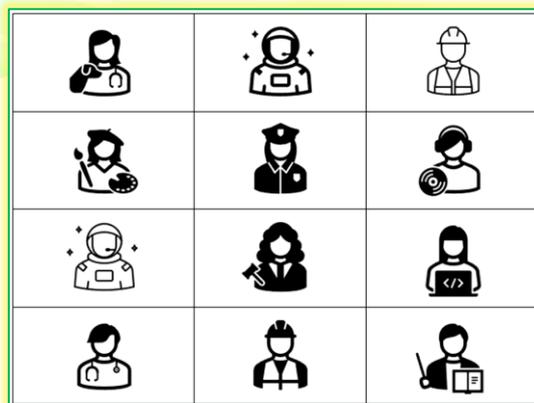


Фартуков В.А., Зборовская М.И.



ФОРМИРОВАНИЕ АЛГОРИТМА МАШИННОГО ОБУЧЕНИЯ ПО УПРАВЛЕНИЮ СИСТЕМОЙ ПОЛИВА

Учебное пособие



Москва, 2024

УДК 631.6:004(075.8)

ББК 40.623.5:32973я75

Ф 25

Рецензенты:

*к.т.н., доцент, и.о. зав. кафедрой сельскохозяйственного водоснабжения, водоотведения, насосов и насосных станций **Али Мунзер Сулейман ИМВХиС имени А.Н. Костякова ФГБОУ ВО РГАУ-МСХА имени К.А. Тимирязева;***

*к.т.н., доцент кафедры инженерных конструкций **Мареева Ольга Викторовна ИМВХиС имени А.Н. Костякова ФГБОУ ВО РГАУ-МСХА имени К.А. Тимирязева;***

Фартуков, В.А. Формирование алгоритма машинного обучения по управлению системой полива: учебное пособие / В.А. Фартуков, М.И. Зборовская, Российский государственный аграрный университет – МСХА имени К. А.Тимирязева. – Москва : РГАУ-МСХА имени К. А. Тимирязева, 2024. – 157 с. - Текст : электронный.

ISBN 978-5-9675-2057-0

Учебное пособие посвящено формированию алгоритма машинного обучения для управления системой полива. Пособие включает введение, основные концепции, этапы разработки и пример реализации.

Система автоматического полива растений позволяет эффективно управлять водными ресурсами, обеспечивая оптимальные условия для роста растений. В данном пособии мы рассмотрим, как можно использовать методы машинного обучения для создания алгоритма, который будет управлять системой полива на основе различных факторов, таких как влажность почвы, температура воздуха и прогноз погоды.

Предназначено для студентов, обучающихся по направлению Строительство 08.04.01 «Речные и подземные гидротехнические сооружения» дисциплина «Интеллектуальные системы в гидротехнике», а также студентов цифровой кафедры, обучающихся по программе «Создание цифровых двойников в гидротехнике».

Рекомендовано к изданию учебно-методической комиссией ИМВХиС ФГБОУ ВО РГАУ-МСХА имени К.А. Тимирязева.

УДК 58/.59(075.8)

ББК 40.1 я 73

ISBN

© Фартуков В.А., Зборовская М.И. 2024



Оглавление

I. ОСНОВНЫЕ КОНЦЕПЦИИ	15
<i>I.1. Система полива</i>	15
<i>I.2. Роль машинного обучения в системах автоматического полива</i>	17
<i>I.3. Сбор данных</i>	18
<i>I.4. Основные концепции машинного обучения</i>	18
<i>I.5. Примеры применения машинного обучения</i>	20
II. СБОР ДАННЫХ	25
<i>II.1 Сбор данных о состоянии почвы, погоде и объеме воды, подаваемого в систему полива</i> 25	
<i>II.1.1. Примеры технологий и компаний:</i>	29
<i>II.2. Данные о погоде</i>	33
<i>II.2.1. Способы сбора данных:</i>	35
<i>II.2.2. Модели прогноза погоды:</i>	45
<i>II.2.3. Полевые наблюдения:</i>	49
<i>II.2.4. Веб-датчики и IoT:</i>	51
<i>II.2.5. Примеры технологий и компаний в метеорологии:</i>	57
<i>II.3. Данные о поливе</i>	63
<i>II.3.1. Примеры технологий и компаний:</i>	65
<i>II.3.2. Примеры исследований и ресурсов</i>	73
III. ЭТАПЫ РАЗРАБОТКИ АЛГОРИТМА	74
<i>III.1. Сбор данных</i>	75
<i>III.2. Предобработка данных</i>	77
<i>III.3. Выбор модели</i>	81
<i>III.4. Обучение модели</i>	83
<i>III.5. Тестирование и валидация</i>	84
<i>III.6. Разработка системы управления</i>	86
<i>III.7. Мониторинг и улучшение</i>	88
IV. ПРИМЕР РЕАЛИЗАЦИИ	92
ЗАКЛЮЧЕНИЕ	Error! Bookmark not defined.
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ	142

ВВЕДЕНИЕ

По прогнозам к 2050 году население мира составит около 9,1 млрд человек, что значительно увеличит и без того высокий мировой спрос на продовольствие. В ближайшие годы ожидается, что производство продовольствия увеличится на 70% во всем мире, а если рассматривать только развивающиеся страны, то оно может увеличиться до 100%. В настоящее время 70% пресной воды, добываемой из водоносных горизонтов, ручьев и озер, используется для орошения сельскохозяйственных культур, и этот процент необходимо будет увеличить, чтобы поддержать ожидаемый рост производства продовольствия [1].

Истощение земель, доступной воды и биоразнообразия в сочетании с изменением климата уже замедляют рост производительности сельского хозяйства, что приводит к опасениям, что в будущем производительность сельского хозяйства будет недостаточной для удовлетворения мирового спроса на продовольствие. Эти факторы, среди прочего, демонстрируют срочность улучшения аспектов сельскохозяйственного производства и позволяют использовать территории, которые не всегда используются для сельскохозяйственной деятельности, такие как городские районы [1].

Наличие больших технологических возможностей становится движущей силой перехода к более интеллектуальным сельскохозяйственным системам, которые при внедрении помогают улучшить качество и количество продовольствия, оптимизируя использование ресурсов, необходимых для его производства, таких как орошение. Это, вместе с возрождением таких видов деятельности, как городское сельское хозяйство, которое приобрело популярность в последние годы и привлекло внимание различных исследователей, может быть правильным способом смягчить прогнозируемый будущий продовольственный кризис [1].

Умное сельское хозяйство, в целом, относится к внедрению новых технологий в управление урожаем для упрощения удаленного мониторинга, оптимизации ресурсов и автоматизации задействованных систем. Эта

концепция связана с концепцией интеллектуальной системы орошения, которая заключается в добавлении интеллекта к автоматическим системам орошения, либо с использованием интеллектуальных методов управления, таких как нечеткая логика или нейронные сети, для улучшения принятия решений по орошению, либо путем включения значительных объемов данных в анализ для улучшения моделирования или управления орошением, используя такие методы, как глубокое обучение или машинное обучение. Интеллектуальные системы орошения стремятся повысить производительность сельского хозяйства, одновременно снижая воздействие сельскохозяйственных культур на окружающую среду путем подачи воды в нужное время, в нужном количестве и в нужном месте каждому растению [1].

Интерес к технологиям, применяемым в орошении, вдохновил на систематический обзор литературы новых и прорывных технологий в городском сельском хозяйстве в публикациях с 2015 по 2021 год, согласно которому наиболее распространенными технологиями в публикациях, а также наиболее применяемыми в орошении, являются IoT, автоматизация и искусственный интеллект [1].

Если почва заранее тщательно подготовлена и периодически подкармливается удобрениями, желательно органическими, урожайность может быть весьма удовлетворительной, что делает еще более интересным изучение того, каким образом использование различных уровней и типов технологий для интеллектуального орошения может помочь выращивать более качественные культуры на почве, одновременно способствуя рациональному использованию воды.

Технологии, связанные с искусственным интеллектом (ИИ), который является дисциплиной компьютерной науки, изучающей и предлагающей алгоритмы для разработки вычислительных решений, имитирующих поведение человека и животных или биологию, представляют собой многообещающие приложения для автоматизации орошения. ИИ не

ограничивается методами, имитирующими только биологическое поведение, но также рассматривает, например, как рассуждает человеческий мозг.

В настоящее время ИИ является одной из наиболее актуальных развивающихся технологий, которая пронизывает многие процессы и разработки в области науки, технологий и бизнеса. Сельское хозяйство в целом выступило в качестве области применения ИИ. В [2], датированной 2020 годом, Руис-Реал проводит систематический обзор ИИ в сельском хозяйстве, где очевидно, что количество публикаций в этой области растет с 2003 года. В том году в базе данных Scopus было зарегистрировано 20 публикаций, а в 2019 году это число выросло до 489 публикаций. Это показывает, как интерес к этой теме постепенно возрастал в течение последнего десятилетия, а увеличение числа публикаций обусловлено тремя улучшениями: (1) ростом данных, связанных с сельскохозяйственными системами, к которым предполагается применять интеллектуальные решения; (2) мощностью алгоритмов ИИ; (3) текущей доступностью быстрых вычислений в мобильных устройствах, датчиках, оборудовании, персональных компьютерах и облаке.

ИИ имеет различные области изучения, и машинное обучение (МО) считается одним из самых важных. Системы МО позволяют компьютерам обучаться с помощью алгоритмов, которые обрабатывают наборы данных и делают прогнозы о них без необходимости явного программирования, и в зависимости от типа приложения и его сложности эти алгоритмы классифицируются на подобласти. В рамках МО выделяется подобласть искусственных нейронных сетей (ИНС), вдохновленная человеческим мозгом и стремящаяся имитировать его сложные функции, в том числе в генерации шаблонов, познании, обучении и принятии решений [3]. ИНС основаны на упрощенных версиях биологической нейронной сети, которая в основном состоит из взаимосвязанных и организованных обрабатывающих единиц, следующих определенной топологии, которая регулярно устанавливается серией узлов, расположенных в нескольких слоях [3].

ИНС обычно используются для регрессии и классификации проблем; перевод языка и изучение шаблонов являются типичными примерами их использования [4]. Глубокое обучение (DL - Deep Learning)) является подобластью МО с определенным сходством с методом ИНС. Этот метод расширяет классическое МО, добавляя сложность к математическим моделям, используемым для обучения, тем самым достигая высокоточных результатов даже с очень сложными данными. Обработка данных в алгоритмах DL включает преобразование данных для обеспечения их иерархического представления через несколько уровней абстракции [5].

Обработка данных в алгоритмах глубокого обучения (DL, deep learning) включает в себя преобразование данных, чтобы они могли быть представлены на различных уровнях абстракции. Давайте разберем это подробнее.

1. Иерархическое представление: В глубоких нейронных сетях данные обрабатываются через несколько слоев, каждый из которых извлекает определенные признаки. На первом слое нейронная сеть может выявлять простые признаки, такие как края или текстуры. На более глубоких слоях сеть может комбинировать эти простые признаки для распознавания более сложных объектов, например лиц или сцен.

2. Преобразование данных: На этапе подготовки данных может происходить ряд преобразований, таких как нормализация (приведение значений к одному масштабу), аугментация (создание новых образцов данных путем изменения существующих), а также кодирование категориальных переменных. Эти преобразования помогают сети лучше обучаться и обобщать информацию.

3. Уровни абстракции: Каждый слой нейронной сети может рассматриваться как уровень абстракции. Например, в сверточных нейронных сетях (CNN) первый слой может обнаруживать края, второй может обнаруживать формы, а третий — более сложные объекты, такие как глаза или уши. Таким образом, на выходе сети мы получаем высокоуровневое представление данных, которое можно использовать для классификации или других задач.

4. Примеры: В задачах обработки изображений, например, сначала могут быть выделены простые геометрические формы, затем — более сложные структуры, а в конечном итоге — объекты, такие как животные или машины. В обработке текста на начальных этапах могут быть выделены слова, затем фразы, а на более высоком уровне — темы или смысловые связи.

В изображениях слои абстракции цветов и форм, среди прочего, созданы для идентификации и классификации шаблонов. Эти слои позволяют обеспечить точность и скорость прогнозирования моделей, а также анализировать сложные данные, такие как изображения, видео, аудио, голос и данные естественного языка.

Таким образом, подход глубокого обучения позволяет эффективно обрабатывать и анализировать данные, извлекая из них полезную информацию на разных уровнях абстракции, что делает его мощным инструментом для решения сложных задач.

Нечеткая логика также считается областью ИИ. Этот термин был введен в 1965 году Лофти Аскером Заде как название систематического способа решения проблем, которые люди обычно решают с помощью здравого смысла и опыта [6,7]. Нечеткая логика возникает как альтернативное решение проблем, которые классическая булева логика не может решить, учитывая, что этот подход может справиться только с переменными, принимающими только двоичные значения, которые обычно математически представлены нулями и единицами. Когда проблема требует использования переменных, которые могут принимать континуум значений, результатом являются выражения, которые нельзя считать полностью ложными или истинными. Вводя понятие «степени» в проверку условия, связанного с проблемой, возможно, что это условие принимает значения, отличные от истинного и ложного. Следовательно, это обеспечивает ценную гибкость для рассуждений и позволяет учитывать неточности и неопределенности, связанные с реальными физическими ситуациями. Нечеткая логика классифицируется как область ИИ, поскольку она используется для моделирования поведения человеческих решений, полученных из опыта с использованием лингвистических переменных и правил ЕСЛИ–ТО, которые имитируют процесс человеческого мышления. Используя формулировки нечеткой логики, интересующие переменные можно классифицировать в наборы, определяемые нечеткими характеристиками, такими как очень высокая, высокая, средняя, низкая или очень низкая влажность или температура. Таким образом, как поведение системы, подлежащее управлению, так и желаемый ответ системы управления можно лингвистически описать для их моделирования и проектирования, даже если в системе нет традиционных математических моделей. С другой стороны, Интернет вещей (IoT), который во многих приложениях взаимосвязан с ИИ, был определен как взаимосвязанная система вычислительных устройств,

механических и цифровых машин, объектов, животных или людей, снабженных уникальными идентификаторами. Это система, способная передавать данные по сети без необходимости взаимодействия человека с человеком или человека с компьютером. Интернет вещей стремится интегрировать физический мир с виртуальным, используя Интернет как средство для общения и извлечения информации [[6,7].

Области применения технологий Интернета вещей обширны и разнообразны, поскольку решения Интернета вещей все больше распространяются практически на все сферы повседневной жизни [34]. Сельское хозяйство является одним из секторов, на который, как ожидается, сильно повлияет прогресс в области Интернета вещей [4]. Даже сейчас Интернет вещей вносит значительный вклад в сельское хозяйство, в том числе в системы орошения и удобрения, метеорологический мониторинг, мониторинг почвы, а также борьбу с болезнями и вредителями и т. д.; ожидается, что в будущем он продолжит оказывать влияние на еще большее количество областей.

Повышение эффективности аграрного сектора в России невозможно без использования информационных технологий, что связано с тенденцией на цифровизацию всех экономических отраслей и сфер общественной жизни. Цифровое сельское хозяйство базируется на современных способах производства сельскохозяйственной продукции и продовольствия с использованием таких технологий как интернет вещей, искусственный интеллект, робототехника, беспилотные летательные аппараты, геоинформационные системы и технологии и др. [8]. Освоение цифровых компетенций — необходимое условие повышения конкурентоспособности в сельском хозяйстве. Основными направлениями этого процесса в АПК (Цифровизация сельскохозяйственного производства России на период 2018-2025 гг.) являются реализация таких проектов как «Цифровые технологии в управлении АПК», «Умное землепользование», «Умное поле», «Умный сад»,

«Умная теплица», «Умная ферма», «Создание отраслевой платформы, направленной на решение задач по развитию цифровизации АПК» (рис. 1)



Рисунок 1. Организационные механизмы реализации Программы «Цифровое (умное) сельское хозяйство»

Цифровое сельское хозяйство требует высокой подготовленности кадров, начиная с обучения в ВУЗе, создание новых моделей развития АПК с точки зрения киберэкономики. В связи с этим, подготовка выпускников ВУЗов для работы в сельском хозяйстве с учетом требований развития цифровизации агропромышленного комплекса стала актуальной задачей.

В аграрном образовании и сельском хозяйстве существуют несколько инновационных решений, которые можно отнести к концепции умного сельского хозяйства: облачные решения и интернет вещей (IoT-платформы); беспилотные транспортные средства; летательные аппараты; датчики и сенсоры; ГЛОНАСС/GPS; геоинформационные системы и технологии дистанционного зондирования [8].

Система автоматического полива растений позволяет эффективно управлять водными ресурсами, обеспечивая оптимальные условия для роста растений. В данном пособии мы рассмотрим, как можно использовать методы

машинного обучения для создания алгоритма, который будет управлять системой полива на основе различных факторов таких как влажность почвы и наличие микроэлементов (получаемых от полевых датчиков (IoT)), температура воздуха и прогноз погоды.

Согласно [9] такие аспекты, как изменение климата и рост населения, представляют собой серьезные проблемы в отраслях, способных производить достаточное количество сельскохозяйственных культур для всех. Это привело к тому, что бизнес-лидеры ищут новые инновационные подходы в целях повышения урожайности своих культур.

Одним из наиболее важных решений, которые сейчас реализуются, является искусственный интеллект (ИИ). ... Искусственный интеллект в сельском хозяйстве — это не только новая тенденция, но и возможность изменить будущее аграрной промышленности. С помощью технологии ИИ ежегодно можно производить большее количество культур, чтобы обеспечить продовольствием постоянно растущее население земли [7].

Помимо технологий с использованием искусственного интеллекта в агропродовольственной сфере не менее актуальной выступает концепция вычислительной сети физических предметов, оснащённых встроенными технологиями для взаимодействия друг с другом или с внешней средой под названием «интернет вещей». Internet of Things (IoT) или интернет вещей — это сеть связанных через интернет объектов, способных собирать данные и обмениваться данными, поступающими со встроенных сервисов.

Внедрение интернета вещей стало возможным за счет широкого распространения интернета, смартфонов, беспроводных сетей, удешевления электронных компонентов и обработки данных. На практике IoT-системы обычно состоят из сети умных устройств и облачной платформы, к которой они подключены. К ним примыкают системы хранения, обработки и защиты собранных датчиками данных.

Данная технология позволяет напрямую взаимодействовать между собой различным устройствам и оборудованию, в которое внедрен интерфейс

IoT. Для сельского хозяйства это, прежде всего, умное производство, точное земледелие, агрономия и управление севооборотом и пр. [9].

Выращивание овощей — это важный аспект сельского хозяйства, который может оказывать значительное влияние на окружающую среду, в частности, на углеродный след. Углеродный след относится к количеству углекислого газа и других парниковых газов, которые выбрасываются в атмосферу в результате человеческой деятельности, включая сельское хозяйство.

Основные аспекты, связанные с углеродным следом при выращивании овощей:

1. Методы выращивания:

Органическое земледелие: использование органических методов может снизить углеродный след, так как они часто требуют меньше химических удобрений и пестицидов, которые производятся с использованием ископаемого топлива.

Интенсивное земледелие: применение тяжелой техники и химических удобрений может увеличивать углеродный след из-за сжигания топлива и выбросов, связанных с производством удобрений.

2. Транспортировка:

Овощи, которые выращиваются далеко от мест потребления, требуют значительных затрат на транспортировку, что увеличивает углеродный след. Локальное выращивание может помочь сократить эти выбросы.

3. Выбор культур:

Некоторые овощи требуют больше ресурсов для роста, чем другие. Например, помидоры и огурцы могут иметь более высокий углеродный след, чем более устойчивые к климату корнеплоды, такие как морковь или свекла.

4. Сезонное выращивание:

Выращивание овощей в их естественный сезон может снизить потребность в теплицах и других энергоемких методах, что также уменьшает углеродный след.

5. Управление почвой:

правильное управление почвой, включая севооборот и использование покровных культур, может помочь в захвате углерода в почве и уменьшении выбросов.

6. Снижение пищевых отходов:

уменьшение отходов на этапе потребления также имеет значение. Если овощи не выбрасываются, это снижает общий углеродный след, связанный с их производством и утилизацией.

Рекомендации для снижения углеродного следа при выращивании овощей:

выбираем органические или местные продукты;

используем методы устойчивого сельского хозяйства, такие как пермакультура или агрономия;

участвуем в локальных фермерских рынках или CSA (Community Supported Agriculture);

развиваем собственный огород, чтобы уменьшить зависимость от коммерческого производства;

старемся минимизировать отходы и перерабатывать остатки пищи.

Таким образом, при правильном подходе к выращиванию овощей можно значительно снизить углеродный след и внести вклад в устойчивое развитие.

I. ОСНОВНЫЕ КОНЦЕПЦИИ

I.1. Система полива

— это механизм, который автоматически регулирует количество воды, подаваемое к растениям.

Автоматизация полива позволяет:

- Сэкономить воду: Использование датчиков для определения уровня влажности почвы позволяет подавать воду только тогда, когда это действительно необходимо.
- Улучшить здоровье растений: Оптимизированный полив помогает избежать как недостатка, так и избытка влаги, что важно для здоровья растений.
- Снизить трудозатраты: Автоматические системы требуют меньше ручного труда для управления поливом.

Как работает система автоматического полива?

Система автоматического полива обычно состоит из следующих компонентов:

- Датчики: Для измерения влажности почвы, температуры воздуха, уровня солнечной радиации и других параметров.
- Контроллер: Устройство, которое принимает данные от датчиков и принимает решения о необходимости полива.
- Система полива: Насосы и трубы, которые подают воду к растениям.

Принципы работы систем автоматического полива

Системы автоматического полива используют различные технологии для мониторинга и управления подачей воды. Основные компоненты таких систем включают:

- Датчики: Устанавливаются в почве для измерения уровня влажности, температуры и других параметров окружающей среды. Эти датчики обеспечивают данные в реальном времени, которые необходимы для принятия решений о поливе.

- Контроллеры: Устройства, которые обрабатывают данные от датчиков и принимают решения о необходимости полива. Современные контроллеры могут использовать алгоритмы для адаптации к изменяющимся условиям, таким как изменение влажности или изменение прогноза погоды.

- Системы распределения воды: Это насосы и трубы, которые обеспечивают подачу воды к растениям. Современные системы могут включать капельный полив, который позволяет минимизировать потери воды и обеспечить более точное распределение.

Важность систем автоматического полива

Системы автоматического полива становятся всё более актуальными в условиях изменения климата, увеличения населения и истощения водных ресурсов. Эффективное управление водными ресурсами является критически важным для обеспечения продовольственной безопасности и устойчивого сельского хозяйства.

1. Увеличение потребности в продовольствии: С ростом населения и урбанизацией увеличивается спрос на продовольствие. По прогнозам ООН, к 2050 году население мира достигнет 9,7 миллиарда человек, что потребует увеличения производства продовольствия на 70%. Эффективные системы полива могут помочь в достижении этой цели, позволяя фермерам оптимально использовать доступные ресурсы.

2. Изменение климата: Изменения климата приводят к изменению погодных условий, включая увеличение частоты и интенсивности засух. Это требует внедрения новых технологий, которые помогут адаптироваться к изменяющимся условиям и минимизировать риски для сельского хозяйства.

3. Истощение водных ресурсов: Вода становится дефицитом в некоторых регионах, и её неэффективное использование может привести к серьёзным последствиям. Автоматизированные системы полива помогают сократить расход воды, обеспечивая при этом достаточное увлажнение почвы.

1.2. Роль машинного обучения в системах автоматического полива

Методы машинного обучения становятся важным инструментом для оптимизации работы систем автоматического полива. Они позволяют анализировать большие объёмы данных и выявлять сложные зависимости, которые могут быть незаметны при традиционном анализе. Вот как машинное обучение может быть использовано в автоматизированных системах полива:

1. Анализ данных: Машинное обучение позволяет анализировать данные, собранные с датчиков, и выявлять паттерны. Например, можно установить, как влажность почвы, температура и уровень солнечной радиации влияют на потребность в воде для различных растений.

2. Прогнозирование: Алгоритмы машинного обучения могут использоваться для прогнозирования потребности в поливе на основе исторических данных и текущих условий. Это позволяет более точно планировать полив и избегать как недостатка, так и избытка влаги.

3. Адаптивные системы: Системы, использующие машинное обучение, могут адаптироваться к изменяющимся условиям, таким как изменения в погоде или изменяющиеся потребности растений. Это позволяет фермерам более эффективно управлять ресурсами и увеличивать урожайность.

4. Интеграция с другими системами: Машинное обучение может быть интегрировано с другими системами, такими как прогнозирование погоды и управление фермами, что позволяет создавать более комплексные и эффективные решения для управления сельским хозяйством.

Таким образом, системы автоматического полива растений играют ключевую роль в обеспечении эффективного управления водными ресурсами и создании оптимальных условий для роста растений. Внедрение методов машинного обучения в эти системы открывает новые возможности для повышения их эффективности и устойчивости. В данном пособии мы рассмотрим, как можно разрабатывать алгоритмы, которые будут управлять системой полива на основе различных факторов, таких как влажность почвы,

температура воздуха и прогноз погоды, а также как эти алгоритмы могут быть реализованы на практике.

1.3. Сбор данных

- для обучения модели машинного обучения необходимы данные, которые могут включать:

- Влажность почвы (в процентах)
- Температуру воздуха (в градусах Цельсия)
- Уровень солнечной радиации
- Прогноз осадков
- Время суток

1.4. Основные концепции машинного обучения

Машинное обучение — это раздел искусственного интеллекта, который занимается созданием алгоритмов, способных обучаться на данных и делать предсказания. То есть машинное обучение (МЛ) — это подмножество искусственного интеллекта (ИИ), которое фокусируется на разработке алгоритмов и статистических моделей, позволяющих компьютерам выполнять задачи без явного программирования. Это достигается путем анализа данных, выявления закономерностей и создания моделей, которые могут делать предсказания или принимать решения на основе новых данных

Типы машинного обучения

Существует несколько основных типов машинного обучения:

- Обучение с учителем (Supervised Learning): В этом подходе модели обучаются на размеченных данных, где каждая запись данных имеет соответствующую метку или результат. Например, для классификации растений по их видам, модель может обучаться на данных о различных характеристиках (например, цвет, размер, форма листьев) и соответствующих им видах.

Пример: Если у вас есть набор данных с характеристиками растений и их видами, вы можете использовать алгоритмы, такие как логистическая регрессия или деревья решений, для предсказания вида растения на основе его характеристик.

- Обучение без учителя (Unsupervised Learning): В этом случае модели обучаются на неразмеченных данных, и задача заключается в выявлении скрытых структур или паттернов в данных. Например, кластеризация может быть использована для группировки растений по схожим характеристикам.

Пример: Алгоритм К-средних может быть использован для группировки растений на основе их характеристик, без предварительной информации о том, к какому виду они принадлежат.

- Обучение с подкреплением (Reinforcement Learning): В этом подходе агент учится через взаимодействие с окружающей средой, получая награды или штрафы за свои действия. Это может быть полезно в ситуациях, где необходимо принимать последовательные решения.

Пример: Алгоритмы обучения с подкреплением могут быть использованы для управления системой полива, где агент (система) получает награду за оптимальное распределение воды.

Этапы разработки модели машинного обучения

Процесс разработки модели машинного обучения обычно включает следующие этапы:

1. *Сбор данных:* Необходимо собрать качественные данные, которые будут использоваться для обучения модели. Это может включать как структурированные данные (например, таблицы с характеристиками), так и неструктурированные данные (например, текст или изображения).

2. *Предобработка данных:* На этом этапе данные очищаются и подготавливаются для обучения. Это может включать удаление пропусков, нормализацию значений и преобразование категориальных данных в числовые.

3. *Выбор алгоритма*: В зависимости от задачи выбирается соответствующий алгоритм машинного обучения. Например, для задачи классификации можно использовать логистическую регрессию, дерево решений или нейронные сети.

4. *Обучение модели*: Модель обучается на обучающей выборке данных, где она находит зависимости и паттерны.

5. *Тестирование и оценка*: Модель тестируется на отдельной тестовой выборке, чтобы оценить её производительность и точность.

6. *Внедрение и мониторинг*: После успешного тестирования модель может быть внедрена в реальную систему, где её эффективность будет постоянно отслеживаться и корректироваться при необходимости.

1.5. Примеры применения машинного обучения

- **Распознавание изображений**: Алгоритмы машинного обучения, такие как свёрточные нейронные сети (CNN), используются для распознавания объектов на изображениях. Это применяется в таких областях, как медицинская диагностика (например, распознавание раковых клеток на снимках) и автономные автомобили (распознавание дорожных знаков и пешеходов).

- **Обработка естественного языка (NLP)**: Алгоритмы машинного обучения используются для анализа и обработки текстов. Примеры включают чат-боты, автоматические переводчики и системы рекомендаций.

- **Финансовый анализ**: Машинное обучение применяется для прогнозирования цен на акции, оценки кредитоспособности и обнаружения мошенничества.

Вот несколько примеров применения машинного обучения в системах полива растений и в сельском хозяйстве в целом, а также ссылки на соответствующие исследования и статьи:

Оптимизация полива с использованием датчиков

Описание: Использование датчиков для мониторинга влажности почвы, температуры и других факторов, а затем применение алгоритмов машинного обучения для предсказания потребности в воде. Например, модели могут анализировать данные о влажности и температуре, чтобы решить, когда и сколько воды необходимо подать.

Пример: В исследовании, проведенном в Университете Калифорнии, были использованы алгоритмы машинного обучения для разработки системы, которая оптимизирует полив в зависимости от состояния почвы и погодных условий [10].

В работе отмечается, что пресная вода необходима для орошения и обеспечения питательными веществами для роста растений, чтобы компенсировать нехватку осадков. При этом сельскохозяйственная деятельность использует около 70% доступной пресной воды. Этот факт подчеркивает важность ответственного управления с использованием интеллектуальных сельскохозяйственных водных технологий. Такое управление опирается на исследования, касающиеся интеграции различных моделей машинного обучения и могут обеспечить оптимальное управление решениями по орошению.

