variables, the allocation is done by the user.

Tensile strength against (cement, age, water)

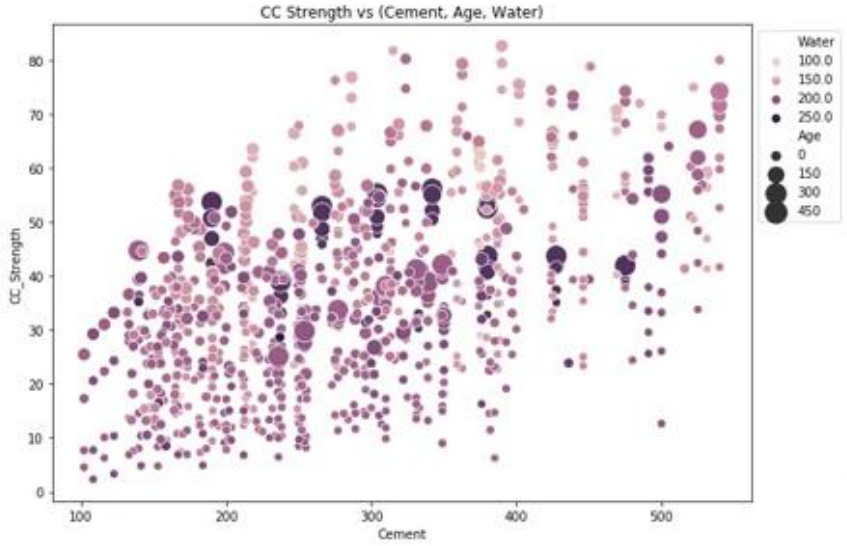

Fig. 3.8 Scatterplot of tensile strength versus cement, age, water

Observations we can make from this graph:

• The compressive strength increases as the amount of cement increases as the points move up as we move to the right on the x-axis.

• Compressive strength increases with age (because dot size represents age), this is not always the case, but may be to some extent.

• Less aged cement requires more cement for higher strength as the smaller dots move up as we move to the right on the x-axis.

• The older the cement, the more water it needs, which can be confirmed by observing the color of the dots. Larger dots of dark color indicate older age and more water.

• The strength of concrete increases when less water is used in its preparation because the dots on the underside (Y-axis) are darker and the dots on the top end (Y-axis) are brighter.

Observations we can make from this graph:

• The compressive strength increases as the amount of cement increases as the points move up as we move to the right on the x-axis.

• Compressive strength increases with age (because dot size represents age), this is not always the case, but may be to some extent.

• Less aged cement requires more cement for higher strength as the smaller dots move up as we move to the right on the x-axis.

• The older the cement, the more water it needs, which can be confirmed by observing the color of the dots. Larger dots of dark color indicate older age and more water.

• The strength of concrete increases when less water is used in its preparation because the dots on the underside (Y-axis) are darker and the dots on the top end (Y-axis) are brighter.

Tensile strength of concrete compared to (fine aggregate, super plasticizer, fly ash)

(Figure 3.9)

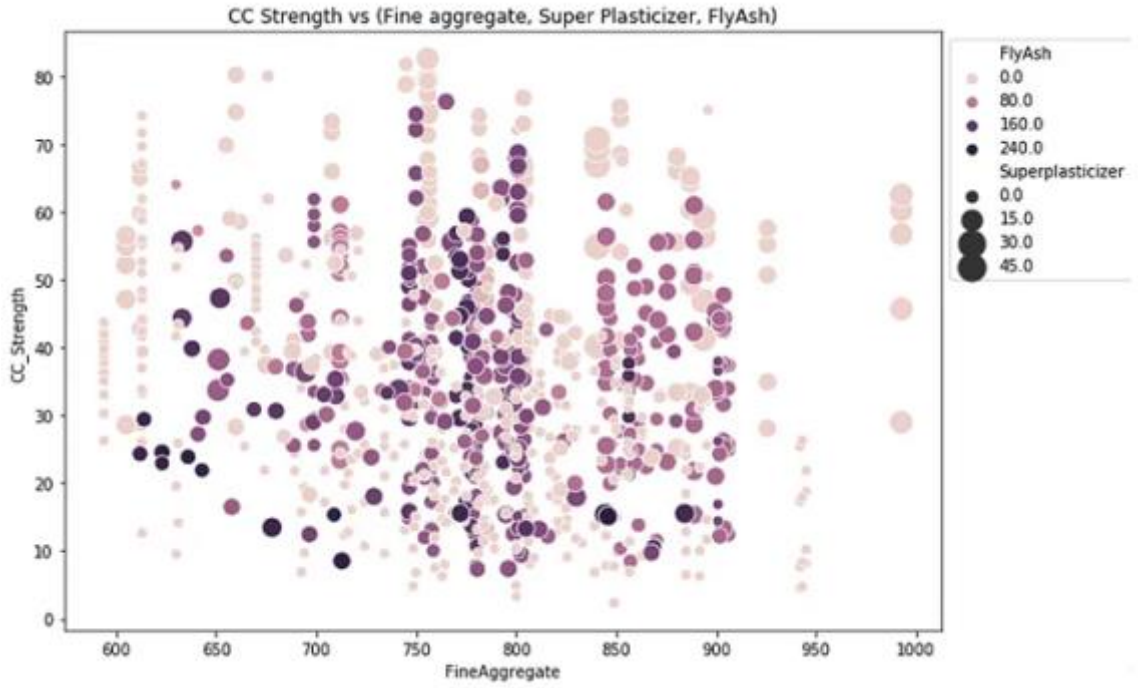

Fig. 3.9 Tensile strength of concrete compared to fine aggregate, super plasticizer, fly ash.