Рассматривается развертывание разработанных моделей машинного обучения для использования фермерами в целях устойчивого управления орошением. Обсуждается, как цифровые решения для сельского хозяйства, такие как мобильные и веб-фреймворки¹ [10]. Веб-платформа может быть интегрирована с базами данных для пользователей для выполнения манипулирования данными, визуализации, аналитики и удаленного управления. Эти веб-приложения могут помочь фермерам принимать

1 Веб-фреймворки — это программные платформы, которые предоставляют разработчикам инструменты и библиотеки для создания веб-приложений и веб-сайтов. Они упрощают процесс разработки, позволяя сосредоточиться на функциональности приложения, а не на рутинных задачах, связанных с разработкой. Веб-фреймворки могут быть как серверными, так и клиентскими. Например Ruby on Rails (Ruby): Фреймворк, который делает акцент на разработку по принципу "сначала делай, потом думай", что позволяет быстро создавать приложения.

Веб-фреймворки значительно упрощают процесс разработки веб-приложений, позволяя разработчикам сосредоточиться на бизнес-логике и функциональности, а не на низкоуровневых деталях. Выбор фреймворка зависит от требований проекта, предпочтений команды разработчиков и других факторов.

решения, связанные с орошением, такие как расчет общего объема используемой воды для орошения и стоимости практики орошения, оценка состояния воды в почве (потребляемой воды) и управление ирригационным оборудованием с дистанционным управлением, среди прочего. Кроме того, интеграция IoT и аналитики больших данных в облачные базы данных (DB), такие как веб-сервис Amazon, Microsoft Azure, облачная платформа Oracle DBGoogle MangoDB Atlas и т.д., предоставила возможность для извлечения сохраненных экспериментальных данных для составления прогноза для фермеров с помощью их мобильной и веб-платформы по удобрениям с учетом требований к орошению, а также маркетингового проекта собранной продукции.

Обсуждение в статье [10]: раздел 2 рассматривает современное состояние использования моделей машинного обучения для точного орошения, в то время как раздел 3 фокусируется на современном состоянии применения цифровых решений, таких как мобильное приложение–веб-фреймворк для интеллектуального управления орошением. Раздел 4 изучает проблемы и возможности применения машинного обучения к системам точного орошения. Рассматриваются будущие тенденции.

Ожидается, что стремление к Индустрии 4.0² в сельском хозяйстве побудит больше исследовательских работ по внедрению технологии цифровых двойников³ в интеллектуальные системы орошения. Ожидается, что интеграция методов машинного обучения и интеграция мобильных и веб-решений принесет много преимуществ как фермерам, так и пользователям.

Умные системы полива

² **Индустрия 4.0** — это концепция, которая включает в себя использование современных технологий, таких как Интернет вещей (IoT), большие данные, искусственный интеллект (AI) и автоматизацию, для создания "умных" производственных процессов. В контексте сельского хозяйства это означает применение этих технологий для повышения эффективности, устойчивости и производительности аграрного сектора.

³ **Цифровой двойник** — это виртуальная модель физического объекта или процесса, которая отражает его характеристики и поведение в реальном времени. В сельском хозяйстве цифровые двойники могут быть использованы для моделирования различных аспектов, таких как: - Поля и растения: Моделирование роста растений, состояния почвы и условий окружающей среды. - Системы орошения: Моделирование работы орошения, включая объем воды, необходимый для оптимального роста растений, и прогнозирование потребностей в воде.

Описание: Интеграция машинного обучения в системы автоматического полива, которые могут адаптироваться к изменяющимся условиям. Эти системы могут использовать данные с метеостанций и датчиков, чтобы принимать решения о поливе в реальном времени.

Пример: В статье «A Smart Irrigation System Powered by IoT and Machine Learning for Optimal Water Management» на сайте "IEEE Xplore"[11] описывается умная система полива, использующая машинное обучение и IoT (Интернет вещей) для оптимизации использования воды в сельском хозяйстве.

В этом инновационном стремлении к более эффективному орошению авторы исследуют совместное использование сенсорных технологий и машинного интеллекта для оптимизации использования воды в сельском хозяйстве. Собирают точные данные о важнейших переменных, включая расход воды, температуру, влажность, влажность почвы и уровень воды каждые 15 дней, оперативно устанавливая датчики на местах для мониторинга их в режиме реального времени. Используют огромное количество доступных данных для обучения моделей машинного обучения, включая рекуррентные нейронные сети (RNN), К ближайших соседей (KNN), искусственные нейронные сети (ANN) и сверточные нейронные сети (CNN), специально для потоков, которые, как ожидается, будут иметь определенное распределение. CNN обладает исключительной точностью при интерпретации визуальных сигналов с точностью декодирования 94,5%, превосходя ANN, KNN и RNN, которые достигают более низкой точности 91,1%, 88,7% и 83,6% соответственно. Исследование выявляет гибкое взаимодействие между данными датчиков и обучением модели, что приводит к тому, что каждая модель обладает уникальными характеристиками. Эти результаты не только демонстрируют точность, с которой машинное обучение может распределять воду, но и свидетельствуют о заметном прогрессе в ресурсосберегающем и устойчивом сельском хозяйстве. Помимо математических открытий, исследования закладывают основу для создания

передовых ирригационных систем, которые плавно интегрируют механические и естественные процессы в сельском хозяйстве.

Машинное обучение имеет широкий спектр применений в области полива растений и сельского хозяйства, позволяя оптимизировать использование водных ресурсов и повышать урожайность. Использование таких технологий помогает фермерам принимать более обоснованные решения и адаптироваться к изменяющимся условиям окружающей среды.



II. СБОР ДАННЫХ

II.1 Сбор данных о состоянии почвы, погоде и объеме воды, подаваемого в систему полива

Сбор данных о состоянии почвы, погоде и объеме воды, подаваемого в систему полива, является важным аспектом для эффективного управления орошением и оптимизации сельскохозяйственного производства. Рассмотрим, как можно собирать эти данные, какие источники использовать, и какие технологии могут помочь в этом процессе. Это могут быть как исторические данные, так и данные в режиме реального времени.

Данные о состоянии почвы могут включать: влажность почвы, температуру почвы, pH почвы, наличие питательных веществ (азот, фосфор, калий и др.)

Способы сбора:

- Сенсоры: Используются например беспроводные сенсоры для мониторинга влажности и температуры почвы типа Capacitive Soil Moisture Sensors. Они могут предоставлять данные о влажности почвы в реальном времени [12] (рис. 1).

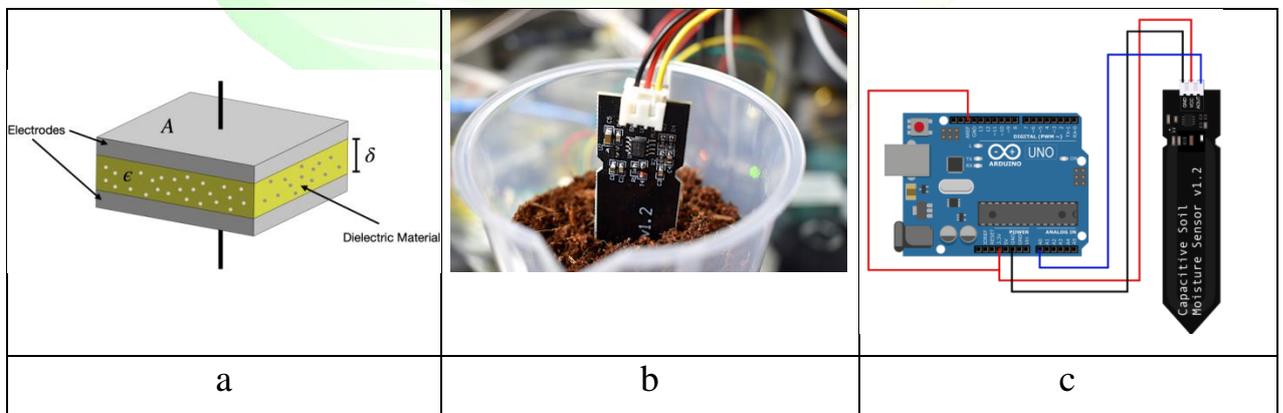


Рисунок 1. Беспроводные сенсоры для мониторинга влажности и температуры почвы типа Capacitive Soil Moisture Sensors

а - Плоский конденсатор, показывающий стандартные компоненты, используемые для расчета емкости для заданного материала и электрического потенциала; b - ёмкостный датчик влажности почвы, вставленный в тестовую почву. Он работает при напряжении 3,3 В, использует интегральную схему таймера 555 и выдает аналоговый сигнал, пропорциональный резонансу; с - подключение ёмкостного датчика влажности почвы к плате Arduino Uno

Согласно [12] влажность почвы можно измерить с помощью различных методов: гравиметрического, ядерного, электромагнитного, тензиометрического, гигрометрического и других. В рассмотренном здесь методе для калибровки электромагнитного датчика влажности почвы ёмкостного типа используется гравиметрический метод. Ёмкостные датчики влажности почвы используют диэлектрический контраст между водой и почвой, где относительная диэлектрическая проницаемость сухой почвы составляет от 2 до 6, а воды — примерно 80. Точное измерение содержания воды в почве необходимо для применения в агрономии и ботанике, где недостаточный или чрезмерный полив может привести к неэффективному использованию ресурсов или их перерасходу. Поскольку в некоторых почвах вода занимает до 60% объёма, в зависимости от удельной пористости почвы, для точного прогнозирования содержания воды необходимо проводить калибровку в каждой среде. К счастью, точность измерительных приборов повышается, а стоимость датчиков снижается. В этом эксперименте платформа Arduino используется для программирования микроконтроллера для считывания аналогового сигнала с ёмкостного датчика, который, в свою очередь, выдаёт напряжение. Обратное значение этого напряжения можно линейно аппроксимировать для приблизительного определения объёмного содержания влаги в почве с помощью гравиметрических методов. Это делается путём измерения объёма и взвешивания сухой и влажной почвы в диапазоне влажности. Этот процесс описан в [12].

Системы IoT: Система IoT собирает данные с сенсоров и передает их на центральный сервер или в облако для анализа.

Система Интернета вещей (IoT) — это сеть взаимосвязанных физических объектов, «вещей», оснащенных встроенными датчиками, программным обеспечением, актуаторами и другими технологиями для обмена данными друг с другом и внешними системами через интернет. Ключевым элементом этой сети является сбор данных с сенсоров и их передача для анализа. Рассмотрим этот процесс подробнее:

Сбор данных с сенсоров:

Типы сенсоров: IoT-системы используют разнообразные сенсоры, измеряющие различные параметры: температуру, влажность, давление, свет, движение, звук, химический состав, геолокацию и многое другое. Выбор сенсоров зависит от конкретного применения системы. Например, система умного дома может использовать датчики температуры, влажности и движения, а система мониторинга окружающей среды — датчики качества воздуха, уровня воды и температуры почвы.

Методы сбора данных: Данные с сенсоров могут собираться различными способами, включая аналого-цифровое преобразование (АЦП), цифровые интерфейсы (I2C, SPI, UART) и беспроводные протоколы связи.

- **Частота сбора данных:** Частота сбора данных может варьироваться от нескольких раз в секунду до нескольких раз в день или даже реже, в зависимости от требований приложения. Высокая частота сбора данных необходима для мониторинга быстро меняющихся параметров, а низкая — для параметров, которые изменяются медленно.

Передача данных:

Беспроводные протоколы связи: Большинство IoT-устройств используют беспроводные протоколы связи для передачи данных на центральный сервер или в облако. Наиболее распространенные протоколы включают:

Wi-Fi: Обеспечивает высокую скорость передачи данных и подходит для устройств, находящихся в зоне действия Wi-Fi сети.

Bluetooth: Используется для передачи данных на короткие расстояния, подходит для устройств с низким энергопотреблением.

Zigbee: Протокол с низким энергопотреблением и высокой надежностью, используется в сетях с большим количеством устройств.

LoRaWAN: Протокол дальнего действия с низким энергопотреблением, идеально подходит для устройств, расположенных на значительном расстоянии от центральной точки.

NB-IoT/LTE-M: Cellular-технологии, обеспечивающие широкий охват и надежную связь, часто используются для устройств, расположенных в удаленных местах.

Проводные соединения: В некоторых случаях для передачи данных могут использоваться проводные соединения, например, Ethernet. Это обеспечивает более высокую скорость и надежность передачи данных, но ограничивает мобильность устройств.

Шлюзы (Gateways): Часто используются шлюзы, которые собирают данные от нескольких устройств и передают их на центральный сервер или в облако. Шлюзы могут выполнять предварительную обработку данных, например, агрегацию или фильтрацию, что снижает нагрузку на центральный сервер.

Анализ данных:

Центральный сервер или облако: Собранные данные передаются на центральный сервер или в облако, где хранятся и обрабатываются.

Методы анализа данных: Для анализа данных используются различные методы, включая:

Анализ временных рядов: Используется для выявления трендов и аномалий в данных, изменяющихся во времени.

Машинное обучение: Используется для построения прогнозных моделей и автоматической классификации данных.

Data mining: Используется для поиска закономерностей и скрытых знаний в больших объемах данных.

Визуализация данных: Результаты анализа данных часто представляются в виде графиков, диаграмм и других визуальных элементов для удобства восприятия.

В целом, процесс сбора, передачи и анализа данных в IoT-системах включает в себя:

1. Датчики: измеряют физические параметры.
2. Устройства: собирают данные с датчиков и передают их.
3. Сеть: обеспечивает передачу данных между устройствами и центральной системой.
4. Центральный сервер/облако: хранит и обрабатывает данные.
5. Аналитические инструменты: обрабатывают данные и предоставляют информацию.
6. Визуализация: представляет информацию в

Этот процесс позволяет получать ценную информацию о состоянии окружающего мира и принимать обоснованные решения на основе данных. Это используется в широком диапазоне приложений, от умных домов и городов до промышленной автоматизации и мониторинга окружающей среды.

II.1.1. Примеры технологий и компаний:

Decagon Devices⁴: Производит сенсоры для измерения влажности и других характеристик почвы.

Soil moisture sensors: Различные модели от производителей, таких как ***Vegetronix*** и ***AquaSpy***.

Давайте подробнее рассмотрим Decagon Devices и датчики влажности почвы, включая примеры от Vegetronix и AquaSpy. Все эти компании и технологии играют важную роль в сельском хозяйстве, научных исследованиях и мониторинге окружающей среды.

⁴ американская компания. Её штаб-квартира находится в Пуллмане, штат Вашингтон, США

Decagon Devices — это компания, специализирующаяся на разработке и производстве высокоточных датчиков и измерительных приборов для различных применений, с сильным фокусом на сельском хозяйстве и научных исследованиях. Их продукция известна своей надежностью, точностью и долговечностью. В области измерения влажности почвы Decagon предлагает широкий спектр решений:

Датчики влажности почвы: Они предлагают различные типы датчиков влажности почвы, основанные на разных принципах измерения: диэлектрической постоянной, частоте, сопротивлении. Это позволяет выбрать датчик, оптимально подходящий для конкретных условий и задач. Разные модели отличаются по размерам, глубине погружения, диапазону измеряемых значений и другим характеристикам.

Измерительные системы: Decagon не только производит датчики, но и предлагает целые системы для измерения влажности почвы, включающие программное обеспечение для сбора, обработки и анализа данных. Это позволяет автоматизировать процесс мониторинга и получать полную картину состояния почвы.

Программное обеспечение: Программное обеспечение Decagon обычно обеспечивает удобный интерфейс для просмотра данных, построения графиков, экспорта данных в различные форматы и выполнения анализа.

Применение: Датчики Decagon используются в различных областях, включая:

Прецизионное земледелие: Оптимизация полива, удобрения и других агротехнических мероприятий на основе данных о влажности почвы.

Научные исследования: Изучение влияния различных факторов на водный баланс почвы.

Мониторинг окружающей среды: Оценка состояния почвы и ее влияния на окружающую среду.

Soil moisture sensors (датчики влажности почвы):

Рынок датчиков влажности почвы весьма разнообразен, и *Vegetronix*⁵ и *AquaSpy*⁶ — это лишь два примера производителей. Ключевые различия между различными моделями и производителями часто заключаются в:

Принцип работы: Как уже упоминалось, датчики могут использовать различные принципы измерения. Диэлектрические датчики измеряют диэлектрическую постоянную почвы, которая коррелирует с влажностью. Тензометрические датчики измеряют изменение напряжения в чувствительном элементе при изменении влажности. Датчики на основе нейтронного рассеяния используются для измерения влажности на больших глубинах.

Диапазон измерения: Датчики могут измерять влажность в разных диапазонах, от очень сухой до полностью насыщенной водой почвы.

Глубина измерения: Датчики могут быть установлены на различных глубинах, в зависимости от задачи.

Размер и конструкция: Датчики могут быть разных размеров и форм, что влияет на их установку и использование.

Цена: Цена датчиков может значительно варьироваться в зависимости от производителя, принципа работы и других характеристик.

Vegetronix: Известна своими датчиками влажности почвы, часто используемыми в исследованиях и сельском хозяйстве. Они, как правило, предлагают высокоточное измерение и долговечность. Конкретные модели могут отличаться по своим техническим характеристикам.

AquaSpy: Предлагает системы измерения влажности почвы, которые отличаются высокой точностью и возможностью получения профилей влажности почвы на разных глубинах. Их системы часто используются для мониторинга больших площадей и обеспечивают более комплексный анализ состояния почвы.

⁵ расположенная в Скалистых горах штата Юта, была основана в 2008 году с целью коммерциализации новой технологии датчиков влажности почвы. <https://vegetronix.com/company/about>

⁶ Сан-Диего, Калифорния

В заключение, выбор конкретной технологии и производителя датчиков влажности почвы зависит от конкретных требований и задач. Decagon Devices, Vegetronix и AquaSpy — это лишь некоторые из многих компаний, предлагающих высококачественные решения в этой области. Для принятия решения необходимо учитывать такие факторы, как точность, диапазон измерений, глубина измерения, цена, совместимость с другими устройствами и программным обеспечением, и долговечность.

Информация о российских производителях датчиков влажности почвы ограничена в открытом доступе по сравнению с западными аналогами. Крупных компаний, специализирующихся именно на этом сегменте, мало. Однако, российские организации и ученые занимаются разработкой и производством подобных устройств, хотя часто это происходит в рамках более широких исследований и разработок в области агрономии, метеорологии и мониторинга окружающей среды.

Что можно сказать о российских датчиках влажности почвы:

- Часто входят в состав более крупных систем: Вместо самостоятельных продуктов, датчики влажности почвы часто являются частью комплексных систем мониторинга, разработанных в российских университетах, НИИ или компаниях, специализирующихся на сельскохозяйственных технологиях или автоматизации. Эти системы могут включать не только датчики влажности, но и другие сенсоры (температуры, освещенности, состава почвы и т.д.), а также программное обеспечение для обработки данных.
- Меньший масштаб производства: По сравнению с зарубежными компаниями, российское производство таких датчиков, скорее всего, имеет меньший масштаб и, как следствие, ограниченный рынок сбыта. Это может

быть связано как с экономическими факторами, так и с недостатком инвестиций в данную область.

- Использование импортных компонентов: Возможно использование импортных компонентов в некоторых моделях российских датчиков, особенно микроэлектроники. Это зависит от доступности и стоимости отечественных аналогов.

- Ориентация на специфические ниши: Российские разработчики могут концентрироваться на создании датчиков, адаптированных к специфическим условиям российского климата и типов почв.

К сожалению, без конкретных названий компаний или институтов сложно предоставить более подробную информацию. Для поиска информации о конкретных российских разработчиках датчиков влажности почвы можно попробовать:

- Поиск в научных базах данных: Искать публикации и патенты, связанные с разработкой датчиков влажности почвы российскими учеными и организациями.

- Поиск в реестрах российских производителей: Проверить базы данных российских производителей измерительной техники.

- Обратиться к специализированным организациям: Связаться с российскими сельскохозяйственными университетами, НИИ и компаниями, занимающимися разработкой и внедрением прецизионных технологий в сельском хозяйстве.

В целом, российский рынок датчиков влажности почвы развивается, но пока уступает по масштабу и разнообразию зарубежным аналогам. Однако, существует потенциал для роста, особенно учитывая важность развития прецизионного земледелия в России.

II.2. Данные о погоде

Что может включать:

- *Температура воздуха*: измеряется в градусах Цельсия или Фаренгейта. Обычно фиксируется на высоте около 1.5 метра от поверхности земли. Измерение: Температура воздуха измеряется с помощью термометров, которые могут быть ртутными, спиртовыми или электронными. Измерения проводятся в тени, чтобы избежать влияния солнечного света.

Значение: Температура воздуха влияет на климатические условия, комфорт человека, а также на сельское хозяйство и экосистемы.

- *Влажность воздуха*: определяет количество водяного пара в воздухе и выражается в процентах. Измеряется с помощью гигрометров. Существует два основных типа: психрометр (использует два термометра, один из которых намочен) и электронные гигрометры.

Значение: Влажность влияет на восприятие температуры (ощущаемая температура), а также на образование облаков и осадков.

- *Скорость и направление ветра*: скорость ветра измеряется в метрах в секунду или километрах в час, а направление — в градусах (относительно севера). Для этого используются анемометры и флюгеры. Анемометры могут быть чашечными или лопастными, а направление измеряется с помощью флюгеров или анеморумбов.

Значение: Ветер влияет на распространение семян, пыльцы, а также на погодные условия, такие как образование штормов.

- *Объем осадков*: измеряется в миллиметрах и фиксирует количество выпавших осадков (дождя, снега и т.д.) за определенный период времени. Для этого применяются дождемеры, которые фиксируют количество выпавшей влаги за определенный период.

Значение: Осадки критически важны для экосистем, сельского хозяйства и водоснабжения.

- *Уровень солнечного излучения*: измеряет количество солнечной энергии, достигающей поверхности земли, и выражается в ваттах на квадратный метр. Используются солнечные радиометры.

Значение: Уровень солнечного излучения влияет на климат, фотосинтез растений и солнечную энергетику.

II.2.1. Способы сбора данных:

- *Метеостанции:* Автоматизированные или ручные станции, расположенные в различных местах, собирающие данные о погоде. Автоматические станции могут работать круглосуточно и передавать данные в реальном времени, тогда как ручные требуют участия человека.

Возможна установка локальных метеостанций для сбора данных о погоде. Многие из них могут передавать данные в реальном времени.

Размещение: Метеостанции могут быть расположены в городах, сельской местности, на высоте или вблизи водоемов для получения разнообразных данных.

- *Спутниковые наблюдения:* Использование спутников для мониторинга облачности, температуры поверхности и других параметров. Спутники могут фиксировать облачность, температуру поверхности океанов и изменения в атмосфере на больших территориях.

Типы спутников: Существуют метеорологические спутники, которые предоставляют данные о погоде, и спутники для климатического мониторинга.

Метеорологические спутники и спутники для климатического мониторинга играют ключевую роль в сборе данных о погоде и изменениях климата. Давайте подробнее рассмотрим их функции, типы и применение.

Метеорологические спутники предназначены для наблюдения за атмосферными явлениями и предоставления данных, необходимых для прогнозирования погоды и мониторинга климатических условий.

Основные типы метеорологических спутников:

1. Геостационарные спутники:

Принцип работы: Эти спутники находятся на высоте около 36,000 километров над экватором и вращаются с такой же скоростью, с какой Земля

вращается вокруг своей оси. Это позволяет им оставаться над одной и той же точкой на поверхности Земли.

Примеры:

GOES (Geostationary Operational Environmental Satellite) — спутники, используемые в США для наблюдения за погодой, облачностью и атмосферными системами.

Meteosat — спутники Европейской организации по метеорологии, которые обеспечивают данные для Европы и Африки.

Основные геостационарные спутники, используемые в России, относятся к серии "Метеор" и "Гонец". Давайте рассмотрим их подробнее.

Спутники серии "Метеор"

1. Метеор-3: первый спутник серии "Метеор-3" был запущен в 1991 году. Он стал первым спутником, который обеспечивал метеорологические данные в реальном времени. Его функции: предназначен для наблюдения за облачностью, температурой, атмосферным давлением и другими метеорологическими параметрами.

2. Метеор-М: спутники этой серии начали запускаться с 2009 года. Они являются более современными и технологически продвинутыми.

Функции: "Метеор-М" обеспечивает высокое качество изображений облаков, а также данные о температуре и влажности на различных высотах атмосферы. Эти спутники могут использоваться для мониторинга природных катастроф, таких как наводнения и ураганы.

Спутники серии "Гонец": представляют собой систему связи, которая также включает метеорологические функции. Они обеспечивают передачу данных и связь между различными метеорологическими станциями и центрами.

Функции: Эти спутники могут передавать информацию о погодных условиях, а также использоваться для мониторинга состояния окружающей среды.

Функции геостационарных спутников:

- Наблюдение за облачностью: Спутники фиксируют облачность и погодные явления, что позволяет метеорологам прогнозировать осадки и другие атмосферные явления.

- Измерение температуры: Геостационарные спутники могут измерять температуру поверхности земли и океана, что критически важно для понимания климатических условий.

- Мониторинг атмосферных явлений: Спутники отслеживают ураганы, тайфуны и другие опасные погодные условия, предоставляя данные в реальном времени для экстренных служб и метеорологических организаций.

- Сбор данных по всей территории: Благодаря своей геостационарной орбите, спутники могут непрерывно наблюдать одну и ту же область, что позволяет получать постоянный поток данных о погоде.

Применение данных. Данные, собранные геостационарными спутниками, используются для:

- Прогнозирования погоды на краткосрочные и среднесрочные сроки.
- Оценки климатических изменений и их влияния на экологию и сельское хозяйство.
- Уведомления о природных катастрофах и поддержания безопасности населения.

Заключение: геостационарные спутники Российской Федерации, такие как "Метеор" и "Гонец", играют важную роль в системах мониторинга погоды и климата. Они обеспечивают необходимую информацию для прогнозирования и анализа атмосферных явлений, что способствует более эффективному реагированию на изменения погоды и защиты населения от природных катастроф.

2. Полярные орбитальные спутники:

- Принцип работы: Эти спутники движутся по полярной орбите, проходя над полюсами и охватывая всю поверхность Земли. Они делают полный оборот вокруг Земли примерно за 90 минут.

Примеры:

- NOAA-20 и Suomi NPP — спутники, предоставляющие данные о температуре поверхности, облачности и других метеорологических параметрах.

- Terra и Aqua — спутники NASA, которые обеспечивают данные для изучения климата и экосистем.

Основные полярные орбитальные спутники РФ:

1. Спутники серии "Метеор-Н":

Описание: Это серия метеорологических спутников, предназначенных для наблюдения за атмосферой и климатом. Спутники "Метеор-Н" обеспечивают высококачественные данные о состоянии атмосферы и поверхности.

Функции: Они собирают информацию о температуре, влажности, облачности, уровне осадков и других метеорологических параметрах.

2. Спутники серии "Космос":

Описание: Эта серия спутников включает различные типы, некоторые из которых предназначены для метеорологических наблюдений и мониторинга окружающей среды.

Функции: Спутники "Космос" могут использоваться для наблюдения за состоянием земной поверхности, а также для мониторинга природных ресурсов и экологической ситуации.

3. Спутники серии "Ресурс":

Описание: Спутники "Ресурс" предназначены для дистанционного зондирования Земли и мониторинга природных ресурсов.

Функции: Эти спутники собирают данные о земной поверхности, включая сельское хозяйство, лесное хозяйство и водные ресурсы. Они также могут использоваться для оценки состояния экосистем.

4. Спутники серии "Гонец" (в некоторых случаях):

Описание: Хотя "Гонец" в первую очередь предназначен для связи, некоторые из них также могут выполнять функции наблюдения за окружающей средой.

Функции: Обеспечивают связь и передачу данных о погодных условиях, а также могут использоваться для мониторинга состояния окружающей среды.

5. Спутники "Канопус":

Описание: Спутники "Канопус" предназначены для дистанционного зондирования Земли и могут использоваться как для метеорологических, так и для экологических наблюдений.

Функции: Они собирают данные о состоянии растительности, водоемов и других природных ресурсов.

Преимущества полярных орбитальных спутников

- Полное покрытие: Полярные орбитальные спутники обеспечивают наблюдение за всей поверхностью Земли, включая удаленные и труднодоступные районы.

- Частота наблюдений: Они могут проходить над одной и той же точкой на Земле несколько раз в день, что позволяет получать актуальные данные о погоде и климате.

- Высокое разрешение: Многие полярные спутники оснащены высокотехнологичными сенсорами, которые обеспечивают высокое разрешение изображений и точные измерения.

Применение данных - данные, собранные полярными орбитальными спутниками, используются для:

- Прогнозирования погоды и климатических изменений.
- Мониторинга природных ресурсов и экосистем.
- Оценки состояния окружающей среды и воздействия человека на природу.
- Поддержания безопасности и реагирования на стихийные бедствия.

Заключение

Полярные орбитальные спутники Российской Федерации являются важными инструментами для метеорологических наблюдений, климатического мониторинга и дистанционного зондирования Земли. Они

обеспечивают необходимую информацию для анализа и прогнозирования погодных условий, что способствует более эффективному управлению природными ресурсами и защитой окружающей среды.

Функции метеорологических спутников:

- Наблюдение за облачностью: Спутники фиксируют облачность, что позволяет прогнозировать осадки и другие погодные явления.

- Измерение температуры: Спутники могут измерять температуру поверхности моря и земли, что важно для понимания погодных условий и климата.

- Мониторинг атмосферных явлений: Они отслеживают ураганы, тайфуны, штормы и другие опасные метеорологические явления в режиме реального времени.

- Сбор данных о влажности: Спутники могут измерять уровень влаги в атмосфере, что важно для прогнозирования осадков.

Спутники для климатического мониторинга

Описание: Спутники для климатического мониторинга специально разработаны для изучения долгосрочных изменений в климатических системах Земли и сбора данных о климатических переменных.