Based on this graph, we see that:

• The compressive strength decreases if the amount of fly ash is increased because the darker dots are concentrated in the area representing low compressive strength.

• The compressive strength increases with the addition of a super plasticizer to the mixture, since the larger the dot, the higher they are on the graph.

We can visually understand 2D, 3D and up to 4D plots (features represented by color and

size) as shown above, we can additionally use Seaborn row and column plotting functions for further analysis, but still we are missing the ability to track all these correlations independently. For this reason, we can turn to machine learning to capture these relationships and better understand the problem.

7.4 Separation of dependent and independent variables

Before we start building the model, we have to split the dataset into two parts,

1. Independent variables contain a list of those variables on which a particular quality depends.
2. A dependent variable is one that depends on the values of other variables.

We do not use complete data to create the model. Some data is randomly selected and stored to check the quality of the model. This is known as the test data and the rest of the data is called the training data on which the model is built. Typically 70% of the data is used as training data and the remaining 30% is used as test data.

7.5 Model building

After preparing the data, we can fit different models to the training data and compare their performance to select an algorithm with good performance. Since this is a regression problem, we can use the RMSE (Root Mean Squared Error) and the R^2 score as scoring measures.

7.6 Linear regression

We'll start with linear regression since it's an algorithm for jumping to any regression problem. The algorithm tries to form a linear relationship between the input functions and the target variable, i.e. it corresponds to the straight line given by the formula:

$$y = W * X + b = \sum_{i=1}^{n} w_i * x_i + b$$

Formula 3.1 - Linear Regression

Where w_i corresponds to the feature coefficient x_i.

The magnitude of these coefficients can be further controlled using the regularization conditions for the cost functions. Adding the sum of the coefficient values will cause the coefficients to be close to zero, this variant of linear regression is called Lasso regression. Adding the sum of the squares of the coefficients to the cost function will result in the coefficients being in the same range, and this variation is called ridge regression. Both of these options help reduce the complexity of the model and therefore reduce the chance of overfitting the data.

As a result of data processing, we obtain the following values (Fig. 3.10):

Model	RMSE	R2
LinearRegression	10.29	0.57
LassoRegression	10.68	0.54
RidgeRegression	10.29	0.57

Fig. 3.10 Results of processing linear regression data

There is not much difference between the performance of these three algorithms, we can plot the coefficients assigned by the three algorithms to the functions (Figure 3.11).

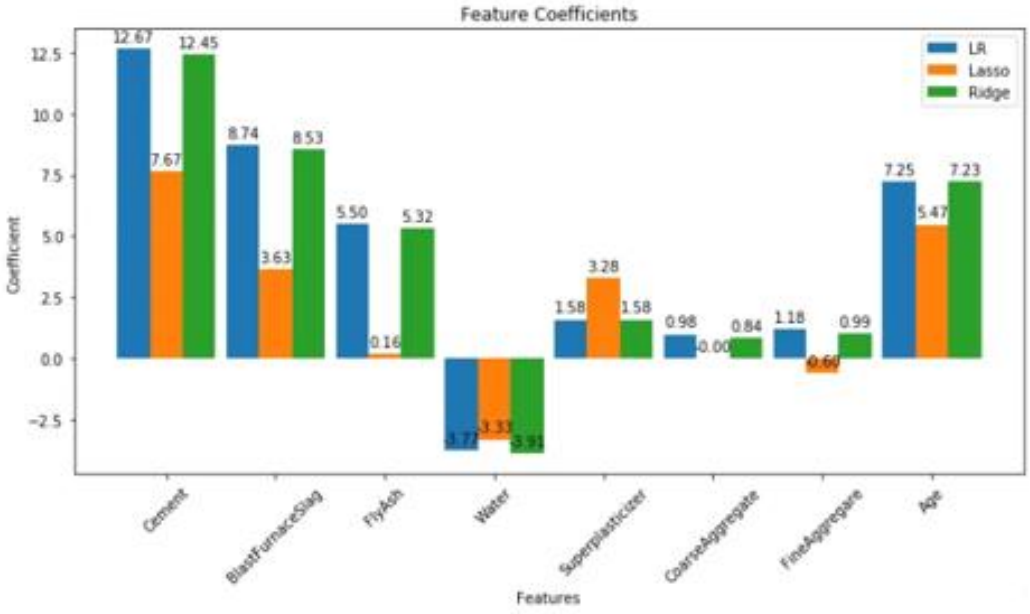

Fig. 3.11 Feature Importance of Linear Regression Algorithms

As you can see in the figure, Lasso regression pushes the coefficients towards zero, and the coefficients of regular linear regression and ridge regression are almost the same.