Основные типы спутников для климатического мониторинга:

1. Спутники для наблюдения за климатическими переменными:

Эти спутники собирают данные о температуре, влажности, уровне углекислого газа, концентрации парниковых газов и других климатических факторов.

Примеры:

OCO-2 (Orbiting Carbon Observatory-2) — спутник, предназначенный для наблюдения за углекислым газом в атмосфере.

GOSAT (Greenhouse Gases Observing Satellite) — японский спутник для мониторинга парниковых газов.

Спутники для климатического мониторинга играют ключевую роль в сборе данных о состоянии атмосферы, поверхности Земли и изменениях климата. В Российской Федерации используется несколько основных типов спутников, которые занимаются мониторингом климата. Давайте рассмотрим их подробнее:

1. *Спутники серии "Метеор"*. Описание: Это серия метеорологических спутников, предназначенных для наблюдения за состоянием атмосферы и климатом.

Функции: Спутники "Метеор" обеспечивают сбор данных о температуре, влажности, облачности, уровне осадков и других метеорологических параметрах. Они помогают в прогнозировании погоды и мониторинге климатических изменений.

2. *Спутники серии "Канопус"*. Описание: Спутники "Канопус" предназначены для дистанционного зондирования Земли и могут использоваться как для метеорологических, так и для экологических наблюдений.

Функции: собирают данные о состоянии растительности, водоемов и других природных ресурсов, что важно для оценки состояния экосистем и сельского хозяйства.

3. *Спутники серии "Ресурс"*. Описание: спутники "Ресурс" также предназначены для дистанционного зондирования Земли и мониторинга природных ресурсов. Функции: собирают данные о земной поверхности, включая информацию о сельском хозяйстве, лесном хозяйстве и водных ресурсах. Они могут использоваться для оценки состояния экосистем и мониторинга изменений ландшафта.

4. *Спутники серии "Гонец"*. - Описание: спутники "Гонец" в первую очередь предназначены для связи, некоторые из них могут выполнять функции наблюдения за окружающей средой.

Функции: Они могут передавать информацию о погодных условиях, а также использоваться для мониторинга состояния окружающей среды и передачи данных от метеостанций.

5. *Спутники серии "Спектр"*. Описание: Спутники "Спектр" предназначены для изучения атмосферы и поверхности Земли с использованием спектроскопических методов.

Функции: собирают данные о концентрации парниковых газов, аэрозолей, облаков, а также о состоянии озонового слоя, что критически важно для понимания климатических изменений.

6. *Спутники для наблюдения за изменениями в ледниках и морском льде*. Описание: некоторые спутники, такие как "Союз", предназначены для мониторинга состояния ледников и морского льда.

Функции: Они фиксируют изменения в ледниках и морском льде, что позволяет исследовать последствия глобального потепления и изменения климата.

Применение данных спутников для климатического мониторинга

- Мониторинг изменений климата: Данные помогают в оценке изменений температуры, уровня осадков и других климатических параметров на протяжении времени.

- Исследование экосистем: Спутники позволяют изучать влияние изменения климата на экосистемы, сельское хозяйство и природные ресурсы.

- Оценка воздействия человека: Данные о земной поверхности помогают оценить, как деятельность человека влияет на климат и окружающую среду.

- Прогнозирование и предупреждение: Спутники помогают в прогнозировании экстремальных погодных явлений, таких как ураганы и наводнения, что позволяет лучше готовиться к ним.

Заключение

Спутники для климатического мониторинга в России играют важную роль в понимании и анализе климатических изменений, а также в

обеспечении устойчивого использования природных ресурсов. Они предоставляют критически важные данные для метеорологических служб, ученых и государственных структур, занимающихся охраной окружающей среды и управлением ресурсами.

Спутники для наблюдения за изменениями в экосистемах:

Эти спутники фиксируют изменения в растительности, ледниках, уровнях воды и других природных ресурсах.

Примеры:

Landsat — серия спутников, которые предоставляют данные о земной поверхности, включая использование земель и изменения в экосистемах.

Sentinel — спутники программы Copernicus, предназначенные для мониторинга окружающей среды и изменений климата.

Функции спутников для климатического мониторинга:

- Долгосрочные наблюдения: Спутники собирают данные на протяжении многих лет, что позволяет исследовать изменения климата и выявлять тенденции.

- Изучение воздействия изменения климата: Данные помогают в оценке влияния изменения климата на экосистемы, сельское хозяйство и водные ресурсы.

- Мониторинг ледников и морского льда: Спутники фиксируют изменения в ледниках и морском льде, что важно для понимания глобального потепления.

- Измерение уровня моря: Спутники могут измерять изменения уровня моря, что важно для оценки рисков затопления и изменений в экосистемах.

Заключение

Метеорологические спутники и спутники для климатического мониторинга являются важными инструментами для изучения погоды и климата. Они предоставляют данные, которые помогают в прогнозировании погоды, понимании климатических изменений и разработке стратегий по адаптации к этим изменениям. С помощью этих технологий ученые и

метеорологи могут более эффективно реагировать на вызовы, связанные с изменением климата и экстремальными погодными явлениями.

Радарные системы: Позволяют отслеживать осадки и их движение.

Радарные системы в метеорологии используются для наблюдения за атмосферными явлениями, такими как осадки, грозы, ураганы и другие метеорологические условия. Они работают на основе радиолокации, которая позволяет измерять расстояние до объектов и их скорость. Вот несколько ключевых аспектов радарных систем:

1. Принцип работы

Радарная система излучает радиоволны, которые отражаются от объектов, таких как капли воды или кристаллы льда в облаках. Радар принимает отраженные сигналы и анализирует их, чтобы определить:

- Расстояние до объекта (по времени, за которое сигнал возвращается).
- Скорость объекта (по эффекту Доплера: изменение частоты сигнала в зависимости от движения объекта).
- Интенсивность осадков (по количеству отраженного сигнала).

2. Типы радаров

- Доплеровские радары: Они измеряют скорость движения осадков и могут определять направление ветра, что помогает в прогнозировании гроз и других опасных метеорологических явлений.

- Радарные системы с синтезированной апертурой (SAR): используются для более детального изучения поверхности Земли и могут работать независимо от погодных условий.

3. Применение

- Прогнозирование погоды: Радарные данные используются для создания карт осадков и прогнозов погоды.

- Мониторинг штормов и ураганов: Помогает в отслеживании и предупреждении о сильных метеорологических явлениях.

- Изучение климатических изменений: Долгосрочные данные радаров могут использоваться для анализа изменений в осадках и других климатических параметрах.

4. Системы в России

В России используются различные радарные системы, такие как:

- Метеорологические радиолокационные станции: Они обеспечивают данные для прогнозирования погоды на региональном уровне.

- Система "Радар-3000": Это одна из современных систем, применяемых для мониторинга осадков и штормов.

5. Преимущества и недостатки

Преимущества:

- Высокая точность в измерении осадков.
- Возможность отслеживания динамики атмосферных явлений в реальном времени.

Недостатки:

- Ограниченная дальность действия (обычно до 200-300 км).
- Проблемы с точностью в сложных метеорологических условиях (например, при наличии сильных отражений от рельефа).

Радарные системы играют важную роль в современной метеорологии, обеспечивая точные и своевременные данные для прогнозирования погоды и предупреждения о стихийных бедствиях.

II.2.2. Модели прогноза погоды:

Компьютерные модели, которые используют данные для прогнозирования погоды на основе физических законов.

Это сложные математические и физические модели, которые используют данные о текущем состоянии атмосферы для предсказания её будущего поведения. Эти модели основаны на физических законах, таких как уравнения Навье-Стокса, которые описывают движение жидкости (в данном случае — воздуха), а также уравнения состояния, которые связывают давление, температуру и плотность воздуха.

Основные виды моделей прогноза погоды:

1. Глобальные модели: Эти модели охватывают всю Землю и предоставляют прогнозы для больших регионов. Примеры включают модели GFS (Global Forecast System) и ECMWF (European Centre for Medium-Range Weather Forecasts). Они обычно имеют низкое пространственное разрешение, но могут делать прогнозы на длительные сроки (до 10-15 дней и более).

2. Региональные модели: Эти модели фокусируются на более узких географических областях и имеют более высокое пространственное разрешение. Примером может служить модель WRF (Weather Research and Forecasting), которая используется для детальных прогнозов в определенных регионах.

3. Модели численного прогноза: Это модели, которые используют численные методы для решения уравнений, описывающих атмосферные процессы. Они требуют значительных вычислительных ресурсов и могут быть запущены на суперкомпьютерах.

Процесс прогнозирования:

1. Сбор данных: Для начала прогноза необходимо собрать данные о текущем состоянии атмосферы. Это включает в себя информацию о температуре, влажности, давлении, скорости и направлении ветра, а также данные с метеорологических спутников и радаров.

2. Инициализация модели: Собранные данные вводятся в модель, чтобы задать начальные условия.

3. Численное моделирование: Модель решает уравнения, описывающие атмосферные процессы, на основе начальных условий. Это делается с использованием численных методов, которые позволяют оценить состояние атмосферы на будущие моменты времени.

4. Постобработка и интерпретация: Результаты численного моделирования обрабатываются, чтобы представить их в удобном для

восприятия виде. Это может включать в себя создание карт с прогнозами температуры, осадков, ветра и других параметров.

Ограничения моделей:

- Неопределенность: Атмосфера является сложной и хаотичной системой, что делает долгосрочные прогнозы менее точными. Обычно точность прогноза снижается с увеличением временного горизонта.

- Моделирование процессов: Некоторые атмосферные процессы, такие как облакообразование и конвекция, сложно моделировать, что может влиять на точность прогноза.

- Доступность данных: Качество прогноза зависит от доступных данных. В некоторых регионах может быть недостаток метеорологических станций.

Несмотря на эти ограничения, компьютерные модели прогноза погоды значительно улучшили нашу способность предсказывать погодные условия и являются важным инструментом в метеорологии.

В России для прогноза погоды используются как глобальные, так и региональные модели. Глобальные модели предоставляют данные о состоянии атмосферы на больших масштабах и играют важную роль в обеспечении метеорологических служб необходимой информацией. Вот несколько ключевых моментов о глобальных моделях погоды, используемых в России:

1. Глобальная модель ГМВ (Глобальная метеорологическая модель)

- Создатель: ГМВ разрабатывается и применяется в Российском гидрометеорологическом центре (РГМЦ).

- Функции: Модель охватывает всю планету и используется для долгосрочного прогнозирования погоды, климатического моделирования и других метеорологических задач.

- Параметризация: Модель учитывает физические процессы в атмосфере, такие как конвекция, облакообразование и радиационные процессы.

2. Модель GFS (Global Forecast System)

- Создатель: Модель GFS разрабатывается Национальной океанической и атмосферной администрацией США (NOAA).

- Использование в России: Хотя это не российская модель, GFS широко используется в российских метеорологических службах для получения дополнительных данных и валидации прогноза.

3. Модель ECMWF (European Centre for Medium-Range Weather Forecasts)

- Создатель: ECMWF — это европейская модель, которая также используется в России.

- Функции: Модель предоставляет прогнозы на среднесрочные сроки (до 10-15 дней) и известна высокой точностью.

4. Роль суперкомпьютеров

Современные глобальные модели требуют значительных вычислительных мощностей, и в России для их запуска используются мощные суперкомпьютеры. Например, в РГМЦ и других научных учреждениях страны работают суперкомпьютеры, которые позволяют обрабатывать большие объемы данных и выполнять сложные расчеты.

5. Прогнозирование погоды и климатические исследования

Глобальные модели также используются для климатических исследований, включая оценку воздействия изменения климата на различные регионы России. Это помогает в разработке адаптационных стратегий и управлении природными ресурсами.

6. Интеграция данных

Российские метеорологические службы интегрируют данные из различных глобальных моделей, чтобы улучшить точность своих прогнозов. Это включает в себя использование спутниковых данных, данных с метеорологических станций и других источников.

Заключение

Глобальные модели погоды играют важную роль в прогнозировании погоды в России. Они позволяют получать информацию о состоянии атмосферы на больших масштабах и обеспечивают основу для более детальных региональных прогнозов. Совершенствование этих моделей и использование новых технологий продолжают улучшать точность и надежность метеорологических прогнозов.

II.2.3. Полевые наблюдения:

Сбор данных вручную метеорологами или исследователями на месте.

Полевые наблюдения — это важный аспект метеорологии и климатологии, заключающийся в сборе данных о состоянии атмосферы и окружающей среды вручную метеорологами или исследователями на месте. Этот метод позволяет получить высококачественные и точные данные, которые могут дополнять или проверять данные, полученные с помощью автоматических метеорологических станций и спутников. Вот основные моменты, касающиеся полевых наблюдений:

1. Цели полевых наблюдений

- Сбор данных о погоде: Полевые наблюдения используются для измерения основных метеорологических параметров, таких как температура, влажность, давление, скорость и направление ветра, осадки и облачность.

- Исследования климатических изменений: Данные, собранные в ходе полевых наблюдений, могут помочь в понимании долгосрочных климатических изменений и их воздействия на экосистемы.

- Изучение атмосферных процессов: Полевые наблюдения позволяют исследовать локальные атмосферные явления, такие как шторма, туманы и другие метеорологические явления.

2. Методы сбора данных

- Ручные измерения: Метеорологи используют различные инструменты для ручного измерения метеорологических параметров. Например,

термометры для измерения температуры, барометры для давления, анемометры для ветра и гигрометры для влажности.

- Наблюдения за погодными явлениями: Исследователи также записывают визуальные наблюдения, такие как тип облаков, наличие осадков, видимость и другие метеорологические явления.

- Сбор образцов: В некоторых случаях проводятся сборы образцов воздуха, воды или почвы для дальнейшего анализа.

3. Инструменты и оборудование

- Метеорологические станции: В полевых условиях могут устанавливаться временные метеорологические станции, которые оборудованы необходимыми инструментами для автоматического или полуавтоматического сбора данных.

- Портативные устройства: Используются различные портативные устройства и инструменты, которые позволяют собирать данные в труднодоступных местах.

4. Преимущества полевых наблюдений

- Высокая точность: Ручные измерения могут быть более точными, чем автоматические, особенно в сложных климатических условиях.

- Контекстуальная информация: Полевые наблюдения позволяют собирать контекстуальную информацию о местных условиях, которые могут быть недоступны для автоматических систем.

- Валидация данных: Ручные наблюдения помогают проверить и откорректировать данные, полученные с помощью автоматических метеорологических станций и спутников.

5. Недостатки и ограничения

- Трудоемкость: Сбор данных вручную требует значительных временных и трудовых затрат, а также может быть ограничен погодными условиями.

- Ограниченная пространственная охват: Полевые наблюдения обычно проводятся в ограниченном географическом районе, что может не давать полной картины глобальных или региональных метеорологических условий.

- Человеческий фактор: Ошибки могут возникать из-за человеческого фактора, включая неправильные измерения или записи.

6. Применение данных

- Научные исследования: Данные, собранные в ходе полевых наблюдений, используются для научных исследований, в том числе в области метеорологии, экологии и климатологии.

- Прогнозирование погоды: Полевые наблюдения могут быть использованы для улучшения моделей прогноза погоды и повышения их точности.

- Мониторинг окружающей среды: Данные помогают отслеживать изменения в окружающей среде и оценивать влияние изменений климата на экосистемы.

Заключение

Полевые наблюдения остаются важным методом сбора данных в метеорологии и климатологии. Они обеспечивают ценную информацию, которая помогает понимать атмосферные процессы и улучшать прогнозы погоды. Несмотря на свои ограничения, полевые наблюдения дополняют автоматизированные методы и играют ключевую роль в научных исследованиях.

II.2.4. Веб-датчики и IoT:

Использование интернет-технологий для сбора и передачи данных о погоде с помощью различных сенсоров, установленных в городах и на открытых территориях.

Веб-датчики и технологии Интернета вещей (IoT) представляют собой современный подход к сбору и передаче данных о погоде с помощью различных сенсоров, установленных в городах и на открытых территориях.

Эти технологии позволяют собирать данные в реальном времени, что значительно улучшает качество метеорологических прогнозов и мониторинга. Рассмотрим подробнее основные аспекты использования веб-датчиков и IoT в метеорологии:

Что такое веб-датчики и IoT?

- Веб-датчики: Это устройства, оснащенные различными сенсорами, которые могут измерять параметры окружающей среды, такие как температура, влажность, атмосферное давление, скорость и направление ветра, уровень осадков и другие метеорологические показатели. Данные с этих датчиков могут передаваться через интернет для дальнейшей обработки и анализа.

- Интернет вещей (IoT): Это концепция, при которой физические объекты (включая веб-датчики) подключаются к интернету и могут обмениваться данными друг с другом и с центральными системами. IoT позволяет собирать, анализировать и визуализировать данные в реальном времени, что открывает новые возможности для мониторинга и управления.

2. Как работают веб-датчики и IoT?

- Сенсоры: Веб-датчики содержат различные сенсоры, которые измеряют метеорологические параметры. Например, термометры для измерения температуры, гигрометры для влажности, анемометры для скорости ветра и т.д.

- Передача данных: Собранные данные передаются через беспроводные сети (например, Wi-Fi, LTE, LoRaWAN) на облачные серверы или локальные системы. Это позволяет обеспечить доступ к данным в реальном времени из любой точки мира.

- Обработка и анализ данных: Данные, собранные с веб-датчиков, могут обрабатываться с помощью программного обеспечения для анализа и визуализации. Это может включать в себя создание графиков, карт и отчетов о текущем состоянии погоды.

3. Преимущества использования веб-датчиков и IoT

- Доступность данных в реальном времени: Веб-датчики обеспечивают возможность получения актуальной информации о погоде, что особенно важно для городского планирования, сельского хозяйства и управления чрезвычайными ситуациями.

- Расширение сети наблюдений: Установка веб-датчиков в различных местах позволяет значительно увеличить количество точек наблюдения, что улучшает пространственное покрытие и точность прогнозов.

- Экономичность: Веб-датчики часто являются более доступными по стоимости по сравнению с традиционными метеорологическими станциями, что позволяет организовать более широкую сеть наблюдений.

- Автоматизация: Процесс сбора и передачи данных автоматизирован, что снижает вероятность ошибок, связанных с ручным вводом данных.

4. Применение веб-датчиков и IoT в метеорологии

- Городское метеорологическое наблюдение: Установка датчиков в городах позволяет собирать данные о микро-климате, что полезно для управления городским планированием, контроля загрязнения и оценки влияния городской застройки на климат.

- Агрометеорология: В сельском хозяйстве веб-датчики могут использоваться для мониторинга погодных условий, что помогает фермерам принимать обоснованные решения о посеве, поливе и защите растений.

- Мониторинг экстремальных погодных явлений: Веб-датчики могут быть установлены в зонах риска (например, вблизи рек, склонов и т.д.) для мониторинга условий, способствующих наводнениям, оползням и другим опасным явлениям. Это позволяет своевременно предупреждать население и принимать меры по предотвращению бедствий.

- Научные исследования: Данные с веб-датчиков могут использоваться в научных исследованиях для изучения климатических изменений, атмосферных процессов и взаимодействия между климатом и экосистемами.

Заключение

Веб-датчики и технологии IoT открывают новые горизонты для сбора и анализа метеорологических данных. Их использование позволяет значительно улучшить качество прогнозов погоды, расширить сеть наблюдений и обеспечить доступ к актуальной информации в реальном времени. С развитием технологий и снижением стоимости сенсоров, использование веб-датчиков будет только расти, что принесет пользу как исследователям, так и обществу в целом.

Онлайн-сервисы: Используем API метеорологических служб, таких как OpenWeatherMap или WeatherAPI, для получения исторических и текущих данных о погоде.

Онлайн-сервисы для получения метеорологических данных, такие как OpenWeatherMap и WeatherAPI, предоставляют удобный доступ к информации о погоде через API (Application Programming Interface). Эти сервисы позволяют разработчикам интегрировать данные о погоде в свои приложения, сайты и системы. Рассмотрим подробнее, как работают такие API, их функциональность и применение.

1. Что такое API метеорологических служб?

API метеорологических служб — это интерфейсы, которые позволяют пользователям (разработчикам) запрашивать данные о погоде и получать их в структурированном формате (например, JSON или XML). Это упрощает интеграцию метеорологических данных в различные приложения, веб-сайты и системы.

2. Основные функции API метеорологических служб

- Текущая погода: API позволяет получать данные о текущих погодных условиях для заданного местоположения. Это может включать температуру, влажность, атмосферное давление, скорость и направление ветра, а также описание погодных условий (ясно, облачно, дождь и т.д.).

- Прогноз погоды: Многие API предоставляют возможность запрашивать краткосрочные и долгосрочные прогнозы погоды. Это может быть прогноз на несколько часов или дней вперед.

- Исторические данные: Некоторые API позволяют получать доступ к историческим данным о погоде, что полезно для анализа и исследований. Это может включать данные о температуре, осадках и других метеорологических параметрах за прошлые дни, месяцы или годы.

- Данные о климате: Некоторые сервисы предлагают доступ к климатическим данным, которые могут помочь в понимании долгосрочных тенденций и изменений климата в определенных регионах.

- Атмосферные явления: API могут предоставлять данные о специфических атмосферных явлениях, таких как осадки, штормы, ураганы и другие экстремальные погодные условия.

3. Примеры популярных метеорологических API

- OpenWeatherMap: Один из наиболее популярных метеорологических API, предлагающий доступ к текущей погоде, прогнозам, историческим данным и другим метеорологическим параметрам. OpenWeatherMap предлагает как бесплатные, так и платные тарифы, в зависимости от объема запросов и доступных функций.

- WeatherAPI: Этот API также предоставляет данные о текущей погоде, прогнозах и исторических данных. WeatherAPI предлагает простой и удобный интерфейс и поддерживает множество языков программирования.

- Weatherbit: Weatherbit API предлагает доступ к текущим и прогнозируемым данным о погоде, а также к историческим данным. Сервис также предлагает возможности для получения данных о качестве воздуха.

- Climacell (сейчас Tomorrow.io): Этот API предлагает высококачественные данные о погоде, включая точные прогнозы и данные о микроклимате, что делает его полезным для различных приложений.

4. Как использовать метеорологические API

- Регистрация: Обычно для использования API требуется регистрация на сайте сервиса. После регистрации пользователь получает API-ключ, который необходим для аутентификации запросов.

- Формирование запросов: Разработчики формируют HTTP-запросы к API, указывая необходимые параметры, такие как местоположение (город, координаты), тип запрашиваемых данных (текущая погода, прогноз и т.д.) и другие настройки.

- Получение ответов: API возвращает данные в структурированном формате (например, JSON), которые затем можно использовать в приложении. Ответ включает запрашиваемую информацию, а также может содержать дополнительные метаданные.

- Обработка данных: Разработчики обрабатывают полученные данные, чтобы отобразить их пользователям в удобном формате, например, на веб-сайте или в мобильном приложении.

5. Применение онлайн-сервисов в разработке

- Мобильные приложения: Разработчики могут интегрировать метеорологические API в мобильные приложения для предоставления пользователям актуальной информации о погоде.

- Веб-сайты: Сайты могут отображать данные о текущей погоде и прогнозах, используя API для получения информации.

- Аналитические платформы: Веб-сервисы могут использовать метеорологические данные для анализа и визуализации, что может быть полезно в различных отраслях, таких как сельское хозяйство, логистика и туризм.

- Умные устройства: IoT-устройства могут использовать данные о погоде для автоматизации процессов, например управления системами отопления или полива в зависимости от погодных условий.

6. Заключение

Онлайн-сервисы и API метеорологических служб предоставляют мощные инструменты для доступа к данным о погоде. Они позволяют

разработчикам интегрировать актуальную информацию в свои приложения и системы, улучшая пользовательский опыт и предоставляя ценную информацию о погодных условиях. С помощью таких технологий можно создать множество полезных приложений, которые помогут пользователям принимать обоснованные решения на основе погодных данных.

Каждый из этих методов имеет свои преимущества и недостатки, и часто данные собираются с использованием нескольких методов для повышения точности и надежности.

2.2.5. Примеры технологий и компаний в метеорологии:

1. Davis Instruments: Производит метеостанции для сельского хозяйства.

Общая информация:

Davis Instruments — это американская компания, основанная в 1963 году, которая специализируется на разработке и производстве метеорологических инструментов и оборудования. Компания известна своими высококачественными метеостанциями, которые используются как в профессиональных, так и в любительских целях, включая сельское хозяйство, метеорологические исследования и климатический мониторинг.

Продукты:

- Метеостанции: Davis предлагает различные модели метеостанций, такие как Vantage Vue и Vantage Pro2, которые могут измерять температуру, влажность, атмосферное давление, скорость и направление ветра, уровень осадков и солнечную радиацию. Эти станции могут работать как в автономном режиме, так и быть подключены к интернету для удаленного мониторинга.

- Датчики и аксессуары: Компания также производит дополнительные датчики для расширения функциональности метеостанций, такие как датчики температуры почвы, датчики UV-излучения и другие.

- Системы для сельского хозяйства: Продукция Davis Instruments широко используется в сельском хозяйстве для мониторинга погодных условий, что позволяет фермерам принимать обоснованные решения о поливе, защите растений и планировании сельскохозяйственных работ.

Технологии:

- Беспроводные технологии: Метеостанции Davis используют беспроводные технологии для передачи данных, что упрощает установку и использование.

- Интернет-вещей (IoT): Некоторые модели метеостанций могут подключаться к интернету, позволяя пользователям удаленно отслеживать данные и получать уведомления о погодных условиях через мобильные приложения или веб-интерфейсы.

Применение:

- Сельское хозяйство: Использование метеостанций Davis помогает фермерам оптимизировать свои процессы, снижая риски, связанные с неблагоприятными погодными условиями.

- Научные исследования: Метеостанции используются в университетах и исследовательских институтах для изучения климатических изменений и локальных атмосферных процессов.

2. OpenWeatherMap

Общая информация:

OpenWeatherMap (OWM) — это онлайн-сервис, основанный в 2014 году, который предоставляет доступ к метеорологическим данным через API. Сервис охватывает широкий спектр метеорологических параметров и предлагает как бесплатные, так и платные тарифы в зависимости от объема запросов и уровня доступа к данным.

Продукты и услуги:

- API для текущей погоды: OpenWeatherMap предлагает API, который позволяет получать данные о текущих погодных условиях для заданного

местоположения. Это включает в себя температуру, влажность, скорость ветра, атмосферное давление и описание погодных условий.

- Прогноз погоды: API предоставляет доступ к краткосрочным и долгосрочным прогнозам погоды, включая данные на 5, 16 и 30 дней вперед.

- Исторические данные: OpenWeatherMap также предлагает доступ к историческим данным о погоде, что полезно для аналитики и исследований.

- Данные о качестве воздуха: Сервис предоставляет информацию о качестве воздуха, включая уровни загрязняющих веществ, что может быть полезно для мониторинга здоровья и экологии.

Технологии:

- RESTful API: OpenWeatherMap использует RESTful архитектуру для обмена данными, что делает интеграцию с другими приложениями и системами простой и удобной.

- Поддержка различных форматов: API возвращает данные в различных форматах, включая JSON и XML, что позволяет разработчикам легко обрабатывать и интегрировать информацию в свои приложения.

Применение:

- Мобильные приложения: Разработчики могут использовать OpenWeatherMap для интеграции актуальных данных о погоде в мобильные приложения, что позволяет пользователям получать обновления о погодных условиях в реальном времени.

- Веб-сайты: Сайты могут отображать данные о погоде, используя API для получения информации о текущих условиях и прогнозах.

- Аналитические платформы: Данные OpenWeatherMap могут быть использованы для анализа и визуализации погоды, что полезно в различных отраслях, таких как транспорт, сельское хозяйство и энергетика.

Компании, такие как Davis Instruments и OpenWeatherMap, играют ключевую роль в области метеорологии, предоставляя технологии и данные, которые помогают пользователям принимать обоснованные решения на

основе погодных условий. Davis Instruments фокусируется на производстве высококачественных метеостанций для различных отраслей, тогда как OpenWeatherMap предлагает доступ к богатому массиву метеорологических данных через API, что делает информацию о погоде доступной для разработчиков и пользователей по всему миру. Эти технологии и сервисы продолжают развиваться, что открывает новые возможности для мониторинга и управления погодными условиями.

В области метеорологии и климатологии существует множество компаний, которые играют ключевую роль, предоставляя технологии, данные и услуги для прогнозирования погоды и мониторинга климатических изменений. Вот некоторые из них:

1. The Weather Company (IBM Weather)

- Общая информация: The Weather Company, часть IBM, предоставляет метеорологические данные и прогнозы для бизнеса и потребителей. Компания известна своими высококачественными прогнозами и аналитикой погоды.

- Услуги: Прогнозы погоды, данные о текущих условиях, API для разработчиков, решения для бизнеса, включая управление рисками и оптимизацию логистики.

2. AccuWeather

- Общая информация: AccuWeather — это крупная метеорологическая компания, предоставляющая прогнозы погоды и метеорологическую информацию для пользователей по всему миру.

- Услуги: Текущие условия, долгосрочные прогнозы, мобильные приложения, API для интеграции данных о погоде в другие приложения и веб-сайты.

3. MeteoGroup

- Общая информация: MeteoGroup — это международная метеорологическая компания, предлагающая услуги прогнозирования погоды

и специализированные решения для различных отраслей, включая энергетику, транспорт и сельское хозяйство.

- Услуги: Прогнозы погоды, метеорологические консультации, данные о климате, решения для управления рисками.

4. Weather Underground

- Общая информация: Weather Underground, часть The Weather Company, предлагает данные о погоде, основанные на сообщениях пользователей и метеорологических станциях.

- Услуги: Платформа для обмена данными о погоде, доступ к текущим условиям, прогнозам и историческим данным.

5. Climacell (Tomorrow.io)

- Общая информация: Climacell, переименованный в Tomorrow.io, предлагает высококачественные метеорологические данные и прогнозы, используя технологии машинного обучения и IoT.

- Услуги: Прогнозы на основе данных с сенсоров, API для разработчиков, данные о микроклимате и качество воздуха.