We can further see what the predictions are by plotting the true values and the predicted values (Figure 3.12).

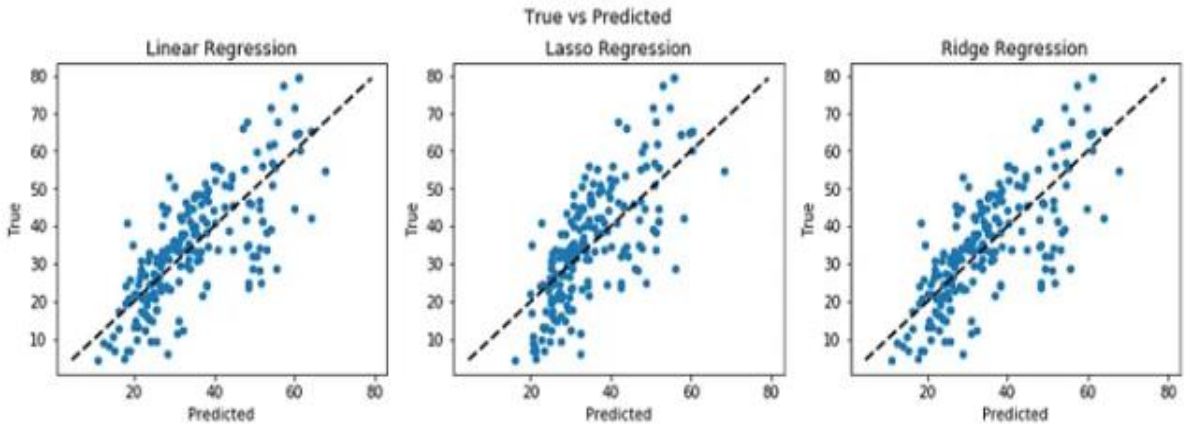

Fig. 3.12 Scatter plots to test linear regression prediction values

If the predicted values and the target values are equal, then the points on the scatter plot will lie on a straight line. As we can see here, none of the models predict compressive strength correctly.

7.5 Decision trees

The decision tree algorithm presents data in a tree structure where each node represents a decision made about a feature. In this case, this algorithm will give better performance because we have a lot of zeros in some of the input functions, as seen from their distribution in the pair plot above (Figure 3.13). This will help decision trees build trees based on some feature conditions, which can further improve performance.

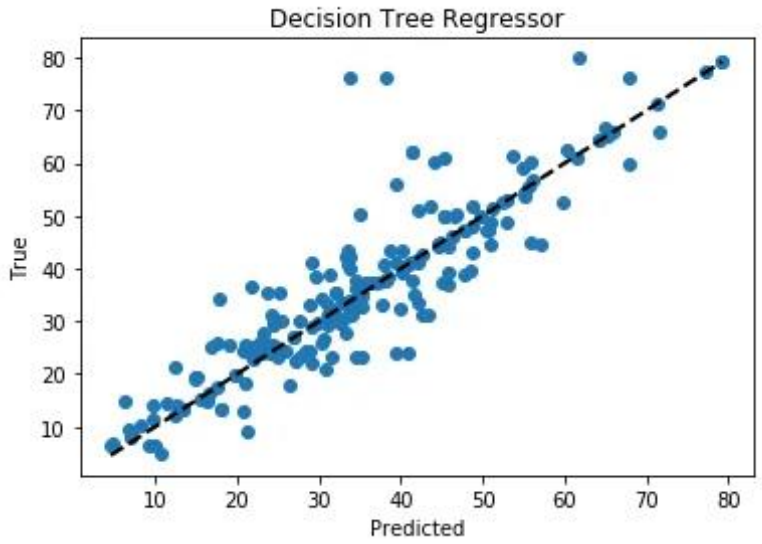

Fig. 3.13 Scatterplot to test decision tree prediction values

The root mean squared error (RMSE) has been reduced from 10.29 to 7.31, so the decision tree regressor has improved performance significantly. This can be observed on the chart, as more points are closer to the line (Fig.3.14).

Model	RMSE	R2
Decision Tree Regressor	7.31	0.78

Fig. 3.14 Decision Tree Data Processing Results

7.6 Random forests

Using the decision tree regressor has improved our performance, we can further improve performance by merging more trees. Random Forest Regressor trains randomly initialized trees with random subsets of data selected from the training data, this will make our model more robust (Figure 3.15).