6. NOAA (Национальное управление океанических и атмосферных исследований, США)

- Общая информация: NOAA — это правительственное агентство США, которое занимается метеорологическими исследованиями, наблюдениями и прогнозами.

- Услуги: Открытые данные о погоде, климате, предупреждения о погодных явлениях, исследования и научные публикации.

7. EUMETSAT

- Общая информация: Европейская организация по эксплуатации метеорологических спутников (EUMETSAT) отвечает за предоставление данных с метеорологических спутников для стран Европы.

- Услуги: Спутниковые данные для мониторинга погоды и климата, продукты для прогнозирования погоды и анализа климатических изменений.

8. DTN

- Общая информация: DTN предоставляет услуги прогнозирования погоды и аналитики для различных отраслей, включая сельское хозяйство, энергетику и транспорт.

- Услуги: Специализированные прогнозы, данные о погоде в реальном времени, аналитика для бизнеса.

9. SkyMate

- Общая информация: SkyMate предлагает метеорологические услуги и технологии для морской и авиационной отраслей.

- Услуги: Прогнозы погоды для судов и самолетов, системы для мониторинга и управления погодными условиями.

10. AerisWeather

- Общая информация: AerisWeather предоставляет API и метеорологические данные для разработчиков и компаний.

- Услуги: Доступ к данным о погоде, включая текущие условия, прогнозы и исторические данные, интеграция с приложениями и системами.

Эти компании представляют собой лишь часть обширной экосистемы, работающей в области метеорологии и климатологии. Каждая из них вносит свой вклад в развитие технологий, сбор и анализ метеорологических данных, а также в предоставление услуг для различных отраслей и пользователей. С учетом изменений климата и растущих потребностей в точной метеорологической информации, роль таких компаний будет продолжать расти.

В России также есть несколько компаний и организаций, которые занимаются метеорологией и предоставлением погодных услуг. Вот некоторые из них:

1. Гидрометецентр России: Это федеральное учреждение, которое занимается мониторингом и прогнозированием погоды на территории России. Они предоставляют актуальные данные о погоде, климате и природных явлениях.

2. Метеосервис: Это коммерческая компания, которая предоставляет услуги по прогнозированию погоды, включая мобильные приложения и веб-сервисы. Она предлагает прогнозы для различных регионов России и мира.

3. Погодные сайты и приложения: Существуют различные платформы, такие как "Яндекс.Погода" и "Гисметео", которые предлагают прогнозы погоды и метеорологические данные для пользователей. Эти сервисы часто используют данные от Гидрометеоцентра и других источников.

4. Научные учреждения: В России есть несколько научных организаций, занимающихся метеорологическими исследованиями, такие как Институт физики атмосферы РАН, которые разрабатывают модели и исследуют климатические изменения.

5. Стартапы и инновационные компании: В последние годы в России появляются стартапы, которые используют новые технологии, такие как IoT и машинное обучение, для улучшения метеорологических прогнозов и сбора данных.

Эти компании и организации играют важную роль в обеспечении населения актуальной информацией о погоде и климате в России.

II.3. Данные о поливе

Что могут включать:

1. Объем воды, подаваемой в систему полива

- Этот параметр показывает, сколько воды используется для орошения растений. Он может варьироваться в зависимости от типа растений, их потребностей и климатических условий.

- Регулярный мониторинг объема воды помогает предотвратить как недостаток, так и избыток полива, что может негативно сказаться на здоровье растений.

2. Время и частота полива

- Время, в течение которого осуществляется полив, и его частота играют ключевую роль в обеспечении оптимальных условий для роста

растений. Например, полив в утренние или вечерние часы может уменьшить испарение и повысить эффективность использования воды.

- Частота полива зависит от погодных условий, типа почвы и потребностей конкретных культур.

3. Тип используемой системы орошения (капельное, дождевание и т.д.).

Существует несколько типов систем полива, включая:

- Капельное орошение: позволяет точно подавать воду к корням растений, минимизируя потери от испарения.

- Дождевание: имитирует естественный дождь и может использоваться для больших площадей.

- Подземное орошение: система, которая подает воду прямо в корневую зону растений через трубопроводы, расположенные под землей.

4. Способы сбора:

Автоматизированные системы полива могут быть оснащены датчиками и контроллерами, которые позволяют отслеживать объем подаваемой воды в реальном времени. Эти системы могут управляться с помощью программного обеспечения, что позволяет адаптировать параметры полива в зависимости от погодных условий и состояния почвы.

Примеры таких систем включают Rain Bird, Hunter и другие.

5. Системы управления поливом:

- Установка расходомеров позволяет точно измерять объем воды, подаваемой в систему орошения. Эти устройства могут быть интегрированы в системы управления поливом и предоставлять данные для дальнейшего анализа.

- Используются также датчики влажности почвы, которые помогают определить, когда необходимо проводить полив.

- Сенсоры расхода воды: Используйте расходомеры для измерения объема воды, подаваемой в систему орошения.

Установите системы автоматического полива, которые могут отслеживать объем подаваемой воды и передавать эти данные в реальном времени.

6. Обработка и анализ данных

После сбора данных важно их обработать и проанализировать. Это можно сделать с помощью:

- Платформ для аналитики: Используйте платформы для анализа данных, такие как Microsoft Azure, Google Cloud или AWS, которые предлагают инструменты для обработки больших объемов данных.

- Машинное обучение: Применяйте алгоритмы машинного обучения для предсказания потребностей в поливе на основе собранных данных.

Данные о поливе являются важным аспектом управления водными ресурсами в сельском хозяйстве и садоводстве. Они помогают оптимизировать использование воды, повышать урожайность и сохранять окружающую среду. Рассмотрим подробнее, какие данные могут быть собраны, какие методы сбора данных существуют, а также ссылки на литературу, где можно узнать больше об этих аспектах.

Систематический подход к сбору данных о поливе и их анализу позволяет значительно повысить эффективность использования воды, что особенно актуально в условиях глобального изменения климата и нехватки водных ресурсов.

13

II.3.1. Примеры технологий и компаний:

- Rain Bird: Производит системы управления поливом и автоматические контроллеры.

- Hunter Industries: Предлагает решения для автоматического полива и мониторинга.

Рассмотрим подробнее компании Rain Bird и Hunter Industries, их продукцию и технологии, которые они предлагают для систем автоматического полива.



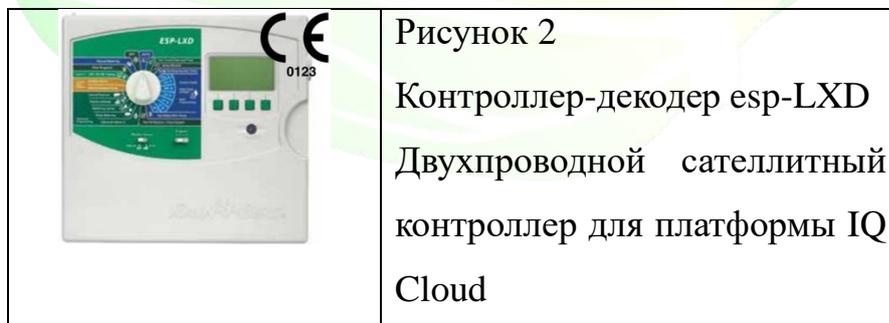
Rain Bird — один из ведущих производителей систем управления поливом и автоматических контроллеров. Rain Bird была основана в 1933 году в США и с тех пор зарекомендовала себя как один из лидеров и надежный поставщик в области орошения и управления поливом. Компания изначально разработала первый спринклер для орошения, который стал основой для дальнейших инноваций в этой области. Сегодня Rain Bird предлагает широкий ассортимент продуктов и решений, которые используются как в сельском хозяйстве, так и в ландшафтном дизайне [13].

Продукция Rain Bird:

1. Автоматические контроллеры (рис. 2):

- Настраиваемые контроллеры: Rain Bird предлагает контроллеры, которые можно настраивать под конкретные нужды пользователя. Они могут включать различные программы полива, учитывающие тип растений, сезонные изменения и погодные условия.

- Умные контроллеры: Некоторые модели могут подключаться к интернету, позволяя пользователям управлять поливом через мобильные приложения и получать данные о погоде для оптимизации графиков полива.



2. Датчики и модули:

- Датчики дождя: Эти устройства предотвращают полив во время дождя, что помогает экономить воду и предотвращает избыток влаги в почве



(рис. 3).

Датч
ики
вла
жно

сти почвы: Позволяют оценить уровень влажности в почве и автоматически регулировать полив в зависимости от потребностей растений (рис. 4).



3. Системы орошения:

- Системы капельного полива: Rain Bird предлагает решения для капельного полива, которые обеспечивают эффективное распределение воды, минимизируя потери за счет испарения и стока. Это решение обеспечивает точное распределение воды к корням растений, минимизируя потери воды за счет испарения и стока.

- Распределительные системы: включают спринклеры и форсунки, которые могут использоваться как для ландшафтного дизайна, так и для сельского хозяйства. Спринклерные системы Rain Bird предлагает широкий ассортимент спринклеров для различных нужд, включая как стационарные, так и подвижные модели.

4. Программное обеспечение:

- Rain Bird предлагает программное обеспечение для проектирования и управления системами полива, позволяющее пользователям оптимизировать свои системы и максимально эффективно использовать ресурсы.

5. Аксессуары и компоненты:

- В ассортименте Rain Bird имеются различные аксессуары, такие как клапаны, фильтры, соединительные элементы и другие компоненты, которые помогают создавать надежные и эффективные системы полива.

Инновации и устойчивое развитие

Rain Bird активно работает над инновациями в области технологий полива, стремясь к устойчивому развитию и эффективному использованию водных ресурсов. Компания подчеркивает важность экономии воды и предлагает решения, которые помогают фермерам и владельцам участков минимизировать расход воды, улучшая при этом урожайность и здоровье растений.

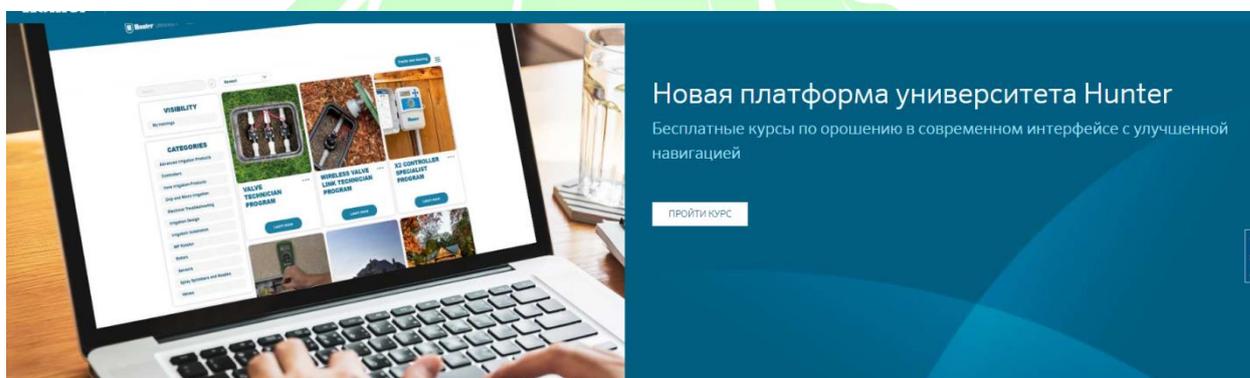
Заключение

Rain Bird помогают пользователям оптимизировать свои системы и достигать лучших результатов в садоводстве и сельском хозяйстве. С их помощью можно не только эффективно поливать растения, но и существенно

экономить ресурсы, что особенно важно в условиях глобальных вызовов, связанных с изменением климата и нехваткой воды.

Hunter Industries

Hunter Industries — компания известна своими инновациями и высоким качеством продукции. Специализируется на производстве систем автоматического полива и ландшафтного оборудования. Она была основана в 1981 году в Сан-Диего, штат Калифорния, и с тех пор зарекомендовала себя как один из ведущих производителей в этой области [14].



<https://www.hunterindustries.com/ru>

1. Системы полива: Hunter предлагает широкий ассортимент продукции для автоматического полива, включая роторные и стационарные спринклеры, капельные системы, а также контроллеры и таймеры для управления поливом (продукция Hunter Industries):

1.1. Автоматические контроллеры:

- Программируемые контроллеры: Hunter предлагает разнообразные контроллеры, которые позволяют настраивать расписание полива, учитывая тип растений и климатические условия.

- Wi-Fi контроллеры: Некоторые модели могут подключаться к Wi-Fi, что позволяет управлять системами полива через мобильные приложения и интегрироваться с метеорологическими данными.

1.2. Датчики:

- Датчики дождя и ветра: Эти устройства помогают предотвратить ненужный полив, автоматически отключая систему при дождливой или ветреной погоде.

- Датчики влажности почвы: Эти датчики могут интегрироваться с контроллерами для автоматической настройки полива на основе реального состояния почвы. SOIL-CLIK™ - система оценки влажности почвы от Hunter.

Soil-Clik – это система оценки влажности почвы, предотвращающая излишний полив, когда почва уже мокрая. Система Soil-Clik разработана для использования с контроллерами Hunter с размыкающим входящим сигналом с датчика или с любыми контролирующими системами с питанием от переменного тока и функционирует, размыкая общий провод, идущий к клапанам (рис. 5) [14].

Расположение в корневой зоне

При использовании в дерне датчик следует располагать в корневой зоне, на глубине приблизительно 15 см (это значение может быть изменено в соответствии с состоянием дерна).

При использовании на участке с деревьями и кустарником выберите более глубокое расположение в соответствии с корневой зоной. В зоне с только что высаженными саженцами располагайте датчик в родном грунте на глубине приблизительно половины корневого кома.

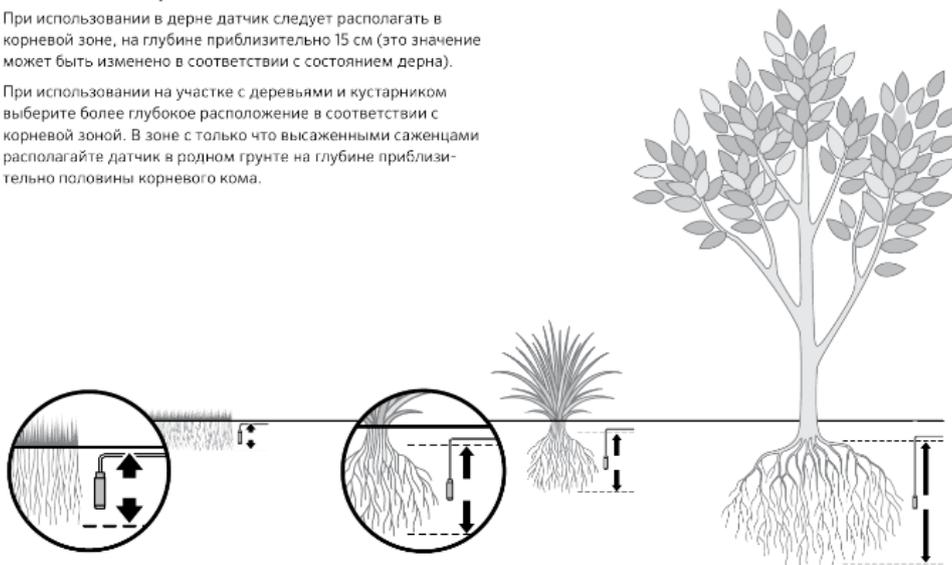


Рисунок 5. Soil-Clik – это система оценки влажности почвы.

Система Soil-Clik используется для установки желаемого уровня влажности почвы. Этот уровень можно изменить с помощью кнопок + и -. Когда желаемый уровень влажности будет достигнут, Soil-Clik прервет полив либо через входящий сигнал контроллера с датчика, либо «разомкнув» общий провод к участку полива (рис. 6) [14].



1.3. Системы орошения:

- Спринклерные системы: Hunter предлагает широкий выбор спринклеров для различных нужд, включая модели для больших земельных участков и небольших садов.

- Капельные системы: Эффективное решение для полива растений, которое минимизирует испарение и обеспечивает точное распределение воды (рис. 6) [14].

1.4. Аксессуары и компоненты:

- Hunter производит множество аксессуаров, таких как фильтры, клапаны и соединительные элементы, которые помогают создать эффективные и надежные системы полива.

Заключение

Обе компании, Rain Bird и Hunter Industries, предлагают широкий ассортимент продуктов для автоматического полива, обеспечивая пользователям возможность адаптировать системы под свои конкретные нужды. Их инновационные решения, такие как умные контроллеры и датчики, помогают оптимизировать расход воды и повышают эффективность

полива, что особенно важно в условиях современных требований к устойчивому сельскому хозяйству и экологии.

2. Ландшафтное оборудование: Компания также производит решения для управления ландшафтом, такие как дренажные системы и оборудование для управления дождевой водой.

3. Системы управления: Hunter разрабатывает продвинутые системы управления поливом, которые позволяют пользователям оптимизировать расход воды и управлять поливом удаленно, используя технологии IoT.

Инновации и технологии

Hunter Industries известна своими инновациями. Компания активно разрабатывает новые технологии, такие как системы с использованием сенсоров, которые помогают определять уровень влажности почвы и оптимизировать полив в зависимости от погодных условий. Это позволяет значительно экономить воду и повышать эффективность орошения.

Устойчивое развитие

Hunter Industries также уделяет большое внимание вопросам устойчивого развития и экологии. Они стремятся создавать продукты, которые помогают пользователям сократить потребление воды и минимизировать воздействие на окружающую среду.

Глобальное присутствие

Хотя компания была основана в США, Hunter Industries расширила свое присутствие на международном рынке и сейчас предлагает свою продукцию в более чем 100 странах мира. Она имеет дистрибьюторов и партнеров по всему миру, что позволяет ей обслуживать широкий круг клиентов, включая как частных пользователей, так и коммерческие организации.

Hunter Industries является одним из лидеров в области автоматического полива и ландшафтного оборудования, предлагая своим клиентам высококачественные и инновационные решения. С акцентом на устойчивое развитие и технологические инновации, компания продолжает оставаться на переднем крае отрасли.

Сбор данных о состоянии почвы, погоде и объеме воды для систем полива является важной частью современного сельского хозяйства. Используя технологии IoT, сенсоры и онлайн-сервисы, можно существенно повысить эффективность орошения и оптимизировать использование водных ресурсов.

II.3.2. Примеры исследований и ресурсов

- "Precision Agriculture": Это концепция, которая включает в себя использование технологий для повышения эффективности сельского хозяйства. Смотрите, например, исследования из журналов, таких как "Precision Agriculture" и "Computers and Electronics in Agriculture".

- Российские исследования: Поиск в российских научных базах, таких как eLIBRARY.RU, может предоставить доступ к статьям о современных технологиях в агрономии и применении IoT в сельском хозяйстве.



III. ЭТАПЫ РАЗРАБОТКИ АЛГОРИТМА

Этапы разработки алгоритма — это последовательные шаги, которые помогают структурировать процесс создания алгоритма для решения конкретной задачи. Эти этапы могут варьироваться в зависимости от сложности задачи и подхода к разработке, но обычно включают следующие основные шаги:

1. Определение задачи

На этом этапе необходимо чётко сформулировать задачу, которую нужно решить. Важно понять:

- Какова цель алгоритма?
- Какие входные данные будут использоваться?
- Каковы ожидаемые выходные данные?

2. Анализ требований

После определения задачи следует проанализировать требования к алгоритму:

- Какие ограничения существуют (например, по времени или памяти)?
- Каковы условия, при которых алгоритм должен работать?
- Какие возможные ошибки могут возникнуть, и как их обрабатывать?

3. Проектирование алгоритма

На этом этапе разрабатывается структура алгоритма:

- Выбор подхода к решению (например, жадный алгоритм, динамическое программирование, перебор и т.д.).
- Определение последовательности шагов, необходимых для достижения цели.
- Создание блок-схемы или псевдокода для визуализации алгоритма.

4. Реализация алгоритма

На этом этапе алгоритм переводится в код на выбранном языке программирования:

- Написание кода согласно проекту.
- Использование структур данных и библиотек, если это необходимо.

5. Тестирование алгоритма

После реализации необходимо протестировать алгоритм:

- Проверка на корректность работы с различными входными данными.
- Тестирование на крайние случаи и на больших объёмах данных.
- Устранение ошибок и оптимизация алгоритма при необходимости.

6. Документация

Создание документации, которая описывает:

- Как использовать алгоритм.
- Описание входных и выходных данных.
- Описание времени выполнения и требований к памяти.

7. Поддержка и оптимизация

После внедрения алгоритма может потребоваться его поддержка:

- Исправление обнаруженных ошибок.
- Оптимизация в случае изменения требований или появления новых данных.
- Обновление документации.

Эти этапы помогают организовать процесс разработки алгоритма и обеспечивают его эффективность и надёжность. Важно помнить, что *разработка алгоритма — это итеративный процесс*, и на любом этапе может понадобиться возвращаться назад для уточнения или изменения предыдущих шагов.

III.1. Сбор данных

Для эффективного капельного полива с использованием датчиков по влажности почвы и состава микроэлементов необходимо собирать и анализировать ряд данных. Вот основные параметры и данные, которые могут быть полезными:

1. Данные о влажности почвы

- Текущая влажность почвы: Измеряется в процентах или в м³ воды на м³ почвы. Данные помогут определить, насколько почва влажна и требуется ли полив.

- Оптимальный уровень влажности: Знание оптимального уровня влажности для конкретных культур поможет в планировании полива.

2. Данные о составе почвы

- Содержание микроэлементов: Измерения таких элементов, как азот (N), фосфор (P), калий (K), кальций (Ca), магний (Mg) и других необходимых для роста растений.

- pH почвы: Уровень pH может влиять на доступность питательных веществ для растений.

- Содержание органических веществ: Это влияет на удержание влаги и доступность питательных веществ.

3. Данные о погоде

- Температура воздуха: Влияет на испарение влаги из почвы и растений.

- Влажность воздуха: Высокая влажность может уменьшить потребность в поливе.

- Скорость и направление ветра: Ветер может увеличить испарение воды из почвы и растений.

- Осадки: Данные о количестве и частоте дождей помогут избежать избыточного полива.

- Солнечная радиация: Уровень солнечного света влияет на испарение и потребность растений в воде.

4. Данные о растениях

- Состояние растений: Здоровье и рост растений, которые могут быть оценены по визуальным признакам или с помощью датчиков.

- Потребность в воде: Разные культуры имеют разные потребности в воде в зависимости от стадии роста.

5. Данные о системе полива

- Объем подаваемой воды: Мониторинг объема воды, подаваемой через капельную систему.
- Давление воды в системе: Давление может влиять на эффективность распределения воды через капельницы.
- Эффективность полива: Оценка, насколько эффективно система полива распределяет воду по полю.

6. Исторические данные

- Исторические данные о поливе: Сравнение текущих данных с историческими показателями поможет выявить тенденции и оптимизировать полив.

Заключение

Сбор и анализ этих данных с использованием датчиков и других технологий позволит оптимизировать систему капельного полива, обеспечивая растения необходимым количеством воды и питательных веществ, что в свою очередь повысит урожайность и снизит затраты.

III.2. Предобработка данных

При анализе и моделировании систем полива овощей предобработка данных играет важную роль, поскольку качество данных напрямую влияет на результаты анализа и эффективность моделей.

Этапы предобработки данных: очистка: удаление пропусков и аномалий; нормализация: приведение данных к одному масштабу и разделение на обучающую и тестовую выборки.

Удаление пропусков:

Пропуски могут возникать по разным причинам, например, из-за ошибок при сборе данных или технических сбоев. В зависимости от объема пропусков можно использовать разные подходы:

- Удаление строк с пропусками: если пропусков немного, можно просто удалить соответствующие строки.

- Замена пропусков: можно заменить пропуски средним, медианой или модой по соответствующему признаку. Также возможно использование более

сложных методов, таких как интерполяция или imputation (например, KNN imputation).

KNN Imputation (импутация с использованием метода K ближайших соседей) — это метод заполнения пропущенных значений в наборе данных, основанный на сходстве между наблюдениями. Этот метод использует информацию от "ближайших соседей" для предсказания значений, которые отсутствуют в данных. Давайте рассмотрим основные аспекты KNN импутации.

Принцип работы KNN Imputation

1. Определение расстояний:

- Для каждой записи с пропущенным значением вычисляются расстояния до всех других записей в наборе данных. Обычно для этого используются такие метрики, как:
 - Евклидово расстояние: наиболее распространенная метрика.
 - Манхэттенское расстояние: может быть использовано в случае, если данные имеют различную шкалу.
 - Косинусное расстояние: полезно для текстовых данных или векторов.

2. Поиск K ближайших соседей:

- После вычисления расстояний выбираются K ближайших наблюдений (соседей) к текущему наблюдению, для которого необходимо заполнить пропущенное значение. Значение K — это гиперпараметр, который нужно подбирать.

3. Заполнение пропуска:

- Пропущенное значение может быть заполнено несколькими способами:
 - Среднее значение: для числовых данных пропуск может быть заменен средним значением K соседей.
 - Мода: для категориальных данных пропуск может быть заменен наиболее частым значением среди K соседей.
 - Взвешенное среднее: можно использовать взвешенное среднее, где ближайшие соседи имеют больший вес.

Преимущества KNN Imputation

- Учет локальной структуры данных: KNN импутация учитывает сходство между записями, что может привести к более точным значениям, чем простая замена средним или медианой.
- Гибкость: KNN может быть использован как для числовых, так и для категориальных данных.

Недостатки KNN Imputation

- Время вычисления: KNN может быть вычислительно затратным, особенно для больших наборов данных, так как необходимо вычислять расстояния между всеми парами наблюдений.
- Чувствительность к выбору K: Выбор значения K может значительно повлиять на результаты импутации. Слишком малое значение K может привести к переобучению, а слишком большое — к недообучению.

- Проблемы с высоко размерными данными: В высоко размерных пространствах расстояния между точками становятся менее информативными, что может снизить качество импутации.

Применение KNN Imputation

KNN импутация может быть полезной в различных областях, таких как:

- Медицинские исследования: заполнение пропусков в данных о пациентах.

- Финансовый анализ: работа с пропущенными данными о транзакциях.
- Обработка изображений: заполнение пропусков в пикселях.
- Анализ данных и машинное обучение: предобработка данных перед обучением моделей.

Пример реализации KNN Imputation

В Python для выполнения KNN импутации можно использовать библиотеку scikit-learn, которая предоставляет класс KNNImputer. Вот пример использования:

```
python
import numpy as np
import pandas as pd
from sklearn.impute import KNNImputer

# Пример данных с пропущенными значениями
data = np.array([[1, 2, np.nan],
                 [3, 4, 5],
                 [np.nan, 6, 7],
                 [8, np.nan, 9]])

# Создание DataFrame
df = pd.DataFrame(data, columns=['A', 'B', 'C'])

# Инициализация KNNImputer с K=2
imputer = KNNImputer(n_neighbors=2)

# Применение импутации
imputed_data = imputer.fit_transform(df)

# Результат
imputed_df = pd.DataFrame(imputed_data,
                           columns=df.columns)
print(imputed_df)
```

В этом примере создается набор данных с пропущенными значениями, и KNN импутация используется для заполнения этих пропусков с использованием 2 ближайших соседей. Результат выводится в виде DataFrame с заполненными значениями.

Обнаружение и удаление аномалий:

- Аномалии могут быть вызваны ошибками в измерениях или нехарактерными значениями. Для выявления аномалий можно использовать:
 - Статистические методы: такие как Z-оценка или IQR (межквартильный размах).
 - Визуализация: графики, такие как ящики с усами (box plots), могут помочь выявить выбросы.

Нормализация данных:

Нормализация — это процесс приведения данных к одному масштабу, что особенно важно, если данные имеют разные единицы измерения или диапазоны. Это позволяет улучшить работу алгоритмов машинного обучения, которые чувствительны к масштабированию данных.

Нормализация подразумевает изменение диапазонов в данных без изменения формы распределения,

Стандартизация изменяет форму распределения данных (приводится к нормальному распределению).

Обычно достаточно нормализовать данные. Например, в глубоком обучении (Deep Learning) требуется перевести цвета изображений RGB из диапазона 0-255 к диапазону 0-1.

Основные методы нормализации включают:

Min-Max Scaling;

- Z-оценка (Standardization):

- Приведение данных к нормальному распределению со средним 0 и стандартным отклонением 1

- Логарифмическое преобразование:

- Используется, если данные имеют большой разброс или экспоненциальное распределение. Логарифмирование может помочь снизить влияние выбросов.

3. Разделение данных на обучающую и тестовую выборки

Разделение данных на обучающую и тестовую выборки — это важный шаг для оценки эффективности моделей. Обычно данные делятся на две части:

- Обучающая выборка:

- Обычно составляет от 70% до 80% от общего объема данных. Эта выборка используется для обучения модели, позволяя ей находить закономерности в данных.

- Тестовая выборка:

- Составляет оставшиеся 20% до 30% данных. Эта выборка используется для проверки качества модели и ее способности обобщать на новые, невидимые данные.

Способы разделения:

- Случайное разделение:

- Данные случайным образом делятся на обучающую и тестовую выборки. Это наиболее распространенный метод.

- Стратифицированное разделение:

- Используется, если данные имеют несбалансированные классы (например, разные виды овощей). В этом случае сохраняется пропорция классов в обеих выборках.

Заключение

Предобработка данных — это критически важный этап в анализе данных и машинном обучении, особенно в контексте полива овощей. Очистка данных помогает устранить шум и искажения, нормализация позволяет привести данные к единому масштабу, а разделение на обучающую и тестовую выборки обеспечивает надежную оценку модели. Эти шаги помогают обеспечить более точные и надежные результаты при анализе и прогнозировании, что в итоге может привести к более эффективным методам полива и повышению урожайности овощей.

III.3. Выбор модели

Выберите подходящую модель машинного обучения. Для задачи управления поливом можно использовать:

- Регрессионные модели (линейная регрессия, регрессия на основе деревьев решений)

- Модели на основе нейронных сетей

- Алгоритмы, такие как Random Forest или Gradient Boosting

1. Регрессионные модели

- Линейная регрессия: Это простая и интерпретируемая модель, которая может быть использована для прогнозирования уровня полива на основе различных факторов, таких как температура, влажность, тип почвы и т.д. Однако линейная регрессия предполагает линейную зависимость между входными и выходными переменными, что может быть ограничением, если данные имеют сложные нелинейные зависимости.

- Регрессия на основе деревьев решений: Деревья решений могут захватывать как линейные, так и нелинейные зависимости. Они работают путем разбивки данных на подгруппы на основе значений признаков, что позволяет легко интерпретировать результаты. Однако деревья решений могут быть подвержены переобучению, если не использовать регуляризацию или ансамблевые методы.

2. Модели на основе нейронных сетей

- Нейронные сети: Модели глубокого обучения, такие как многослойные перцептроны (MLP), могут быть полезны для более сложных задач управления поливом, особенно если у вас есть большие объемы данных. Они способны выявлять сложные паттерны и взаимосвязи в данных. Однако они требуют большего объема данных для обучения и могут быть менее интерпретируемыми, чем более простые модели.

3. Алгоритмы, такие как Random Forest или Gradient Boosting

- Random Forest: Это ансамблевый метод, который использует множество деревьев решений для улучшения точности предсказаний. Он работает путем создания множества деревьев на случайных подвыборках данных и объединения их предсказаний. Random Forest хорошо справляется с переобучением и может обрабатывать как числовые, так и категориальные данные.

- Gradient Boosting: Этот метод также использует ансамбли деревьев, но создает деревья последовательно, где каждое новое дерево пытается исправить ошибки предыдущих. Gradient Boosting, как правило, дает более

высокую точность, чем Random Forest, но требует более тщательной настройки гиперпараметров.

Выбор модели

Выбор подходящей модели зависит от нескольких факторов:

- Объем и качество данных: Если у вас много данных с разнообразными признаками, нейронные сети могут быть хорошим выбором. Если данных меньше, то Random Forest или регрессионные модели могут работать лучше.
- Сложность задачи: Если задача требует выявления сложных паттернов, стоит рассмотреть модели на основе нейронных сетей или градиентного бустинга.
- Интерпретируемость: Если важна интерпретируемость результатов, линейная регрессия или деревья решений могут быть предпочтительными.

Заключение

Для управления поливом можно начать с простых моделей, таких как линейная регрессия или деревья решений, и постепенно переходить к более сложным методам, таким как Random Forest или нейронные сети, в зависимости от результатов и сложности данных. Важно проводить кросс-валидацию и тестирование моделей, чтобы оценить их производительность и выбрать наилучший подход для вашей задачи.

III.4. Обучение модели

Обучите выбранную модель на обучающей выборке, используя соответствующие метрики для оценки её эффективности.

Этапы обучения модели

1. Подготовка данных:

- Разделите ваши данные на обучающую и тестовую выборки. Обычно используется соотношение 70% на 30% или 80% на 20%.

- Проверьте данные на наличие пропусков и аномалий, и при необходимости выполните предобработку (например, импутацию пропусков, нормализацию или стандартизацию).

2. Выбор модели:

- В зависимости от ваших данных и цели, выберите одну из моделей, например, Random Forest или Gradient Boosting.

3. Обучение модели:

- Используйте обучающую выборку для обучения модели. В Python это можно сделать с использованием библиотеки scikit-learn.

III.5. Тестирование и валидация

Проведите тестирование модели на тестовой выборке и оцените её точность. Используйте метрики, такие как RMSE (среднеквадратичная ошибка) или MAE (средняя абсолютная ошибка).

Оценка эффективности модели:

- После обучения модели, примените её к тестовой выборке и оцените её эффективность с помощью соответствующих метрик. Для задач регрессии часто используются метрики, такие как:

- Mean Absolute Error (MAE): средняя абсолютная ошибка.
- Mean Squared Error (MSE): средняя квадратичная ошибка.
- R-squared (R^2): коэффициент детерминации.

Пример кода на Python

Ниже приведен пример кода, который иллюстрирует процесс обучения модели с использованием Random Forest для задачи управления поливом.

```
python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

# Загрузка данных
data = pd.read_csv('path_to_your_dataset.csv')

# Предположим, что в данных есть признаки и целевая переменная
X = data.drop('target_column', axis=1) # Признаки
y = data['target_column'] # Целевая переменная

# Разделение данных на обучающую и тестовую выборки
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Создание и обучение модели Random Forest
model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Прогнозирование на тестовой выборке
y_pred = model.predict(X_test)

# Оценка производительности модели
mae = mean_absolute_error(y_test, y_pred)
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f'Mean Absolute Error (MAE): {mae}')
print(f'Mean Squared Error (MSE): {mse}')
print(f'R-squared (R²): {r2}')
```

Объяснение кода

- Импорт библиотек: Мы используем pandas для работы с данными, train_test_split для разделения данных, RandomForestRegressor для создания модели и метрики для оценки производительности.
- Загрузка данных: Загрузка данных из CSV-файла.
- Разделение данных: Данные разделяются на обучающую и тестовую выборки.
- Обучение модели: Модель RandomForestRegressor обучается на обучающей выборке.
- Прогнозирование: Модель применяется к тестовой выборке для получения предсказаний.
- Оценка метрик: Рассчитываются и выводятся метрики MAE, MSE и R² для оценки эффективности модели.

Заключение

Этот процесс можно адаптировать в зависимости от ваших данных и модели. Важно экспериментировать с различными моделями и гиперпараметрами для достижения наилучшей производительности.

III.6. Разработка системы управления

На основе обученной модели создайте систему, которая будет принимать данные в реальном времени и управлять поливом в зависимости от предсказаний модели.

Разработка системы управления поливом на основе обученной модели машинного обучения включает несколько ключевых этапов. Система должна принимать данные в реальном времени, делать предсказания и управлять процессом полива. Вот основные шаги для реализации такой системы:

Этапы разработки системы управления поливом

1. Сбор данных в реальном времени:

- Настройте датчики для мониторинга условий, таких как влажность почвы, температура воздуха, уровень осадков и другие факторы, влияющие на потребность в поливе.

- Обеспечьте передачу данных от датчиков в вашу систему (например, через API или MQTT).

2. Интеграция обученной модели:

- Загрузите обученную модель в вашу систему, чтобы она могла делать предсказания на основе новых данных.

- В Python это можно сделать с помощью библиотеки `joblib` или `pickle`.

3. Предсказание и управление поливом:

- Используйте модель для предсказания потребности в поливе на основе данных, полученных от датчиков.

- Настройте систему управления поливом, чтобы включать или отключать насосы или клапаны в зависимости от предсказаний модели.

4. Мониторинг и обратная связь:

- Реализуйте систему мониторинга, чтобы отслеживать эффективность полива и корректировать модель и систему управления по мере необходимости.

- Храните данные о поливе и условиях окружающей среды для дальнейшего анализа и улучшения модели.

Пример кода для системы управления поливом

Ниже приведен упрощенный пример кода, который демонстрирует, как можно интегрировать обученную модель в систему управления поливом:

```
python
import joblib
import time
import random # Для симуляции данных от датчиков

# Загрузка обученной модели
model = joblib.load('path_to_your_trained_model.pkl')

# Функция для получения данных от датчиков (симуляция)
def get_sensor_data():
    # В реальной системе данные будут получаться от реальных датчиков
    data = {
        'soil_moisture': random.uniform(0, 100), # Уровень влажности почвы
        'temperature': random.uniform(15, 35), # Температура воздуха
        # Добавьте другие необходимые параметры
    }
    return data

# Функция для управления поливом
def control_irrigation(prediction):
    if prediction > threshold: # threshold - пороговое значение для полива
        print("Включение полива...")
        # Логика для включения полива (например, управление насосами)
    else:
        print("Полив не требуется.")

# Основной цикл системы управления поливом
threshold = 50 # Пример порогового значения
while True:
    # Получение данных от датчиков
    sensor_data = get_sensor_data()

    # Подготовка данных для модели
    input_data = [[sensor_data['soil_moisture'], sensor_data['temperature']]]

    # Предсказание потребности в поливе
    prediction = model.predict(input_data)[0]

    # Управление поливом на основе предсказания
    control_irrigation(prediction)

    # Задержка перед следующим циклом (например, 10 минут)
    time.sleep(600)
```

Объяснение кода

- Загрузка модели: Модель загружается из файла с помощью joblib.

- Получение данных от датчиков: В функции `get_sensor_data` симулируются данные, которые в реальной системе будут получаться от физических датчиков.
- Управление поливом: Функция `control_irrigation` принимает предсказание и включает или отключает полив в зависимости от порогового значения.
- Основной цикл: Бесконечный цикл, который получает данные от датчиков, делает предсказание с помощью модели и управляет поливом. Задержка между циклами позволяет контролировать частоту обновления данных.

Заключение

Эта система может быть расширена и дополнена различными функциями, такими как:

- Хранение данных в базе данных для дальнейшего анализа.
- Визуализация данных и предсказаний.
- Улучшение модели на основе новых данных (обучение с подкреплением).

В реальных условиях также важно учитывать надежность системы и возможность обработки ошибок, например, при сбое датчиков или проблемах с подключением.

III.7. Мониторинг и улучшение

После внедрения системы необходимо регулярно мониторить её эффективность и вносить коррективы в модель по мере накопления новых данных.

Мониторинг и улучшение системы управления поливом — это важные этапы, которые помогают поддерживать её эффективность и адаптироваться к изменениям условий. Здесь описаны ключевые шаги для организации мониторинга и улучшения модели.

Этапы мониторинга и улучшения

1. Сбор данных о производительности:

- Регулярно собирайте данные о работе системы: количество полива, уровень влажности почвы, погодные условия, состояние растений и т.д.- Храните эти данные в базе данных или в файлах для дальнейшего анализа.

2. Оценка эффективности модели:

- Используйте метрики, такие как MAE, MSE или R^2 , для оценки точности предсказаний модели на новых данных.

- Сравните предсказания модели с фактическими потребностями в поливе, чтобы понять, насколько хорошо она работает.

3. Анализ ошибок:

- Проанализируйте случаи, когда модель сделала неправильные предсказания. Это может помочь выявить проблемы в данных или в самой модели.

- Обратите внимание на условия, при которых происходят ошибки, и подумайте, какие дополнительные данные можно собрать для улучшения модели.

4. Обновление модели:

- По мере накопления новых данных, периодически переобучайте модель. Это поможет улучшить её точность и адаптировать к изменениям условий.

- Рассмотрите возможность использования методов активного обучения, чтобы выбирать наиболее информативные примеры для обучения.

5. Тестирование новых моделей:

- Экспериментируйте с различными моделями и гиперпараметрами, чтобы найти наилучший вариант для вашей задачи.

- Используйте кросс-валидацию для оценки производительности различных моделей и выбора наилучшей.

6. Обратная связь и улучшения:

- Внедрите механизмы обратной связи, чтобы пользователи системы могли сообщать о проблемах или предложениях по улучшению.

- Регулярно пересматривайте и улучшайте процессы сбора данных, предобработки и обучения модели.

Пример мониторинга и улучшения

Вот пример кода, который демонстрирует, как можно реализовать мониторинг и обновление модели на основе новых данных:

```
python
import pandas as pd
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
from sklearn.ensemble import RandomForestRegressor
import joblib

# Загрузка данных о производительности (например, из базы данных или CSV)
new_data = pd.read_csv('new_data.csv')

# Предположим, что в данных есть признаки и целевая переменная
X_new = new_data.drop('target_column', axis=1) # Признаки
y_new = new_data['target_column'] # Целевая переменная

# Загрузка существующей модели
model = joblib.load('path_to_your_trained_model.pkl')

# Прогнозирование на новых данных
y_pred = model.predict(X_new)

# Оценка производительности модели
mae = mean_absolute_error(y_new, y_pred)
mse = mean_squared_error(y_new, y_pred)
r2 = r2_score(y_new, y_pred)

print(f'Mean Absolute Error (MAE): {mae}')
print(f'Mean Squared Error (MSE): {mse}')
print(f'R-squared (R²): {r2}')

# Переобучение модели, если эффективность снизилась
if mae > threshold: # threshold - пороговое значение для MAE
    print("Переобучение модели на новых данных...")
    model.fit(X_new, y_new)
    joblib.dump(model, 'path_to_your_trained_model.pkl') # Сохранение обновленной модели
```

Объяснение кода

- Сбор новых данных: Загружаем новые данные, которые были собраны в ходе работы системы.
- Прогнозирование и оценка: Используем старую модель для предсказания и вычисляем метрики для оценки её производительности.
- Переобучение: Если метрики показывают, что модель работает плохо, производится переобучение на новых данных, и обновленная модель сохраняется.

Заключение

Регулярный мониторинг и корректировка модели — это жизненно важные аспекты для поддержания эффективности системы управления поливом. Важно не только следить за производительностью, но и активно реагировать на изменения в данных и условиях, чтобы система могла адаптироваться и улучшаться со временем.



IV. ПРИМЕР РЕАЛИЗАЦИИ

Пример 1.

Вот пример простого алгоритма на Python с использованием библиотеки `scikit-learn` для регрессии, который может быть использован для предсказания необходимости полива [15].

```
python
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

# Создаем пример данных
data = {
    'temperature': [22, 25, 30, 20, 18, 27, 29, 24, 23, 21],
    'humidity': [30, 45, 20, 50, 60, 35, 25, 40, 55, 70],
    'sunlight': [5, 6, 8, 4, 3, 7, 9, 5, 6, 4],
    'water_needed': [1, 1, 1, 0, 0, 1, 1, 1, 0, 0] # 1 -
    # нужно полить, 0 - не нужно
}

# Создаем DataFrame
df = pd.DataFrame(data)

# Определяем признаки и целевую переменную
X = df[['temperature', 'humidity', 'sunlight']]
y = df['water_needed']

# Разделяем данные на обучающую и тестовую выборки
X_train, X_test, y_train, y_test = train_test_split(X, y,
    test_size=0.2, random_state=42)

# Создаем и обучаем модель линейной регрессии
model = LinearRegression()
model.fit(X_train, y_train)

# Делаем предсказания
y_pred = model.predict(X_test)

# Оценка модели
mse = mean_squared_error(y_test, y_pred)
print(f'Mean Squared Error: {mse}')

# Выводим предсказания
for i in range(len(y_pred)):
    print(f'Факторы: {X_test.iloc[i].values}, Предсказание:
    {y_pred[i]}, Реальность: {y_test.iloc[i]}')
```

Объяснение кода:

1. Импортируем библиотеки: Мы импортируем необходимые библиотеки, такие как `numpy`, `pandas`, и `scikit-learn`.

2. Создаем данные: В данном примере мы создаем небольшой набор данных с тремя признаками (температура, влажность и солнечный свет) и целевой переменной (необходимость полива).
3. Создаем DataFrame: Мы используем pandas для создания DataFrame из нашего набора данных.
4. Определяем признаки и целевую переменную: X — это матрица признаков, а y — это целевая переменная.
5. Разделяем данные: Мы используем `train_test_split` для разделения данных на обучающую и тестовую выборки.
6. Создаем и обучаем модель: Мы создаем модель линейной регрессии и обучаем её на обучающей выборке.
7. Делаем предсказания: Мы используем модель для предсказания необходимости полива на тестовой выборке.
8. Оценка модели: Мы вычисляем среднюю квадратичную ошибку (MSE) для оценки точности модели.
9. Выводим предсказания: Мы выводим предсказанные значения и реальные значения для сравнения.

Этот пример является упрощённым и предназначен для иллюстрации использования линейной регрессии. В реальных сценариях данные могут быть более сложными, и для улучшения предсказаний можно использовать более сложные модели, а также проводить дополнительную предобработку данных.

Пример 2.

Создадим пример алгоритма регрессии на Python с использованием библиотеки `scikit-learn`, чтобы предсказывать необходимость полива растений на основе факторов, таких как температура, влажность, наличие удобрений и уровень солнечного света.

Пример кода

```
python
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

# Создаем пример данных
data = {
    'temperature': [22, 25, 30, 20, 18, 27, 29, 24, 23, 21],
    'humidity': [30, 45, 20, 50, 60, 35, 25, 40, 55, 70],
    'fertilizer': [1, 0, 1, 0, 1, 0, 0, 1, 1, 0], # 1 -
удобрения есть, 0 - нет
    'sunlight': [5, 6, 8, 4, 3, 7, 9, 5, 6, 4],
    'water_needed': [1, 1, 1, 0, 0, 1, 1, 1, 0, 0] # 1 -
нужно полить, 0 - не нужно
}

# Создаем DataFrame
df = pd.DataFrame(data)

# Определяем признаки и целевую переменную
X = df[['temperature', 'humidity', 'fertilizer', 'sunlight']]
y = df['water_needed']

# Разделяем данные на обучающую и тестовую выборки
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

# Создаем и обучаем модель линейной регрессии
model = LinearRegression()
model.fit(X_train, y_train)

# Делаем предсказания
y_pred = model.predict(X_test)

# Оценка модели
mse = mean_squared_error(y_test, y_pred)
print(f'Mean Squared Error: {mse}')

# Выводим предсказания
for i in range(len(y_pred)):
    print(f'Факторы: {X_test.iloc[i].values}, Предсказание:
{y_pred[i]:.2f}, Реальность: {y_test.iloc[i]}')
```

Объяснение кода:

1. Импортируем библиотеки: Мы импортируем необходимые библиотеки, такие как `numpy`, `pandas`, и `scikit-learn`.
2. Создаем данные: Мы создаем небольшой набор данных с четырьмя признаками (температура, влажность, наличие удобрений и солнечный свет) и целевой переменной (необходимость полива).
3. Создаем DataFrame: Мы используем `pandas` для создания DataFrame из нашего набора данных.
4. Определяем признаки и целевую переменную: X — это матрица признаков, а y — это целевая переменная.
5. Разделяем данные: Мы используем `train_test_split` для разделения данных на обучающую и тестовую выборки.
6. Создаем и обучаем модель: Мы создаем модель линейной регрессии и обучаем её на обучающей выборке.
7. Делаем предсказания: Мы используем модель для предсказания необходимости полива на тестовой выборке.
8. Оценка модели: Мы вычисляем среднюю квадратичную ошибку (MSE) для оценки точности модели.
9. Выводим предсказания: Мы выводим предсказанные значения и реальные значения для сравнения.

Примечание:

- В этом примере используется линейная регрессия, но для задач классификации (где результат может быть "нужно полить" или "не нужно поливать") лучше использовать алгоритмы классификации, такие как логистическая регрессия или деревья решений.
- В реальных сценариях данные могут быть более сложными и могут требовать дополнительной предобработки, чтобы улучшить качество предсказаний.

Рассмотрим такой пример.

Пример 3.

Рассмотрим формирование алгоритма машинного обучения по управлению системой полива.

Введение. Почти 40% мировых запасов продовольствия производится за счёт орошения. Во всём мире нехватка воды для орошения из-за конкуренции между промышленностью и городским потреблением угрожает продовольственной безопасности [16 с. 7].

В настоящее время экстремальные погодные явления, такие как заморозки, град, тепловые волны, процентиль осадков и периоды засухи, влияют на глобальную продовольственную безопасность, ограничивая потенциал производства неорошаемых и орошаемых сельскохозяйственных культур. Во всём мире, по оценкам, к 2050 году более 50% пахотных земель будут иметь проблемы с качеством почвы [17 с. 7 - 8].

Овощи играют важную роль в обеспечении продовольственной безопасности, но они скоропортящиеся и чувствительны к непредсказуемым и экстремальным изменениям климата [17 с. 9]. При этом планирование орошения овощных культур представляет собой некоторые уникальные проблемы из-за разнообразия видов культур, интенсивных севооборотов и количества полей, которые необходимо обрабатывать, а также конкурирующей операций в агротехнологии. Достижения в датчиках влажности почвы, беспроводной связи, измерениях ET, дистанционном зондировании, компьютерных технологиях и облачных вычислениях, машинном обучении открывают множество потенциальных возможностей для разработки надёжных консультативных инструментов по орошению, которые помогут фермерам точно определить и удовлетворить потребности овощных культур в воде. И это требует постоянных инноваций для разработки простых в использовании инструментов, которые будут широко приняты овощеводами (рис. 7 и 8). [17 с. 108, 18].

Развитие новых технологий измерения составляющих водного баланса орошаемых земель позволило использовать современные приборы и оборудование, а также базы данных дистанционного зондирования для создания системы оперативного мониторинга орошаемых земель [11].

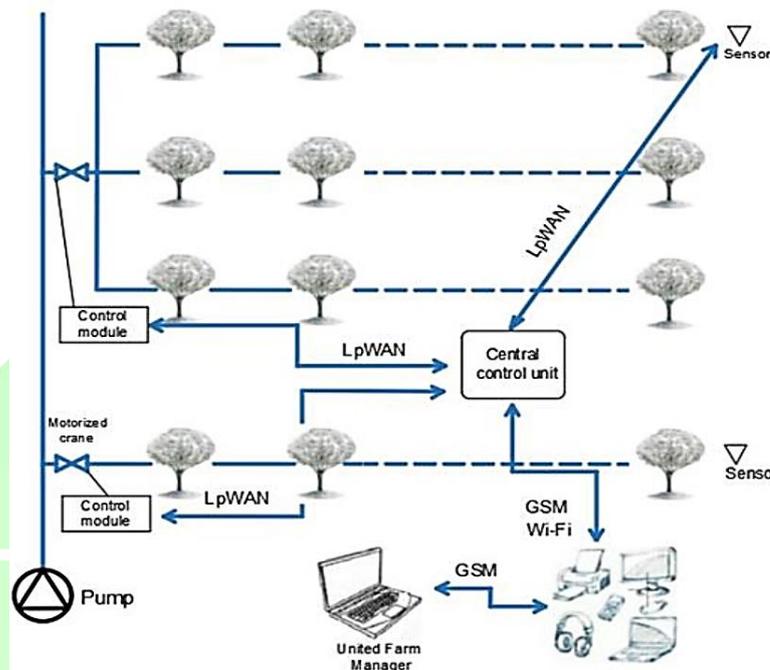


Рисунок 7. Схема орошения путем объединения участков (капустное поле) [18].



Рисунок 8. Многофункциональный датчик (sensor), установленный в поле [18].

Схема орошения для капустного поля, основанная на объединении участков, может быть организована с использованием различных методов орошения, таких как капельное орошение, дождевание или подповерхностное орошение. Капельное орошение подходит для экономии воды и обеспечения точечного полива. Устанавливаются капельные трубки

вдоль рядов капусты. При проектировании схемы орошения можно выполнить объединение участков: соединить участки для оптимального использования системы. Например, можно создать одну общую сеть для нескольких соседних участков. (рис. 7 и 8). Для контроля устанавливаем автоматические системы управления, которые будут контролировать уровень воды и время полива. Необходимо вносить изменения в график полива в зависимости от погодных условий и стадии роста капусты. Обработка полученных данных и использование машинного обучения способствуют эффективному решению данной задачи.

Материалы и методы исследований.

Тема формирования *алгоритма машинного обучения* для управления системой полива сегодня очень актуальная и важная.

С теоретической стороны машинное обучение (МО) – дисциплина, находящаяся на пересечении математической статистики, численных методов оптимизации, теории вероятностей, а также дискретного анализа. С помощью методов МО происходит решение задачи извлечения знаний из данных, которой занимается активно формирующаяся область «Интеллектуальный анализ данных» (DataMining) [19 стр. 07].

Сложность работы в сфере DataMining обуславливается неполнотой данных, которые при этом могут иметь гигантские объемы. Для их обработки требуются специальные программные средства: инструменты преобразования сырых данных в информацию, а информации в знания. Кроме того, алгоритмы обработки данных должны иметь возможность обучаться по прецедентам [19 стр. 09].

Для формирования алгоритма машинного обучения по управлению системой полива необходимы следующие шаги:

1. *Сбор данных*; 2. *Подготовка данных*: объединяем данные из разных таблиц или файлов. Выбираем наиболее важные и релевантные признаки для анализа. Это помогает упростить модель и улучшить её точность. 3. *Определяем* цель нашего алгоритма машинного обучения. Например,

оптимальное использование воды или минимизации затрат на полив. 4. *Выбор модели* машинного обучения, которая наилучшим образом подходит для нашей задачи. Некоторые популярные модели для задач управления системой полива включают решающие деревья, случайные леса или нейронные сети. 5. *Разделение данных* на обучающую и тестовую выборки. 6. *Обучение модели*: обучаем выбранную модель на обучающей выборке. 7. *Оценка модели*: оцениваем производительность модели на тестовой выборке. Используем метрики, такие как точность, полнота, F1-мера или среднеквадратичная ошибка, чтобы оценить, насколько хорошо модель выполняет задачу. 8. *Настройка и оптимизация* модели. 9. *Внедрение модели* после достижения удовлетворительных результатов в систему полива. Настраиваем систему для автоматического принятия решений на основе результатов модели. 10. *Мониторинг и обновление*: регулярно мониторим производительность модели и обновляем её при необходимости. Следим за новыми данными и изменяющимися условиями, чтобы модель оставалась актуальной и эффективной.

В технологии машинного обучения находим некоторое приближение неизвестного параметра на основе исходных данных. Это приближение позволит найти единственное лучшее приближение некоторого количества интересующих нас параметров: - один параметр; - вектор параметров (линейная регрессия); - целая функция.

Основной задачей при машинном обучении, является попытка предсказать переменную Y по заданному входному вектору X . При оценке функций необходимо нахождение приближения F с помощью модели или оценки \hat{F} , где функция оценки \hat{F} — это точечная оценка в функциональном пространстве. Необходимо, чтобы функция оценки имела малое смещение и низкую дисперсию.

При оценке максимального правдоподобия (MLE — англ. maximum likelihood estimation), из выборки данных, таких как среднее значение или дисперсия, вероятность получения наблюдаемых данных должна быть

максимальна. Так как число обучающих выборок стремиться к бесконечности, то оценка максимального правдоподобия приближается к истинному значению параметра и тем самым определяет степень сходимости. Оценка максимального правдоподобия предпочтительна для машинного обучения после определения сходимости и эффективности. В контексте машинного обучения и статистики сходимость и эффективность — это два важных свойства оценок, которые помогают определить, насколько хорошо методы работают.

1. Сходимость относится к тому, как оценка (например, параметр модели) ведет себя по мере увеличения объема выборки данных. В частности, сходимость может означать:

- Сходимость по вероятности: оценка сходится к истинному значению параметра, когда размер выборки стремится к бесконечности. То есть, по мере увеличения объема данных, оценка становится все более близкой к истинному значению.

- Сходимость по распределению: Оценка, после нормализации, имеет распределение, которое приближается к нормальному распределению, когда размер выборки велик.

- Сходимость в L2-норме: Оценка сходится в смысле среднеквадратичной ошибки, что означает, что среднее значение квадратов отклонений оценок от истинного значения стремится к нулю.

2. Эффективность оценок относится к их способности минимизировать ошибку по сравнению с другими возможными оценками. Основные моменты:

- Эффективность по Крамеру-Рао: Эффективная оценка достигает нижнего предела дисперсии, установленного теоремой Крамера-Рао. Это означает, что в классе несмещенных оценок эффективная оценка имеет наименьшую возможную дисперсию.

- Сравнение с другими оценками: Эффективные оценки обеспечивают наилучшее соотношение между смещением и дисперсией. То есть, они имеют

минимальное смещение и минимальную дисперсию среди всех возможных оценок.

- Скорость сходимости: Эффективные оценки также могут сходиться быстрее к истинному значению по сравнению с менее эффективными оценками.

Таким образом сходимость гарантирует, что с увеличением данных модель будет улучшаться, а эффективность позволяет выбирать лучшие модели и методы, которые дают наименьшие ошибки предсказания.

В обучающих сетях ошибка определяется как разница между фактическим значением y и прогнозируемым значением \hat{y} . Функция, которая используется для вычисления этой ошибки, является функцией потерь или функцией ошибки.

Функции потерь, применяемые в машинном обучении, включая среднюю квадратичную ошибку, получены из метода максимального правдоподобия.

Результаты исследований и обсуждения.

Проведем разведочный анализ (EDA — англ. Exploratory Data Analysis). Он является важным этапом в подготовке данных и позволяет получить первичное представление о свойствах и структуре данных. Важно применять различные методы и визуализации для получения полного представления о данных перед моделированием.

Метод, который применен для обучения и оптимизации модели, является *метод пакетного градиентного спуска*, который использует все обучающие данные для обновления параметров модели в каждой итерации.

В алгоритме градиентного спуска определяется ошибка для каждой итерации обучения случайных параметров модели, продолжая при этом обновлять параметры, с целью получения минимальных значений.

Рассмотрен вариант подготовки данных, полученных от модуля химического анализа.

Краткая информация по типам данных `water.info ()`.