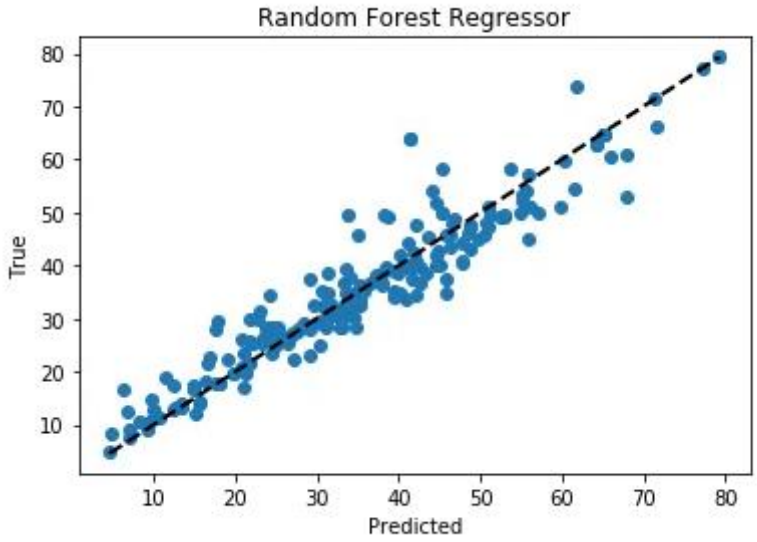

Fig. 3.15 Scatterplot to test random forest prediction values

RMSE has been further reduced by merging several trees (Figure 3.16). We can plot feature importance for tree models. Feature importance indicates how important a feature is to the model when making predictions.

Cement and age are considered to be the most important characteristics of tree models. Fly ash, coarse and fine aggregates are the least important factors in predicting the strength of concrete (Figure 3.17).

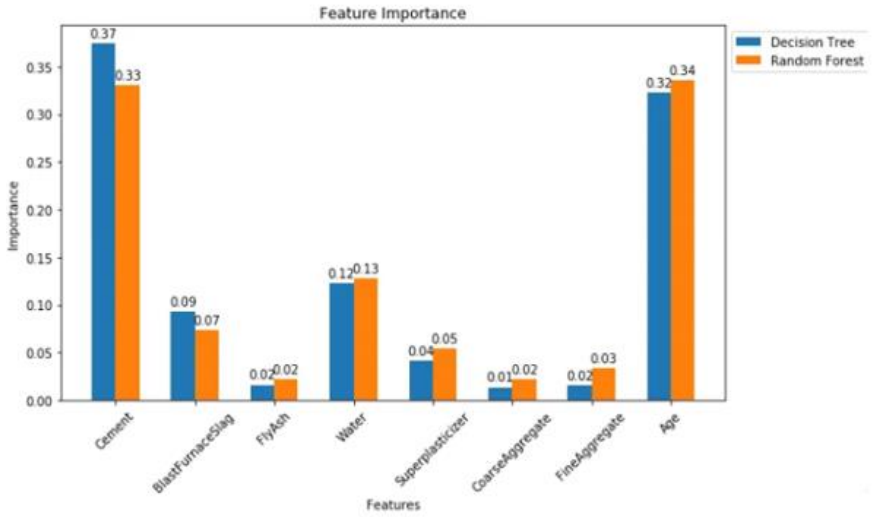

Fig. 3.17 Feature Importance of Decision Tree and Random Forest Algorithms

7.7 Comparison:

Finally, let's compare the results of all algorithms (Figure 3.18).

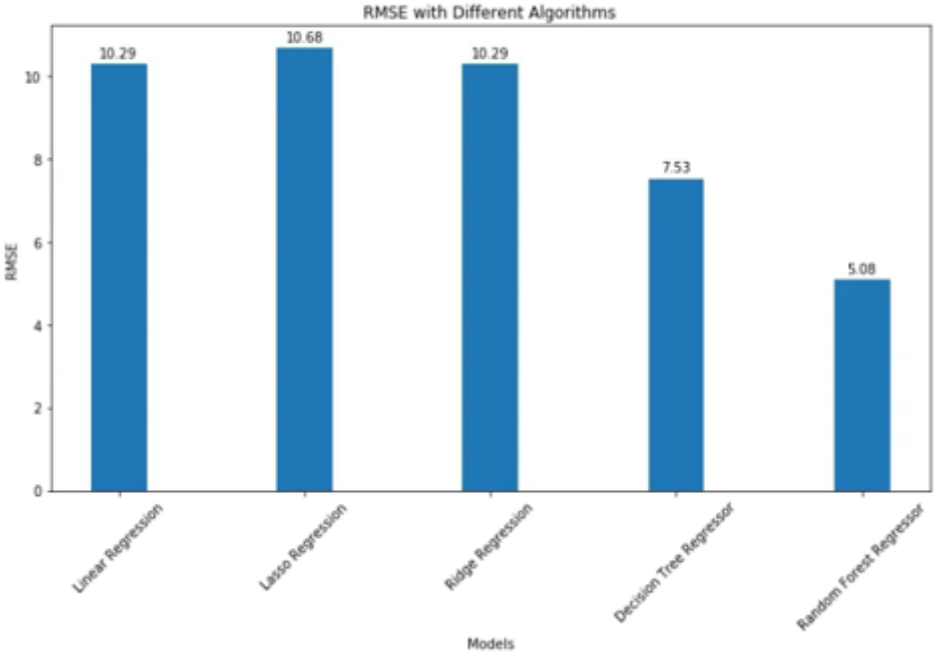

Fig. 3.18 Comparison of the results of all algorithms (RMSE - standard deviation)