Команда `Water.info ()` предоставляет информацию о воде в форме `DataFrame pandas` с 10 столбцами и 3276 записями (таблица 1). Сводка данных включает количество записей, количество ненулевых значений для каждого столбца и типы данных столбцов. Также в таблице 1 предоставляется информация об использовании памяти `DataFrame`:

Таблица 1

Информация о воде в форме `DataFrame pandas`

<Class 'pandas.core.frame.DataFrame'>

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3276 entries, 0 to 3275
Data columns (total 10 columns):
#   Column                Non-Null Count  Dtype
---  -
0   ph                    2785 non-null   float64
1   Hardness              3276 non-null   float64
2   Solids                3276 non-null   float64
3   Chloramines          3276 non-null   float64
4   Sulfate               2495 non-null   float64
5   Conductivity         3276 non-null   float64
6   Organic_carbon       3276 non-null   float64
7   Trihalomethanes     3114 non-null   float64
8   Turbidity            3276 non-null   float64
9   Potability           3276 non-null   int64
dtypes: float64(9), int64(1)
memory usage: 256.1 KB
```

Основные представленные показатели воды (таблица 2) – это данные о pH, жесткости воды, содержании твёрдых веществ, хлораминов, сульфатов, проводимости воды, содержании органического углерода, тригалометанов, о мутности и потребляемости (питьевой пригодности) воды.

Таблица 2

Основные показатели воды

	pH	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic	Trihalomethanes
std	1.59	32.88	8768.57	1.58	41.42	80.82	3.31	16.17
min	0.00	47.43	320.94	0.35	129.00	181.48	2.20	0.74
25%	6.09	176.85	15666.69	6.13	307.69	365.73	12.07	55.84
50%	7.04	196.97	20927.83	7.13	333.07	421.88	14.22	66.62
75%	8.06	216.66	27332.76	8.11	359.95	481.79	16.56	77.343
max	14.00	323.12	61227.20	13.13	481.03	753.34	28.30	124.00

Обсуждение.

Представленный в таблице 3 код отображает коробчатые диаграммы для каждой переменной в наборе данных (рисунок 3). Он создает сетку подграфиков 3x3, каждая из которых представляет отдельную переменную. Ось Y каждого подграфика представляет значения соответствующей переменной, а ось X представляет диапазон значений.

Ниже в таблице 3 приведен модифицированный код для создания коробчатых диаграмм:

Таблица 3

Модифицированный код для создания коробчатых диаграмм

```
python
import matplotlib.pyplot as plt
import seaborn as sns

fig, ax = plt.subplots(3, 3, figsize=(14, 14))
fig.suptitle('Outliers Analysis')

sns.boxplot(y=water['ph'], ax=ax[0, 0])
sns.boxplot(y=water['Hardness'], ax=ax[0, 1])
sns.boxplot(y=water['Solids'], ax=ax[0, 2])
sns.boxplot(y=water['Chloramines'], ax=ax[1, 0])
sns.boxplot(y=water['Sulfate'], ax=ax[1, 1])
sns.boxplot(y=water['Conductivity'], ax=ax[1, 2])
sns.boxplot(y=water['Organic_carbon'], ax=ax[2, 0])
sns.boxplot(y=water['Trihalomethanes'], ax=ax[2, 1])
sns.boxplot(y=water['Turbidity'], ax=ax[2, 2])

plt.show()
```

Анализ данных в виде коробчатых диаграмм представлен на рисунке 9 отражает тот факт, что основные данные находятся в допустимых диапазонах.

Предоставленный в таблице 4 код создает круговую диаграмму (рисунок 4) для анализа зависимой переменной «Potability» («Непригодная» (Не питьевая) и «Пригодная» (Питьевая) вода) согласно набору данных таблицы 3. Для создания диаграммы он использует библиотеку Plotly. На диаграмме отображается количество образцов в каждой категории, а процентное соотношение и метка отображаются за пределами секторов круговой диаграммы.

В таблице 4 представлен модифицированный код для создания круговой диаграммы.

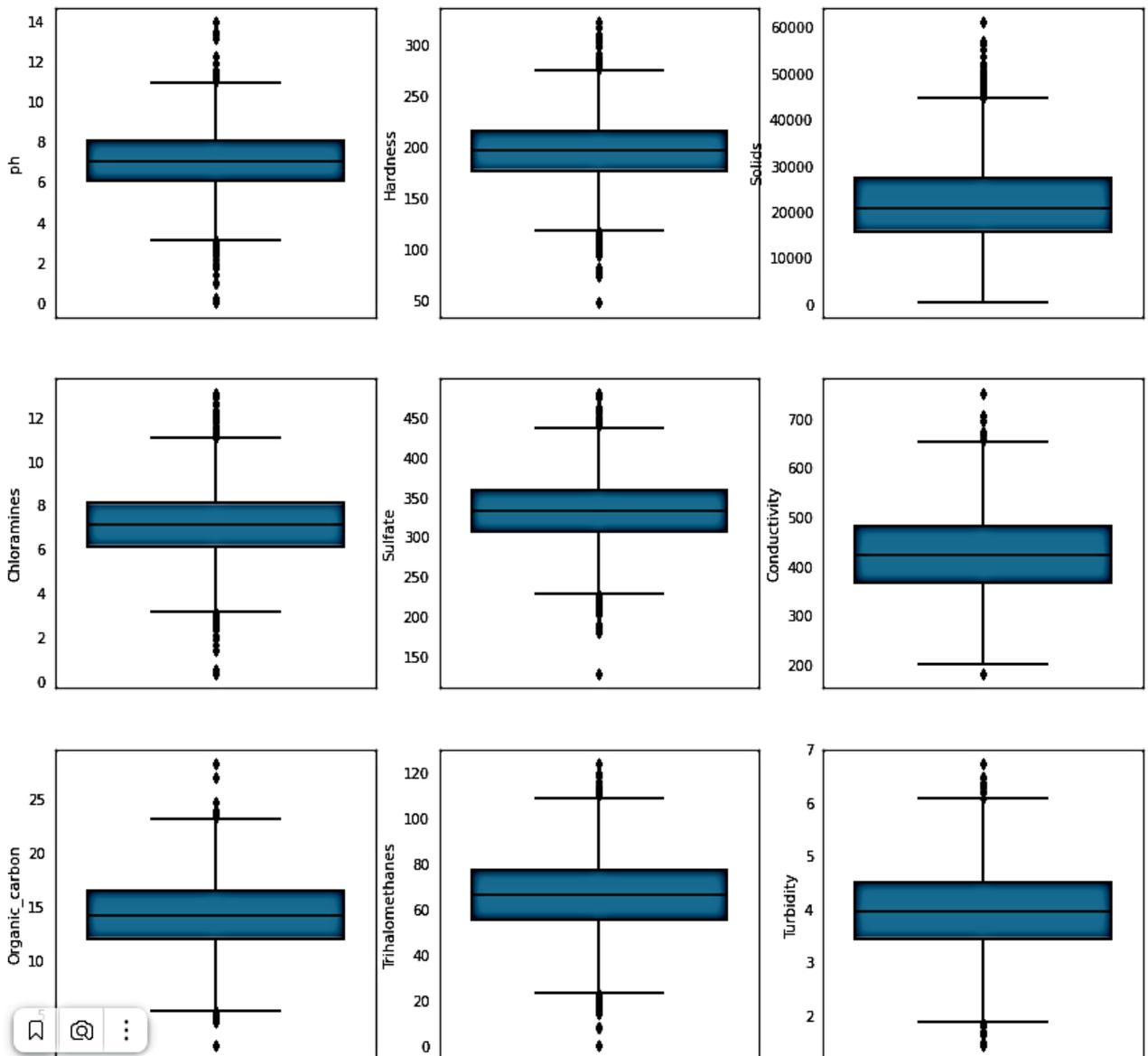


Рисунок 9. Основные данные находятся в допустимых диапазонах

Модифицированный код для создания коробчатых диаграмм

```
python
import plotly.express as px

p = water['Potability'].value_counts()

fig = px.pie(p, values="Potability", names=["Непригодная", "Пригодная"],
            hole=0.33, opacity=0.85,
            labels={"label": "Potability", "Potability": "Number of
Samples"})

fig.update_layout(title=dict(text="Pie Chart - Potability"))
fig.update_traces(textposition="outside", textinfo="percent+label")

fig.show()
```

На рисунке 10 представлена круговая диаграмма характеристик пригодности воды для питья.

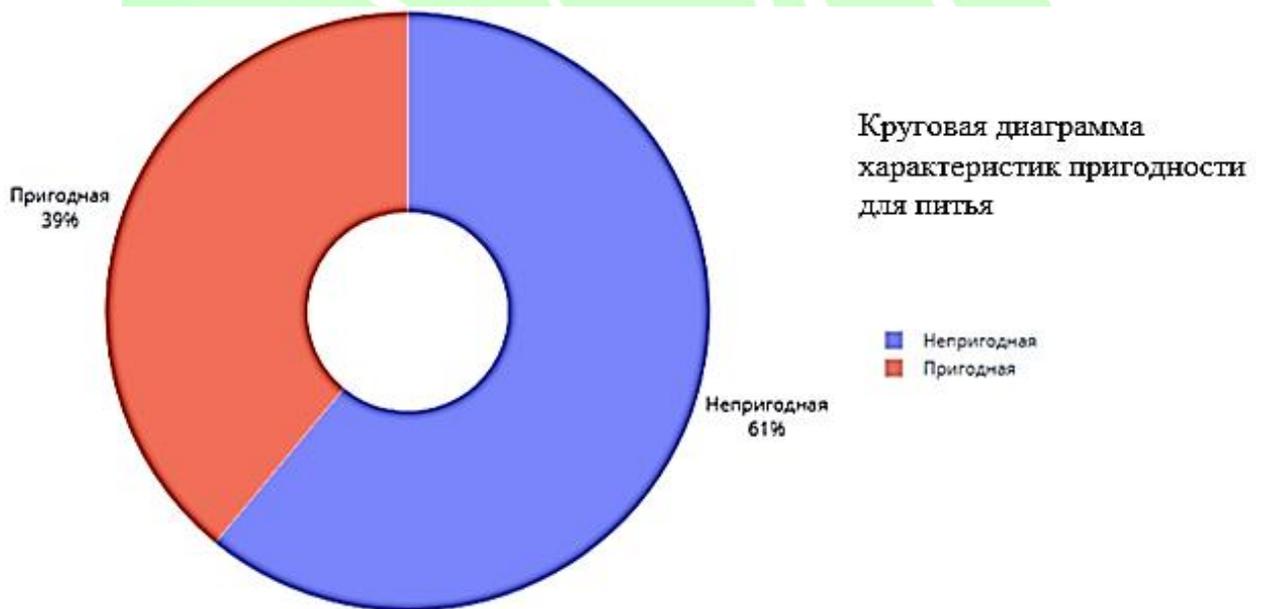


Рисунок 10. Круговая диаграмма характеристик пригодности воды для питья.

По результатам круговой диаграммы можно легко определить, как распределены образцы воды по классам пригодности для питья. Круговая диаграмма не позволяет увидеть детали внутри каждого класса, поэтому для более подробного анализа может потребоваться использование других видов визуализаций или статистических методов.

Предоставленный в таблице 5 код выполняет анализ зависимых переменных и отображает матрицу корреляции между признаками в данных о воде с использованием тепловой карты (heatmap).

Таблица 5

```
Анализ зависимых переменных, корреляция между признаками.
sns.set_style('white')
plt.figure(figsize = (12,10))
matrix = np.triu(water.corr())
sns.heatmap(water.corr(),annot = True, fmt='.1g', mask=matrix,
vmin=-1,
vmax=1, center= 0, cmap= 'coolwarm', cbar=False)
```

Анализ зависимых переменных, корреляция между признаками в данных о воде показаны на рисунке 11.

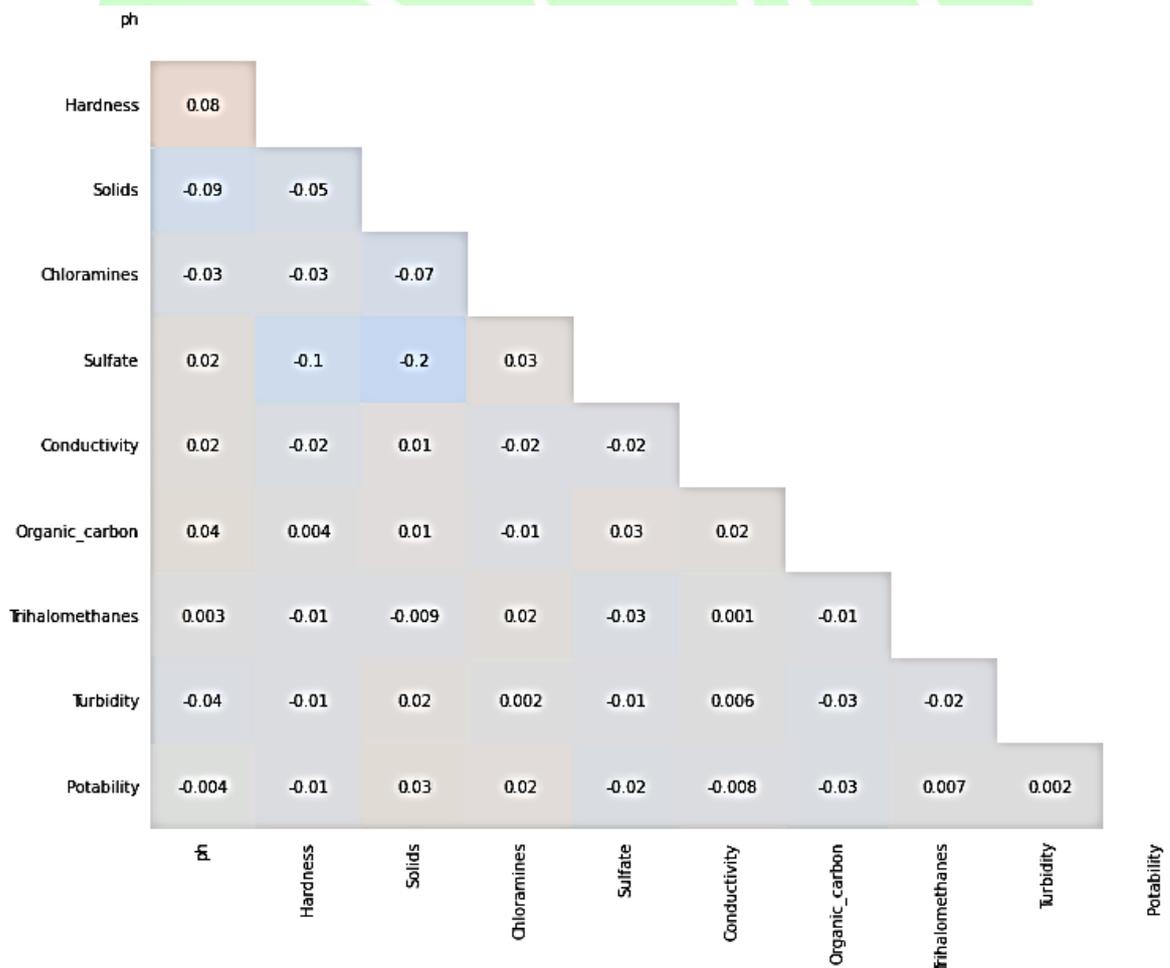


Рисунок 11. Матрица корреляции между признаками в данных о воде с использованием тепловой карты (heatmap).

Интерпретация результатов:

1. Тепловая карта (heatmap) показывает корреляцию между всеми парами признаков в наборе данных о воде. Каждая ячейка на карте представляет собой коэффициент корреляции между соответствующими признаками.

2. Значения коэффициента корреляции могут варьироваться от -1 до 1.

- Корреляция близкая к 1 означает положительную линейную зависимость между признаками.

- Корреляция близкая к -1 означает отрицательную линейную зависимость.

- Корреляция около 0 означает отсутствие линейной зависимости между признаками.

Обратим внимание на сильные корреляции (близкие к 1 или -1), так как они могут свидетельствовать о наличии линейной зависимости между признаками, что может потребовать дополнительного анализа при построении моделей машинного обучения.

3. Чем ярче цвет ячейки на тепловой карте, тем сильнее корреляция между соответствующими признаками.

- Если на тепловой карте есть блоки ярких цветов (какие-то признаки сильно коррелируют между собой), это может указывать на наличие мультиколлинеарности, что может быть проблемой при моделировании.

- После анализа корреляции можно рассмотреть исключение сильно коррелирующих признаков, чтобы избежать проблемы мультиколлинеарности.

- Также можно использовать информацию о корреляции для выбора наиболее информативных признаков для построения модели машинного обучения.

Величина коэффициента корреляции между переменными не больше 0,20 - слабая корреляция и в расчете не участвует.

Код (рис.12а) создает столбчатую диаграмму (countplot) для переменной "Potability" из набора данных "water".

Функция countplot из библиотеки Seaborn используется для отображения количества наблюдений в каждой категории переменной,

относящихся к различным значениям переменной "Potability" (питьевая пригодность воды).

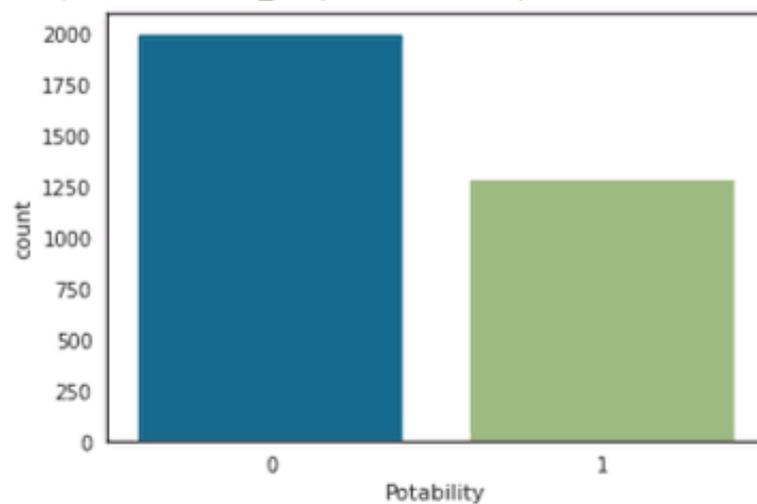
В столбчатой диаграмме наблюдается дисбаланс классов (как и в круговой диаграмме) - выполняем анализ данных и принимаем соответствующие меры.

Код `water.nunique ()` (рис. 12) используется для вычисления количества уникальных значений в каждом столбце данного набора данных. Результат показывает, сколько различных значений содержится в каждом признаке. Например, столбец "Potability" имеет 2 уникальных значения, что может означать, что это бинарный признак рисунок 12.

```
fig = plt.figure(figsize = (6,4))  
sns.countplot(x=water['Potability'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f0037d37050>
```

a)



б)

Рисунок 12. Столбчатая диаграмма (countplot) для переменной "Potability" из набора данных "water".

```

water.nunique ()
  ph          2785
  Hardness    3276
  Solids       3276
  Chloramines  3276
  Sulfate      2495
  Conductivity 3276
  Organic_carbon 3276
  Trihalomethanes 3114
  Turbidity    3276
  Potability   2
  dtype: int64    9:48 ✓✓

```

Рисунок 13. Код water.nunique()

Код `msno.matrix(water)` относится к библиотеке `missingno`, которая предназначена для визуализации пропущенных значений в наборе данных рисунок 8.

```

msno.matrix(water)
plt.show()    9:49 ✓✓

```

Рисунок 14. Код msno.matrix(water)

Функция `msno.matrix()` создает матричный график (рисунок 15), где строки представляют отдельные точки данных, а столбцы - признаки. Пропущенные значения отображаются белыми линиями, в то время как наличие значений отображается цветными линиями. Это помогает быстро оценить, где в данных присутствуют пропущенные значения.

Использование этих кодов помогает лучше понять набор данных, выявить проблемы с пропущенными значениями и принять соответствующие меры по их обработке.

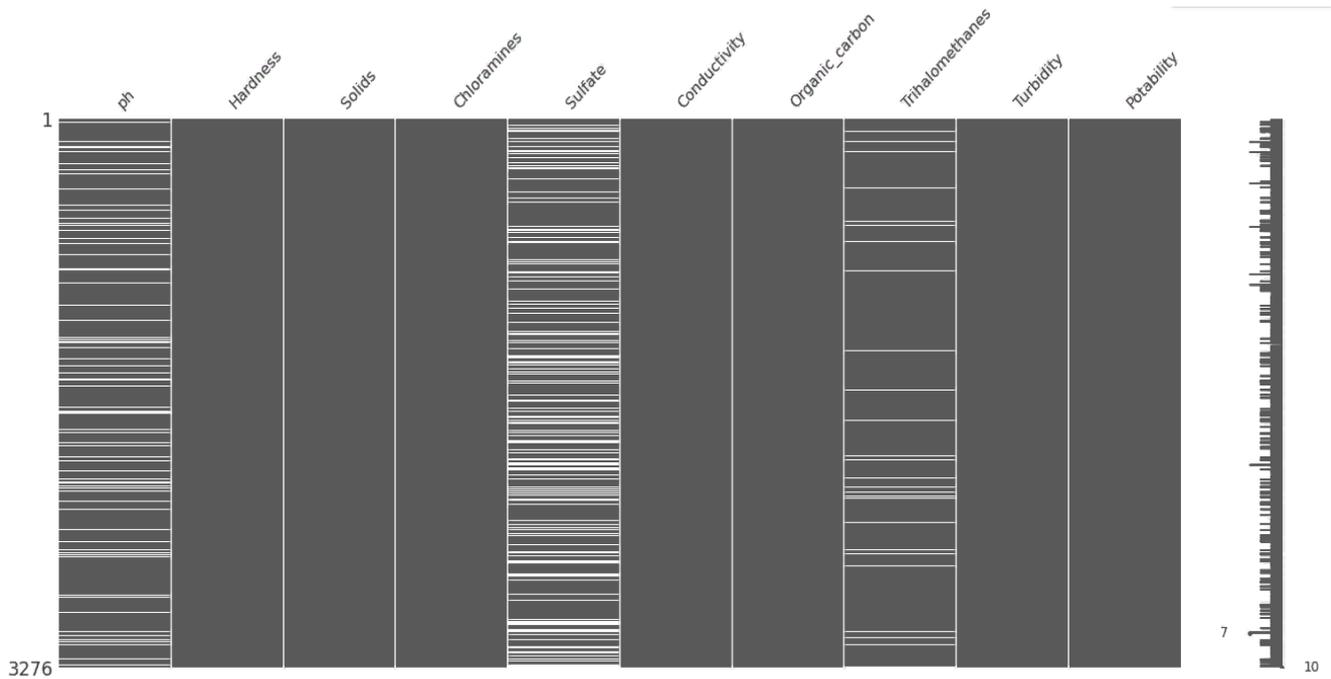


Рисунок 15. Матричный график функции `msno.matrix()`

В столбцах pH, Sulfate, Trihalomethanes матричного графика функции `msno.matrix()` имеются отсутствующие значения, проводим исправление (рис. 16).

Заполняем отсутствующие значения в столбцах "ph", "Sulfate" и "Trihalomethanes" их средними значениями. Используя среднее значение для заполнения пропущенных данных, обеспечиваем сохранение общего распределения данных.

Варианты визуализации результатов представлены на рисунках 17а и 17 б.

```

ph          491
Hardness    0
Solids      0
Chloramines 0
Sulfate     781
Conductivity 0
Organic_carbon 0
Trihalomethanes 162
Turbidity   0
Potability  0
dtype: int64

```

```

water["ph"].fillna(value = water["ph"].mean(), inplace = True)
water["Sulfate"].fillna(value = water["Sulfate"].mean(), inplace = True)
water["Trihalomethanes"].fillna(value =
water["Trihalomethanes"].mean(), inplace = True)

```

Рисунок 16. Исправление матричного графика функции `msno.matrix()`

Вывод. Этот код (рис. 17) выполняет визуализацию с использованием ядерной оценки плотности для сравнения распределений значений признаков между питьевой и не питьевой водой. Он может быть улучшен с помощью других типов визуализаций для более наглядного представления результатов. Выбор конкретного типа визуализации зависит от целей и вопросов, которые необходимо исследовать. Нас интересует различие в распределении значений признаков между питьевой и не питьевой водой, `boxplot` может быть более подходящим вариантом.

Далее переходим к подготовке данных для машинного обучения по трем вариантам (1. Random Forest; 2. Градиентный бустинг; 3. XGBoost) [6].

Материал представлен в части 2 статьи.

```

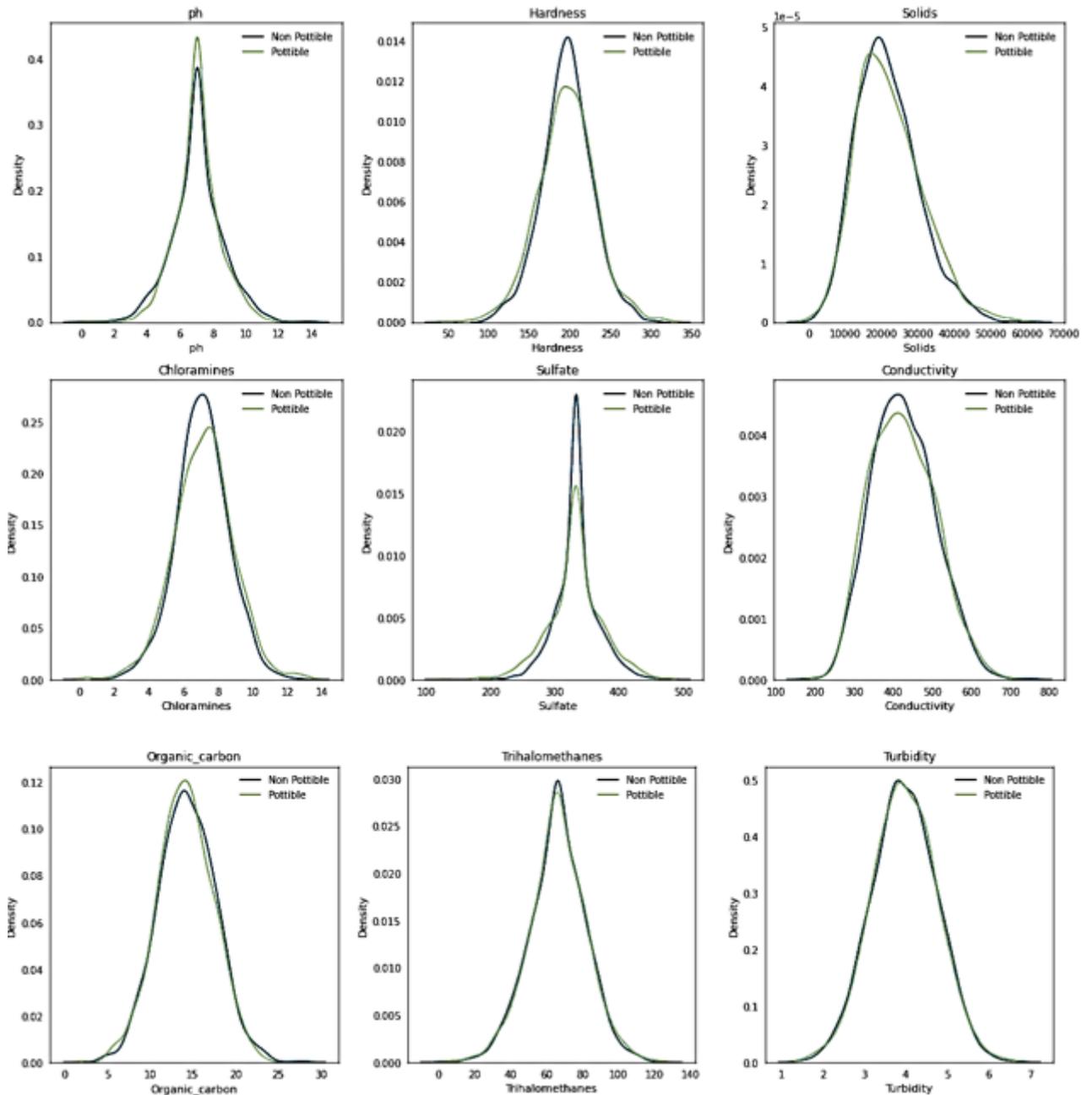
Варианты визуализации результатов
non_pottible = water[water['Potability'] == 0]
pottible = water[water['Potability'] == 1]
plt.figure(figsize=(15,15))

for ax,col in enumerate(water.columns[:9]):
    plt.subplot(3,3,ax+1)
    plt.title(col)
    sns.kdeplot(x=non_pottible[col],label = "Non Pottible")
    sns.kdeplot(x=pottible[col],label = "Pottible")
    plt.legend()

plt.tight_layout()

```

9:54 ✓ a)



б)

Рисунок 17. Варианты визуализации результатов

Использование машинного обучения для управления системой полива может значительно повысить эффективность использования водных ресурсов и улучшить здоровье растений. В данном пособии мы рассмотрели основные этапы разработки алгоритма и привели пример реализации. Вы можете адаптировать и расширять этот алгоритм в зависимости от ваших конкретных условий и данных.

Это учебное пособие можно использовать как основу для дальнейшего изучения и разработки систем управления поливом с использованием методов машинного обучения.

Мы получили набор данных, выявили и решили проблемы с пропущенными значениями и приняли соответствующие меры по их обработке [11,12].

Далее рассматриваем подготовку данных для машинного обучения по трем вариантам: 1. Random Forest; 2. Градиентный бустинг; 3. Xgboost.

1. *Random Forest*

- Обработка пропусков: заполните или удалите пропуски в данных, так как Random Forest может работать с некоторыми пропусками, но лучше избегать их.

- Кодирование категориальных переменных: Преобразуйте категориальные переменные в числовые с помощью методов, таких как one-hot encoding или label encoding.

- Масштабирование данных: обычно не требуется, но может быть полезно для улучшения интерпретируемости.

- Разделение на обучающую и тестовую выборки: разделите данные на обучающую и тестовую выборки (например, 80/20), чтобы оценить производительность модели.

2. *Градиентный бустинг*

- Обработка пропусков: как и в случае с Random Forest, пропуски нужно обработать, но градиентный бустинг может лучше справляться с некоторыми типами пропусков.

- Кодирование категориальных переменных: Используйте one-hot encoding или другие методы кодирования для категориальных переменных, так как многие реализации градиентного бустинга не поддерживают категориальные данные напрямую.

- Масштабирование данных: может быть полезно, особенно если используете алгоритмы, чувствительные к масштабу, но не является обязательным.

- Разделение на обучающую и тестовую выборки: обязательно разделите данные, чтобы избежать переобучения.

3. PyCaret

- Импорт данных: PyCaret упрощает процесс подготовки данных, начиная с импорта данных в DataFrame.

- Обработка пропусков: PyCaret автоматически обрабатывает пропуски, предлагая варианты, такие как заполнение средним или медианой.

- Кодирование категориальных переменных: PyCaret автоматически кодирует категориальные переменные, что упрощает процесс.

- Масштабирование данных: PyCaret предлагает автоматическое масштабирование, и вы можете выбрать, хотите ли вы масштабировать данные или нет.

- Разделение на обучающую и тестовую выборки: PyCaret автоматически разделяет данные на обучающую и тестовую выборки, что упрощает процесс работы с моделями.

Эти шаги помогают подготовить данные для машинного обучения с использованием каждого из этих методов.

Результаты исследований и обсуждения.

Выполним несколько шагов по подготовке данных для машинного обучения, включая разделение данных на обучающую и тестовую выборки, а также нормализацию.

Разделение данных на признаки и целевую переменную.

Создаем матрицу признаков `X`, удаляя столбец "Potability" из DataFrame `water`. Целевая переменная `y` — это значения столбца "Potability" (рис. 18).

Используем функцию `train_test_split` из библиотеки `sklearn`, чтобы разделить данные на обучающую и тестовую выборки. Выполняем проверку размеров выборок и нормализуем данные (рис. 18).

Подготовка данных для машинного обучения