7.8 Conclusion

We analyzed the compressive strength data and used machine learning to predict the compressive strength of concrete. We used linear regression and its variants, decision trees and random forests to make predictions and compare their performance. Random Forest Regressor has the lowest RMSE and is a good choice for this problem

Приложение 2 (скрипт на Python)

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

%matplotlib inline
```

Loading the Data

:

```
data = pd.read_excel("data/Concrete_Data.xls")

len(data)
1030
```

:

```
data.head()
```

	Cement (component 1)(kg in a m^3 mixture)	Blast Furnace Slag (component 2)(kg in a m^3 mixture)	Fly Ash (component 3)(kg in a m^3 mixture)	Water (component 4)(kg in a m^3 mixture)	Superplasticizer (component 5)(kg in a m^3 mixture)	Coarse Aggregate (component 6)(kg in a m^3 mixture)	Fine Aggregate (component 7)(kg in a m^3 mixture)	Age (day)	Concrete compressive strength(MPa, megapascals)
0	540.0	0.0	0.0	162.0	2.5	1040.0	676.0	28	79.986111
1	540.0	0.0	0.0	162.0	2.5	1055.0	676.0	28	61.887366
2	332.5	142.5	0.0	228.0	0.0	932.0	594.0	270	40.269535
3	332.5	142.5	0.0	228.0	0.0	932.0	594.0	365	41.052780
4	198.6	132.4	0.0	192.0	0.0	978.4	825.5	360	44.296075

Simplifying Column names, since they appear to be too lengthy.

In [5]:

```
req_col_names = ["Cement", "BlastFurnaceSlag", "FlyAsh", "Water",
"Superplasticizer",
                 "CoarseAggregate", "FineAggregate", "Age", "CC_Strength"]
curr_col_names = list(data.columns)

mapper = {}
for i, name in enumerate(curr_col_names):
    mapper[name] = req_col_names[i]

data = data.rename(columns=mapper)

data.head()
```

	Cement	BlastFurnaceSlag	FlyAsh	Water	Superplasticizer	CoarseAggregate	FineAggregate	Age	CC_Strength
0	540.0	0.0	0.0	162.0	2.5	1040.0	676.0	28	79.986111

	Cement	BlastFurnaceSlag	FlyAsh	Water	Superplasticizer	CoarseAggregate	FineAggregate	Age	CC_Strength
1	540.0	0.0	0.0	162.0	2.5	1055.0	676.0	28	61.887366
2	332.5	142.5	0.0	228.0	0.0	932.0	594.0	270	40.269535
3	332.5	142.5	0.0	228.0	0.0	932.0	594.0	365	41.052780
4	198.6	132.4	0.0	192.0	0.0	978.4	825.5	360	44.296075

Checking for 'null' values

```
data.isna().sum()
```

```
Cement              0
BlastFurnaceSlag    0
FlyAsh              0
Water               0
Superplasticizer    0
CoarseAggregate     0
FineAggregate       0
Age                 0
CC_Strength         0
dtype: int64
```

There are no null values in the data.

EDA

Exploring the data.

:

```
data.describe()
```

	Cement	BlastFurnaceSlag	FlyAsh	Water	Superplasticizer	CoarseAggregate	FineAggregate	Age	CC_Strength
count	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000	1030.000000
mean	281.165631	73.895485	54.187136	181.566359	6.203112	972.918592	773.578883	45.662136	35.817836
std	104.50	86.279104	63.996	21.355	5.973492	77.753818	80.17542	63.169	16.7056

	Cement	BlastFurnaceSlag	FlyAsh	Water	Superplasticizer	CoarseAggregate	FineAggregate	Age	CC_Strength
	7142		469	567			7	912	79
min	102.000000	0.000000	0.000000	121.750000	0.000000	801.000000	594.000000	1.000000	2.331808
25%	192.375000	0.000000	0.000000	164.900000	0.000000	932.000000	730.950000	7.000000	23.707115
50%	272.900000	22.000000	0.000000	185.000000	6.350000	968.000000	779.510000	28.000000	34.442774
75%	350.000000	142.950000	118.270000	192.000000	10.160000	1029.400000	824.000000	56.000000	46.136287
max	540.000000	359.400000	200.100000	247.000000	32.200000	1145.000000	992.600000	365.000000	82.599225

sns.scatterplot(y="CC_Strength", x="Cement", hue="Water",size="Age", data=data, ax=ax, sizes=(50, 300))

```
sns.scatterplot(y="CC_Strength", x="FineAggregate", hue="FlyAsh", size="Superplasticizer", data=data, ax=ax, sizes=(50, 300))
```

```
X = data.iloc[:,:-1] # Features
y = data.iloc[:,-1] # Target X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=2) sc = StandardScaler() X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