```
python
from sklearn.model_selection import train_test_split
import numpy as np

X = water.drop("Potability", axis=1).values
y = water["Potability"].values

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=1)

print("X - обучающий набор:", X_train.shape)
print("X - тестовый набор:", X_test.shape)
print("y - обучающий набор:", y_train.shape)
print("y - тестовый набор:", y_test.shape)

# Нормализация данных
X_min = np.min(X_train)
X_max = np.max(X_train)
X_train = (X_train - X_min) / (X_max - X_min)
X_test = (X_test - X_min) / (X_max - X_min)
```

Рисунок 18. Разделение данных на признаки и целевую переменную

При использовании алгоритмов машинного обучения, таких как 1. Random Forest, 2. Градиентный бустинг и 3. Русарет, важным шагом является подготовка данных. Необходимо нормализовать данные перед обучением моделей. Пользователь заполняет пропущенные значения в столбцах средними значениями, визуализирует распределение данных и подготавливает данные для машинного обучения (рис. 19).

1. Подготовка данных для машинного обучения *Random Forest*:

```

1. Случайный лес
from sklearn.ensemble import RandomForestClassifier
models = [("DTC", DecisionTreeClassifier(max_depth = 3)),
          ("RF", RandomForestClassifier())]
finalResults = []
cmList = []
for name, model in models:
    model.fit(X_train, y_train) # train
    model_result = model.predict(X_test) # prediction
    score = precision_score(y_test, model_result)
    cm = confusion_matrix(y_test, model_result)

    finalResults.append((name, score))
    cmList.append((name, cm))
finalResults
[('DTC', 0.711864406779661), ('RF', 0.6974789915966386)]

Визуализация дерева решений
for name, i in cmList:
    plt.figure()
    sns.heatmap(i, annot = True, linewidths = 0.8, fmt = ".1f")
    plt.title(name)
    plt.show()

```

Рисунок 19. Процесс обучения и оценки моделей машинного обучения

Random Forest

Код на рисунке 2 демонстрирует процесс обучения и оценки моделей машинного обучения, а именно классификаторов: дерева решений и случайного леса (Random Forest). Визуализацию дерева решений можно увидеть на рис. 20.

В конце в списке `finalResults`, имеем следующие результаты (имена моделей и их точности) (рис. 3):

- DTC (Decision Tree Classifier) имеет точность 0.7119 (или 71.19%).
- RF (Random Forest Classifier) имеет точность 0.6975 (или 69.75%).

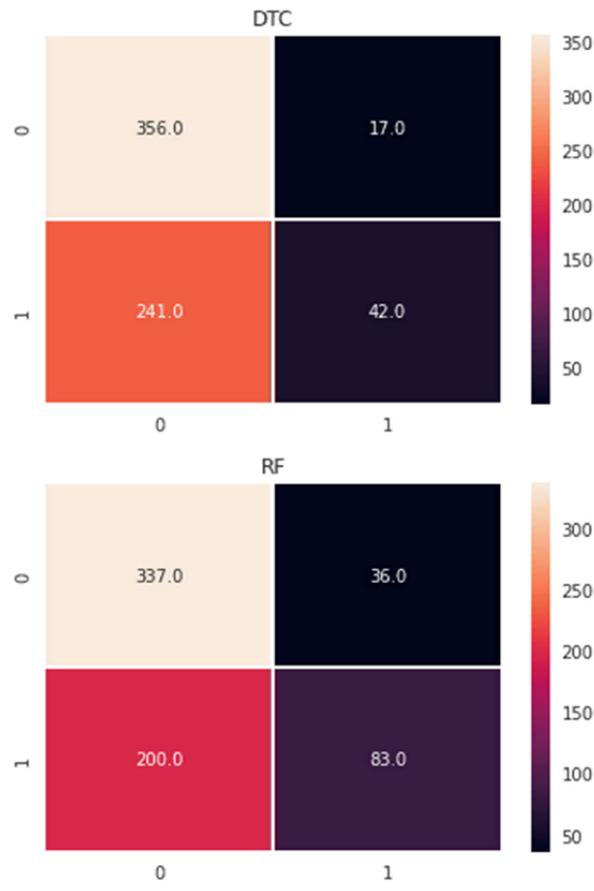


Рисунок 20. Матрицы путаницы

2. Градиентный бустинг (Gradient Boosting) —мощный метод машинного обучения, который используется для решения задач регрессии и классификации. Он основан на концепции ансамблевого обучения, где несколько слабых моделей объединяются для создания более сильной модели. В контексте градиентного бустинга слабые модели обычно представляют собой простые модели, такие как решающие деревья (decision trees), которые имеют низкую производительность, когда используются в одиночку. Таким образом, решающее дерево — это простая модель, которая принимает решения, основываясь на последовательных вопросах о значениях признаков (например, "больше ли значение X, чем 5?"). Это делает их легкими для понимания и интерпретации (рис. 21 и 22).

Эти модели называются "слабыми" или "слабыми учениками" (weak learners) из-за их высокой предрасположенности к ошибкам по нескольким причинам:

1. переобучение: небольшие деревья (например, деревья с глубиной 1-3, известные как "stumps") могут быть чувствительны к шуму в данных и могут переобучаться на ограниченном наборе данных. Они могут хорошо работать на обучающем наборе, но плохо обобщать на новых данных.

2. низкая предсказательная мощность: в одиночку простые деревья могут не захватывать сложные зависимости в данных. Например, одно дерево может не иметь возможности моделировать нелинейные взаимодействия между признаками.

3. комбинирование моделей: градиентный бустинг использует идею, что, комбинируя множество слабых моделей, можно создать мощную модель, которая будет более устойчивой и точной, чем любая из отдельных моделей. Каждое следующее дерево обучается на ошибках предыдущих, что позволяет постепенно улучшать качество предсказаний.

Основная идея заключается в том, чтобы последовательно обучать модели на ошибках предыдущих моделей, улучшая предсказания на каждом шаге.

Основные концепции градиентного бустинга:

1. Слабые модели: обычно используются простые модели, такие как решающие деревья (decision trees).

2. Обучение на остатках: Каждая новая модель (или "бустер") обучается на остатках (ошибках) предыдущих моделей. Таким образом, каждая новая модель пытается исправить ошибки предыдущих.

3. Градиентный спуск: метод использует градиентный спуск для минимизации функции потерь. Это позволяет находить оптимальные параметры для каждой модели, чтобы минимизировать общую ошибку.

4. Комбинирование моделей: все модели объединяются с помощью взвешенной суммы, чтобы сделать окончательное предсказание. Веса могут быть равными или зависеть от производительности каждой модели. При этом отмечаем легкость в комбинировании: различные слабые модели (в данном случае, решающие деревья) могут быть легко объединены для создания более

сильной модели. Каждое новое дерево добавляется к ансамблю, и его задача — исправить ошибки предыдущих деревьев.

Преимущества градиентного бустинга:

- Высокая точность: градиентный бустинг часто показывает отличные результаты и может outperform более простые методы.

- Гибкость: метод может использоваться с различными функциями потерь, что делает его универсальным для разных задач.

- Устойчивость к переобучению: при правильной настройке параметров (например, регуляризации) градиентный бустинг может быть менее подвержен переобучению. Склонность к переобучению: если дерево слишком глубокое, оно может запомнить данные, вместо того чтобы обобщать. Однако, если вы ограничите глубину дерева (например, с помощью параметра `max_depth`), вы получите модель, которая будет иметь высокую предрасположенность к ошибкам, но в то же время будет достаточно гибкой для захвата некоторых закономерностей в данных.

Недостатки градиентного бустинга:

- Выбор гиперпараметров: градиентный бустинг имеет много гиперпараметров, которые могут быть сложными для настройки.

- Время обучения: обучение может занять много времени, особенно на больших наборах данных, поскольку каждая новая модель обучается последовательно.

Популярные реализации:

- XGBoost: оптимизированная библиотека для градиентного бустинга, которая предоставляет высокую производительность и гибкость.

- LightGBM: быстрая и эффективная библиотека, разработанная для работы с большими данными.

- CatBoost: библиотека, специально разработанная для работы с категориальными признаками, без необходимости предварительного кодирования.

Таким образом, идея состоит в том, чтобы использовать множество слабых моделей для создания одного сильного предсказателя, который будет лучше, чем каждая из отдельных моделей. Это и есть суть градиентного бустинга: поэтапное улучшение модели за счет обучения на ошибках предыдущих моделей.

```

from sklearn.ensemble import GradientBoostingClassifier
gb = GradientBoostingClassifier(n_estimators=100, learning_rate=1.0,
                               max_depth=3, random_state=0)
gb.fit(X_train,y_train)
y_pred = gb.predict(X_test)
gb_acc = accuracy_score(y_test, y_pred)

print ("The accuracy for GradientBoostingClassifier is", gb_acc)
print (classification_report(y_test, y_pred))

The accuracy for GradientBoostingClassifier is 0.5807926829268293
precision recall f1-score support

 0   0.61   0.71   0.66   373
 1   0.52   0.42   0.46   283

accuracy          0.58   656
macro avg   0.57   0.56   0.56   656
weighted avg   0.57   0.58   0.57   656

GBC_cm = confusion_matrix(y_test, y_pred)
sns.heatmap(GBC_cm/np.sum(GBC_cm), annot = True, fmt = '0.2%',
            cmap = 'Blues')
<matplotlib.axes._subplots.AxesSubplot at 0x7f00379c6c10>

```

Рисунок 21. Градиентный бустинг

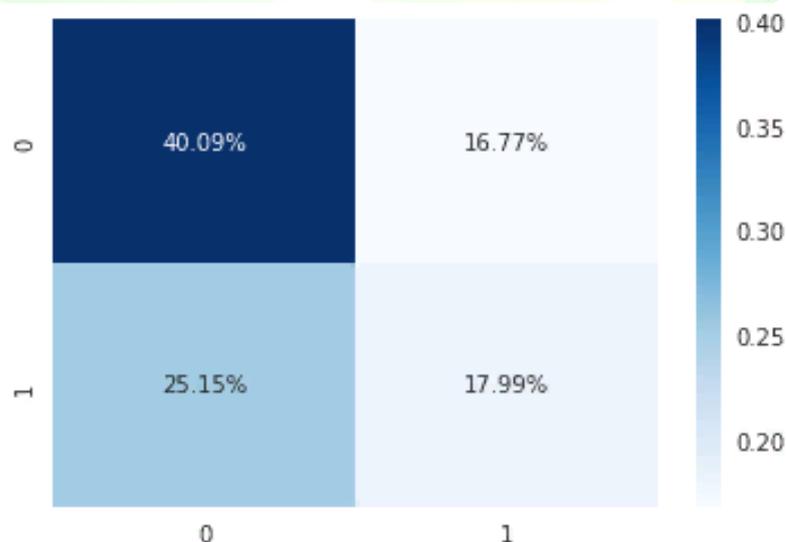


Рисунок 22. Матрица путаницы

3. PyCaret — это мощная библиотека Python для автоматизации процессов машинного обучения. Она значительно упрощает задачи подготовки данных, выбора моделей, их обучения и оценки.

Используем PyCaret для подготовки данных для моделирования и предсказаний. Он разделяет данные на две части: 95% для моделирования и 5% для предсказаний. Затем обновляет индексы и выводит информацию о разбиении данных (рис. 23).

```

Оставляем в датасете 95% данных для дальнейшего
моделирования
water_pc = water.sample(frac=0.95,random_state=1)
Оставляем 5% данных от датасета для предсказаний
water_pc_us = water.drop(water_pc.index)
Обновляем индексы
water_pc.reset_index(inplace=True, drop=True)
water_pc_us.reset_index(inplace=True, drop=True)

Проверяем разбиение
Print ('Исходные данные: ' + str(water.shape))
Print ('Данные для моделирования: ' + str(water_pc.shape))
Print ('Данные для предсказаний: ' + str(water_pc_us.shape))
water_pc.head(2)

```

Исходные данные: (3276, 10)
Данные для моделирования: (3112, 10)
Данные для предсказаний: (164, 10)

ph	Hardness Turbidity	Solids Potability	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes
0	6.063355 69.038454	160.767311 3.467337	14775.145596 0	7.484104	6.220936	305.828553	327.270239 12.309016
1	7.080795 51.284401	233.020134 4.284879	27071.118618 0	6.220936	6.220936	298.112645	357.119622 16.768945

```

water_pc_us.head(2)

```

ph	Hardness Turbidity	Solids Potability	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes
0	6.347272 75.071617	186.732881 4.376348	41065.234765 0	9.629596	9.629596	364.487687	516.743282 11.539781
1	NaN 4.133423	227.435048 0	22305.567414	10.333918	Naan	554.820086	16.331693 45.382815

Рисунок 23. PyCaret для подготовки данных для моделирования и предсказаний

Разделили данные на части для моделирования и предсказаний и данные готовы для использования в модели машинного обучения с помощью `PyCaret`.

Код использует функцию `setup` из библиотеки `PyCaret` для подготовки данных к моделированию. Это мощный инструмент, который автоматизирует процесс предобработки данных и настройки модели. После выполнения `setup` можно использовать другие функции `PyCaret` для обучения моделей, их оценки и выбора наилучшей модели для данных.

Настроили обработку данных и моделирование с использованием `PyCaret`. Решили проблему дисбаланса классов, используя метод `SMOTE`, и теперь создаем модель с помощью `Extra Trees Classifier (ETC)` с помощью команды `ETC = create_model('et')`.

Эта настройка позволяет эффективно обрабатывать отсутствующие значения, нормализовать данные и применять различные преобразования перед обучением моделей машинного обучения. Использование `SMOTE` для решения проблемы дисбаланса классов — хорошая практика для улучшения производительности модели на несбалансированных наборах данных.

Созданием модели `Extra Trees Classifier` с использованием библиотеки `PyCaret` (рис. 24).

Создаем модель `Extra Trees Classifier`

```
ETC = create_model('et')
```

Результаты метрик для модели на различных фолдах выглядят следующим образом (таблица 3):

```

mod=setup(water_pc, target = 'Potability', session_id=234,
silent=True, fix_imbalance=True)

  Description Value
0 session_id 234
1 Target Potability
2 Target Type Binary
3 Label Encoded None
4 Original Data (3112, 10)
5 Missing Values False
6 Numeric Features 9
7 Categorical Features 0
8 Ordinal Features False
9 High Cardinality Features False
10 High Cardinality Method None
11 Transformed Train Set (2178, 9)
12 Transformed Test Set (934, 9)
13 Shuffle Train-Test True
14 Stratify Train-Test False
15 Fold Generator StratifiedKFold
16 Fold Number 10
17 CPU Jobs -1
18 Use GPU False
19 Log Experiment False
20 Experiment Name clf-default-name
21 USI d03e
22 Imputation Type simple
23 Iterative Imputation Iteration None
24 Numeric Imputer mean
25 Iterative Imputation Numeric Model None
26 Categorical Imputer constant
27 Iterative Imputation Categorical Model None
28 Unknown Categoricals Handling least_frequent
29 Normalize False
30 Normalize Method None
31 Transformation False
32 Transformation Method None
33 PCA False
34 PCA Method None
35 PCA Components None
36 Ignore Low Variance False
37 Combine Rare Levels False
38 Rare Level Threshold None
39 Numeric Binning False
40 Remove Outliers False
41 Outliers Threshold None
42 Remove Multicollinearity False
43 Multicollinearity Threshold None
44 Remove Perfect Collinearity True
45 Clustering False
46 Clustering Iteration None
47 Polynomial Features False
48 Polynomial Degree None
49 Trigonometry Features False
50 Polynomial Threshold None
51 Group Features False
52 Feature Selection False
53 Feature Selection Method classic
54 Features Selection Threshold None
55 Feature Interaction False
56 Feature Ratio False
57 Interaction Threshold None
58 Fix Imbalance True
59 Fix Imbalance Method SMOTE

```

Рисунок 24. Создаем модель Extra Trees Classifier

Результаты метрик для модели на различных фолдах

Таблица 3

Fold	Accuracy	AUC	Recall	Precision	F1	Kappa	MCC
0	0.6651	0.7154	0.4706	0.5882	0.5229	0.2698	0.2738
1	0.6789	0.7184	0.4405	0.6167	0.5139	0.2840	0.2929
2	0.6284	0.6534	0.4881	0.5190	0.5031	0.2068	0.2071
3	0.5963	0.6086	0.4167	0.4730	0.4430	0.1285	0.1291
4	0.6651	0.7310	0.5000	0.5753	0.5350	0.2754	0.2770
5	0.6651	0.6753	0.4762	0.5797	0.5229	0.2687	0.2718
6	0.6101	0.5981	0.3929	0.4925	0.4371	0.1446	0.1467
7	0.6422	0.6446	0.5000	0.5385	0.5185	0.2345	0.2349
8	0.6037	0.6323	0.3690	0.4844	0.4189	0.1265	0.1292
9	0.6728	0.7319	0.5238	0.5867	0.5535	0.2966	0.2978
Mean	0.6428	0.6709	0.4578	0.5454	0.4969	0.2235	0.2260
Std	0.0294	0.0482	0.0484	0.0482	0.0440	0.0640	0.0649

Результаты и обсуждения

Средние значения метрик показывают, что модель имеет среднюю точность и полноту. Есть возможность улучшить модель, проведя дополнительную настройку гиперпараметров или анализируя важность признаков для улучшения результатов (рис. 25).

Plot model(ETC,scale=1.3)

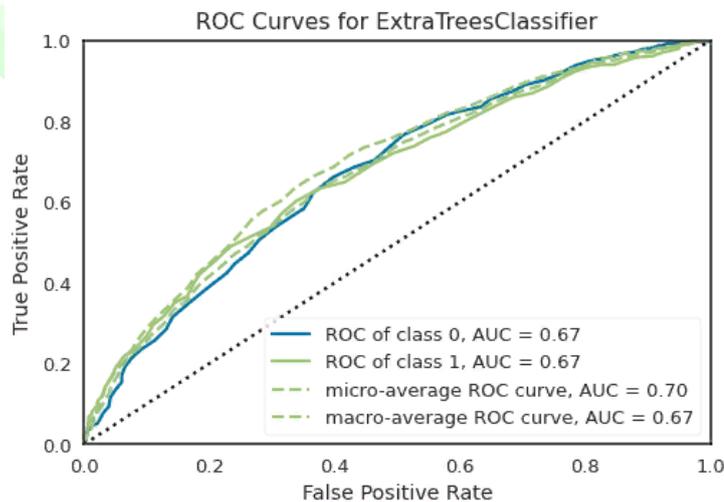


Рисунок 25. График, значимых параметров Ph для модели.

Чтобы создать график значимости параметров Ph для вашей модели, необходимо получить список важных параметров и их важности, а затем построить график, отображающий эту информацию.

Для анализа значимости параметров (фичей) в модели можно использовать график важности признаков (рис. 26). График показывает, какие параметры оказывают наибольшее влияние на прогнозы модели. Обычно важность признаков измеряется с помощью метрик, таких как Gini importance или SHAP values.

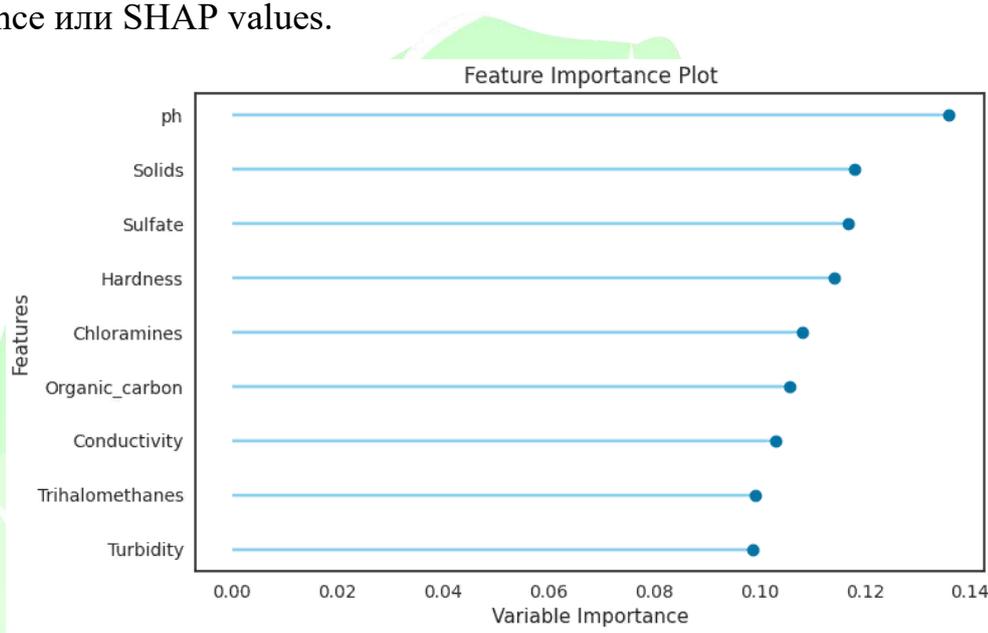


Рисунок 26. График важности признаков

Далее создаем модель Random Forest

`Classifier RFC = create_model('rf')`.

Таблица 4

Модель классификатора случайного леса (Random Forest Classifier) с использованием PyCaret

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
Fold							
0	0.5917	0.6759	0.4588	0.4756	0.4671	0.1364	0.1364
1	0.6606	0.7086	0.4524	0.5758	0.5067	0.2536	0.2578
2	0.5963	0.6386	0.4762	0.4762	0.4762	0.1478	0.1478
3	0.5963	0.6026	0.4405	0.4744	0.4568	0.1363	0.1366
4	0.6743	0.7110	0.5238	0.5867	0.5535	0.2984	0.2996
5	0.6651	0.7021	0.5595	0.5663	0.5629	0.2915	0.2915
6	0.6239	0.6261	0.4048	0.5152	0.4533	0.1729	0.1758
7	0.6330	0.6461	0.5000	0.5250	0.5122	0.2184	0.2185
8	0.6083	0.6237	0.5238	0.4944	0.5087	0.1835	0.1837
9	0.6866	0.7244	0.6071	0.5930	0.6000	0.3425	0.3425
Mean	0.6336	0.6659	0.4947	0.5282	0.5097	0.2181	0.2190
Std	0.0338	0.0415	0.0575	0.0459	0.0467	0.0708	0.0709

Мы создали модель классификатора случайного леса (Random Forest Classifier) с использованием PyCaret. Метрики оценки модели показывают производительность модели на различных выборках данных. Средние значения дают общее представление о производительности модели (таблица 4).

Точность Accuracy - доля правильно предсказанных наблюдений (как положительных, так и отрицательных) к общему числу наблюдений;

AUC (Area Under the Curve) - площадь под кривой ROC (Receiver Operating Characteristic). AUC измеряет способность модели различать классы. Значение AUC варьируется от 0 до 1, где 1 означает идеальную модель, а 0.5 — случайное угадывание. Чем выше AUC, тем лучше модель;

полнота (Recall) (чувствительность) — это доля правильно предсказанных положительных наблюдений к числу фактически положительных наблюдений,

точность (Precision) — это доля правильно предсказанных положительных наблюдений к общему числу предсказанных положительных наблюдений,

F1-мера — гармоническое среднее между точностью и полнотой. F1-мера полезна, когда необходимо учитывать как ложные срабатывания, так и пропуски - баланс между этими двумя метриками;

Карра (Cohen's Kappa) — это статистическая мера, которая оценивает согласие между двумя оценщиками (или между моделью и истинными метками), учитывая случайное согласие. Значение Карра варьируется от -1 до 1, где 1 означает идеальное согласие, 0 - случайное согласие, а отрицательные значения указывают на худшее согласие, чем случайное.

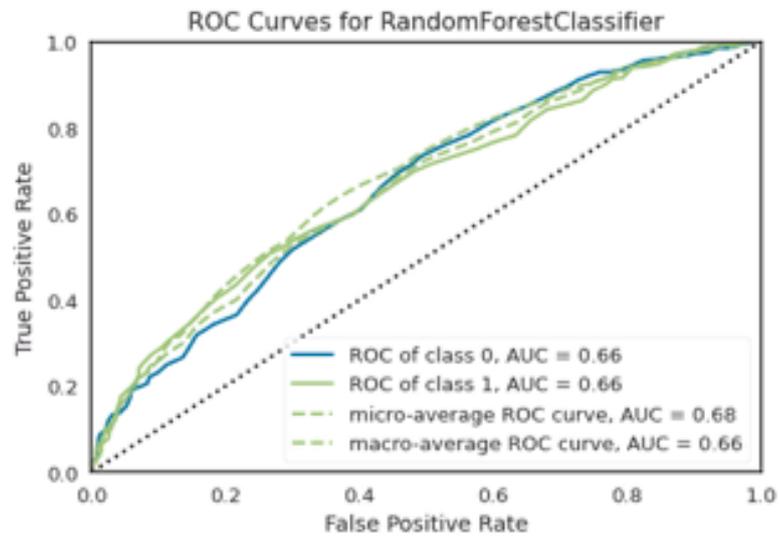
MCC (Matthews Correlation Coefficient) — это мера, которая учитывает все четыре компонента матрицы путаницы (TP - истинно положительные, TN - истинно отрицательные, FP - ложно положительные, FN - ложно отрицательные) и может быть использована как для сбалансированных, так и для несбалансированных наборов данных. Значение MCC варьируется от -1 до 1, где 1 означает идеальное предсказание, 0 — случайное предсказание, и -1 — полное несогласие.

Эти метрики помогают оценить качество моделей машинного обучения и выбрать наиболее подходящие в зависимости от конкретной задачи.

Исходя из метрик, считаем, что модель показывает умеренно хорошие результаты, с возможностью улучшения.

Функция `plot_model` позволяет визуализировать различные аспекты модели, такие как важность признаков, границы принятия решений и другие. Построив график модели случайного леса, можно получить представление о том, как модель принимает решения и какие признаки более важны для прогнозирования (рис. 27).