Приложение 3 (скрипт на Python)

```
coeff_lr = lr.coef_
coeff_lasso = lasso.coef_
coeff_ridge = ridge.coef_ labels = req_col_names[:-1]
x = np.arange(len(labels))
width = 0.3 fig, ax = plt.subplots(figsize=(10,6))
rects1 = ax.bar(x - 2*(width/2), coeff_lr, width, label='LR')
rects2 = ax.bar(x, coeff_lasso, width, label='Lasso')
rects3 = ax.bar(x + 2*(width/2), coeff_ridge, width, label='Ridge')
ax.set_ylabel('Coefficient')
ax.set_xlabel('Features')
ax.set_title('Feature Coefficients')
ax.set_xticks(x)
ax.set_xticklabels(labels, rotation=45)
ax.legend() def autolabel(rects):
   """Attach a text label above each bar in *rects*, displaying its
height."""
   for rect in rects:
      height = rect.get_height()
      ax.annotate('{:.2f}'.format(height), xy=(rect.get_x() +
rect.get_width() / 2, height), xytext=(0, 3), textcoords="offset
points", ha='center', va='bottom') autolabel(rects1)
autolabel(rects2)
autolabel(rects3)
fig.tight_layout()
plt.show()


fig, (ax1, ax2, ax3) = plt.subplots(1,3, figsize=(12,4))
ax1.scatter(y_pred_lr, y_test, s=20)
ax1.plot([y_test.min(), y_test.max()], [y_test.min(),
y_test.max()], 'k--', lw=2)
ax1.set_ylabel("True")
ax1.set_xlabel("Predicted")
ax1.set_title("Linear Regression")
ax2.scatter(y_pred_lasso, y_test, s=20) ax2.plot([y_test.min(),
y_test.max()], [y_test.min(), y_test.max()], 'k--', lw=2)
ax2.set_ylabel("True")
ax2.set_xlabel("Predicted")
ax2.set_title("Lasso Regression")
ax3.scatter(y_pred_ridge, y_test, s=20) ax3.plot([y_test.min(),
y_test.max()], [y_test.min(), y_test.max()], 'k--', lw=2)
ax3.set_ylabel("True")
ax3.set_xlabel("Predicted")
ax3.set_title("Ridge Regression")
```

```
fig.suptitle("True vs Predicted")
fig.tight_layout(rect=[0, 0.03, 1, 0.95])



from sklearn.tree import DecisionTreeRegressor
dtr = DecisionTreeRegressor()
dtr.fit(X_train, y_train)
y_pred_dtr = dtr.predict(X_test) print("Model\t\t\t\t RMSE \t\t
R2")
print("""Decision Tree Regressor \t {:.2f} \t\t{:.2f}""".format(
np.sqrt(mean_squared_error(y_test, y_pred_dtr)), r2_score(y_test,
y_pred_dtr))) plt.scatter(y_test, y_pred_dtr)
plt.plot([y_test.min(), y_test.max()], [y_test.min(),
y_test.max()], 'k--', lw=2)
plt.xlabel("Predicted")
plt.ylabel("True")
plt.title("Decision Tree Regressor") plt.show()



from sklearn.ensemble import RandomForestRegressor rfr =
RandomForestRegressor(n_estimators=100)
rfr.fit(X_train, y_train) y_pred_rfr = rfr.predict(X_test)
print("Model\t\t\t\t RMSE \t\t R2") print("""Random Forest
Regressor \t {:.2f} \t\t{:.2f}""".format(
np.sqrt(mean_squared_error(y_test, y_pred_rfr)), r2_score(y_test,
y_pred_rfr))) plt.scatter(y_test, y_pred_rfr)
plt.plot([y_test.min(), y_test.max()], [y_test.min(),
y_test.max()], 'k--', lw=2)
plt.xlabel("Predicted")
plt.ylabel("True")
plt.title("Random Forest Regressor")
plt.show()



feature_dtr = dtr.feature_importances_
feature_rfr = rfr.feature_importances_ labels = req_col_names[:-1]
x = np.arange(len(labels))
width = 0.3
fig, ax = plt.subplots(figsize=(10,6))
rects1 = ax.bar(x-(width/2), feature_dtr, width, label='Decision
Tree')
rects2 = ax.bar(x+(width/2), feature_rfr, width, label='Random
Forest')
ax.set_ylabel('Importance')
ax.set_xlabel('Features')
```

```
ax.set_title('Feature Importance')
ax.set_xticks(x)
ax.set_xticklabels(labels, rotation=45)
ax.legend(loc="upper left", bbox_to_anchor=(1,1))
autolabel(rects1)
autolabel(rects2)
fig.tight_layout()
plt.show()



models = [lr, lasso, ridge, dtr, rfr]
names = ["Linear Regression", "Lasso Regression", "Ridge
Regression", "Decision Tree Regressor", "Random Forest Regressor"]
rmses = [] for model in models:
   rmses.append(np.sqrt(mean_squared_error(y_test,
model.predict(X_test)))) x = np.arange(len(names))
width = 0.3
fig, ax = plt.subplots(figsize=(10,7))
rects = ax.bar(x, rmses, width)
ax.set_ylabel('RMSE')
ax.set_xlabel('Models')
ax.set_title('RMSE with Different Algorithms')
ax.set_xticks(x)
ax.set_xticklabels(names, rotation=45)
autolabel(rects)
fig.tight_layout()
plt.show()
```

Список использованных источников и литературы

1. M. Pala *et al.*
Appraisal of long-term effects of fly ash and silica fume on compressive strength of concrete by neural networks.
Constr. Build. Mater.
(2007)

2.U. Atici
Prediction of the strength of mineral admixture concrete using multivariable regression analysis and an artificial neural networkə
Expert Syst. Appl.
(2011)

3.V. Nilsen *et al.*
Prediction of concrete coefficient of thermal expansion and other properties using machine learning.
Constr. Build. Mater.
(2019)

4.S. Chithra *et al.*
A comparative study on the compressive strength prediction models for High Performance Concrete containing nano silica and copper slag using regression analysis and Artificial Neural Networks.
Constr. Build. Mater.
(2016)

5.H. Naderpour *et al.*
Compressive strength prediction of environmentally friendly concrete using artificial neural networks.
Journal of Building Engineering.

(2018)

6.H. Ling *et al.*

Combination of Support Vector Machine and K-Fold cross validation to predict compressive strength of concrete in marine environment.

Constr. Build. Mater.

(2019)

7.Z.H. Duan *et al.*

Prediction of compressive strength of recycled aggregate concrete using artificial neural networks

Constr. Build. Mater.

(2013)

8.J. Sobhani *et al.*

Prediction of the compressive strength of no-slump concrete: A comparative study of regression, neural network and ANFIS models.

Constr. Build. Mater.

(2010)

9.A.K. Al-Shamiri *et al.*

Modeling the compressive strength of high-strength concrete: An extreme learning approach

Constr. Build. Mater.

(2019)

10.H.I. Erdal

Two-level and hybrid ensembles of decision trees for high performance concrete compressive strength prediction.

Eng. Appl. Artif. Intel.

(2013)

11.Q. Han *et al.*

A generalized method to predict the compressive strength of high-performance concrete by improved random forest algorithm.

Constr. Build. Mater.

(2019)

12.B. Ahmadi-Nedushan
An optimized instance based learning algorithm for estimation of compressive strength of concrete.
Eng. Appl. Artif. Intel.
(2012)
13.U. Anyaoha *et al.*
Soft computing in estimating the compressive strength for high-performance concrete via concrete composition appraisal.
Constr. Build. Mater.
(2020)
14.Z.M. Yaseen *et al.*
Predicting compressive strength of lightweight foamed concrete using extreme learning machine model.
Adv. Eng. Softw.
(2018)
15.A.M. Diab *et al.*
Prediction of concrete compressive strength due to long term sulfate attack using neural network.
16.Alex. Eng. J.
(2014)
17.A. Öztaş *et al.*
Predicting the compressive strength and slump of high strength concrete using neural network.
Constr. Build. Mater.
(2006)
18. Обучение с учителем // Машинное обучение [Электронный ресурс]. – URL: http://www.machinelearning.ru/wiki/index.php?title=%D0%9E%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5_%D1%81_%D1%83%D1%87%D0%B8%D1%82%D0%B5%D0%BB%D0%B5%D0%BC (дата обращения: 11.05.2023).
19. Обучение нейросети с учителем, без учителя, с подкреплением — в чем отличие? Какой алгоритм лучше? // Neurohive [Электронный ресурс]. – URL: https://neurohive.io/ru/osnovy-data-science/obuchenie-s-uchitelem-bezuchitelja-s-podkrepleniem/ (дата обращения: 13.04.2023).

20. Машинное обучение применили для помощи анастезиологам // Neurohive [Электронный ресурс]. – URL: https://neurohive.io/ru/gotovyeprilozhenija/mashinnoe-obuchenie-primenili-dlya-pomoshhi-anasteziologam/ (дата обращения: 14.02.2023).

21. Модель обучили находить оптимальную схему лечения // Neurohive [Электронный ресурс]. – URL: https://neurohive.io/ru/papers/modelobuchili-nahodit-optimalnuju-shemu-lecheniya/ (дата обращения: 18.04.2023).

22. Машинное обучение: методы и способы // OSP – Гид по технологиям цифровой трансформации [Электронный ресурс]. – URL: https://www.osp.ru/cio/2018/05/13054535 (дата обращения: 12.04.2023).

23. Дерево решений // Loginom [Электронный ресурс]. – URL: https://wiki.loginom.ru/articles/decision-trees.html (дата обращения: 1.03.2023).

24. Логистическая регрессия (Logistic Regression) // Loginom [Электронный ресурс]. – URL: https://wiki.loginom.ru/articles/logisticregression.html (дата обращения: 3.03.2022). 108

25. Как работает случайный лес? // Nuancesprog [Электронный ресурс]. – URL: https://nuancesprog.ru/p/6160/ (дата обращения: 8.03.2023).