```
plot_model(RFC,scale=1.3)
```



```
plot_model(tuned_blend,scale=1.3)
```

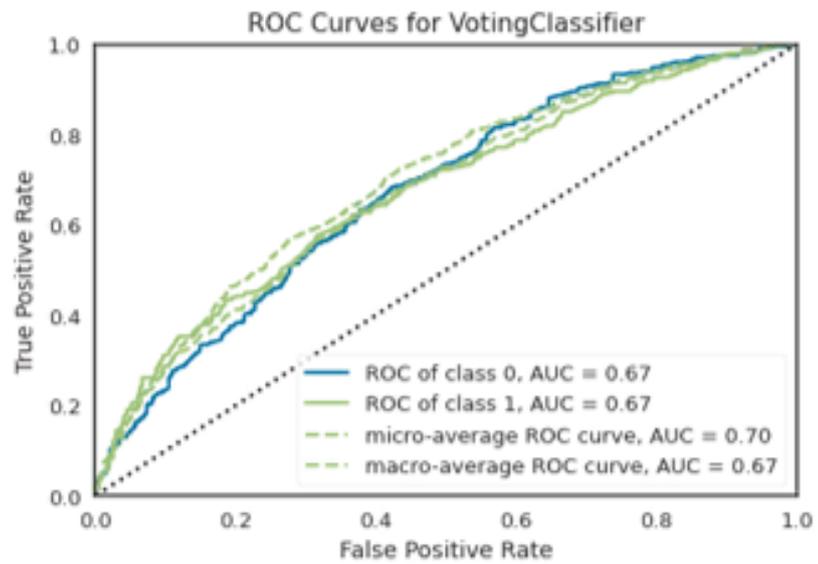


Рисунок 27. График модели случайного леса

График Precision(Точность) - Recall(Полнота) (рис. 28).

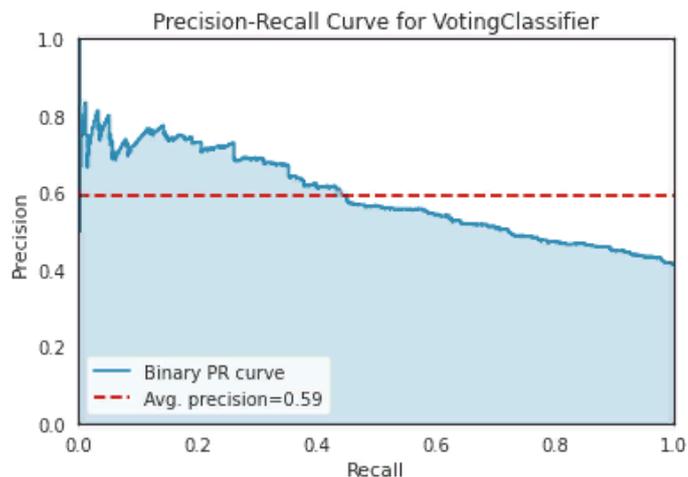


Рисунок 28. График модели случайного леса

По результатам расчетов получили матрицу ошибок.

`plot_model(tuned_blend, plot = 'confusion_matrix')` (рис. 29).

Код, использует функцию `plot_model` для визуализации матрицы ошибок (confusion matrix) для нашей модели. Матрица ошибок позволяет оценить производительность классификационной модели, показывая количество верно и неверно классифицированных образцов для каждого класса.

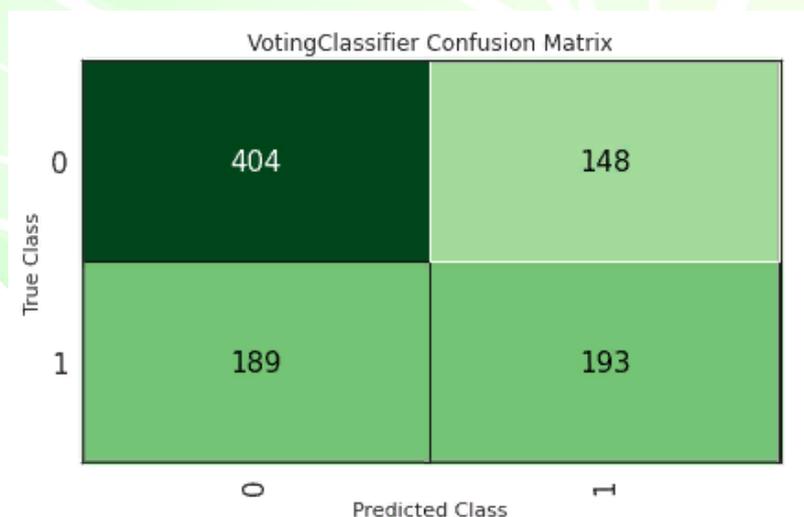


Рисунок 29. Матрица ошибок

Формируем окончательную модель

```

Формируем окончательную модель
final_blend = finalize_model(tuned_blend)
print(final_blend)
plot_model(final_blend,scale=1.3)

```

Код использует функцию `finalize_model` для формирования окончательной модели на основе вашей настроенной модели `tuned_blend`. Затем визуализируем эту окончательную модель с помощью `plot_model`.

После завершения модели с помощью `finalize_model`, получаем окончательную версию модели, которая будет использоваться для предсказаний на новых данных (рис. 30,31).

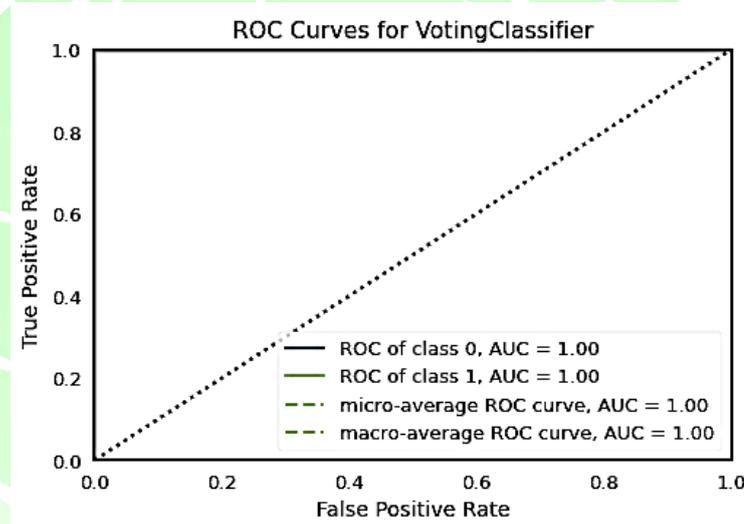


Рисунок 30. Версия модели для предсказаний на новых данных

```
plot_model (final_blend, plot = 'confusion_matrix')
```

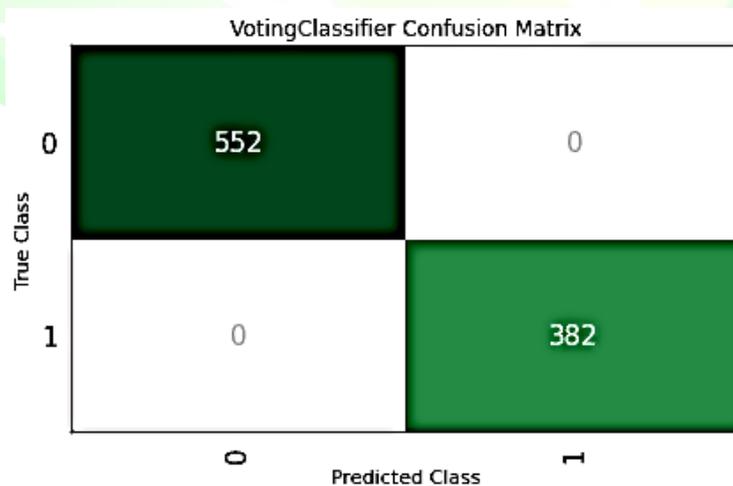


Рисунок 31. Показатели метрик модели.

Определяем предсказание на 5% данных, которые не попали в машинное обучение.

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability	Label	Score
0	6.347272	186.732881	41065.234765	9.629596	364.487687	516.743282	11.539781	75.071617	4.376348	0	1	0.5318
1	7.080795	227.435048	22305.567414	10.333918	333.775777	554.820086	16.331693	45.382815	4.133423	0	0	0.6147
2	6.514415	198.767351	21218.702871	8.670937	323.596349	413.290450	14.900000	79.847843	5.200885	0	1	0.5106
3	7.077874	220.670540	23873.378210	6.181840	341.987234	333.050786	10.661799	90.297770	4.764508	0	0	0.5423
4	6.506488	202.285938	17087.076902	6.201983	326.662368	481.771934	16.535186	46.205748	3.548776	0	1	0.5617
5	4.965774	220.848639	21802.744898	6.250578	311.598136	386.413332	18.035238	53.908177	3.421767	0	0	0.6012
6	6.579969	197.226981	17657.084176	4.506293	290.289894	254.393276	9.847655	73.749716	4.575932	0	1	0.5488
7	6.906992	216.785291	19991.239149	7.302352	340.106028	356.459137	9.614106	61.122644	2.124759	0	1	0.5224
8	6.769573	168.937072	11067.181148	7.121264	305.794950	289.542460	18.545903	62.001571	5.220540	0	0	0.5288
9	4.814136	205.214041	17650.405049	8.121080	350.487939	414.030709	10.999416	47.402666	5.190852	0	0	0.5247
10	9.406326	216.762159	27948.589445	6.156111	355.473035	347.983078	16.340715	32.108292	3.097278	0	0	0.8553
11	7.080795	169.974849	23403.637304	8.519730	333.775777	475.573562	12.924107	50.861913	2.747313	1	0	0.6036
12	5.191834	211.094185	14866.103863	7.238691	323.109285	388.716580	17.780000	83.505406	3.859801	1	1	0.5923
13	9.058628	155.357557	15737.468938	7.251979	343.233977	463.389256	17.854470	83.175189	3.652134	1	0	0.7047
14	5.862641	185.065220	44069.272158	4.382721	412.690111	331.570139	15.306079	59.605812	5.507421	1	1	0.5359
15	7.821971	157.181545	23318.187976	9.084040	324.851659	294.636531	17.186766	72.568376	3.532958	1	0	0.5046
16	9.161089	119.841905	26618.059637	6.105573	343.289234	379.725182	10.899926	54.580736	4.888061	1	0	0.5130
17	9.597192	167.088976	21153.322827	7.944469	346.075016	335.197608	14.347676	50.642447	5.145153	1	0	0.7283
18	7.748655	239.788278	29331.236633	10.713097	217.000607	441.529547	16.389384	66.396293	2.511810	1	1	0.7176
19	7.080795	286.201763	46931.884293	7.440024	262.526456	557.421879	14.471650	74.043860	4.120931	1	1	0.7423

Для предсказания на 5% данных, которые не использовались в обучении модели, используем вашу обученную модель и применяем её к этим данным. Используем функцию `predict()` нашей модели, чтобы получить предсказания для этих данных.

Проверка точности предсказаний по метрике сбалансированная F-мера

Проверка точности предсказаний по метрике сбалансированная F-мера

```
from rucaret.utils import check_metric
check_metric(actual=unseen_predictions['Potability'],
              prediction=unseen_predictions['Label'],metric='F1')
Точность составила 0.4754
Сохраняем модель
save_model(final_blend,'final_blend 04.07.2022')
```

Transformation Pipeline and Model Successfully Saved

```
(Pipeline(memory=None,
 steps=[('dtypes',
         DataTypes_Auto_infer(categorical_features=[],
                               display_types=False, features_todrop=[],
                               id_columns=[],
                               ml_usecase='classification',
                               numerical_features=[],
                               target='Potability', time_features=[])),
        ('imputer',
         Simple_Imputer(categorical_strategy='not_available',
                        fill_value_categorical=None,
                        fill_value_numerical=None,
                        numeric_...
                        max_leaf_nodes=None,
                        max_samples=None,
                        min_impurity_decrease=0.0,
                        min_impurity_split=None,
                        min_samples_leaf=1,
                        min_samples_split=2,
                        min_weight_fraction_leaf=0.0,
                        n_estimators=100,
                        n_jobs=-1,
                        oob_score=False,
                        random_state=234,
                        verbose=0,
                        warm_start=False))),
        flatten_transform=True, n_jobs=-1,
        verbose=False, voting='soft',
        weights=[0.5700000000000001, 0.64]]),
 verbose=False), 'final_blend 04.07.2022.pkl')
```

Метрика сбалансированной F-меры для ваших предсказаний составила 0.4754. Это показатель, который учитывает как точность, так и полноту модели.

Мы сохранили нашу модель с названием "finalblend 04.07.2022" в файле "finalblend 04.07.2022.pkl".

Повторная загрузка сохраненной модели

```
saved_blend = load_model('final_blend 04.07.2022')
```

Transformation Pipeline and Model Successfully Loaded

Проверка на идентичность новых предсказаний по сохраненной модели

```
new_prediction = predict_model(saved_blend, data=water_pc_us)
```

```
new_prediction.to_excel('new_water_predictions.xls')
```

```
new_prediction.head(20).
```

Проверка и загрузка сохраненной модели прошли успешно. Для оценки новых предсказаний с использованием сохраненной модели можно использовать код, который предоставлен. Результаты предсказаний сохранены в файле 'newwaterpredictions.xls'. Первые 20 строк предсказаний показаны в таблице 3, включая различные метрики качества модели, такие как Accuracy, AUC, Recall, Precision, F1, Каппа, MCC, а также значения признаков и предсказанные метки.

Используя соответствующий код, отобразим первые 20 строк предсказаний в виде таблицы с метриками качества модели, значениями признаков и предсказанными метками. Для этого нам понадобится. Давайте начнем с загрузки данных и анализа результатов.

Сравнение результатов машинного обучения различными методами.

Был проанализирован датасет о пригодности потребления воды, с учетом его физических параметров.

В ходе EDA были выявлены пустые данные. Далее была проведена нормализация данных и проверка на зависимость переменных.

Таблица 5

Метрики качества модели (Accuracy, AUC, Recall, Precision, F1, Kappa, MCC), а также значения признаков и предсказанные метки

Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC					
0 Voting Classifier	0.6098	0.6279	0.5273	0.4328	0.4754	0.1695	0.1716					
ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability	Label		
0	6.347272	186.732881	41065.234765	9.629596	364.487687	516.743282	11.539781	75.071617	4.376348	0	1	0.5318
1	7.080795	227.435048	22305.567414	10.333918	333.775777	554.820086	16.331693	45.382815	4.133423	0	0	0.6147
2	6.514415	198.767351	21218.702871	8.670937	323.596349	413.290450	14.900000	79.847843	5.200885	0	1	0.5106
3	7.077874	220.670540	23873.378210	6.181840	341.987234	333.050786	10.661799	90.297770	4.764508	0	0	0.5423
4	6.506488	202.285938	17087.076902	6.201983	326.662368	481.771934	16.535186	46.205748	3.548776	0	1	0.5617
5	4.965774	220.848639	21802.744898	6.250578	311.598136	386.413332	18.035238	53.908177	3.421767	0	0	0.6012
6	6.579969	197.226981	17657.084176	4.506293	290.289894	254.393276	9.847655	73.749716	4.575932	0	1	0.5488
7	6.906992	216.785291	19991.239149	7.302352	340.106028	356.459137	9.614106	61.122644	2.124759	0	1	0.5224
8	6.769573	168.937072	11067.181148	7.121264	305.794950	289.542460	18.545903	62.001571	5.220540	0	0	0.5288
9	4.814136	205.214041	17650.405049	8.121080	350.487939	414.030709	10.999416	47.402666	5.190852	0	0	0.5247
10	9.406326	216.762159	27948.589445	6.156111	355.473035	347.983078	16.340715	32.108292	3.097278	0	0	0.8553
11	7.080795	169.974849	23403.637304	8.519730	333.775777	475.573562	12.924107	50.861913	2.747313	1	0	0.6036
12	5.191834	211.094185	14866.103863	7.238691	323.109285	388.716580	17.780000	83.505406	3.859801	1	1	0.5923
13	9.058628	155.357557	15737.468938	7.251979	343.233977	463.389256	17.854470	83.175189	3.652134	1	0	0.7047
14	5.862641	185.065220	44069.272158	4.382721	412.690111	331.570139	15.306079	59.605812	5.507421	1	1	0.5359

```
15 7.821971 157.181545 23318.187976 9.084040 324.851659
294.636531 17.186766 72.568376 3.532958 1 0 0.5046
16 9.161089 119.841905 26618.059637 6.105573 343.289234
379.725182 10.899926 54.580736 4.888061 1 0 0.5130
17 9.597192 167.088976 21153.322827 7.944469 346.075016
335.197608 14.347676 50.642447 5.145153 1 0 0.7283
18 7.748655 239.788278 29331.236633 10.713097 217.000607
441.529547 16.389384 66.396293 2.511810 1 1 0.7176
19 7.080795 286.201763 46931.884293 7.440024 262.526456
557.421879 14.471650 74.043860 4.120931 1 1 0.7423
```

Выводы

В ходе работы использовались библиотеки Sklearn и CatBoost, а также библиотеки с низким кодом PyCaret. Наилучшие результаты показали алгоритмы дерева решений RandomForestClassifier, DecisionTreeClassifier. Наиболее значимым признаком является pH, который является важным параметром при оценке кислотно-щелочного баланса воды. Это также индикатор кислотного или щелочного состояния воды. На этом процесс машинного обучения по оценке качества воды и её изменения завершен. Дальнейший контроль за изменением качества воды повторяется и по мере накопления данных, поступающих в файл data.set от датчиков, производится новое обучение модели.

Заключение

Согласно [20] комплексное внедрение хотя бы трёх элементов точного земледелия позволяет экономить 20–40 % средств на операции по сравнению с традиционными методами, «умное» сельское хозяйство, согласно мировому рейтингу потенциального позитивного эффекта глобальных технологий, занимает первое место в мире.

В работе [10] представлена динамика количества публикаций об интеллектуальных ирригационных системах с течением времени – рисунок 32. Исследование свидетельствует о растущем интересе к теме интеллектуальных ирригационных систем.

Как показано на рисунке 32 [10], страной, проявляющей наибольший интерес к данной теме, является Индия (71 публикация), выделенная

красным. Она намного превосходит соседнюю страну, поскольку за исследуемый период в Индии было выпущено 51% всех публикаций. За Индией по порядку следуют США (12 публикаций), Индонезия (9 публикаций), Бразилия (8 публикаций) и Китай (8 публикаций), все они отмечены оранжевым на рисунке 32.

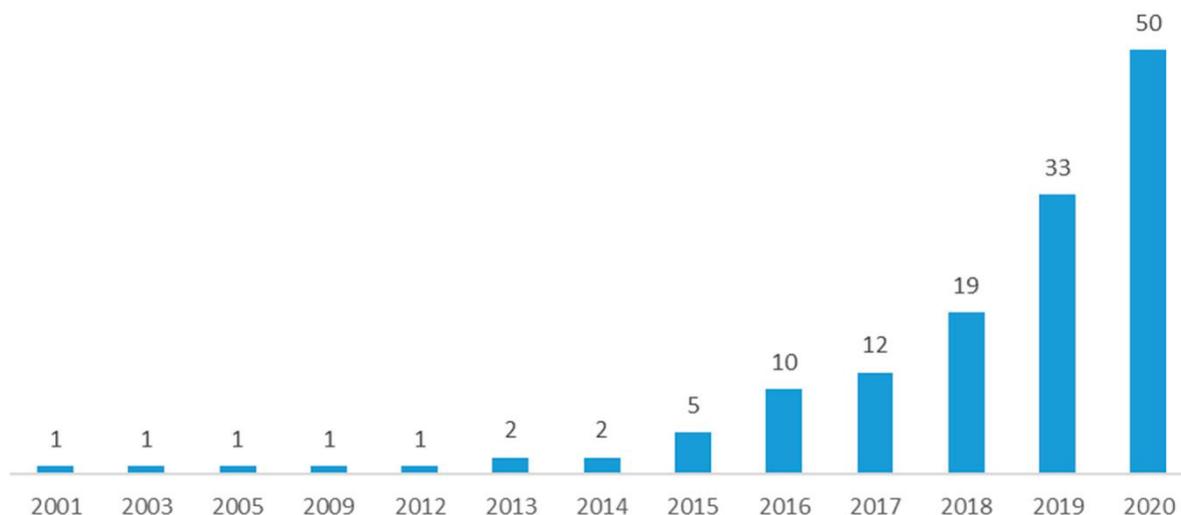


Рисунок 32. Динамика количества публикаций об интеллектуальных ирригационных системах с течением времени [2]



Рисунок 33. Количество публикаций на страну [10]

На рисунке 4 показано облако слов, состоящее из предложений, которые считаются значимой комбинацией совпадающих слов, повторяющихся в статьях Scopus [10]. Самые крупные элементы на рисунке — это те, которые встречаются чаще всего — это “нейро-нечеткая система”, “реальное время”, “водопользование”, “водные ресурсы”, “изменение климата” и “низкая стоимость”.

Городское сельское хозяйство было предметом особого интереса в этом обзоре; поэтому был проведен анализ именно этого вида сельского хозяйства. 28 публикаций относятся к городскому сельскому хозяйству из 170 и становятся очевидными пропорции двух наиболее используемых в настоящее время технологий для городского сельского хозяйства. Это IoT, что свидетельствует о полезности этой технологии в городских сельскохозяйственных условиях и подтверждает гипотезу о простоте использования сетевых технологий в городах, а также машинное обучение и нечеткая логика.

Присутствие термина «больших данных» определяет их как дополнительную технологию, используемую в интеллектуальных системах орошения.

Представлены публикации по типу техники орошения. Наиболее часто используются дождевание, капельные, спринклерные разбрызгиватели.

Относительно небольшое количество публикаций об интеллектуальных системах орошения в городском сельском хозяйстве при выращивании почвенных культур означает, что у будущих исследователей есть возможность предложить решения для орошения в таких системах. Одной из возможных альтернатив является адаптация решений ирригационных систем, которые уже были опробованы в сельском хозяйстве, к городскому сельскому хозяйству [10].

Интернет вещей — это прорывная технология во многих секторах, включая сельское хозяйство. Внедрение Интернета вещей в ирригационных системах сочетается со многими технологиями благодаря преимуществам, связанным с представлением физических элементов в виде данных, а также простоте получения и сбора данных с датчиков и передачи их либо в облако, либо между встроенными системами. Встроенные системы — это компьютерные системы, которые выполняют определенную задачу внутри машины или более обширной электрической системы.

Есть предложения использовать ML и IoT для внедрения ирригационной системы, которая учитывает переменные влажности почвы, температуры, относительной влажности и pH для своей работы. Кроме того, система обеспечивает прогнозирование того, какие культуры можно сажать в зависимости от почвы и погодных условий, чтобы избежать потерь перед уборкой. Показано, с какой легкостью IoT позволяет осуществлять мониторинг и запись данных с датчиков для последующего управления.

Есть пример, где реализована сеть датчиков, использующая интернет вещей и большие данные и генерирующая большой объем данных, который растет экспоненциально со временем, что требует нетрадиционных приложений компьютерной обработки для правильной обработки данных. Например, он используется для орошения сельскохозяйственных культур открытого грунта и для сравнения трех алгоритмов ML для прогнозирования влажности почвы. В этих моделях система нечеткой логики отвечает за управление поливом. Есть предложения использования алгоритма ML для системы орошения сельскохозяйственных культур. Кроме того, платформа IoT используется для подключения физических устройств системы к мобильному приложению для визуализации интересующих данных [10].

Как упоминалось во введении, ML полезен в сельском хозяйстве, поскольку позволяет компьютерам извлекать информацию из имеющихся данных, таких как различные погодные параметры, непрерывно измеряемые метеостанциями, или результаты измерений, полученных в результате наблюдения за урожаем в течение длительного времени. ML использует эти данные в своих интересах, добавляя их в математические алгоритмы, которые предназначены для прогнозирования или классификации некоторых интересующих переменных. Например, величина суммарного испарения может быть использована для оценки необходимых периодов орошения сельскохозяйственных культур.

Стало очевидным, что встроенные системы предпочтительнее при реализации прототипов интеллектуальных систем орошения, в которых

используются технологии такие как IoT, ML, ANNs (Artificial Neural Networks, искусственные нейронные сети) и DL (Deep Learning, глубокое обучение).

IoT — это базовая технология, часто используемая в альтернативах ирригационным системам, включающих интеллектуальное управление. Это связано с тем, что эта технология позволяет отображать физические объекты в Интернете, а передача данных между устройствами упрощается. Таким образом, процессы сбора данных, мониторинга и дистанционного управления в предлагаемых ирригационных системах упрощаются.

Ирригационные системы, использующие ML, ANNs и DL, могут использоваться во многих случаях для реализации инновационных предложений, таких как методы машинного зрения или технологии прогнозирования поведения представляющих интерес переменных, таких как влажность. Такие инновационные предложения возможны, учитывая, что ML, ANNs и DL позволяют обрабатывать значительный объем данных.

С другой стороны, нечеткая логика — это другой тип технологии, который неоднократно упоминается в литературе, связанной с ML, ANNs и DL. В этих случаях обычно за управление орошением отвечает система нечеткой логики. Ирригационная система, использующая ML, ANNs или DL, является очень жизнеспособным вариантом при наличии большого и надежного объема данных. Однако, если есть необходимость внедрить систему орошения с нуля и у вас есть практические знания в данной области, система нечеткой логики - лучший вариант. Этот подход также легко допускает оптимизацию, как показано в различных публикациях по моделям ANFIS.

ANFIS (Adaptive Neuro-Fuzzy Inference System) — это гибридная модель, которая сочетает в себе элементы нейронных сетей и нечеткой логики. Она используется для решения задач, где требуется обработка неопределенности и приближенное представление знаний, таких как прогнозирование, классификация и управление.

Исследования интеллектуальных систем орошения имеют решающее значение для дальнейшего совершенствования сельскохозяйственных процессов.

Важным элементом для авторов этой работы является возможность применения метода к сельскохозяйственным системам в городских условиях. В этом смысле информация, собранная из проанализированных здесь работ, подразумевает, что интеллектуальные ирригационные системы для сельского хозяйства, с необходимыми модификациями, в высокой степени воспроизводимы для городского сельского хозяйства (особенно для культур, использующих почву в качестве субстрата), поскольку в городах есть все необходимые ресурсы, такие как пространство, вода и электричество. Однако эти ресурсы обычно сопряжены со значительными затратами или более ограничены, чем в сельском хозяйстве.

Интересно отметить относительно низкую частоту сообщений о технологиях интеллектуального орошения в городских условиях по сравнению с сельскими районами, несмотря на преимущества, которыми обладают города для внедрения интеллектуальных систем орошения. Это неудивительно, поскольку большинство сельскохозяйственных культур выращивается в сельской местности, но это привлекает внимание к потенциалу развития такого типа систем в городском сельском хозяйстве [10].

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Vallejo-Gómez, D.; Osorio, M.; Hincapié, C.A. Smart Irrigation Systems in Agriculture: A Systematic Review. *Agronomy* 2023, 13, 342. <https://doi.org/10.3390/agronomy13020342>
<https://www.mdpi.com/2073-4395/13/2/342> 15
2. Ruiz-Real J. L. et al. A look at the past, present and future research trends of artificial intelligence in agriculture // *Agronomy*. – 2020. – Т. 10. – №. 11. – С. 1839.
https://scholar.google.com/scholar_lookup?title=A+Look+at+the+Past,+Present+and+Future+Research+Trends+of+Artificial+Intelligence+in+Agriculture&author=Ruiz-Real,+J.L.&author=Uribe-Toril,+J.&author=Arriaza,+J.A.T.&author=de+Valenciano,+J.P.&publication_year=2020&journal=Agronomy&volume=10&pages=1839&doi=10.3390/AGRONOMY10111839
3. . Liakos K. G. et al. Machine Learning in Agriculture: A Review / *Sensors*. – 2018. – Т. 18. – С. 2674.
https://www.scienceopen.com/document_file/270206fe-42f2-4006-a7b7-64938225fe10/PubMedCentral/270206fe-42f2-4006-a7b7-64938225fe10.pdf
4. González García C. et al. A review of artificial intelligence in the internet of things // *International Journal Of Interactive Multimedia And Artificial Intelligence*, 5. – 2019.
https://www.researchgate.net/publication/324048060_A_Review_of_Artificial_Intelligence_in_the_Internet_of_Things
5. Kamilaris A., Prenafeta-Boldú F. X. Deep learning in agriculture: A survey // *Computers and electronics in agriculture*. – 2018. – Т. 147. – С. 70-90.
<https://arxiv.org/pdf/1807.11809>
6. Мукешимана К. Применение модели нечёткой логики для оценки качества передачи данных в сетях индустриального интернета вещей. *Компьютерные инструменты в образовании*, 2019 № 1: 29–42
<http://ipo.spb.ru/journal>
doi:10.32603/2071-2340-2019-1-29-42
<http://ipo.spb.ru/journal/index.php?article/2121/>
7. Khatri V. Application of Fuzzy logic in water irrigation system // *Int. Res. J. Eng. Technol.* – 2018. – Т. 5. – №. 4. – С. 3372.
8. Чернов, Р. В. Необходимость применения цифровых технологий при подготовке специалистов АПК / Р. В. Чернов, Н. Г. Казыдуб // *Профессиональное самоопределение молодежи инновационного региона: проблемы и перспективы* : Сборник статей по материалам Всероссийской (национальной) научно-практической конференции, Красноярск, 15–29 ноября 2021 года. – Красноярск: Красноярский государственный аграрный университет, 2022. – С. 376-379. – EDN IENWVW.

9. Москалев, С. М. Искусственный интеллект и интернет вещей как инновационные методы совершенствования агропромышленного сектора / С. М. Москалев, Н. В. Клименок-Кудинова // Известия Санкт-Петербургского государственного аграрного университета. – 2018. – № 52. – С. 121-130. – EDN YNDQGL

10. Precision Irrigation Management Using Machine Learning and Digital Farming Solutions by Emmanuel Abiodun Abioye, Oliver Hensel, Travis J. Esau, Olakunle Elijah, Mohamad Shukri Zainal Abidin, Ajibade Sylvester Ayobami, Omosun Yerima and Abozar Nasirahmadi11. . Review. AgriEngineering 2022, 4, 70-103; <https://doi.org/10.3390/agriengineering4010006>

11. A Smart Irrigation System Powered by IoT and Machine Learning for Optimal Water Management by Ritika Dhyani; Nikita Manne; Jyoti Garg; Deepak Motwani; Ashok Kumar Shrivastava; Meenakshi Sharma. 2024 4th International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE) 14-15 May 2024. Page(s):1801 - 1805. DOI: 10.1109/ICACITE60783.2024.10617429

<https://ieeexplore.ieee.org/document/8259674>

12. Joshua Hrisko. Capacitive Soil Moisture Sensor Theory, Calibration, and Testing. Maker Portal LLC New York, NY July 5, 2020. https://www.researchgate.net/publication/342751186_Capacitive_Soil_Moisture_Sensor_Theory_Calibration_and_Testing

13. Rain Bird. The Intelligent Use of Water.™ Оборудование для ландшафтного орошения. Каталог. 148стр.

14. Датчик влажности почвы HUNTER Soil-clik. Руководство по установке. 24 стр.

15. Андреас Мюллер, Сара Гвидо. Введение в машинное обучение с помощью Python. Руководство для специалистов по работе с данными. Москва 2016-2017. – 393с.

16. Al-Ghobari, H.M.; Dewidar, A.Z. Integrating deficit integration into surface and subsurface drip irrigation as a strategy to save water in arid regions. Agric. Water Manag. 2018, 2019, 55-61.

17. Федосов А. Ю., Меньших А.М., Иванова М. И., Рубцов А. А. Инновационные технологии орошения овощных культур. – М.: изд-во «Ким Л. А.», 2021. - 306 с.

18. Fartukov, V. A. Multifunctional multilevel moisture sensor / V. A. Fartukov, V. A. Zimnyukov, M. I. Zborovskaya // Nexo Revista Científica. – 2022. – Vol. 35, No. 02. – P. 580-587. – <https://doi.org/10.5377/nexo.v35i02.14638> . – EDN MDXZPO

19. M.M. Maina, M.S.M. Amin and M.A. Yazid Web geographic information system decision support system for irrigation water management: a review article.

Acta Agriculturae Scandinavica, Section B — Soil & Plant Science, 2014 Vol. 64, No. 4, 283–293, <http://dx.doi.org/10.1080/09064710.2014.896935>

20. Районированная цифра. Цифровизация АПК // Тематическое приложение к ежедневной деловой газете РБК. 2019. №112 (3067). [Электронный ресурс]. Режим доступа: <https://plus.rbc.ru> › pdf › rbcplus_1



**Фартуков Василий Александрович
Зборовская Марина Ильинична**

**ФОРМИРОВАНИЕ АЛГОРИТМА
МАШИННОГО ОБУЧЕНИЯ
ПО УПРАВЛЕНИЮ СИСТЕМОЙ ПОЛИВА**

Учебное пособие

Обложка художника
Компьютерная верстка
Корректор

Подписано в печать Формат 60×84 1/16
Бумага писчая Гарнитура шрифта «Times New Roman» Печать офсетная
Печ. л. Тираж экз. Изд. Заказ Тип заказа

Отпечатано в типографии