26. Оценка качества в задачах классификации // Университет ИТМО [Электронный ресурс]. – URL: https://neerc.ifmo.ru/wiki/index.php?title=%D0%9E%D1%86%D0%B5%D0%BD%D0%BA%D0%B0_%D0%BA%D0%B0%D1%87%D0%B5%D1%81%D1%82%D0%B2%D0%B0_%D0%B2_%D0%B7%D0%B0%D0%B4%D0%B0%D1%87%D0%B0%D1%85_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D0%B8 (дата обращения: 8.04.2022).

27. Интерпретируй это: метод SHAP в Data Science // Чернобровов Алексей [Электронный ресурс]. – URL: https://chernobrovov.ru/articles/interpretiruj-eto-metod-shap-v-data-science.html (дата обращения: 16.04.2023).

28. Основные инструменты анализа данных. Откройте для себя список из 14 лучших программ и инструментов анализа // Xmldatafeed [Электронный ресурс]. – URL: https://xmldatafeed.com/osnovnye-instrumentyanaliza-dannyh-otkrojte-dlya-sebya-spisok-iz-14-luchshih-programm-iinstrumentov-analiza/ (дата обращения: 25.04.2023).

29. Choosing a Better Framework // Tutorials point [Электронный ресурс]. – URL: https://www.tutorialspoint.com/python_web_development_

libraries/python_web_development_libraries_choosing_a_better_framework.htm (дата обращения: 22.04.2023).

30. Диаграмма Исикавы [Электронный ресурс]: сайт. – URL: https://up-pro.ru/encyclopedia/diagramma-isikavy/ (дата обращения: 22.05.2023).

31. История диаграммы Ганта [Электронный ресурс] / Юлия Челянова и Евгений Пикулев. — Электрон. текстовые дан. — Режим доступа: http://gibtech.ru/blog/discus?entry_id=177 (дата обращения: 22.05.2022).

32. «Трудовой кодекс Российской Федерации» от 30.12.2001 N 197- ФЗ (ред. от 25.02.2022) (с изм. и доп., вступ. в силу с 01.03.2022).

33. Федеральный закон «О специальной оценке условий труда» от 28.12.2013 N 426-ФЗ. 36. ГОСТ 12.2.032-78 «Рабочее место при выполнении работ сидя».

34. СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания.

35. ГОСТ Р 50923-96 Дисплеи. Рабочее место оператора.

36. СП 52.13330.2016 Естественное и искусственное освещение.

37. Безопасность жизнедеятельности. Расчёт искусственного освещения. Методические указания к выполнению индивидуальных заданий для студентов дневного и заочного обучения всех направлений и специальностей ТПУ. – Томск: Изд. ТПУ, 2008. – 20 с.

38. ГОСТ Р 12.2.143-2009 Система стандартов безопасности труда. Системы фотолюминесцентные эвакуационные. Требования и методы контроля.

39. МР 2.2.9.2311 – 07 «Профилактика стрессового состояния работников при различных видах профессиональной деятельности».

40. ГОСТ 12.1.030-81 ССБТ Защитное заземление, зануление.

41. ГОСТ Р 12.1.019-2009 ССБТ. Электробезопасность. Общие требования и номенклатура видов защиты.

42. ГОСТ Р 50948-2001 Средства отображения информации индивидуального пользования. Общие эргономические требования и требования безопасности. 110

43. Федеральный закон №89 от 1998 г. «Об отходах производства и потребления». Глава III, ст. № 9. — 1988. — С. 39. 47. ГОСТ Р 53692-2009 Ресурсосбережение. Обращение с отходами. Этапы технологического цикла отходов — введ. впервые 15.12.2009. — Москва: Стандартинформ, 2011. — С. 20.

44. Федеральный классификационный каталог отходов (с изменениями на 4 октября 2021 года) [Электронный ресурс]. – 2021. – Режим доступа: http://www.consultant.ru/document/cons_doc_LAW_218071/, свободный.

45. Постановление Правительства РФ от 28.12.2020 N 2314 «Об утверждении Правил обращения с отходами производства и потребления в части осветительных устройств, электрических ламп, ненадлежащие сбор, накопление, использование, обезвреживание, транспортирование и размещение которых может повлечь причинение вреда жизни, здоровью граждан, вреда животным, растениям и окружающей среде».

46. N 123-ФЗ от 22.07.2008 (ред. от 30.04.2021) Технический регламент о требованиях пожарной безопасности. 51. Правила устройства электроустановок [Электронный ресурс]. – Режим доступа: https://docs.cntd.ru/document/1200030216, свободный.

47. Правила по охране труда при эксплуатации электроустановок [Электронный ресурс]. – Режим доступа: https://docs.cntd.ru/document/573264184, свободный.

48. СП 12.13130.2009 Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности.

49. Критерии отнесения объектов, оказывающих негативное воздействие на окружающую среду, к объектам I, II, III и IV категорий [Электронный ресурс]. – Режим доступа: https://docs.cntd.ru/document/573292854, свободный